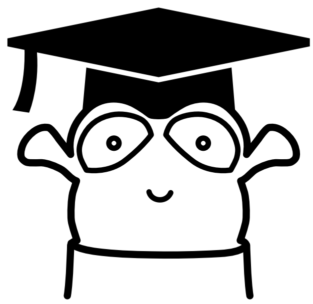


Real-Time Generation and Adaptation of Social Companion Robot Behaviors

Dissertation

zur Erlangung des Doktorgrades an der
Fakultät für Angewandte Informatik
der Universtität Augsburg

vorgelegt von

Hannes Ritschel

2022

Erstgutachterin	Prof. Dr. Elisabeth André
Zweitgutachter	Prof. Dr. Björn Schuller
Drittgutachterin	Prof. Dr. Catherine Pelachaud

Tag der mündlichen Prüfung: 7. Dezember 2022

To my parents.

Abstract

Social robots will be part of our future homes. They will assist us in everyday tasks, entertain us, and provide helpful advice. However, the technology still faces challenges that must be overcome to equip the machine with social competencies and make it a socially intelligent and accepted housemate.

An essential skill of every social robot is verbal and non-verbal communication. In contrast to voice assistants, smartphones, and smart home technology, which are already part of many people's lives today, social robots have an embodiment that raises expectations towards the machine. Their anthropomorphic or zoomorphic appearance suggests they can communicate naturally with speech, gestures, or facial expressions and understand corresponding human behaviors. In addition, robots also need to consider individual users' preferences: everybody is shaped by their culture, social norms, and life experiences, resulting in different expectations towards communication with a robot. However, robots do not have human intuition – they must be equipped with the corresponding algorithmic solutions to these problems.

This thesis investigates the use of reinforcement learning to adapt the robot's verbal and non-verbal communication to the user's needs and preferences. Such non-functional adaptation of the robot's behaviors primarily aims to improve the user experience and the robot's perceived social intelligence. The literature has not yet provided a holistic view of the overall challenge: real-time adaptation requires control over the robot's multimodal behavior generation, an understanding of human feedback, and an algorithmic basis for machine learning. Thus, this thesis develops a conceptual framework for designing real-time non-functional social robot behavior adaptation with reinforcement learning. It provides a higher-level view from the system designer's perspective and guidance from the start to the end. It illustrates the process of modeling, simulating, and evaluating such adaptation processes. Specifically, it guides the integration of human feedback and social signals to equip the machine with social awareness.

The conceptual framework is put into practice for several use cases, resulting in technical proofs of concept and research prototypes. They are evaluated in the lab and in in-situ studies. These approaches address typical activities in domestic environments, focussing on the robot's expression of personality, persona, politeness, and humor. Within this scope, the robot adapts its spoken utterances, prosody, and animations based on human explicit or implicit feedback.

Zusammenfassung

Soziale Roboter werden Teil unseres zukünftigen Zuhauses sein. Sie werden uns bei alltäglichen Aufgaben unterstützen, uns unterhalten und uns mit hilfreichen Ratschlägen versorgen. Noch gibt es allerdings technische Herausforderungen, die zunächst überwunden werden müssen, um die Maschine mit sozialen Kompetenzen auszustatten und zu einem sozial intelligenten und akzeptierten Mitbewohner zu machen.

Eine wesentliche Fähigkeit eines jeden sozialen Roboters ist die verbale und nonverbale Kommunikation. Im Gegensatz zu Sprachassistenten, Smartphones und Smart-Home-Technologien, die bereits heute Teil des Lebens vieler Menschen sind, haben soziale Roboter eine Verkörperung, die Erwartungen an die Maschine weckt. Ihr anthropomorphes oder zoomorphes Aussehen legt nahe, dass sie in der Lage sind, auf natürliche Weise mit Sprache, Gestik oder Mimik zu kommunizieren, aber auch entsprechende menschliche Kommunikation zu verstehen. Darüber hinaus müssen Roboter auch die individuellen Vorlieben der Benutzer berücksichtigen. So ist jeder Mensch von seiner Kultur, sozialen Normen und eigenen Lebenserfahrungen geprägt, was zu unterschiedlichen Erwartungen an die Kommunikation mit einem Roboter führt. Roboter haben jedoch keine menschliche Intuition – sie müssen mit entsprechenden Algorithmen für diese Probleme ausgestattet werden.

In dieser Arbeit wird der Einsatz von bestärkendem Lernen untersucht, um die verbale und nonverbale Kommunikation des Roboters an die Bedürfnisse und Vorlieben des Benutzers anzupassen. Eine solche nicht-funktionale Anpassung des Roboterverhaltens zielt in erster Linie darauf ab, das Benutzererlebnis und die wahrgenommene soziale Intelligenz des Roboters zu verbessern. Die Literatur bietet bisher keine ganzheitliche Sicht auf diese Herausforderung: Echtzeitanpassung erfordert die Kontrolle über die multimodale Verhaltenserzeugung des Roboters, ein Verständnis des menschlichen Feedbacks und eine algorithmische Basis für maschinelles Lernen. Daher wird in dieser Arbeit ein konzeptioneller Rahmen für die Gestaltung von nicht-funktionaler Anpassung der Kommunikation sozialer Roboter mit bestärkendem Lernen entwickelt. Er bietet eine übergeordnete Sichtweise aus der Perspektive des Systemdesigners und eine Anleitung vom Anfang bis zum Ende. Er veranschaulicht den Prozess der Modellierung, Simulation und Evaluierung solcher Anpassungsprozesse. Insbesondere wird auf die Integration von menschlichem Feedback und sozialen Signalen eingegangen, um die Maschine mit sozialem Bewusstsein auszustatten.

Der konzeptionelle Rahmen wird für mehrere Anwendungsfälle in die Praxis umgesetzt, was zu technischen Konzeptnachweisen und Forschungsprototypen führt, die in Labor- und In-situ-Studien evaluiert werden. Diese Ansätze befassen sich mit typischen Aktivitäten in häuslichen Umgebungen, wobei der Schwerpunkt auf dem Ausdruck der Persönlichkeit, dem Persona, der Höflichkeit und dem Humor des Roboters liegt. In diesem Rahmen passt der Roboter seine Sprache, Prosodie, und Animationen auf Basis expliziten oder impliziten menschlichen Feedbacks an.

Acknowledgments

I would particularly like to thank my supervisor, Prof. Dr. Elisabeth André, for her constant support and advice. She provided valuable suggestions and comments from the initial idea to the final thesis. Thank you for giving me the oppertunity to work on social robots at your lab. I would also like to thank Prof. Dr. Björn Schuller and Prof. Dr. Catherine Pelachaud for consenting to volunteer as reviewers. Thanks for spending your precious time on this thesis. Thanks also go to current and former colleagues at the HCAI lab for the good cooperation and inspiring conversations. I am especially thankful to Dr. Andreas Seiderer, Dr. Simon Flutura, Katharina Weitz, and Thomas Kiderle, for the refreshing chats and discussions over the years. Finally, I am deeply grateful to my parents for supporting my studies from the outset.

Contents

Acronyms

API application programming interface

ASR automatic speech recognition

CI confidence interval

GUI graphical user interface

HCI human-computer interaction

HRI human-robot interaction

MDP Markov decision process

NLG natural language generation

NLP natural language processing

PGRL policy gradient reinforcement learning

POMDP partially observable Markov decision process

RL reinforcement learning

RMSE root mean-squared error

SSI Social Signals Interpretation

SSML speech synthesis markup language

SSP social signal processing

TAMER Training an Agent Manually via Evaluative Reinforcement

TTS text-to-speech

UCB upper-confidence bound

URBI Universal Robot Body Interface

VSM Visual SceneMaker

WoZ Wizard of Oz

Part I.

Motivation and Background

1. Introduction

Over the past decades, robots have evolved from big, bulky, and expensive machines to more compact, affordable, versatile, and mobile devices. Even though they have not yet entered our daily lives everywhere, the underlying technology has made massive progress. One of the key challenges in human-robot interaction (HRI) has always been to equip these machines with an intuitive, understanding, and pleasing interface for humans, intending to make them *social* robots. Early on, science found that people like to interact with robots that exhibit familiar, humanoid behaviors. No wonder, then, that many social robots are not only designed to have an anthropomorphic or zoomorphic appearance with eyes, ears, a mouth, arms, hands, legs, and feet, but also to communicate with us via associated communication channels.

In general, *communication* is an essential and multifaceted social competence. Humans use verbal and non-verbal communication channels, such as speech, gestures, facial expressions, and gaze. These behaviors allow for very flexible and complex communication in almost any environment, be it auditory, visual or haptic, or a combination of two or more. For humans, *generating*, *perceiving*, and *interpreting* all these cues may be a matter of course, as we learn and develop from an early age. However, it is by no means so for the machine. Social robots need to master various skills that humans often take for granted to be accepted and perceived as a "social" entity.

Apart from the ability to express oneself with speech, gestures, and the like, everybody has their individual personality and communication style, which results, i.a., from learned social norms, social background, culture, and more. For example, in plays of the time of Shakespeare, there existed two addresses: the familiar address "thou" and the formal address "you". The German counterpart is the use of the formal address "Sie" in contrast to the informal "Du", both of which are still in use today as an expression of politeness depending on whether communication happens with a stranger or a friend. Communication also differs between generations in terms of constantly changing vocabulary and neologisms, social status, in-group markers to express group membership, and much more. Beyond language, communication also differs between cultures in terms of directness and expressiveness of gestures, showing or hiding emotions, and much more. These experiences result in individual expectations towards our dialog partners – and social robots! – on what *we* expect how a human or robot should communicate with us.

As a consequence, social robots do not only need tools for verbal and non-verbal communication, but they also need to *adapt* their behaviors to the individual user's needs and preferences. Humans use their intuition to evaluate whether their communication attempts are expedient or not. For this purpose, we perceive verbal and non-verbal behaviors of our dialog partner, including subliminal *social signals*: the way *how* one reacts in terms of prosody, gaze behavior, facial expression, gestures, and posture communicates much more than the spoken words in isolation. Thus, social robots should also be able to sense, process, and interpret such signals as *feedback* for adaptation.

With human intelligence being a result of experience collected during a lifelong learning process, it seems logical to look at social robot adaptation from the perspective of machine learning, and “artificial intelligence”. Do algorithms allow the machine to adapt to the individual user, to learn about their preferences, and are they able to generate personalized behaviors? Learning and adaptation have been crucial to success in human evolution, and thus it is also one opportunity for social robots to master their communication with humans.

1.1. Motivation

Social robots will be part of our future domestic environments – similar to digital voice assistants, smartphones, and smart home technology, which are already part of many people’s everyday lives. Social robots at home will need to provide us with many features, including entertainment, assistance, information retrieval, and communication, but also an attractive embodiment and personality. de Graaf and Allouch (2013) identify usefulness, enjoyment, companionship, behavioral control, sociability, and adaptability as essential variables for robot acceptance. Generating robot behaviors that mimic human-like behaviors and suggest social intelligence, such as humor and politeness, and their adaptation to the individual user’s needs and preferences, are up-to-date challenges for HRI in our own four walls.

The literature addresses various forms of user-adaptive interaction. There is *functional* adaptation, focussing on *what* the robot does. In a goal-oriented interaction, such as a sorting task, the robot can adapt its actions to the user’s desired sorting criteria. Should it sort according to object color, size, or shape? In which order? On the other side, there is *non-functional* adaptation, focussing on *how* the robot executes its actions – independently of the interaction goal. Should it move fast or slow? Should it give comments on the procedure, or should it ask more or less politely for help in case of problems?

In addition, there are also different approaches for implementing adaptation. For example, the robot’s behaviors can be initially configured based on a user questionnaire before starting the interaction. Many studies, which address the adaptation of verbal and non-verbal robot behaviors, use this approach. On the other side, the more complicated approach is to adapt the robot’s behaviors during the interaction. Then, the robot must ask the user for feedback or infer the user’s preferences or needs automatically.

Most of the literature and studies focus on functional adaptation; there are fewer works and insights about non-functional adaptation (see Figure 1.1). Those works, which realize non-functional adaptation, often configure the robot once in the beginning but do not implement learning during the interaction. However, a real-time adaptation approach is desirable for the robot’s autonomy and for disburdening the user from continuously providing feedback. It becomes even more challenging when the context is not a goal-oriented interaction. Without measurable data from the task, the robot needs other sources of feedback to adapt to the user.

This thesis addresses this research gap: how to realize non-functional adaptation of verbal and non-verbal social robot behaviors in real-time to the individual user when feedback cannot be inferred directly from the task? Apart from demanding feedback from the user during the interaction explicitly with a prompt, a key technology for

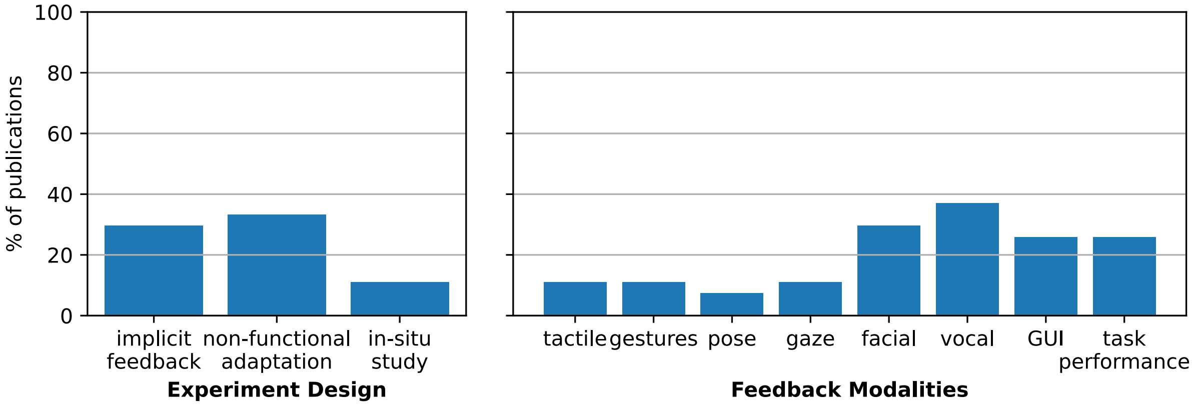


Figure 1.1.: Overview of the amount of literature addressing non-functional adaptation, implicit user feedback, and used feedback modalities. See also chapter 6.

autonomous adaptation is social signal processing (SSP). Human social signals cover different modalities and are a continuous source of information about the user's affective state, engagement, attention, and more (see Figure 1.1). They occur in any interaction because human communication is multimodal and happens to a large extent subconsciously. The inclusion of human social signals as implicit feedback and their combination with a suitable machine learning approach are key challenges to solve.

However, the path of real-time adaptation is a rocky one. Several questions arise concerning acquiring feedback based on human social signals and the algorithmic basis needed to drive the adaptation process. Besides SSP, there are more difficulties to solve for adaptation: which signals are most related to the adaptation goal, how, and when should the robot measure them? How can the sensed data be integrated into a real-time adaptation process? At the same time, HRI has specific requirements on the algorithmic basis: the robot should learn in real-time and be autonomous without expert input. It needs to operate in an uncertain environment where external influences might bias the user's feedback. Adaptation should happen over an extended period to react to changing user needs and preferences, and – ideally – adaptation should happen in the background to not interrupt the interaction.

First attempts have been made in the literature, addressing single aspects without a holistic view of the overall problem. A general overview and guidance on the whole procedure are still needed. This thesis will fill this gap with conceptual, technical, and empirical contributions and insights. It will provide a higher-level view on non-functional social robot behavior adaptation from the system designer's perspective. It will guide the reader from the start to the end by breaking down the complete process of designing a real-time non-functional adaptation process for social robots. The overall structure splits up the problem in

1. generating robot behaviors that allow for adaptation (*what* to adapt?), and
2. designing the algorithmic approach for implementing adaptation (*how* to adapt?).

Figure 1.2 illustrates the interrelation: the robot communicates generated behaviors to the user with the help of the output modalities provided by the hardware. The user provides explicit or implicit feedback – including social signals – to the adaptation process,

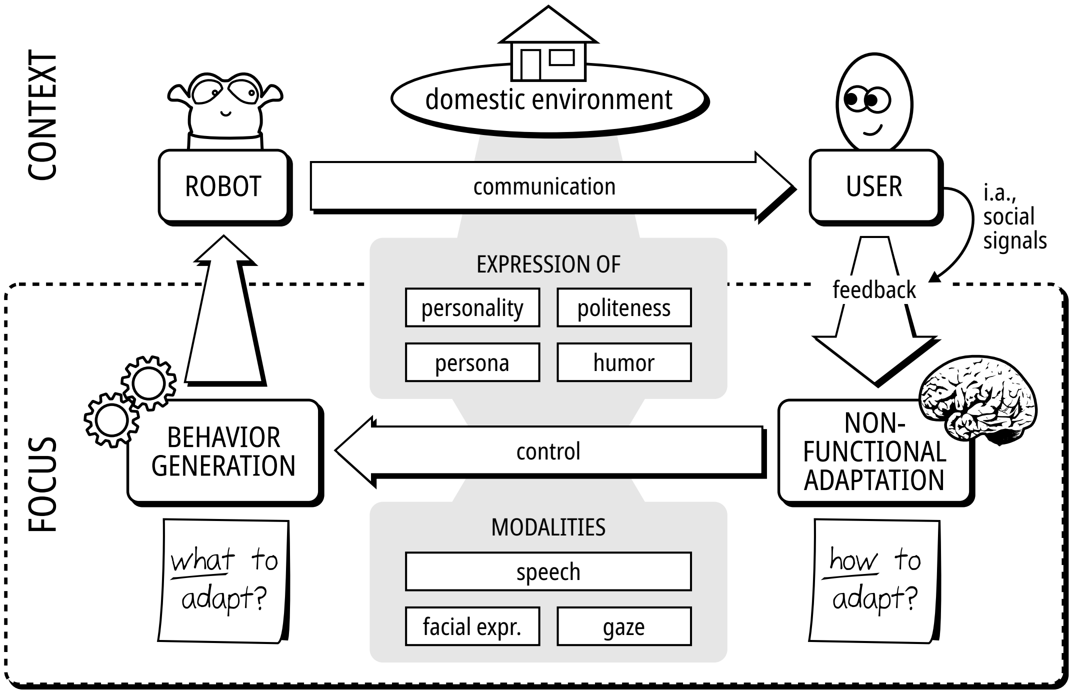


Figure 1.2.: The motivation and the high-level overview of the contributions and context of this thesis.

which then controls and manipulates the generation of the following robot behaviors. This loop repeats over and over again.

The overall idea is a generalized structure with three building blocks for designing such an adaptation process. During *problem modeling*, the system designer lays out the theoretical and general structure of the adaptation approach in consideration of the adaptation goal and problem-specific properties. Specifically, this step also defines the inclusion of human social signals in the adaptation process. Afterward, *simulations* allow for a technical evaluation of the model, its convergence, tweaking parameters, and more. In particular, a simulation can train the robot without human interaction and, as a result, provide the base knowledge for the robot's subsequent interaction with the user. The final *human evaluation* gives insights into the performance of the adaptation approach and resulting impacts on user experience in real HRI. In contrast to simulations, the knowledge gained from human evaluations allows the robot to learn the unexpected: the robot can learn actual user preferences only in real interactions.

This thesis will provide details on critical key considerations and connections between problem modeling, simulations, and human evaluations. It proposes to use the reinforcement learning (RL) framework (Sutton and Barto, 2018) as the algorithmic basis, which has gained increased attention in recent years for adaptation and personalization (den Hengst et al., 2020). RL relies on trial and error: the robot gives several *actions* a try and evaluates which of them is the most effective in a given *state* based on the user's feedback – the *reward*. RL fulfills the requirements mentioned above for non-functional adaptation in HRI: it allows for continuous, autonomous, and real-time adaptation during interaction in uncertain environments. Details will include human feedback and social signals in the state space and the reward signal of a RL agent. They aim to provide the research

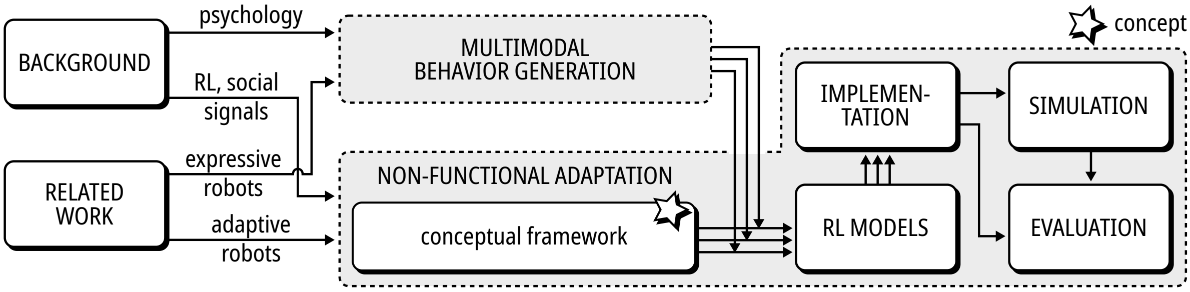


Figure 1.3.: The workflow of this thesis.

community with a conceptual framework for equipping robots with "social awareness" of their human counterparts and creating socially intelligent robot behaviors.

Apart from these conceptual contributions, this thesis will develop and implement such adaptation models and present resulting research prototypes. Given the domestic environment as an application area, the work will focus on generating robot behaviors that portray essential behavior variables. Different types of personality, politeness, persona, and even humor (see Figure 1.2) will be explored, motivated by their interrelation and importance for equipping the robot with socially intelligent behaviors. Several experiments will simulate and evaluate the proposed models in the lab and in-situ studies based on the implemented prototypes.

All in all, this thesis aims to fill the research gap of real-time and non-functional social robot behavior adaptation and enrich the state-of-the-art research with

1. a conceptual framework for modeling real-time non-functional social robot behavior adaptation from start to finish,
2. instantiations of the framework as reusable and robot-independent models of user-adaptive robot communication in domestic environments,
3. proofs of concept and research prototypes as a technical basis for evaluating the implemented models, and
4. insights into their performance and impacts on user experience in simulations, lab, and in-situ studies.

1.2. Research Objectives

The research objectives listed below will be addressed with a top-down approach (see Figure 1.3). The psychological background and the literature about social robot behaviors will serve as a baseline and inspiration for implementing the robot's multimodal communication. The RL theory, background about social signals, and the literature about adaptive social robots will serve as input to a conceptual framework for modeling real-time non-functional social robot adaptation with RL and explicit or implicit human feedback. New RL models will be implemented, simulated, and evaluated based on the conceptual framework and behavior generation approaches.

1.2.1. Generation of Multimodal Socially Intelligent Robot Behaviors

In domestic environments, social robots are integrated into the users' everyday lives. The robot needs expressive behaviors to communicate its intentions and internal state to the user. Moreover, how the robot communicates will also shape its perceived personality, persona, and social intelligence, such as when being polite or using humor. However, how can a social robot generate and use such behaviors?

A particular challenge for embodied agents, such as social robots and virtual agents, is consistency (Gratch et al., 2002). Humans quickly recognize when an embodied agent's verbal and non-verbal behaviors do not match. Studies demonstrate that such inconsistencies impact user experience negatively (Isbister and Nass, 2000). Another challenge is believability. For example, generating robot humor requires a finely tuned synchronization of several modalities to produce a believable and funny performance.

The literature has not yet extensively explored the expression of different robot personas, and most of the literature about robot politeness and humor relies on scripted robot behaviors. Thus, this thesis will investigate the real-time generation of multimodal and socially intelligent robot behaviors. It will

1. review the literature and related works to identify human behaviors that allow for consistent and believable replication. Based on these insights, the thesis will
2. develop models for expressing personality, persona, politeness, irony, and telling jokes with social robots. The models will
3. transfer the identified verbal and non-verbal human behaviors to the machine by augmenting and synchronizing speech with prosody, facial expression, and gaze.

The feasibility of the models will be demonstrated with the Reeti robot. Since many social robots can talk, use facial expressions, and gaze, the implementation of the models will also be transferrable to other robotic hardware. The models will provide the basis for investigating non-functional social robot behavior adaptation (see below).

1.2.2. A Conceptual Framework for Modeling Non-Functional Adaptation

When modeling user-adaptive interaction, several questions must be answered: what aspect of the interaction is adapted based on which information about the task or user, and much more. Unfortunately, there is no generalized overview or guideline for modeling real-time non-functional social robot behavior adaptation in the literature. Thus, a central contribution of this thesis will be a conceptual framework that provides the system designer with a holistic view of the overall problem. After providing the foundations of RL and user-adaptive interaction, the thesis will

1. review the literature about adaptive social robots with RL and human feedback,
2. develop a conceptual framework that covers every step in the process of designing, implementing, simulating, and evaluating models for socially-aware and real-time non-functional social robot behavior adaptation with RL, and

3. analyze the overall structure and general approach for social robot adaptation and identify connections between the theory and practice.

The framework will help system designers make design decisions based on step-by-step instructions and answers to common questions. The conceptual framework will

- describe the roles and responsibilities of the user, robot, and system designer in a user-adaptive interaction process,
- identify requirements and desirable properties of such an adaptation process,
- break down the procedure from start to finish, including problem modeling, simulation, and evaluation,
- provide tips and tricks on the design of the RL action space for maintaining consistency in the robot's behaviors,
- integrate the user in the RL framework, including their state and reactions,
- explain human explicit and implicit feedback (in specific, human social signals) in the RL loop, and
- point out algorithmic considerations and implications.

1.2.3. Models for Real-Time Behavior Adaptation with Human Feedback

The conceptual framework and models for multimodal behavior generation will culminate in several case studies of non-functional social robot behavior adaptation. The thesis will work out several models for user-adaptive robot communication. They allow for getting insights into their performance and impacts on user experience. The models will put the conceptual framework into practice and provide concrete examples for non-functional adaptation of the robot's expressed personality, persona, politeness, and humor. They will

- implement real-time non-functional social robot adaptation with
- increasing complexity, from explicit feedback to implicit feedback based on human social signals by
- optimizing single or multiple generation parameters and
- combining multiple communication modalities to a larger whole.

The models will be designed as generic as possible to implement independently of specific hardware and software. The benefit of the conceptual framework will be its guidance during the development of each model, covering the details of every aspect of the RL problem. Human feedback will be an essential distinguishing feature: explicit and implicit human feedback will be used in different ways to illustrate the range of use cases and applications of the conceptual framework.

The developed adaptation models will be implemented based on the Reeti robot hardware as proofs of concept and research prototypes, illustrating the technical implementation of non-functional social robot behavior adaptation to future developers. Afterward, the models will be simulated and evaluated (see below).

1.2.4. Simulation of User Reactions and Different Types of Noise

The conceptual framework will regard simulations as integral to developing user-adaptive interaction processes based on RL. Simulations will serve as a first, time-, effort-, and cost-saving step in evaluating non-functional adaptation models. Before evaluating the models with humans, this thesis will simulate the adaptation approaches, primarily for technical evaluation. Simulations will, i.a.,

- inspect whether the model converges,
- optimize learning parameters, and
- investigate the model's sensitivity to different types of noise, including biased human feedback and noise introduced by SSP.

While humans are part of user-adaptive interaction processes, simulations must work without real human interaction. Several questions arise regarding the implementation of simulations. How to replace the human with a simulated user, how to equip simulations with artificial preferences, and how to simulate human reactions during the adaptation process? The conceptual framework will answer these questions, while the developed simulations will illustrate the technical implementation.

In a subsequent human evaluation (see below) in the lab or domestic environment, different types of noise will bias human feedback – specifically but not exclusively when relying on human social signals as feedback. Thus, the impact of different types of noise on the learning agent's performance will be explored based on the simulations.

1.2.5. Identification of User Preferences and Impacts on User Experience

Finally, the developed and simulated models will be evaluated in the lab and in in-situ studies to verify the proposed non-functional adaptation approaches. The resulting in-situ and lab prototypes will include the implemented mechanisms for adaptation and behavior generation, SSP, automated adaptation measurements, and more. An autonomous evaluation platform will be developed and deployed in participants' homes. It will implement several applications for a companion robot, including health-related recommendations, entertainment, information retrieval, and communication.

During the human evaluation, the robot will adapt its behaviors to the individual users and identify their preferences. The thesis will report the resulting impacts of non-functional adaptation on the participants' user experience.

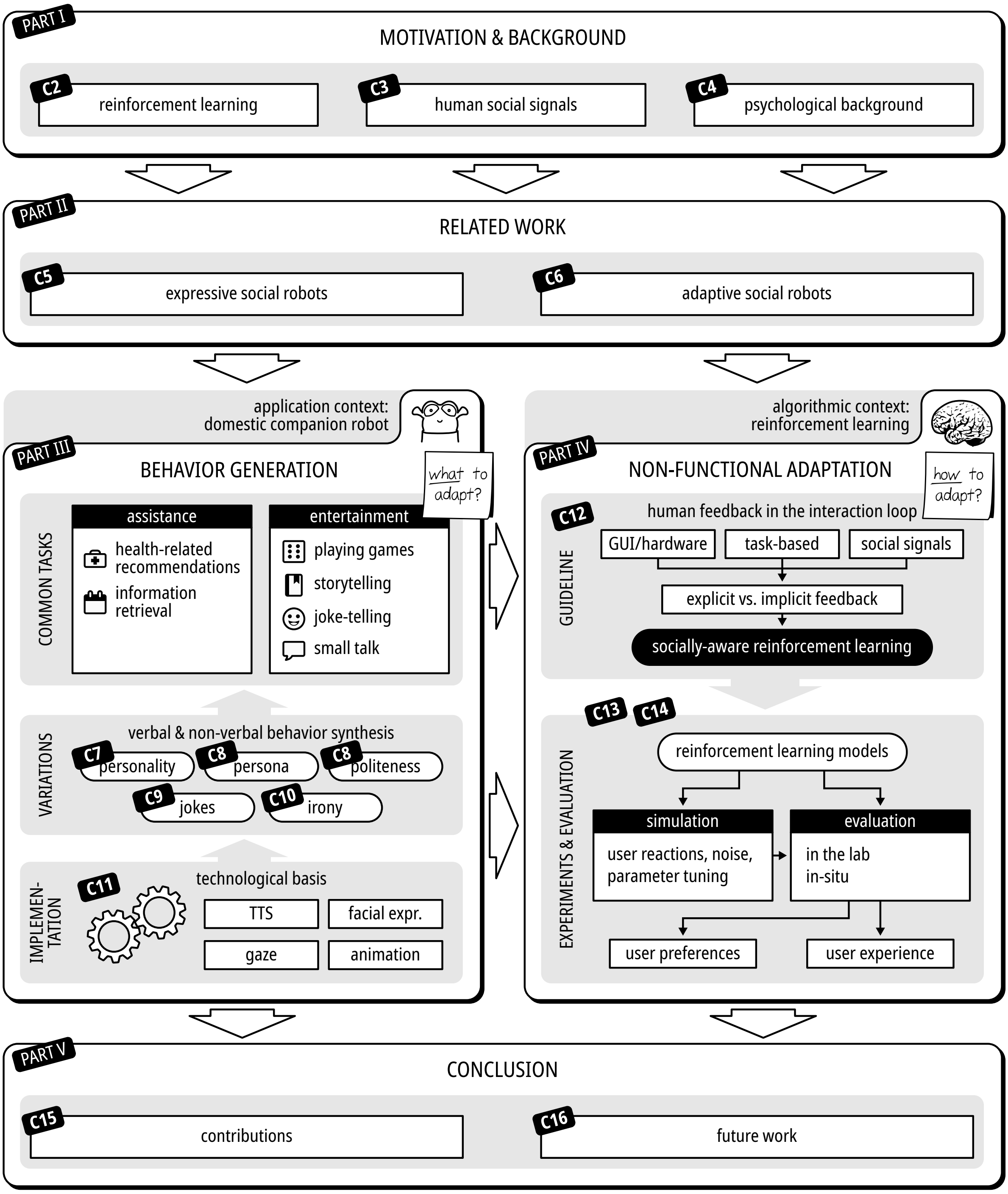


Figure 1.4.: Overview of the contents and structure of this thesis with corresponding chapter numbers.

1.3. Overview

Figure 1.4 gives an overview of this thesis. The rest of Part I deals with the theoretical background. Chapter 2 gives an introduction to RL. It covers an overview of definitions, problem modeling, properties, typical parameters, and algorithms. In addition, it presents common approaches for measuring and plotting performance. Chapter 3 is about human social signals, which serve as the basis for implementing multimodal robot behaviors and for incorporating human feedback for adaptation. An overview of non-verbal communication is given, including gestures, posture, body movement, facial expression, gaze, and paralinguistic behaviors. Chapter 4 deals with the psychological background. It covers personality, politeness theory, and different types of humor and their expression by humans.

Part II gives an overview of the literature. Chapter 5 introduces social robots, their abilities for sensing and producing verbal and non-verbal behaviors, and experiments in domestic environments. Related works cover the expression of personality, persona, politeness, and humor with different modalities. Chapter 6 covers the adaptation of social robots. It introduces user-adaptive interaction, outlining different types, criteria, and metrics. An overview of RL for social robot adaptation is given. The focus is on experiment designs, algorithms, feedback modalities, and the use of human social signals. The chapter terminates with limitations and research gaps in the presented literature.

Part III (*what* to adapt) presents novel concepts and techniques for expressing and generating multimodal robot behaviors in the context of a domestic companion robot. Chapter 7 focuses on expressing personality in the context of storytelling. Chapter 8 presents a domestic companion robot with assistive and entertainment applications. Afterward, it focuses on the expression of politeness and different personas in the context of health-related recommendations, board games, and information retrieval. Chapter 9 presents approaches for multimodal robot joke-telling. Chapter 10 presents an approach for generating robot irony in the context of small talk. Chapter 11 gives an overview of the developed technical basis for generating multimodal robot behaviors.

Part IV (*how* to adapt) presents different non-functional adaptation approaches. Chapter 12 details on algorithmic considerations, and several challenges resulting from the requirements of real-time HRI. It presents the conceptual framework for modeling, simulating, and evaluating the adaptation of multimodal social robot behaviors based on RL and human feedback. *Socially-aware* RL describes the concept of embedding human social signals in such adaptation processes. Chapter 13 focuses on explicit human feedback. It covers an experiment, simulation, and in-situ study results for adapting the robot's persona and politeness. Chapter 14 is about implicit human feedback. It presents experiments, simulations, and study results for the storytelling and joke-telling scenario.

Finally, Part V summarizes the contributions of this thesis in chapter 15 and open questions in chapter 16, which are subject for future work and research.

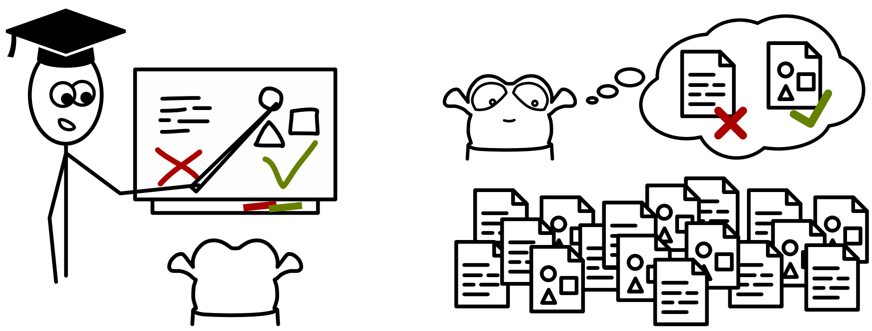
2. Reinforcement Learning

There are three categories of machine learning: supervised, unsupervised, and reinforcement learning. Figure 2.1 illustrates the first two approaches: *Supervised learning* provides instructive feedback from an expert; in contrast, *unsupervised learning* requires the system to extract structure or features from observed data on its own. In the context of robots and HRI, many tasks are so-called *control problems*, which require real-time decision-making. The robot must decide which action to take at a specific time in a particular situation. Moreover, robots typically are not constantly supervised by human experts, need to act autonomously, and thus need to decide how to behave on their own in real-time. For these types of problems, *reinforcement learning* (Sutton and Barto, 2018) has become increasingly popular. In RL, no expert is involved, and the learner neither receives labeled data nor is its goal to extract hidden structure.

This section gives an overview of the theoretical RL basics and algorithms used in this thesis. It focuses on model-free, value-based methods, which encode the agent's knowledge as floating point values. See chapter 12 for special requirements on applying RL in the context of HRI and corresponding experiments and studies in chapters 13 and 14 for adaptation of social robot behaviors with RL. The book by Sutton and Barto, 2018 serves as the basis for the content and formal notation.

2.1. Overview

RL is a class of machine learning algorithms for solving control problems. It is based on *trial and error*: the learning system, a so-called *agent*, iteratively investigates different *actions* in different situations, called *states*. The agent learns which action is best in which state based on the only scalar feedback it receives: the *reward* signal (see Figure 2.2). These properties make an RL agent completely autonomous. It does not



(a) Supervised learning. (b) Unsupervised learning.

Figure 2.1.: Supervised learning and unsupervised learning.

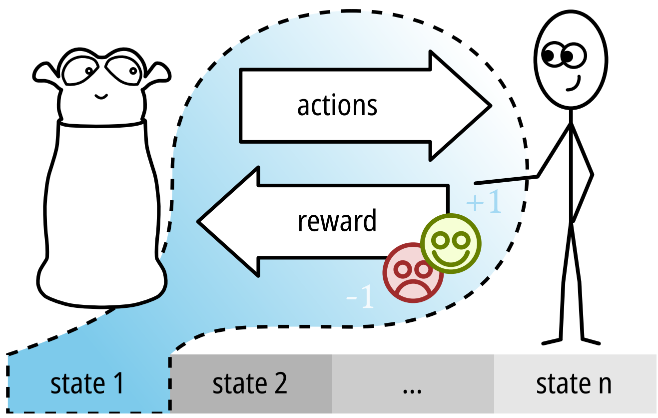


Figure 2.2.: An RL agent autonomously determines which action to take in different states based on a scalar reward signal.

need instructive feedback from (human) experts but figures out a solution based on its collected *experience* during the learning process. The experience includes information about observed state-action pairs linked to rewards. The system designer must specify the following information in order to make use of RL:

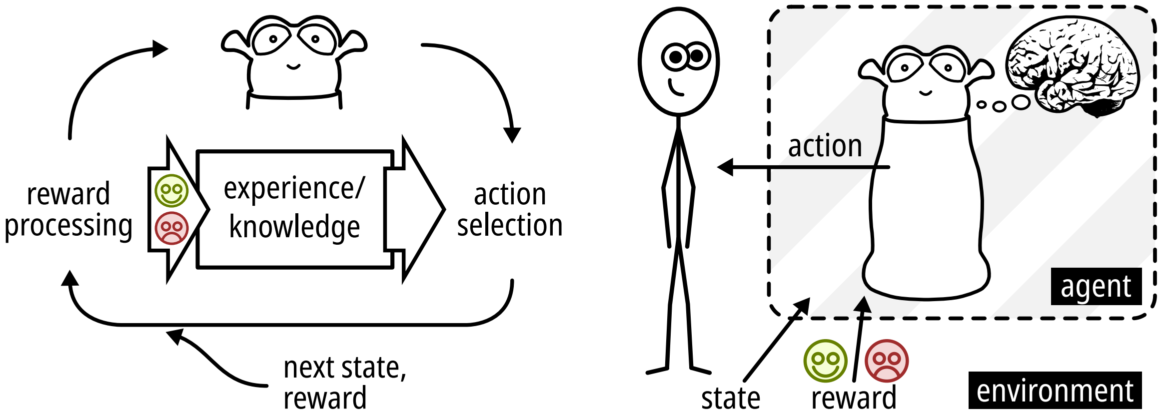
- the *action space* defines the tools for solving the problem,
- the *state space* defines relevant features and their manifestations for distinguishing different situations and
- the *reward function* encodes for each situation whether the agent does something expedient or not.

The agent does *not* know how to behave in which situation upfront, but this knowledge is the result of the learning process. It determines a solution based on trial and error and observing received rewards. The learning agent determines an optimal *policy* over time, which maps each state to an optimal action. Since RL works iteratively, the learning progress is split up into consecutive *time steps* $t = 0, 1, 2, \dots,$ which occur one after the other.

The learning process consists of an (in)finite loop. Each iteration/time step involves two basic tasks, as illustrated in Figure 2.3(a):

1. Select an action in the current state. After execution, observe the reward signal and next state.
2. Update the knowledge based on the last state, selected action, next state, and received reward.

With each learning step, the agent makes more observations about cause and effect in terms of executed actions in different states and received rewards. Its experience increases gradually, the policy gets refined, and the agent's behavior becomes more and more efficient.



(a) Learning loop. (b) Agent and environment.

Figure 2.3.: Tasks and boundaries of an RL agent.

Agent and Environment

The running system, which executes the learning loop, is called the *agent*. Its competence is the selection and execution of actions and the calculation and update of the internal knowledge (see Figure 2.3(b)). This process works differently depending on the implemented RL algorithm, but the overall idea is always the same as described above.

The learning agent does not calculate the current state, next state, and reward signal by itself. This data comes from the *environment* beyond the agent's system boundary. The agent does not influence the received reward or state transitions. However, the agent needs some internal representation of states, actions, and rewards. It associates observed data from the environment with actions to learn an optimal policy. The agent's system boundary is blurry when implementing simulations that also need to simulate the environmental response. The general idea of splitting agent and environment is to swap the learning part with another implementation (e.g., another algorithm) while the environment's logic remains unchanged.

2.2. Problem Modeling

RL is all about solving control problems. Given the problem, the designer of the RL agent needs to think about (1) which skills are necessary to reach the goal, (2) which factors or parameters of the environment will change over time, and (3) how to measure whether the agent reaches the goal. These preliminary considerations are some of the most important ones in the whole process of RL. The agent will be unable to find a solution if the system designer makes errors. For example, including irrelevant data makes the agent potentially less efficient, and the agent will find a solution more slowly. Thus, analyzing the problem in detail before running experiments is essential.

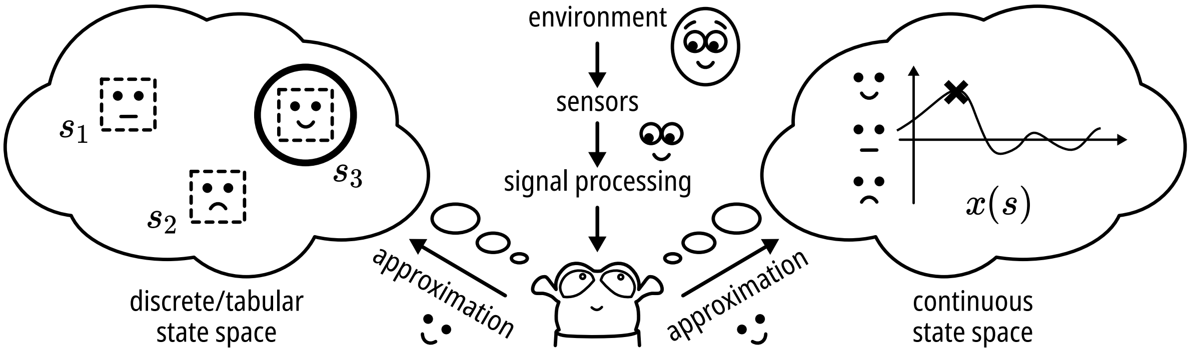


Figure 2.4.: States can be represented discretely or continuously.

2.2.1. Action Space

Interaction with and manipulation of the environment is only possible with the help of the agent's actions. Actions can vary in different states. For example, it might be helpful to reduce the set of actions for states where the execution of specific actions does not make sense a priori. In this thesis, discrete actions are used, which results in a finite set of actions:

$$\mathcal{A}(s) = \{a_1, a_2, a_3, \dots\}$$

In each time step t, an action $A_t \in \mathcal{A}(S_t)$ is selected and executed. The agent cannot directly influence the outcome of the action, but the environment determines the logic of what happens when the agent executes the action in a specific situation. The outcome can be deterministic or nondeterministic (see section 2.4.2).

2.2.2. State Space

The state space allows the agent to distinguish different situations. States are collections of *features*, which serve (1) as an abstracted representation of the environment, as well as (2) the only opportunity to encode knowledge about the task progress. The latter is the result of the *Markov property*: the current state must encode all past data (i.e., the history of past observations relevant to the agent). The set of states $\mathcal{S}$ is defined as follows:

$$\mathcal{S} = \{s_1, s_2, s_3, \dots\}$$

In each time step t, the state $S_t \in \mathcal{S}$ is active, followed by the next state S_{t+1} after action execution. It is important to note that states are interdependent: the execution of an action a_1 in s_1 may lead to the next state s_2; however, the execution of another action a_2 in s_1 could lead to another state s_3. The agent's subsequent observations highly depend on its previous actions while traversing the state space. Metaphorically speaking, the agent walks along different paths through the state space during learning, where each learning step starts in the current state S_t and leads to the next state S_{t+1}. Future learning steps always depend on past states and actions. The power of a RL agent lies in optimizing these paths and selecting actions that lead towards the goal by the shortest route.

Figure 2.4 illustrates two options for encoding states. The set of states can either be discrete or continuous: a discrete state space is defined by a finite set of n states

$\mathcal{S} = s_1, s_2, \ldots, s_n$ ($n = 3$ in Figure 2.4). In contrast, continuous state spaces are infinite and the current state s_t must be described by functions, which take the raw data (e.g., floating point features) of state s as input ($x(s)$ in Figure 2.4). This approach is called *function approximation*. One can also use function approximation for describing finite sets of states, which has the benefit of better generalization (see also section 2.8).

In the case of a discrete set of states, the agent must make abstractions when dealing with complex values. For example, a continuous floating point feature in the range $[0; 1]$ must be discretized. It is necessary to split up the infinite number of possible values in a finite number of intervals, for example, $[0; 0.5]$ and $[0.5; 1.0]$, which reduces its precision significantly. Figure 2.4 presents another example, where the human's smile intensity is the single feature of the state space. Discretization would split this feature into three different manifestations, resulting in a state space with three states s_1, s_2 and s_3. A loss of accuracy is the result since the learning agent can only distinguish a limited amount of manifestations for each feature.

Determining the relevant environment's and task's features for inclusion in the state space is the system designer's task – as is the identification of necessary actions. States are the only option for the agent to distinguish different situations and recognize similar situations. Like the set of actions, irrelevant data may decrease performance, and forgetting relevant data might cause a complete learning failure. The so-called *curse of dimensionality* is a problem, especially for discrete state spaces. The state space exponentially grows when adding new features, which is undesirable due to limited memory and computing time.

2.2.3. Reward, Values, and Policies

After selecting an action A_t in the current state S_t, the agent receives the reward. This numeric value $R_{t+1} \in \mathbb{R}$ serves as an indication of how good or bad this action performs in S_t. Due to the lack of expert feedback, this is the only input available for calculating an optimal *policy* over time (see below). However, rewards do not always serve as an immediate indication of success or failure because rewards may be delayed. The agent's current action selection can influence future events. For example, it might receive a high positive or negative reward only later, caused by the sequence of preceding actions.

For value-based RL approaches, the observed rewards are summed up over time intelligently for calculating weighted averages. These averages estimate how good actions perform while taking into consideration what will happen in the time steps after the current action execution. The average value is called $Q(s, a)$ and is estimated for each state-action pair (s, a) with $s \in \mathcal{S}$ and $a \in \mathcal{A}$. Q can be implemented as a 2D lookup table or hash map, which associates one floating point value with each pair (s, a) if there is a finite number of states. The calculation differs from algorithm to algorithm.

At the end of the iterative calculation process, the Q values converge towards the *real values* q_*. The real values are unknown to the agent; solving the control problem is exactly the calculation of these values. In simulations, the simulation environment/program logic may know all $q_*(s, a)$ (e.g., it might assign random numbers). Some performance measurements presented in section 2.7 rely on them.

In general, the term *reward* does not necessarily mean a positive signal for the agent. R

can be of any value, positive or negative, large or small. It is also possible to give negative rewards exclusively. What matters is the difference between rewards and values for different actions, not necessarily the actual value itself.

A *policy* can be inferred from the Q values at each time step t. It maps each state to its greedy action and thus tells the agent which action is the best to take based on the agent's past observations. This data is not stored separately since it automatically results from the current Q values by calculating $\text{argmax}_a Q_t(s, a)$. With each learning step and increasing experience, the policy becomes better when the Q values converge to the real values q_*. There is at least one *optimal policy*, which is the best policy. Multiple optimal policies might assign more than one optimal action to each state. In this case, it does not matter which one of the optimal actions to take. The optimal policy changes over time if the problem is nonstationary (see section 2.4.1): as soon as the actions' real values q_* change, the optimal policy changes, too.

2.3. Exploitation vs. Exploration

An RL agent solves a control problem, meaning it needs to make decisions during runtime. In each learning/time step t, the agent selects an action and executes it. The decision of which action to take is of central importance since it directly impacts the agent's performance. Action selection requires the agent to balance *exploitation* and *exploration*. Both aspects are crucial for optimal performance, but in detail, it entirely depends on the problem (see section 2.4.1).

Exploitation means that the agent maximizes the expected reward by picking the action with the maximum estimated value. Exploitation is always based on the agent's observations up to time t since the estimated values Q are calculated iteratively. In the case of exploitation, an agent's action selection is *greedy*, and those actions with the maximum value are called greedy actions.

Exploration is the counterpart to exploitation. It aims to explore nongreedy actions, which appear to be suboptimal or might not have been used so far. Selecting suboptimal actions every once in a while is done by intention and is essential for optimal performance. First, the real values q_* might change over time in nonstationary tasks (see section 2.4.1), which means that optimal actions change over time. In addition, rewards can be noisy, so the current estimated values Q might be biased. Thus, the next best solution based on the current experience does not need to be the overall best solution. In most cases, exploration is the only opportunity to become aware of and react to such changes.

2.3.1. greedy

A greedy agent completely ignores exploration. It focuses on maximizing collected rewards exclusively and thus always picks the action with the maximum estimated value $Q_t(s, a)$ in state s at time step t according to the current policy:

$$A_t = \underset{a}{\text{argmax}}\, Q_t(s, a)$$

There are a few exceptions for which a greedy approach is sufficient (see section 2.4.1), but generally, it is not suitable for learning. Typically, greedy action selection results in overall suboptimal behavior because the agent does not care about alternatives due to the lack of exploration and experience.

2.3.2. ϵ-greedy

In order to integrate and balance both exploitation and exploration, ϵ-greedy is a hybrid approach based on the probability $0 \leq \epsilon \leq 1$. In each time step t, the agent draws a random number x. A random action gets selected if the number is smaller than the probability ϵ. Otherwise, the agent uses the greedy approach:

$$A_t = \begin{cases} \text{a random action}, & \text{for } x < \epsilon \\ \operatorname{argmax}_a Q_t(s, a), & \text{for } x \geq \epsilon \end{cases}$$

One can control the amount of exploration by setting ϵ to a lower or higher value. Typically, ϵ is small. One can also adjust epsilon during runtime, starting with a higher value and decreasing it throughout an episode (see section 2.4.1). Often, random actions are selected according to a uniform distribution in the case of exploration. Thus, the agent does not take its experience or uncertainty about specific actions into account.

For $\epsilon = 0$, the action selection is identical to greedy, and the agent performs exploitation all the time. For $\epsilon = 1$, the agent performs exploration exclusively due to random action selection in each learning step.

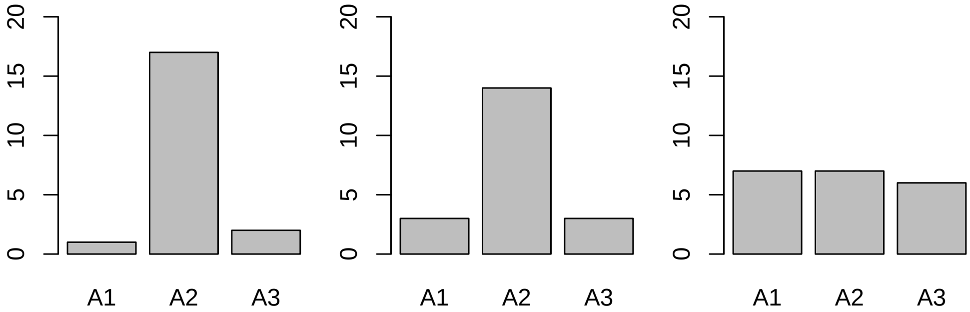
2.3.3. UCB

Upper-confidence bound (UCB) action selection is a more intelligent approach than ϵ-greedy for stationary problems (see section 2.4.1).[1] It considers the uncertainty of all actions by encouraging exploration of those actions, which have been used less often. The agent stores a counter N for each state-action pair (s, a) and increments the counter each time action a is selected in state s. This counter measures uncertainty about the action's estimated value: the rarer it is executed compared to other actions, the higher the uncertainty. While the general idea is a greedy approach, which again works by finding the maximum action, the counter N is part of the calculation. For each action, the square root of the natural logarithm of the current time step t, divided by the number of action executions $N_t(s, a)$, is added to the current estimated value $Q_t(s, a)$. By doing so, each action gets a benefit in proportion to its uncertainty:

$$A_t = \operatorname*{argmax}_a \left[Q_t(s, a) + c\sqrt{\frac{\ln t}{N_t(s, a)}} \right]$$

The constant c controls the influence of uncertainty. Figure 2.5 plots the amount of action selections using UCB for an agent with three actions after 20 time steps. The

[1] Garivier and Moulines (2008) present *discounted* and *sliding-window* UCB for nonstationary environments.



(a) $c = 0.1$ (b) $c = 1$ (c) $c = 10$

Figure 2.5.: UCB's parameter c: smaller values result in more exploitation (selection of the greedy action A2), greater values encourage exploration (action selection is more balanced/evenly distributed).

smaller the value c, the more often the greedy action is selected; the greater the value, the more uniform the resulting action distribution.

An example: for $c = 0.1$ (see Figure 2.5(a)), the greedy action A2 is used almost all the time. With increasing c (see Figures 2.5(b) and 2.5(c)), nongreedy actions are used more frequently. Thus, c controls the balance between exploitation and exploration: a greater c encourages exploration and reduces uncertainty, but it also reduces performance at the same time since the greedy action is selected rarer. For $c = 0$, this results in exclusively greedy behavior, just like when using $\epsilon = 0$ with ϵ-greedy.

One must be aware that c must be chosen depending on the magnitude of the real values q_*. A small c is sufficient if the real values are in range $[-1; 1]$, as is the case in Figure 2.5. However, if the real values are in the range $[-1000; 1000]$, the effect of UCB will not occur until c is big enough. Otherwise, the value of the added root is too small and thus negligible compared to the estimated Q values.

2.4. Problem Properties

Several properties classify different RL problems. Analyzing the problem and identifying its characteristics is essential for selecting and configuring suitable learning algorithms. Similar to problem modeling, this decides on the agent's learning success.

2.4.1. Stationary vs. Nonstationary

One problem property is stationarity, which relates directly to the real values q_*, which are unknown to the agent. In the case of a *stationary* problem, the q_* are fixed and never change during the experiment. The contrary is true for *nonstationary* problems, where the q_* can change at any time *during* the learning process. It means that estimated Q values become invalid as soon as the corresponding q_* change since the new q_* values can be entirely off the previous ones.

Figure 2.6 illustrates an example. The dashed lines represent the real values q_*, and the solid lines represent the estimated values Q over time. In the upper plot (stationary

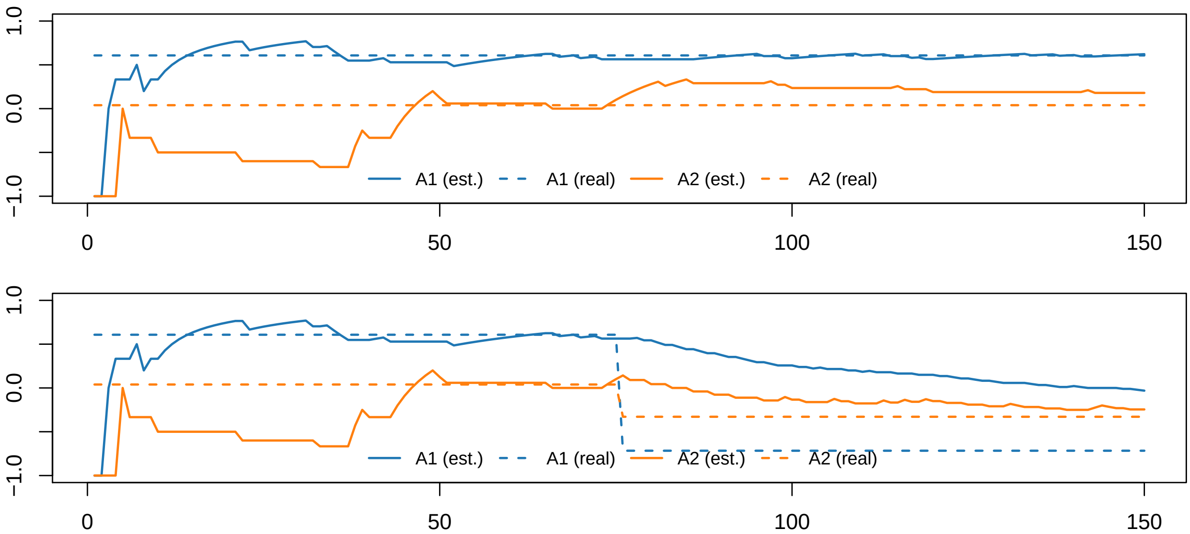


Figure 2.6.: Top: real values do not change (stationary problem). Bottom: real values change at step 75 (nonstationary problem), and the agent cannot adapt accordingly. Both use noisy rewards.

problem), the q_* values are constant from the beginning until the end of the experiment. Due to the stationarity, the agent can approximate the real values by observing rewards, thus calculating increasingly more accurate estimates over time. The reduced distance between the dashed and solid lines demonstrates the convergence towards the real values q_*. Due to noisy rewards, temporary divergence is possible, of course. With action A2 being the nongreedy action, its estimates are worse since it is selected less often. In general, fewer data typically results in less accurate estimates in a nondeterministic environment (see section 2.4.2).

In contrast, the lower plot results from a nonstationary problem. The q_* values change in the middle of the experiment at time step 75. Action A1, the greedy action up to this time, becomes the nongreedy action since A2 has the maximum value from that moment. For the implementation, a simple bandit algorithm (see section 2.6.1) calculates Q values based on the average of all collected rewards for each action. As illustrated by the plot, this is a severe problem since the agent cannot learn the new greedy action until the end of the experiment. The learned values are entirely off the q_* values. Moreover, for the example at hand, the identified greedy action is wrong as soon as the q_* change. The learning process is not able to correct the estimates accordingly. Thus, (non)stationarity is of central importance for selecting an appropriate learning algorithm and balancing exploration and exploitation. See section section 2.5 for more detailed implications.

2.4.2. Deterministic vs. Nondeterministic

An essential property concerning the environment is determinism or the lack of it. The environmental response in state s to the agent's selected action a consists of a state-reward pair (s', r) with $s' \in \mathcal{S}$ and $r \in \mathbb{R}$. A deterministic environment always responds with the same pair (s', r) to a given state-action tuple (s, a). Both the reward and next state are deterministic then.

In the case of a nondeterministic environment, the environmental response can vary. Both the reward and next state may differ; thus, given the current state-action tuple (s, a), different responses (s', r) can occur throughout an experiment. This property is essential to consider during the design of the RL agent since the chosen RL algorithm must be able to cope with nondeterministic environments.

2.4.3. Episodic vs. Continuing

RL problems are categorized in two distinct types: *episodic* and *continuing* tasks. In episodic tasks, the state space includes so-called *terminal states*, which stop the experiment immediately as soon as the agent enters them. Terminal states occur when the task comes to a natural end because the agent has reached its goal and solved the problem. One example of these problems is a navigation task to a specific destination point. Once the agent reaches the goal, there are no actions for continuing the task after that. Conversely, the agent may also fail halfway through the experiment (e.g., by falling into a trap), and, as a result, finishing the task might be impossible. Thus, when reaching a terminal state, the task terminates immediately and starts over. One pass from the initial until a terminal state is called an *episode*. Episodic tasks thus always come to a natural end sooner or later and then start from one of the initial states again – but the agent keeps the experience and knowledge it has acquired so far so that it can perform better and better in the next episodes.

Continuing tasks do not come to a natural end and do not have terminal states. The agent keeps acting forever while improving its behavior. In practice, continuing tasks can be stopped after some time to extract an optimal/greedy policy.

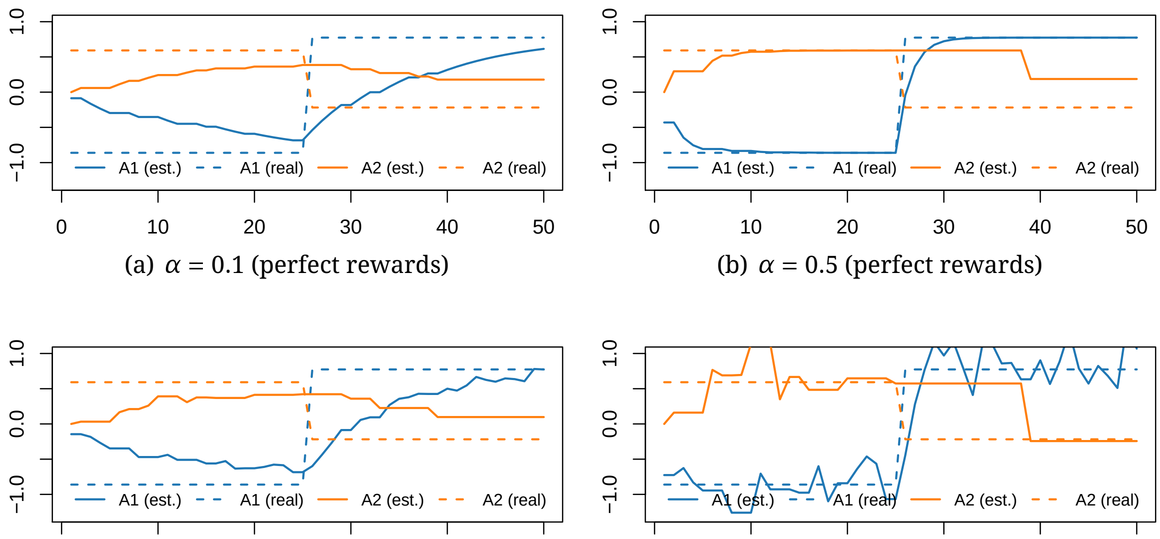
2.5. Common Parameters

RL algorithms typically share a set of common parameters that control the agent's behavior and learning progress. They include the learning rate α, exploration rate ϵ, and discount factor γ. This section gives an overview of these parameters and how different values influence the learning process.

2.5.1. Learning Rate

The learning rate $\alpha \in [0; 1]$ controls how fast an agent learns. As described in section 2.6, each value-based RL algorithm iteratively calculates new value estimates based on the received reward. In the case of an RL event, the current value estimates are updated based on the new experience. The learning rate α determines the fraction of change, i.e., how much of the new experience is applied to the current estimates.

Figure 2.7 illustrates different configurations for a deterministic and nondeterministic environment for a nonstationary problem (real values change at time step 25). In a deterministic environment with perfect rewards and no noise, the difference between a lower value $\alpha = 0.1$ (Figure 2.7(a)) and a higher value $\alpha = 0.5$ (Figure 2.7(b)) is obvious: a higher value increases learning speed and the estimated values (solid lines) converge more quickly towards the real values (dashed lines). However, problems occur when



(a) $\alpha = 0.1$ (perfect rewards)

(b) $\alpha = 0.5$ (perfect rewards)

(c) $\alpha = 0.1$ (noisy rewards)

(d) $\alpha = 0.5$ (noisy rewards)

Figure 2.7.: Top: a greater learning rate makes learning faster. Bottom: However, a greater learning rate is also more prone to noisy rewards.

introducing noise to the system. As an example, Figures 2.7(c) and 2.7(d) compare the same learning rates in a nondeterministic environment. The rewards are not perfect but randomized according to a normal distribution. Thus, the learning rate has a significant impact on the estimated values. The external noise also affects the estimated values when increasing the learning rate: the more noise in the reward, the more noise in the estimated values.

It is not desirable to set α to a large number because many RL environments are nondeterministic or nonstationary. Otherwise, divergence and very unstable learning might be the result. Typically, a small learning rate is used for more stable learning. One can decrease the learning rate over time for stationary tasks. However, this is impossible for nonstationary tasks. The learning rate must be constant and greater than zero for nonstationary tasks. Otherwise, changes to the q_* values cannot be correctly estimated by the learning agent when α is too small.

2.5.2. Exploration Rate

As described in section 2.3, exploration is typically randomized to a certain degree to address the exploration-exploitation dilemma. In the case of ϵ-greedy and related action selection approaches, the parameter $\epsilon \in [0; 1]$ controls the amount of exploration. One can decrease the exploration rate over time for deterministic and stationary problems. However, minimal exploration is crucial for nonstationary or nondeterministic problems.

In the case of nondeterministic problems, the agent might find a suboptimal solution when exploration stops too early because there might be better actions that the agent has not yet identified. A similar problem occurs with nonstationary problems: when exploration stops completely, the learned policy might be suboptimal as soon as the real values q_* change because the agent will never try alternative actions. Thus, a common practice is either to keep a constant, small exploration rate or to start with a greater ϵ to encourage more exploration at the beginning and to reduce it over time.

2.5.3. Discount Factor

The new value of the current state-action pair (s, a) is calculated based on the estimated value of the following state s'. This approach is called *bootstrapping*. The discount factor $\gamma \in [0; 1]$ controls how much the estimated value of the next state s' contributes to the whole calculation. Typically, γ is set to a value near 1, e.g., 0.9. Smaller discount factors focus the agent primarily on immediate rewards, resulting in a more short-sighted behavior. The theory of RL requires $\gamma < 1$ to guarantee convergence.

2.6. Basic Algorithms

2.6.1. k-armed Bandit Problems

In stateless environments, where the agent does not need to distinguish different situations, multi-armed bandit problems (Sutton and Barto, 2018) are a reduced form of RL. Since there is no notion of state the agent explores k different actions from the set of actions $\mathcal{A}$ with $|\mathcal{A}| = k$. Its goal is to estimate each of the actions' real, unknown values q_*. In each time step t, the agent selects an action $A_t \in \mathcal{A}$ (see section 2.3) from the set of actions $\mathcal{A}$, executes it and observes the received scalar reward R_{t+1} to update the estimated action value $Q_t(A_t)$.

The Q value is updated based on the received reward after executing an action according to one of the strategies presented in section 2.3. In a stationary task, the agent calculates the average of the received rewards over time. For each action, a counter N is incremented each time the action is selected. In each time step t, the following algorithm computes the average incrementally:

$$Q(A_t) \leftarrow Q(A_t) + \frac{1}{N(A_t)} \left[R_{t+1} - Q(A_t)\right]$$

While averaging works for stationary tasks, nonstationary tasks require constant adaptation. The following algorithm uses a fixed learning rate α to update the Q value:

$$Q(A_t) \leftarrow Q(A_t) + \alpha \left[R_{t+1} - Q(A_t)\right]$$

2.6.2. Associative Search

Multi-armed bandits do not distinguish between different states. However, when there are different but *independent* problems, one can use a set of different multi-armed bandits (one for each problem) to solve each problem on its own. However, this requires that the problem be identifiable because the agent must choose the appropriate bandit for the current problem. In combination, all agents form an *associative search*: the corresponding agent is chosen and used for action selection and value estimation depending on the problem.

The difference to regular RL problems is that, usually, *states* are interdependent. They describe and belong to the same problem. Associative search only explores different sets of actions in different contexts, which do not influence each other: all bandits describe and solve different – stateless – problems. This approach is used in chapter 13.

2.6.3. Q-learning

A well-known, tabular approach for state-based RL is the Q-learning algorithm used in chapter 14. As with k-armed bandit problems, the agent selects an action $A_t \in \mathcal{A}(s)$ in each time step t according to section 2.3. Each state can have different actions, depending on the task. Thus, the set of actions $\mathcal{A}$ depends on the current state s. In contrast to multi-armed bandit problems, Q values need to be saved for all state-action pairs (s, a) with $s \in \mathcal{S}$ and $a \in \mathcal{A}(s)$. After selecting and executing the action A_t the agent observes the environmental response (S_{t+1}, R_{t+1}), consisting of the next state S_{t+1} and the scalar reward R_{t+1}. The estimated value $Q(S_t, A_t)$ updates according to the following algorithm:

$$Q(S_t, A_t) \leftarrow Q(S_t, A_t) + \alpha \left[R_{t+1} + \gamma \max_a Q(S_{t+1}, a) - Q(S_t, A_t) \right]$$

The update formula includes the current value $Q(S_t, A_t)$, the received reward R_{t+1}, the discount factor γ (see section 2.5.3) and the maximum Q value $\max_a Q(S_{t+1}, a)$ of the *next* state S_{t+1}. The maximum value is identified by iterating over all actions $a \in \mathcal{A}(S_{t+1})$. It is important to note that the action a, which maximizes this term, does not need to be the same action executed in the next time step $t + 1$. The agent assumes that it acts optimally in each time step. While this is not necessarily the case, e.g., because of exploration (random action selection), this is a fundamental property of the Q-learning algorithm. Such behavior is known as *off-policy* learning because the executed action is not necessarily identical to the one used for learning.

As already observed in the update formula in section 2.6.1, the learning rate α controls to which degree the observed *TD-error* $R_{t+1} + \gamma \max_a Q(S_{t+1}, a) - Q(S_t, A_t)$ is taken into account. The TD-error measures the difference between the observed reward, discounted future return $\gamma \max_a Q(S_{t+1}, a)$ and the current estimate $Q(S_t, A_t)$. By iteratively accumulating TD-errors, the estimated value $Q(S_t, A_t)$ moves a fraction toward the new observations.

The programmer can use a two-dimensional array or a hash map to represent the Q values. The former typically reserves memory for every possible state-action combination upfront; the latter saves memory temporarily by adding elements to the hash map only when they occur during an interaction at the expense of additional processing power for the hash map lookup. Dependent on the problem, some state-action pairs potentially never occur.

2.7. Measuring Performance

There are several approaches for measuring an agent's learning performance. Some of these measurements only apply in simulated environments because they require knowledge of the actual values q_*, which are typically unknown in real environments. However, there are also ways to measure performance, which work in simulated and real interactions, e.g., when relying on rewards.

It is important to note that the problem at hand restricts the set of measurable data. Some measurements require episodic tasks (see section 2.4.3), e.g., when accumulating or averaging data of a whole episode. Such measurements cannot be applied to continuing tasks because they never come to an end. Thus, a common approach for continuing tasks is to stop the experiment after several steps and then re-run it several times. Then, measurements are done by averaging multiple runs of potentially distinct agents. In addition, some measures work for both episodic and continuing tasks.

The learning progress often becomes visible only when averaging over multiple episodes or runs of an experiment. Exploration (see section 2.3) causes suboptimal action selection (and thus, suboptimal performance) when looking at one single episode – in particular, but not exclusively, at the beginning of episodes. Thus, plots typically average over multiple episodes or multiple runs of an experiment.

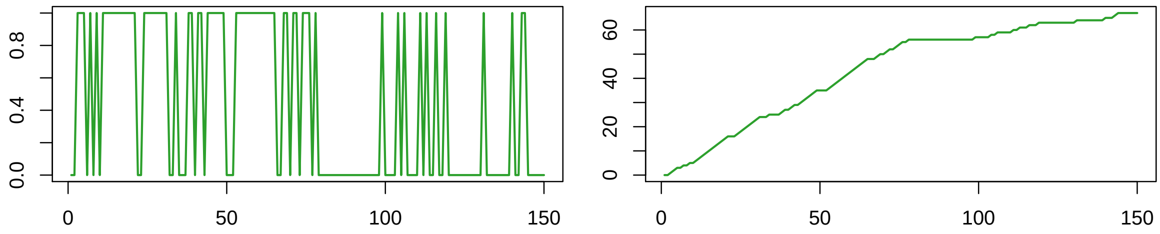
However, visualizing a single episode can be of interest, too. As an example, Figure 2.7 plots the estimated values of a two-armed bandit problem for one simulation run, depending on the learning rate α. It illustrates a continuing task that stops after 50 steps. Figures 2.7(c) and 2.7(d) visualize the impact of noisy rewards in combination with a lower or higher α value, which results in the choppy lines. When averaging over multiple runs, the line plot would be much smoother and less descriptive concerning the agent's actual behavior in one run. In contrast, plots of single episodes typically have no informative value concerning the agent's overall performance. In general, the agent's overall or averaged performance is typically more of interest than individual episodes or time steps since exploration and the resulting random behavior is an inherent part of RL. However, especially when the human is involved in the learning process, such as in the context of HRI, this single episode performance is relevant, too (see chapter 12).

The following measures examine various aspects of performance by focusing on different data. Based on this information, it is possible as a next step to optimize the learning algorithm or its parameters concerning one or more of the measures in order to achieve a problem-specific performance compromise.

2.7.1. Reward

The reward is a standard measure for getting insights into the agent's performance. In general, the received rewards should become greater over time since the central idea of RL is to maximize the expected reward in terms of exploitation (see section 2.3). During the runtime of the experiment, the agent logs received rewards for each time step. Afterward, this data is typically visualized as a line plot, including, but not exclusively, the following approaches:

Actual reward Each point represents the actual reward value R_{t+1} for time step t of one



(a) Actual reward. (b) Cumulative reward.

Figure 2.8.: Reward and cumulative reward for the nonstationary problem from Figure 2.6.

single episode. See Figure 2.8(a) for an example.

Cumulative reward Each point represents the sum of all rewards $\sum_{i=0}^{t+1} R_i$ up to the current time step t of one single episode. In contrast to the actual reward, this has the benefit of visualizing a general tendency whether rewards increase or decrease over time (i.e., whether/how fast the agent's performance increases or decreases during the episode). See Figure 2.8(b) for an example.

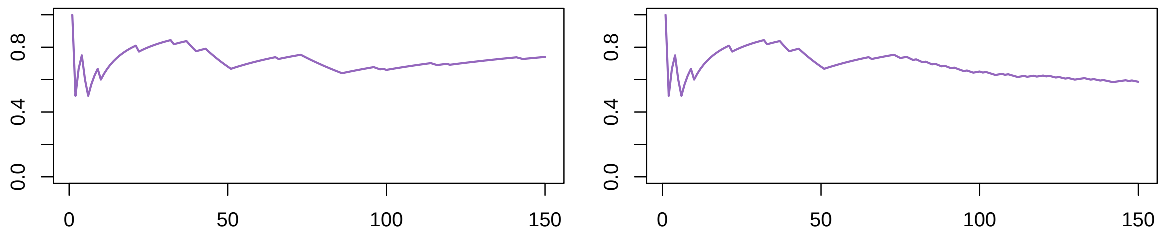
Average reward Each point represents an averaged reward $\overline{\sum_{e=0}^{E} R_{e,t+1}}$ for time step t over multiple episodes E of equal duration. Alternatively, this can also be used for multiple runs E of a continuing problem, which stops after the same number of time steps. See Figure 13.4 for an example.

(Average) Reward per episode Each point represents one episode e. This requires to aggregate all rewards of the specific episode, for example by calculating the sum $\sum_{t=0}^{T-1} R_{e,t+1}$ or average $\overline{\sum_{t=0}^{T-1} R_{e,t+1}}$. Plotting the aggregated values gives insights into the agent's learning progress in the long run (over multiple episodes).

Figures 2.8(a) and 2.8(b) illustrate the actual and cumulative reward for the nonstationary problem from Figure 2.6. Due to the nonstationarity, the actual values q_* change at step 75. In the first half, the agent primarily receives positive rewards. The plot illustrates the actual reward and the steady increase of the cumulative reward. After step 75, positive rewards are rare due to the learning agent's suboptimal action selection. The plot illustrates the flattening of the cumulative reward curve. While this example only uses positive and neutral rewards, negative rewards could also cause a decrease in the cumulative reward. The cumulative reward illustrates the agent's performance loss at time step 75 clearly.

2.7.2. Percentage of Optimal Actions

Another measure of performance is the percentage of optimal actions. It reports how often the agent selects the optimal action (regarding q_*, *not* regarding the agent's estimated values Q). This approach works only in simulated environments because the real values q_* must be known to determine the optimal action. A counter is incremented each time the optimal action is selected. It does not matter whether the reason for selecting the



(a) Stationary problem. (b) Nonstationary problem.

Figure 2.9.: Percentage of optimal actions for the stationary and nonstationary problem from Figure 2.6.

action is exploration or exploitation (see section 2.3). The percentage is calculated for each time step t based on the number of optimal actions selected up to time step t and the total number of actions executed up to time step t.

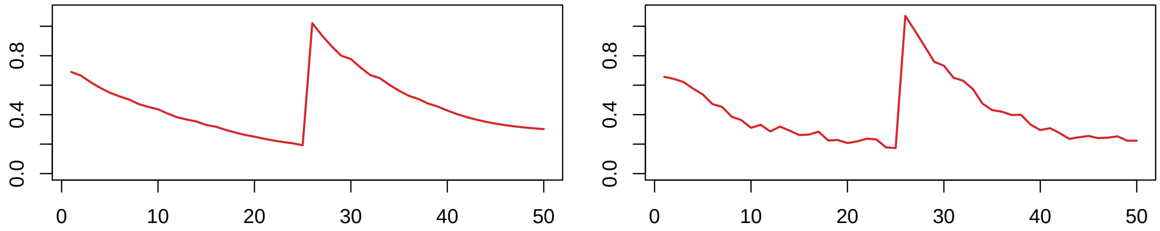
Similar to plotting rewards, one can plot the percentage of optimal actions for a single episode (per time step, see Figure 2.9), as an average over multiple episodes (per time step, see Figure 13.4), or per episode. Typically, the averaged percentage increases over time as the agent learns and performs better and acts more and more greedy (see Figure 13.4). However, it typically never reaches 100 percent due to the necessity of exploration.

Figure 2.9 illustrates the percentage of optimal actions for the stationary and nonstationary problem from Figure 2.6. In both cases, the agent selects the optimal action less than 80 percent of the time since exploration forces the agent to select suboptimal actions. In comparison, the percentage of optimal actions is higher for the stationary problem. Due to the nonstationary problem in Figure 2.9(b), the agent does not identify the new optimal action from time step 75 up to the end of the simulation. As a result, it uses the optimal action only in the case of exploration in the second half (which happens relatively often in this experiment since there are only two actions). Consequently, the percentage of optimal actions decreases in the second half of the experiment.

2.7.3. Root Mean-Squared Error

The root mean-squared error (RMSE) measures the error between all the actions' real values and their current estimations to a specific time t. In simulations, this measure focuses on the agent's actual knowledge and to which degree it differs from the real world. While other measures focus on the agent's behavior, such as the percentage of selected optimal actions or data from beyond the agent's boundaries (average reward), the RMSE is directly connected to the agent's internal representation of actions. While q_* is typically unknown in real-world experiments, the RMSE gives insights into the estimated values' accuracy in simulated environments. Since the RMSE calculates an error, its value should become as small as possible over time. It is calculated for a single episode (per time step), as an average over multiple episodes (per time step), or per episode. One can also create cumulative plots.

First, the sum of the squared errors of all actions $a \in \mathcal{A}$ is calculated. Then, the mean of the squared errors is calculated by dividing by the number of actions. Finally, the root



(a) $\alpha = 0.1$ (perfect rewards)

(b) $\alpha = 0.1$ (noisy rewards)

Figure 2.10.: RMSE for the agents in Figures 2.7(a) and 2.7(c).

is calculated in order to make it return the original units of the comparison:

$$\text{RMSE}_t = \sqrt{\frac{1}{|\mathcal{A}|}\sum_{a\in\mathcal{A}}[q_*(a) - Q_t(a)]^2}$$

Figures 2.10(a) and 2.10(b) plot the RMSE for the two agents illustrated in Figures 2.7(a) and 2.7(c). The RMSE demonstrates the learning progress and effect of noisy rewards. In Figure 2.10(a), the error decreases until step 24 as the two actions' estimated values become more accurate and approach the real values. In step 25, the real values change. Therefore, the RMSE increases abruptly and decreases afterward. Since the agent gets perfect rewards, which are not noisy at all, the curve is very smooth. The effect of noisy rewards is visible in Figure 2.10(b). Since a Gaussian distribution randomizes the rewards, the shape of the line is not smooth but distorted. Each time the reward deviates from the action's real value, this causes an increase in the overall RMSE.

2.7.4. Other Metrics

There are more metrics for measuring performance in RL experiments. These include the (average) cumulative steps per episode, plotting the number of actions to reach a terminal state. For each episode, the number of executed actions is added to the number from the last episode. Since one expects the learning agent to become more efficient over time, i.e., the agent solves the problem with less (or, in the end, the minimal) amount of actions, the curve should flatten over time. However, this metric is only applicable to episodic tasks. The work at hand focuses on continuing tasks that do not have terminal states.

2.8. Linear Function Approximation

Table-based learning algorithms, such as Q-learning, are suitable for smaller problems with a discrete, finite set of states and actions. However, two problems occur for more complex problems:

1. *Curse of dimensionality:* the required memory increases exponentially with each new feature in the state space. The computing time increases, too, since the agent needs to explore more state-action pairs.

2. Missing *generalization*: table-based approaches typically cannot benefit from experience when they face new states. Q values are distinct tuples, and every state-action value stands for itself. There is no notion of similarity between states. Thus, the RL agent needs to learn for every state-action pair from scratch.

Linear and non-linear *function approximation* solve this problem[2]. This thesis uses *linear* function approximation in chapter 14. It represents the state space more compactly and uses generalization to apply existing experience to similar new states. Generalization means that the agent can transfer knowledge between observed states. For example, a state with one feature having the value of 0.4999 and a state with the same feature having the value of 0.5 most probably are very similar concerning this feature. RL algorithms with function approximation have the potential to utilize this similarity by generalizing and thus applying knowledge from previous observations made in similar states to new ones. While most of the RL framework is the same, there are conceptual and notation differences, summarized in the following.

2.8.1. State Space

When using function approximation, one substantial improvement is that the set of states $\mathcal{S}$ can be infinite. There is no need to discretize each feature – the agent can learn based on floating point values. The state is described based on a typically small number d of features $x_1(s), x_2(s), \ldots, x_d(s)$. Each feature $x(s)$ is a function that maps (aspects of) the observed state s to a continuous value. One option for encoding features is the Fourier basis (see Sutton and Barto (2018) for details), which encodes periodic, linear functions as weighted sums of sine and cosine basis functions with different frequencies. The overall number of features d that describes the state is typically much smaller than the dimensionality of a discretized state space with high resolution in table-based learning. The resulting *feature vector* for encoding state s is defined as follows:

$$\mathbf{x}(s) = \{x_1(s), x_2(s), \ldots, x_d(s)\}^\top \text{ with } d \ll |\mathcal{S}|$$

Relevant aspects of the environment are represented as one or more *functions* of the observed state. For the simplified example in Figure 2.4, this could be used for encoding the smile intensity calculated based on an SSP pipeline as illustrated in section 3.4. In table-based learning, the agent has to discretize the continuous user's smile intensity, resulting in a set of distinct manifestations (in this example, sadness, happiness, and neutral). In a simplified manner, the user's smile intensity can be represented as a continuous (non-discretized) input when using function approximation.

2.8.2. Approximation of q_*

In table-based learning, each state-action tuple (s, a) is associated with one floating point Q value. This mechanism does not work for function approximation since the state's features are now encoded as linear functions in the feature vector $\mathbf{x}(s)$. Instead,

[2] In fact, table-based learning approaches are a special form of function approximation.

each component in $\mathbf{x}(s)$ is assigned a component of the *weight vector* $\mathbf{w}$ with the same dimensionality d of $\mathbf{x}(s)$:

$$\mathbf{w} = \{w_1, x_2, \ldots, x_d\}^\top$$

Instead of updating the Q values table, the vector $\mathbf{w}$ gets updated in every learning step. The goal is finding a weight vector $\mathbf{w}$ for every action $a \in \mathcal{A}$ in order to compute a function $\hat{q}(s, a, \mathbf{w})$ that gives an approximation of the optimal action value function $q_*(s, a)$ for every state $s \in \mathcal{S}$. The main idea of linear function approximation is finding the global optimum $\mathbf{w}_*$, which describes as many as possible states. In the context of linear function approximation, the notation $\hat{q}(s, a, \mathbf{w})$ indicates the use of the weight vector. These action values are computed as follows:

$$\hat{q}(s, a, \mathbf{w}) = \mathbf{w}^\top \mathbf{x}(s, a) = \sum_{i=1}^{d} w_i \cdot x_i(s, a)$$

Adjusting the weight vector relies on stochastic gradient descent methods as described in Sutton and Barto (2018). They minimize the mean-squared-error (see section 2.7) between the real and approximated values based on the observed samples (Sutton et al., 2009):

$$\mathbf{w}_{t+1} = \mathbf{w}_t + \alpha \left[q_*(s, a) - \hat{q}(S_t, A_t, \mathbf{w})\right] \nabla \hat{q}(S_t, A_t, \mathbf{w}_t)$$

Since $q_*(s, a)$ is unknown to the agent and similar to section 2.6.3, the following calculation based on the current approximated $\hat{q}$ values and the reward R_{t+1} is used:

$$\mathbf{w}_{t+1} = \mathbf{w}_t + \alpha \left[R_{t+1} + \gamma \max_a \hat{q}(S_{t+1}, a, \mathbf{w}_t) - \hat{q}(S_t, A_t, \mathbf{w}_t)\right] \mathbf{x}(s_t)$$

Greedy action selection happens based on the estimated $\hat{q}$ values:

$$A_t = \operatorname*{argmax}_a \hat{q}(S_t, a, \mathbf{w}_t)$$

2.9. Conclusion

RL is a class of machine learning algorithms based on trial and error. An RL agent learns optimal decision-making for a given environment. The agent iteratively executes actions in different states and observes the environmental response, which consists of a numeric reward and the next state. Over time, the learning process identifies an optimal policy, which maps each state to one or more optimal actions. The calculation of this policy happens solely based on the reward signal and observed state-action pairs, which makes the agent autonomous and independent of expert knowledge. One of the core challenges in RL is balancing exploitation and exploration, i.e., when and how often to take an optimal or suboptimal action based on the observations made so far.

When modeling an RL problem, the system designer's task is to define the action, state space, and reward function. Typical properties of RL problems include (non)stationarity, (non)determinism, and whether it is episodic or continuing. After modeling the problem, a suitable configuration for standard parameters, such as the learning rate, exploration rate, and discount factor, must be chosen. Typical measures for monitoring an RL agent's learning progress include the received rewards, percentage of optimal actions, and root-mean-squared error. This thesis uses stateless algorithms, tabular RL approaches, and linear function approximation for learning problems of different complexity.

This thesis has several reasons for using the RL framework (see also chapter 12). The structure and autonomy of the learning loop fit well into HRI, and it allows for learning during the interaction. Many algorithms can deal with uncertainty about the environment and its reactions, as expected from the human interaction partner. See chapter 12 for more details. For these reasons, the RL framework serves as the basis for the non-functional adaptation of social robot behaviors in Part IV of this thesis.

3. Social Signals

In human-human conversation, natural language is only one modality for sharing and communicating information. Meta information, such as the speaker's current mood, emotion, or interest in the conversation, is often communicated with verbal or non-verbal behavioral cues. Studies by Mehrabian et al. (1971) observed that this meta information significantly contributes to the success of a conversation.

Poggi and D'Errico (2010) describe *signals* as stimuli, which are emitted by an *emitter* and can be interpreted by a *receiver* to extract the signal's *meaning*. Both emitter and sender can be individuals, a group of people, or artificial agents, such as robots or virtual agents. The authors distinguish *informative* and *communicative* signals, which differ because informative signals do not have an intention, goal, or function for conveying information. Instead, they can occur accidentally by a random combination of events. In contrast, emitters of communicative signals have a *goal of communicating*, which might be a conscious intention (such as when using spoken words or symbolic gestures) or an unconscious intention (such as subliminal facial expressions or gestures, which are produced by the emitter automatically with "a low level of awareness"). Poggi and D'Errico also point out that these signals can be influenced by regional or cultural contexts, biological functions, and more.

A *social* signal is "a communicative or informative signal that, either directly or indirectly, conveys information about social actions, social interactions, social emotions, social attitudes, and social relationships" (Poggi and D'Errico, 2010). Different types of verbal and non-verbal social signals simplify interpersonal communication and interaction because they communicate much more than what a person expresses solely via speech. Social signals are not only essential in human-human communication but also play an important role in the context of HRI. Equipping robots with the ability to express, sense, interpret and react to social signals is one goal to (1) simplify interactions, (2) make them more natural and intuitive for humans, and (3) make robots less "socially ignorant" (Pentland, 2005).

This chapter gives an overview of human non-verbal communication. Both synthesizing (i.e., producing artificial social signals) and sensing (i.e., perceiving and interpreting human social signals) are challenges for social robots. Chapter 5 addresses the former: the synthesis of multimodal behaviors relies on verbal and non-verbal communication channels for expressing the robot's internal state or intentions. The end of the chapter at hand addresses the latter: social signal processing techniques allow the machine to sense, process, and interpret these signals to understand the user's state or intentions.

3.1. Terminology

Three terms occur frequently in the context of human non-verbal communication: *behavioral cues*, *social signals* and *social behaviors* (Vinciarelli, Pantic, and Bourlard, 2009). They build on one another and describe concepts on different levels of complexity.

Behavioral Cues

Behavioral cues are the actual stimuli emitted by the human. As illustrated in Figure 3.1, these cues include gaze, facial expression, vocal behaviors, body movements (postures and gestures), interpersonal distance and more (see section 3.3). For the most part, the dialog partner perceives these cues subconsciously. Typically, behavioral cues occur in combination with verbal communication.

Social Signals

One or more behavioral cues in combination produce a social signal. For example, mutual gaze, a forward posture, a raised voice, energetic gesticulation, and lowered eyebrows typically communicate anger. A social signal can be complex because cues might overlap and occur over time. Depending on the concrete characteristics, such as amplitude, speed, and timing, these cues express a variety of meta information about the signal sender, such as the current mood or emotion, agreement or disagreement, attitudes, intentions, and more. For example, lowered eyebrows as a behavioral cue in isolation could also indicate disgust. Therefore, one can identify social signals only by considering all behavioral cues.

Social Behaviors

Vinciarelli, Pantic, and Bourlard define social behaviors as “temporal patterns of non-verbal behavioral cues”. The authors point out the difference between social signals and social behaviors, which lies in their temporal duration. Typically, social signals have a short or medium duration, from milliseconds to minutes. Examples include turn-taking or mirroring. In contrast, social behaviors, such as agreement, politeness, or empathy, last for a longer time, from seconds to minutes or days.

3.2. Categories of Non-Verbal Behavior

Non-verbal behavior is a “continuous source of signals which convey information about feelings, mental state, personality, and other traits of people” (Vinciarelli, Pantic, and Bourlard, 2009). Ekman and Friesen (2010) classify non-verbal behaviors in five categories:

Emblems Typically, emblems are culture-specific and used intentionally with full awareness when verbal communication is impossible. The emblem replaces a verbal message using the body, face, and hands. Examples include the thumbs-up gesture for expression of agreement or praise or making a ring with the thumb and index finger while splaying out the other fingers to express "okay". However, there are also cases of emblem use during verbal communication without being aware of it, e.g., making a fist when being angry.

Illustrators accompany the spoken language. They enrich the verbal content by emphasizing the speech with movements. Examples include pointing at an object or illustrating the size of a huge object by spreading the arms widely. Illustrators might be used with slightly less awareness and intentionality than emblems.

Affect Displays The non-verbal expression of affect happens primarily with the facial muscles, such as biting one's lips. There is a set of typical, mostly culture-independent, facial movements for each primary type of affect, such as happiness, fear, or anger. Other body movements, such as trembling, can result from the underlying affect.

Regulators are related to a conversation. They regulate the interpersonal dialog flow in terms of speaking and listening. For example, direct eye contact is used when expecting an answer from somebody else or to give another person the chance to talk. Typical regulators also include head nods and eyebrow raises. Regulators are less intentional than emblems and illustrators. Typically, the sender expresses them with low awareness.

Adaptors Ekman and Friesen define adaptors as "adaptive efforts to satisfy self or bodily needs, or to perform bodily actions, or to manage emotions, or to develop or maintain prototypic interpersonal contacts, or to learn instrumental activities". They can be motivated or caused by the situation, environment (objects or other persons), or the person's needs. Often, they are expressed with little or no awareness and varying intentionality, depending on their motivation.

As outlined above, these categories vary in cause, intention, and awareness. Non-verbal behaviors accentuate or illustrate spoken words, regulate conversation flow, are used to cope with a situation, and more. Some of these behaviors are very culture-specific, and some are largely culture-independent, such as the expression of affect. The expression and perception of non-verbal behaviors are crucial to human-human and human-robot interaction.

3.3. Non-Verbal Communication Channels

As illustrated in Figure 3.1, this section focuses on non-verbal communication channels. Kendon, Sebeok, and Umiker-Sebeok (2010) define the term *non-verbal communication* as "all of the ways in which communication is effected between persons when in each other's presence, by means other than words." Specifically, they refer to "bodily activity, gesture, facial expression and orientation, posture and spacing, touch and smell" and "those aspects of utterance that can be considered apart from the referential content of what is said". The authors also point out that non-verbal communication is essential to

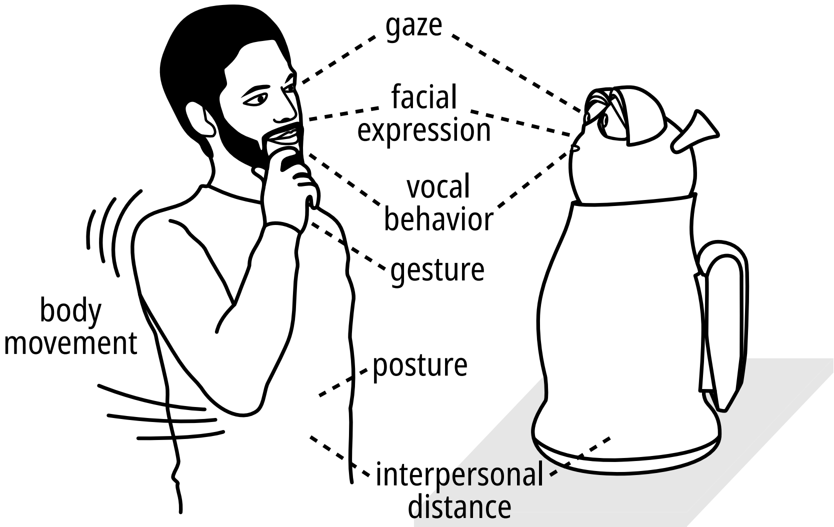


Figure 3.1.: Behavioral cues in HRI. Adapted from Baur (2018).

communicating information, which cannot be achieved in any other way, e.g., using a specific pronunciation, timing, speed, or tone.

As a result, non-verbal behaviors convey personal information about their sender, such as the emotional or mental state, feelings, and personality, but also social relationships between all persons involved. Non-verbal communication is also crucial when the verbal content in isolation is ambiguous or the intended meaning differs from the actual utterance, such as irony (see chapter 10). Thus, non-verbal behaviors are not optional but an inherent part of human and human-robot communication.

Both the *perception* and *generation* of non-verbal behaviors is a challenge for social robots. The robot needs to sense human behavioral cues and interpret them. Only when aware of its counterpart's non-verbal behaviors can it estimate underlying intentions, interests, and emotions and react accordingly. At the same time, the robot must also mimic human social signals to communicate its internal state and intentions more clearly and intuitively.

This work focuses on cues, which are produced and occur typically in the context of HRI, both in terms of perception from the human and expression by the robot: posture, gesture, facial expression, gaze, and vocal behaviors. Vinciarelli, Pantic, and Bourlard (2009) also list physical appearance in terms of height, attractiveness, and body shape as human behavioral cues. This thesis excludes these aspects since the human interaction partner's embodiment cannot be changed. A designer creates the robot's embodiment upfront, so it cannot be changed too. Similar applies to space and environment, which includes distance and seating arrangement, as well as walking in the context of gesture and posture. These aspects are relevant in the context of a mobile robot, which is not the case for the work at hand (see chapter 11). In this thesis, the interaction scenarios are stationary, with the human sitting opposite the robot.

3.3.1. Gestures

Gestures are "conscious or unconscious body movements made mainly with the head, the face alone, or the limbs" (Poyatos, 1984). They often accompany spoken contents, e.g., by moving arms and hands (McNeill, 1985) or affective states, such as shame and embarrassment, e.g., by touching the face (Costa et al., 2001). McNeill (2011) observed that 90 % of body gestures occur during speech as one of the categories in section 3.2. Frequently, gestures regulate interactions (e.g., for turn taking), communicating information without speaking (e.g., the thumbs-up gesture for expression of agreement), greeting (e.g., waving hands), and more (Vinciarelli, Pantic, and Bourlard, 2009). Most gestures are performed consciously. McNeill (2011) classifies gestures in four primary categories, distinguished as imagistic (when they depict imagery) and non-imagistic (when they do not depict imagery) types:

Iconics are gestures with "close formal relationship to the semantic content of speech". They illustrate what is said in spoken language, e.g., raising the hand upwards when talking about a rising balloon. Both speech and gesture must refer to the same content to be iconic, but they can complement each other with information that is not part of the other modality.

Metaphorics are more complex than iconics since they illustrate abstract concepts. They must depict both a *base* (a concrete entity or action) and a *referent* (the concept of a question or answer). This dual structure is the central element of a metaphoric gesture.

Deictics are pointing movements. They are performed with objects or body parts, such as the head, nose, chin, or – typically – with the pointing finger. They do not necessarily refer to concrete entities but any meaning associated with the region of the selected gesture space.

Beats do not express a concrete meaning, but they are rather "small, low energy, rapid flicks of the fingers or hand". Typically, they are performed in the current or rest position of the hand.

Section 14.1 uses human gestures and posture for estimating user engagement, which drives a robot's adaptation process.

3.3.2. Posture

In contrast to gestures, posture is "more static [. . .] and used less as a communicative tool, although they may reveal affective states and social status" (Poyatos, 1984). Postures are typically chosen unconsciously; therefore, they are the most reliable cue concerning a person's actual attitudes in the context of social situations (Vinciarelli, Pantic, and Bourlard, 2009). Vinciarelli, Pantic, and Bourlard point out three properties:

Inclusive vs. non-inclusive Inclusive postures take the presence of others into account, e.g., when facing each other. Non-inclusive postures do the opposite, e.g. when facing the opposite direction.

Face-to-face vs. parallel body orientation Face-to-face interaction, such as when sitting opposite each other, is “more active and engaging”, as it also requires mutual monitoring. When sitting parallel to each other, this may be an indication of less mutual interest.

Congruence vs. incongruence Congruent postures are symmetric, which indicates a “deep psychological involvement”. Mirroring (Chartrand and Bargh, 1999) is an example of symmetric posing, where the communication partners mutually imitate the other. Incongruent postures are asymmetric and are an indication of the opposite.

Example postures are sitting or standing, leaning back or forward. According to Vinciarelli, Pantic, and Bourlard (2009) and McArthur and Baron (1983), postural behavior also includes walking or other movements, as long as they convey social information, such as status, dominance, and affective state. Mehrabian (1969) points out that posture and position indicate attitudes towards communication partners. In this context the author gives an overview of important posture variables: *eye contact*, *body orientation*, *arms-akimbo position* and *trunk relaxation*. For example, a reduced distance indicates more liking, and reduced eye contact indicates disliking. With increasing attitude, female communicators orient their bodies more towards the addressee, similar to more direct eye contact.

The posture is also an indicator of human affect and engagement. For example, leaning back might indicate disengagement and boredom (D’Mello, Chipman, and Graesser, 2007). Section 14.1 uses posture and gestures to implement a robot’s adaptation process based on estimated human engagement.

3.3.3. Facial Expression

The human face has dozens of muscles, providing a very powerful non-verbal communication channel. It communicates emotions (Ekman, 1993), intentions and pain, regulates interpersonal behavior and attitudes (la Torre and Cohn, 2011; Knapp, Hall, and Horgan, 2013). Facial expressions also provide cues about personality and alertness (la Torre and Cohn, 2011). Movements happen very fast, including *micro expressions*, which reveal concealed emotions and are used for exposing liars (Ekman, 2009).

The Facial Action Coding System (FACS) (Ekman and Friesen, 1978) is the de-facto standard for specifying facial expressions and related emotions based on so-called *action units*. For example, anger is expressed by a combination of lowered eyebrows, raised and tightened eyelids and tightened lips; happiness is marked by raised cheeks and pulled lip corners.

Besides emotion recognition, FACS provides a reference for designers and animators for creating realistic facial expressions and animations (Tolba, Al-Arif, and El Horbaty, 2018) in the context of virtual agents and robots (Fong, Nourbakhsh, and Dautenhahn, 2003). Since many robots are appearance-constrained due to their hardware, a one-to-one mapping between FACS and robotic hardware is often impossible. Thus, trade-offs and custom designs must be made (Ribeiro and Paiva, 2012).

Facial expression is of central importance in chapters 9, 10, 11 and section 14.3. On the one hand, synthesizing humorous robot behaviors relies on facial expressions. On the other hand, the user's facial expressions are sensed for estimating their affect in the context of adaptation.

3.3.4. Gaze

The human eye serves multiple purposes at the same time. It senses the emotions, feelings, and intentions of interaction partners, guides social behaviors, and conveys emotional information, all at once. Often, gaze behaviors are triggered subliminally, such as when expressing emotional (Izard, 1991; Adams Jr and Kleck, 2005) and mental state (Teufel et al., 2010). Gaze can also be used intentionally, for example, to guide visual attention (Frischen, Bayliss, and Tipper, 2007), to shift the interaction partner's focus (Senju and Csibra, 2008) or to initiate, monitor, and regulate conversations (Reis and Sprecher, 2009). Eye contact also plays a key role in close relationships for increasing warmth, expressing involvement, and intimacy (Reis and Sprecher, 2009). Gaze duration (Kuzmanovic et al., 2009) and direction are important parameters.

Gaze behavior is based on multiple low-level gaze cues (Ruhland et al., 2014). Saccades, the vestibulo-ocular reflex, smooth pursuit, and vergence all serve the purpose of fixating on static or moving gaze targets. Combined eye-head movements are used when eyeball movements exceed certain thresholds. Eye movements come hand in hand with eye blinks. The frequency of these voluntary or reflexive eyelid movements is related to the cognitive state and activity (Stern, Walrath, and Goldstein, 1984). In human-computer interaction (HCI) and HRI the generation of natural gaze behavior for social robots and virtual agents (Ruhland et al., 2014; Pelachaud and Bilvi, 2003) makes embodied agents appear more lifelike. For example, unblinking faces are visually disconcerting (Ruhland et al., 2014).

Gaze is an essential modality in this work for generating multimodal behaviors for a social robot. This ranges from low-level gaze cues and blinking as described in chapter 11 to humor markers in chapters 9 and 10 and their adaptation in section 14.3.

3.3.5. Vocal Behaviors

Speech is one of the most prominent communication channels in human interaction. However, speech is more than just linguistic content. Every sentence, word and syllable typically is surrounded by *non-verbal* vocal cues: *paralinguistic* or rather *vocal behavior* deals with "*how* something is said, not what is said" (Knapp, Hall, and Horgan, 2013).

Prosody describes sound variations that occur during speech. It includes speech rate, rhythm, pitch, accents, loudness, duration, pauses, variability, resonance, and articulation (Knapp, Hall, and Horgan, 2013). Vinciarelli, Pantic, and Bourlard (2009) summarize this under the term *voice quality*. These variations communicate information beyond spoken language, including mood, emotion, and mental state, as well as humor (see section 4.4.4) and irony (see section 4.4.5). Thus, vocal behaviors complement and impact the linguistic content and can even change its intended meaning.

Beyond that, vocal behaviors also include *linguistic vocalizations*, such as "ehm" or “uhm”. They fill pauses when thinking about an answer for a question or a missing word (Vinciarelli, Pantic, and Bourlard, 2009). They are also used for *back-channeling*, i.e., to express attention or agreement (Shrout and Fiske, 1981). Moreover, there are also *non-linguistic vocalizations* (Vinciarelli, Pantic, and Bourlard, 2009). These standalone vocal cues occur independently of spoken words and include laughter, cries, groans, sighs, or yawns. They also communicate information about the interlocutor, such as emotions or attitudes.

There are also different types of silence. Silence can result from difficulties in talking, the need to think about the following words, hesitation or problems in dealing with a conversation, as well as a sign of respect or ignorance (Vinciarelli, Pantic, and Bourlard, 2009). On top of that, vocal non-verbal behaviors are also used for turn-taking, i.e., for regulating and coordinating the conversation.

Similar to facial expression, vocal behaviors are of central importance in chapters 9, 10 and 11 for producing multimodal, humorous robot behaviors, which are then adapted in section 14.3.

3.4. Social Signal Processing

Pentland (2005) originally pointed out the “social ignorance” of computers since HCI typically uses keyboard, mouse, or touch input and does not process non-verbal communication channels. In Pentland (2007), he came up with the term *social signal processing*, which is the science of capturing and processing human social signals with computers. SSP provides the machine with the ability to *sense* and *interpret* (Vinciarelli, Pantic, and Bourlard, 2009) human *implicit* interaction (i.e., non-verbal behaviors), in addition to traditional *explicit* interaction (e.g., input via mouse, keyboard, speech, touch) (Schmidt, 2000). Vinciarelli, Pantic, and Bourlard (2009) point out that SSP is not only important for anticipating the users’ needs and improving their quality of life. It provides an opportunity for *adapting* the machine based on human social signals, which is of central importance in chapter 14.

In the last decade, SSP has become an essential tool in both HRI and HCI for making interactions more intuitive and natural. Observing and extracting information from the dialog counterparts’ non-verbal behaviors seems intuitive for humans, who often do this subconsciously. However, the machine must actively sense and process this information. Baur (2018) points out four key challenges in SSP:

Synchronization The Machine must process multiple modalities in parallel.

Uncertainty Human communication is relatively chaotic, as compared with computers, which follow defined rules.

Fusion Social cues should not be viewed in isolation, but analysis and interpretation of complex behaviors require combining the data from multiple modalities.

Real-time Constant monitoring and interpretation happen in parallel to the interaction to provide timely feedback.

Over the years, several SSP software packages appeared, which implement these techniques on desktop and mobile platforms. These include the Social Signals Interpretation (SSI) framework (Windows) by Wagner et al. (2013), MobileSSI (Linux, Android) by Flutura et al. (2018) and SSJ (Android) by Damian, Dietz, and André (2018).

The Social Signal Interpretation Framework

Figure 3.2 illustrates a typical pipeline approach for SSP as implemented in the SSI software by Wagner et al. (2013). It records, analyses, and recognizes human behaviors in real time and supports audiovisual and physiological sensors, including (depth) cameras and microphones. The software includes algorithms for synchronizing and handling all data streams, machine learning for online and offline analysis and training of models, filters, feature extraction, fusion, pattern recognition, activity detection, clustering, classification, and an application programming interface (API) for plugin development.

One can create complex pipelines by combining different node types and interconnecting them, as illustrated in the bottom center of Figure 3.2. Typically, the pipeline passes computed results to the application or robot as events over the network (see right side of Figure 3.2).

3.5. Conclusion

Humans use both verbal and non-verbal communication. The spoken language is surrounded by conscious and unconscious non-verbal vocal behaviors, gestures, posture, facial expression, gaze, and spatial body movements. In combination, these behavioral cues generate social signals, which communicate various information about the sender's current mood, emotions, intentions, attitudes, and more.

In HRI, non-verbal behaviors are challenging in two ways: a robot must both be able to (1) produce and express these behaviors on its own and (2) process, interpret and react to the user's behavioral cues. The former often relies on mimicking human behavioral cues that must be tailored to the individual hardware and appearance, limited by the robot's embodiment and actuators. Various SSP techniques addressed the latter in the last decade. They make it possible to complement traditional explicit interaction in terms of keyboard, mouse, touch, and speech input with implicit interaction based on social signals. The SSI software implements these techniques. It synchronizes, processes, interprets and fuses inputs from a wide range of sensors in real-time.

This thesis first deals with the generation of multimodal robot behaviors in Part III with the help of the presented non-verbal communication channels. Building on this, social signals and corresponding SSP techniques provide an essential building block for adapting the generated multimodal robot behaviors to the individual user in Part IV.

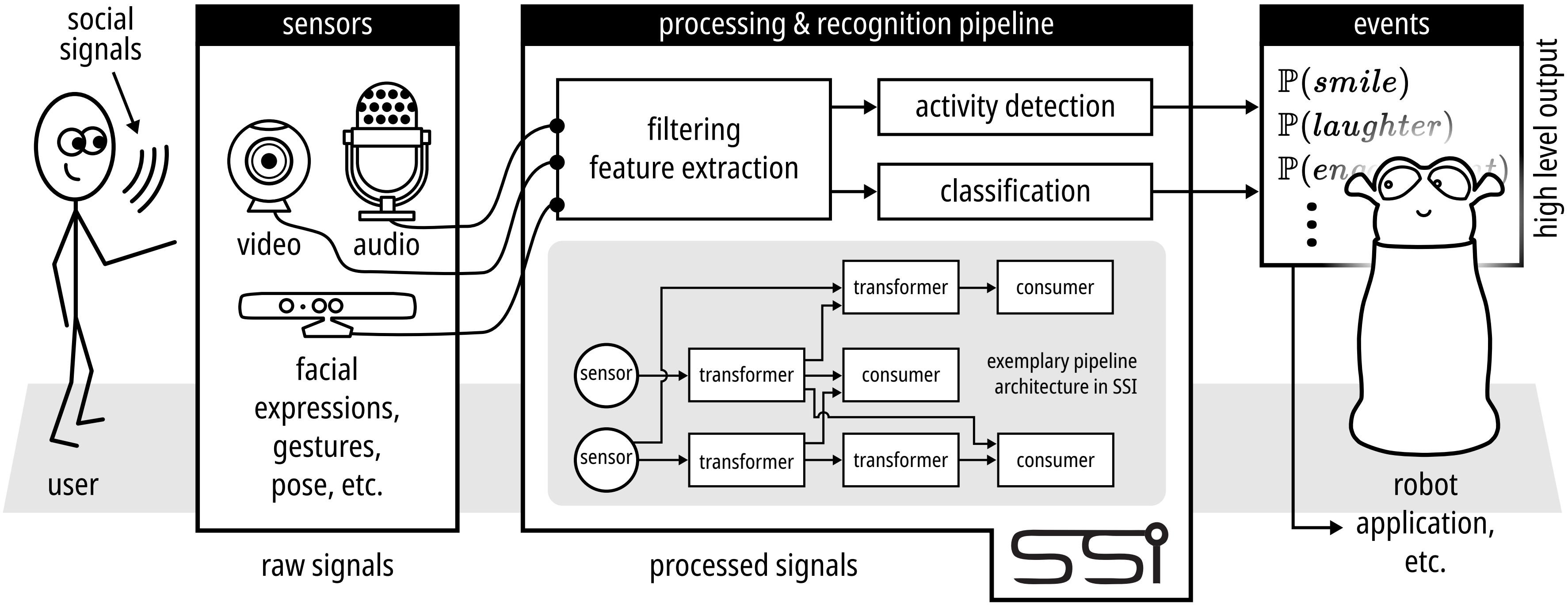


Figure 3.2.: The SSI software uses a pipeline approach. Adapted from Wagner et al. (2013).

4. Psychological Background

A common approach is to mimic human verbal and non-verbal behaviors to realize intuitive communication and interaction with social robots. Therefore, it is essential to get an overview of the psychological foundations of these behaviors.

This chapter introduces the background of personality, persona, politeness, and humor. Personality generally determines how people behave, express themselves, and react. Breazeal (2004) points out that a compelling robot personality makes interaction more interesting and supports establishing a relationship between the human and the robot. Personality is closely linked to persona, politeness, and humor. The focus on these aspects is also motivated by the application contexts assistance and entertainment, which form the framework for this thesis's experiments.

The psychological background is essential for the generation of verbal and non-verbal behaviors *expressing* personality, persona, politeness, and humor in Part III, which aim to equip the robot with socially intelligent behaviors. This chapter also presents theories about the compatibility between particular personality profiles, different forms of politeness, and disparate humor presentation strategies since Part IV addresses heterogeneous findings by adapting the generated robot behaviors within the framework of the presented psychological models and theories.

The contents of this chapter extend the background provided in Ritschel et al. (2019b), Ritschel et al. (2020a), Ritschel et al. (2020b), Janowski, Ritschel, and André (2022), and Kiderle et al. (2021).

4.1. Personality and Interpersonal Stance

Personality describes a person's disposition to respond to specific events in a particular manner. More specifically, this means an individual's behavior patterns that can be observed in a wide range of contexts or their disposition to respond in a certain way when they find themselves in a particular situation (Argyle and Little, 1972). The literature reports several models for describing and measuring human personality, including the *Five-Factor Model* (McCrae and Costa Jr., 2008). Links exist between personality traits and the *interpersonal circumplex* (Horowitz et al., 2006), which models attitudes towards other persons, as well as the politeness theory by Brown and Levinson, 1987, which in turn is linked to the interpersonal stance.

4.1.1. Five-Factor Model

A common theory for describing personality is the Five-Factor Model (McCrae and Costa Jr., 2008), also known as *Big Five*. It defines personality in terms of five dimensions:

Neuroticism describes a person's tendency to experience negative affect (such as distress and anxiety), to act impulsively, or change moods quickly and frequently. Emotionally stable people, however, are calm and relaxed. This factor is also known as *emotional stability*.

Extraversion is associated with outgoing and assertive behavior traits. Extravert people tend to be sociable, talkative, expressive, and active, whereas introverts are more reserved and quiet.

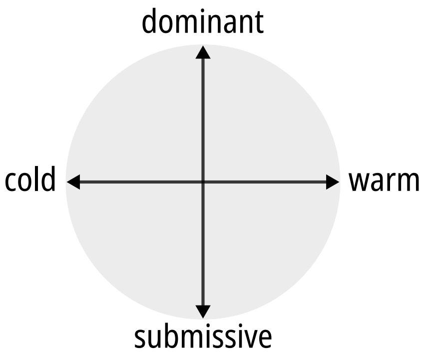
Openness encompasses curiosity, creativity, and intellectuality. Open-minded people have a wide range of interests and think unconventionally. Closed-minded people are generally conservative and unimaginative.

Agreeableness is related to getting along well with others. It encompasses qualities such as being kind, compassionate, forgiving, generous and trusting. In contrast, disagreeable persons tend to criticize others and to be cold-hearted or inconsiderate.

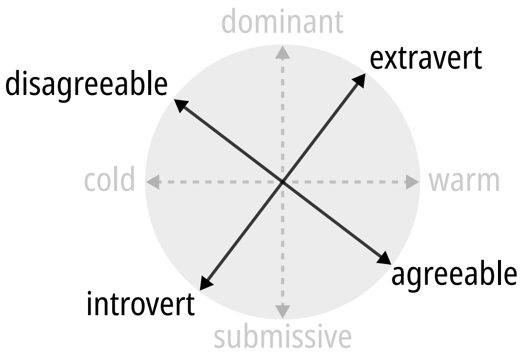
Conscientiousness is concerned with disciplined and responsible qualities. People, who score high in conscientiousness, are generally dutiful and reliable, well-organized and thorough. They are efficient and productive rather than lazy and self-indulgent.

Personality affects human verbal and non-verbal behaviors. For example, extravert people tend to be more energetic. They perform more posture shifts, wider and faster (hand) gestures, have higher head movement frequency, use more facial expressions, and have more mutual gaze with conversational partners than introvert people (Riggio and Friedman, 1986; Lippa, 1998; Knapp, Hall, and Horgan, 2013; Gallaher, 1992; Gifford, 1991; Hassin and Trope, 2000; Jäncke, 1993; Rutter, Morley, and Graham, 1972). Extraverts also tend to prefer a closer interaction distance than introverts (Williams, 1971). The personality influences prosodic parameters while speaking, such as speech rate, silence duration, and pitch (Smith et al., 1975; Scherer, 1979). Moreover, the literature has identified several linguistic features which correlate with different dimensions of personality (Pennebaker and King, 1999; Pennebaker, Mehl, and Niederhoffer, 2003; Mehl, Gosling, and Pennebaker, 2006; Fast and Funder, 2008). The work at hand focuses on spoken language and – in specific – extraversion.

Siegman and Pope (1965), Cope (1969), Scherer (1979), Furnham (1990), Pennebaker and King (1999), Mehl, Gosling, and Pennebaker (2006), and Gill and Oberlander (2019) report linguistic and prosodic markers of extraversion in spoken language. These markers include being more talkative and repetitive at higher speech rates, being less hesitant, making fewer and shorter pauses, using richer and longer words, using less formal language, and more positive emotion words, agreements, and compliments. In contrast, introverts, i.a., tend to use more tentative words (e.g., "perhaps"), hedges (e.g., "somewhat") negations, pauses, formal greetings and first-person singular pronouns as opposed to less formal expressions used by extraverts (Siegman and Pope, 1965; Pennebaker and King, 1999; Oberlander and Gill, 2006). Some of these markers correlate with culture, such as the number of pauses when comparing American and German introverts and extraverts (Scherer, 1979). Mehl, Gosling, and Pennebaker (2006) report



(a) status & affiliation



(b) extraversion & agreeableness

Figure 4.1.: Two pairs of dimensions, which define the IC. Adapted from Janowski, Ritschel, and André (2022).

gender-related variations of linguistic features. Extraverts tend to speak faster with higher, louder, and less monotonous voices than introverts (Hall, 1990; Smith et al., 1975; Pittam, 1994; Gill Woodall and Burgoon, 1983). The work at hand focuses on generating robot utterances with varying degrees of extraversion in chapter 7.

4.1.2. Interpersonal Circumplex

The interpersonal circumplex (IC) (McCrae and Costa, 1989; DeYoung et al., 2013; Horowitz et al., 2006) models attitudes towards other persons. It uses two dimensions:

Status ranges from *submissive* to *dominant* and is usually displayed as the vertical dimension. It is also known as *agency* and describes peoples' tendency to act according to their own will.

Affiliation ranges from *cold* to *warm* and is placed horizontally. It is also known as *communion* and describes a person's social closeness to other people.

McCrae and Costa (1989) and DeYoung et al. (2013) show that status and affiliation are related to extraversion and agreeableness. As explained in section 4.1.1, high extraversion implies sociability and closeness to people, but also assertive and dominant behavior tendencies. Similarly, agreeableness represents social compliance, a combination of warm-hearted and submissive behaviors. According to McCrae and Costa (1989) and DeYoung et al. (2013) these two traits form an alternate pair of axes, which is rotated about 30–45 ° relative to status and affiliation, as shown in Figure 4.1.

4.1.3. Persona

The term *persona* is the Latin root of the term personality and is closely linked to it. While personality refers to "regularities and consistencies in the behavior of individuals in their lives" (Snyder and Ickes, 1985), persona means "a consistent pattern of behavior and attitudes" (Snyder and Ickes, 1985). According to Snyder and Ickes the term persona originates from theaters in ancient Rome, where it referred to the mask worn by an actor. Later, the meaning changed, and the word referred to the character played by the wearer and finally to the actor. Today, the term persona describes "regularities and consistencies in the characters created by actors on the stage" (Snyder and Ickes, 1985), resulting in a fictional personality with varied and stable behavioral and personality patterns (Matthews, Deary, and Whiteman, 2003).

4.2. Interpersonal Compatibility

The literature reports phenomena concerning the compatibility of different peoples' attitudes and personality traits. Interpersonal attraction, i.e., the attraction between individuals, has been investigated for many decades in different populations. The similarity and complementarity attraction paradigm (Byrne, 1997) have emerged as prominent theories for describing interpersonal compatibility:

Similarity attraction attributes compatibility to people with similar attitudes or personality profiles (Byrne and Griffitt, 1969; Wetzel and Insko, 1982; Yeong Tan and Singh, 1995; Montoya and Horton, 2013). According to this theory, an extravert prefers extravert communication partners, and an introvert prefers introvert interlocutors.

Complementarity attraction relies on the opposite assumption, assigning compatibility to people with complementary attitudes or personality profiles (Byrne, 1997; Kristof-Brown, Barrick, and Kay Stevens, 2005; Kausel and Slaughter, 2011; Estroff and Nowicki Jr., 1992; Markey, Funder, and Ozer, 2003). According to this theory an extravert prefers an introvert interlocutor and vice versa.

Similarity and complementarity attraction, as well as other findings, have not only been observed in human interaction but also in HRI. See section 5.2.2 for more details on human-robot personality matching. These heterogeneous findings about interpersonal attraction are an important motivation for autonomous adaptation in section 14.1.

4.3. Politeness Theory

The politeness theory by Brown and Levinson (1987) assumes that every interlocutor has a public self-image (i.e., identity), which is called the *face*. It consists of the following two wants, which are related to the IC according to Oakman, Gifford, and Chlebowsky (2003):

Negative face People desire to be free and autonomous in their actions, without any restrictions by others. This desire resembles the status dimension of the IC, which represents a person's tendency to act autonomously.

Positive face People desire to have other people's approval and appreciation of their own self-image. Similar to the affiliation dimension of the IC, it implies group membership and a social bond with others.

According to Brown and Levinson, 1987, politeness allows mitigating so-called *face threats*, which threaten peoples' negative or positive faces. Negative face threats occur, for example, when putting pressure on the interlocutor (e.g., by ordering/requesting/suggesting/reminding the person to do something). Positive face threats result, for example, from any indication of the speaker not caring about the interlocutor's desires, attitudes, or feelings (such as expressing criticism, disapproval, complaints, or disagreement). Brown and Levinson explain different politeness strategies for minimizing threats in order to preserve the reputation of conversation partners:

Bald on-record strategy does not minimize the threats.

Positive politeness strategies minimize threats to the interlocutor's positive face by indicating that the speaker shares at least some of the interlocutor's wants and that the threat does not mean a "negative evaluation in general of [the interlocutor's] face".

Negative politeness strategies minimize threats to the interlocutor's negative face by expressing understanding for the interlocutor's wants and that the threat "will not (or will only minimally) interfere with the addressee's freedom of action".

Off-record strategies indirectly communicate the speaker's intention, leaving room for negotiation.

Similar to the personality dimensions, politeness is reflected in spoken language. Brown and Levinson provide an extensive list of linguistic politeness strategies. For example, negative politeness strategies include being direct or pessimistic, using questions and hedges, giving deference, and using nominalization. Conversely, positive politeness strategies include noticing, approving, or showing interest in the interlocutor's interests, wants and needs, seeking agreement, using in-group identity markers (e.g., "Buddy"), optimism, and joking. "Please" and "thank you" are common examples of positive politeness. Off-record strategies include ambiguity, tautologies, metaphors, irony, rhetorical questions, over-, and understatements. The personality literature also reports some of these markers, such as questions, hedges, exaggerations, optimism, or pessimism (see section 4.1.1). Furthermore, combined with using jokes as a positive politeness strategy, this also illustrates the link between personality, politeness, and humor. In this thesis, politeness is the subject of chapter 8, which presents a social robot with several politeness strategies in different contexts. The robot adapts the strategies to the individual user's preferences in section 13.1.

4.4. Humor

Humor plays a very important role in human communication. Besides providing entertainment and making conversations more enjoyable, it also has profound benefits: it regulates conversations, helps to establish common ground between the conversational partners, eases communication problems (Binsted et al., 2006) and helps to cope with critique or stress (Nijholt, 2007). Moreover, a shared sense of humor has powerful effects on interpersonal attraction (Murstein and Brust, 1985; Cann, Calhoun, and Banks, 1997; McGee and Shevlin, 2009). There is also a close link between personality and humor: Thorson and Powell (1993) mention that humor is "something either innate or closely related to personality that makes life more enjoyable and worth living". What contributes to personality are also the differences between individuals concerning humor appreciation, which might be "either high or low in personal sense of humor or somewhere in between" (Thorson and Powell, 1993). The individual style or lack of humor shapes personalities and makes them unique, as are individual humor preferences. As outlined in section 4.3, humor may also be used as a positive politeness strategy.

This section provides an overview of selected types of humor and how humans communicate them. The focus is on non-verbal communication. The overview provides the basis for the multimodal behavior generation of humorous robot behaviors in chapters 9 and 10 and their adaptation to the individual user's preferences in section 14.3.

4.4.1. Verbal and Non-Verbal Humor

Humor is a very personal, creative, and multifaceted sign of social intelligence. The literature differentiates two types regarding the form of presentation. *Verbal* humor is limited to "joke-carrying text" (Raskin, 2012), which is performed by the reader as they read it (Norrick, 2004). However, in many cases, humor is presented orally and in combination with non-verbal behaviors, including gestures, facial expressions, shifts in dialect or voice quality, and more (see the following sections and section 3.3). In some cases, humor might not even include spoken language, but consist of a solely non-verbal performance, such as in the case of pantomime. Thus, *non-verbal* humor includes humor, which is "not created, described and expressed by a text" (Raskin, 2012), but "depends on presentation" (Norrick, 2004). In particular, non-verbal humor includes text, too, as long as the text is just one component of the humorous content, such as a joke, and accompanies the non-verbal performance (Norrick, 2004).

4.4.2. Canned and Conversational Humor

The literature distinguishes two types of verbal humor: canned humor and conversational humor (Attardo, Pickering, and Baker, 2011; Dynel, 2009). *Canned* humor is what is commonly associated with jokes. Jokes (see section 4.4.4) are delivered "within a humorous frame and rarely communicate meanings outside it" (Dynel, 2009), i.e., they are contextually dissociated from the conversation topic.

Conversational humor is embedded in a conversation and context-dependent, for example, by referring to the surrounding dialog topic. In contrast to canned humor, it does not rely on a longer narrative. It ranges from "single-word lexemes, phrasemes to whole sentences and even multi-turn exchanges interwoven into non-humorous discourse" (Dynel, 2009). According to Dynel this includes neologisms (new words), witticisms, puns, stylistic figures, such as irony (see section 4.4.5) and puns, and more. Dynel also points out that there are overlaps between some of these categories.

4.4.3. Construction and Presentation

The presentation of humor involves two participants: the speaker and the audience. While the speaker produces the humor, the audience supports the humorist's face (Hay, 2001) by providing feedback, such as laughter, smiling, and more. Thus, the audience also contributes to successful joke delivery and conversational humor.

4.4.3.1. Markers and Factors

Instances of humor rely on a meta message "this is humorous/ironical/sarcastic" (Attardo, Wagner, and Urios-Aparisi, 2013) to enhance the chances that the audience adequately detects the humor and interprets the speaker's performance and intentions correctly (Gironzetti, 2017). This meta message is communicated with humor *markers*, which are linguistic, paralinguistic, or nonverbal elements signaling the presence of humor. Markers are not essential: one can remove them without removing the humor itself, but their absence can affect the recognition of the humor negatively (Attardo, 2000b; Gironzetti, 2017). In contrast, humor *factors* are essential elements. Their removal destroys the humor. Attardo, Wagner, and Urios-Aparisi, 2013 further use the terms *indicators* and *indices* to describe non-essential elements, which are always co-occurring with humor. Similar to markers, indicators can be removed without affecting the humor, while indices unintentionally indicate the presence of humor (Gironzetti, 2017).

The speaker's task is to signal the presence of humor by including multimodal markers in the performance. The markers presented in sections 4.4.4 and 4.4.5 provide the basis for the multimodal generation approaches presented in chapter 9 and chapter 10.

4.4.3.2. Humor Support

The audience's task is to provide the speaker with *humor support* (Hay, 2001). Hay points out that any "reaction from [the] audience that implies appreciation of the humor" is important since it displays acknowledgment and understanding. Attardo, Pickering, and Baker (2011) describe humor support as "conversational strategies used to acknowledge and support humorous utterances".

Hay (2001) and Attardo, Pickering, and Baker (2011) list several humor support strategies. The most apparent strategy is appreciating humor verbally and non-verbally through laughter, smiles, smirks, facial statements, and additional body language. Besides these multimodal behavioral cues, support can also be given by contributing more humor, e.g., by playing along with the joke or witticism initially presented by the speaker and developing it further. Hay mentions that this is often the case for irony and fantasy humor. Humor support also includes echoing (repeating words of the speaker in appreciation), offering sympathy, or contradicting self-deprecating humor (when laughter is an inappropriate response, e.g., in troubles talk). In particular, Attardo, Pickering, and Baker (2011) also include comments on the humorous utterance that express support, such as "Yeah!" and other Meta-Knowledge Resources (see next section). Hay also points out that the audience may display understanding but not provide explicit support (e.g., irony or if the humor is already a support strategy).

Humor support is of central importance in section 14.3, which relies on human audio-visual humor appreciation in terms of laughter and smile for adapting generated social robot humor to the individual user's preferences.

4.4.4. Jokes

Jokes have a two-part structure. The first part is the *setup*. It is a narrative, which typically is decoupled from the non-humorous conversation topic (Attardo, Pickering, and Baker, 2011). The second part is the *punchline*, which engenders unexpected incongruity with the setup (Dynel, 2009) and thus creates humor. Special forms of jokes are riddles and one-liners (Dynel, 2009). Riddles consist of a question followed by an unpredictable or silly answer, such as "Why don't Calculus majors throw house parties? Because you should never drink and derive." One-liners have a very short punchline consisting of only a few words, such as "Never trust atoms; they make up everything."

Sometimes, a negotiating sequence, such as "Do you know this joke?" or "I will tell you a joke", is used by the speaker for introducing the joke (Attardo, Pickering, and Baker, 2011; Canestrari, 2010). After the joke, humor support may be provided by the audience explicitly, e.g., with linguistic expressions, such as "I did not like that one", "That was funny", or "Do not make me laugh!" (Canestrari, 2010). These are examples of explicit verbal instances of *Meta-Knowledge Resource* (Canestrari, 2010). Canestrari uses this term for verbal, non-verbal, and para-verbal signals, which express a humorous intention on the "meta-communicative level".

Multimodal Markers

There is no consensus on whether and which multimodal markers are used for verbal humor (Attardo, Pickering, and Baker, 2011; Gironzetti, 2017). Instead, it seems to depend on the individual speaker and performance. However, the literature reports that professional comedians use markers intentionally, e.g., to emphasize parts of the text and to evoke the impact of the punchline.

Table 4.1 presents a list of multimodal markers from the literature, excluding those of verbal irony (see section 4.4.5). These markers were observed in human conversational humor and canned jokes. One of the most frequent observations is an increase in pitch, volume, speech rate, and a break at the joke's punchline (Pickering et al., 2009; Attardo and Pickering, 2011; Gironzetti, 2017). Also, the literature reports an "atypical" prosody with a minimal pitch range and a special linguistic syntax for the setup of riddles. Comedians may also use other verbal and non-verbal markers, such as controlled vocabulary, facial expression, gestures, and posture (such as head movements in stand-up comedy (Martínez Estrada, 2020)). Sometimes, the speaker laughs or giggles. Facial expressions include smiling and gaze behavior.

4.4.5. Verbal Irony

Verbal irony has been traditionally seen as "a figure of speech which communicates the opposite of what was literally said" (Wilson and Sperber, 1992). Later, Attardo (2000a) and

Table 4.1.: Multimodal markers of verbal humor

Modality	Markers
Prosody	Pitch ↑ (Bauman, 1986; Chafe, 1994; Audrieth, 1998; Norrick, 2001; Wennerstrom, 2001), volume ↑ (Bauman, 1986; Chafe, 1994; Wennerstrom, 2001; Archakis et al., 2010), speech rate ↑ (Norrick, 2001; Wennerstrom, 2001; Archakis et al., 2010), break at punchline (Bauman, 1986; Chafe, 1994; Audrieth, 1998; Archakis et al., 2010), combination of limited pitch range, minor pitch change (syllables, utterance), syntax and content in the setup of punning riddles (Bird, 2011)
Speech	Laughter (Gironzetti, 2017; Pickering et al., 2009; Attardo, Pickering, and Baker, 2011)
Facial expr.	Smile (Gironzetti, 2017; Pickering et al., 2009; Attardo, Pickering, and Baker, 2011) and gaze at the face areas involved in the smile (eyes, mouth) (Gironzetti et al., 2015; Gironzetti, Attardo, and Pickering, 2016), change gaze target to another person (Katevas, Healey, and Harris, 2015)

Dynel (2009) differentiated more types and finer gradations of irony. Dynel categorizes verbal irony as a stylistic figure within the scope of conversational humor. Verbal irony is a type of conversational humor, i.e., it is part of a surrounding conversation or dialog. Attardo (2000a) lists, i.a., the following properties:

1. it is contextually inappropriate,
2. at the same time relevant,
3. it is used intentionally with awareness of the contextual inappropriateness, and
4. the speaker intends that the audience recognizes the points 1–3.

It is important to note that irony *may*, but need not, result in humor. Moreover, there is no clear-cut differentiation between irony and sarcasm, but Attardo (2000a) describes sarcasm as an “overtly aggressive type of irony”. Irony factors and irony markers (Attardo, 2000b) are essential for communicating irony successfully.

4.4.5.1. Irony factor

The irony factor affects the actual meaning of the utterance. This thesis focuses on *ideational reversal irony*, which “arises as a result of negation of a chosen element of the literally expressed meaning or the pragmatic import of the entire utterance” (Dynel, 2014). As a consequence, it is contextually inappropriate but, at the same time, still relevant. Removing this factor would destroy the irony (Attardo, 2000b) since the negation is the core feature of ideational reversal irony.

For example, if person A dislikes chocolate, an ironic utterance would be “I love chocolate”. Person A is aware of this inappropriate utterance because it does not represent the truth about A’s food preferences. At the same time, it is relevant because it addresses the context of A’s liking.

Table 4.2.: Multimodal markers of irony.

Modality	Markers
Language	Exaggerations and understatements (Attardo, 2000b), positive and negative interjections, onomatopoeic expressions for laughter (Carvalho et al., 2009; Frenda, 2016), quotations and heavy punctuation marks (Attardo, 2000b; Carvalho et al., 2009; Frenda, 2016), ellipsis (Attardo, 2000b), hashtags (Valitutti and Veale, 2015; Hee, Lefever, and Hoste, 2018; Cignarella et al., 2018), emojis (Hee, Lefever, and Hoste, 2018; Cignarella et al., 2018)
Facial expr.	Gaze aversion (Williams, Burns, and Harmon, 2009), winking (Attardo, 2000b; Attardo et al., 2003), rolling eyes, wide open eyes and smiling (Attardo et al., 2003)
Prosody	Intonation and nasalization (Attardo, 2000b; Attardo et al., 2003), stress patterns (Attardo, 2000b), speech rate, extra-long pauses and exaggerated intonational patterns (Attardo et al., 2003)
Gestures	Nudges (Attardo, 2000b)

4.4.5.2. Multimodal Markers

The irony factor alone is typically insufficient since the audience might not identify the resulting utterance as irony per se. For example, a person, who does not know the preferences of person A, might think that A loves chocolate if A says "I love chocolate". Thus, verbal and non-verbal markers make the audience aware of the ironic intention by emphasizing and supporting the effect of the irony factor. They include the manipulation of the linguistic content and additional modalities, such as facial expression, prosody, and gestures. While the use of those markers varies and is highly individual, a deviation from the speaker's usual style of communication indicates the presence of irony. For example, speakers use exaggerations, a modified speech rate, rolling eyes, and more to make the counterpart aware of the ironic intention.

Table 4.2 summarizes common multimodal markers of irony from the literature. Several linguistic markers were identified by Attardo (2000b) and Carvalho et al. (2009), including exaggerations (e.g., "really", "utterly"), understatements (e.g., "barely", "almost"), as well as positive and negative interjections (e.g., "Great!", "Super!", "Damn it!", etc.). Interjections in subjective texts express the author's emotions, feelings, and attitudes (Carvalho et al., 2009). For example, a person who dislikes washing the dishes might use a positive interjection and an additional exaggeration: "*Great!* I *really* love washing the dishes on my own." Furthermore, onomatopoeic expressions for laughter (e.g., "haha"), acronyms, such as "lol" (laughing out loud) or "rofl" (rolling on the floor laughing), emoticons (e.g., ";-)"), quotation and heavy punctuation marks (e.g., "!!!") as well as ellipsis ("…") are used in written language (Carvalho et al., 2009).

Regarding prosody, Attardo et al. (2003) list acoustic parameter modulations used when presenting ironic utterances. These atypical speaking behaviors contrast normal speech modulations in terms of pitch, rhythm, and speech rate. For example, the compressed pitch pattern is characterized by a “flat” intonation, causing minimal pitch range while pronouncing the utterance. In contrast, pronounced pitch accents exaggerate the intonation by accentuating words throughout the whole sentence, certain words, or multiple syllables of the same word. Often, they are combined with elongations and stilted pauses.

Typical facial expressions when communicating irony include raised or lowered eyebrows, wide open eyes, squinting or rolling, winking, smiling or a so-called “blank face”, which is perceived as “expressionless”, “emotionless” and “motionless” (Attardo et al., 2003). In addition, gaze aversion is a typical cue accompanying sarcastic statements (Williams, Burns, and Harmon, 2009).

4.5. Conclusion

This chapter presented the theoretical background for personality, politeness, and humor and the links between them. Personality is often described based on the five-factor model or interpersonal circumplex. Persona means a consistent pattern of behavioral and personality patterns. Both politeness and humor are closely linked to personality. The politeness theory suggests different formulations for expressing positive and negative politeness, with some linguistic markers being closely related to the extraversion dimension of the Five-Factor model. In addition, humor has also been reported as a form of positive politeness. The varied forms of canned and conversational humor rely, i.a., on multimodal verbal and non-verbal markers, which the speaker uses to communicate the humorous intention. At the same time, audiovisual feedback from the interlocutor also contributes to the overall performance.

The expression of personality constitutes a basic element of social robots. On top of that, politeness and humor are important aspects to consider in HRI. Politeness and humor are signs of social intelligence, which should also be taken advantage of in HRI. Similar has been shown in the context of conversational agents, where humor makes interactions more natural and enjoyable and increases credibility and acceptance (Nijholt, 2007).

Building on the presented personality, politeness, and humor foundations, Part III deals with the generation of corresponding multimodal social robot behaviors. Motivated by the varying insights from chapter 6 Part IV presents adaptation approaches, which optimize the generated behaviors to the individual user.

Part II.

Related Work

5. Expressive Social Robots

The generation of artificial verbal and non-verbal behaviors is a key challenge for social robots. A social robot's ability to generate these multimodal behaviors depends on its actuators and software. Apart from typical human behaviors, robots can also use non-verbal sounds, LED lights, or text and graphics on displays for communication. However, the robot's behaviors must be implemented for each machine considering its hardware and software. For example, many robots do not have humanoid extremities, which excludes communication via gestures.

This chapter first gives an overview of *social* robots, their embodiment, and communication channels. It includes important research projects in domestic environments focusing on typical assistive functions. Afterward, the chapter details the generation of multimodal robot behaviors. The focus is on the expression of personality, persona, politeness, and humor with verbal and non-verbal behaviors in the literature. Politeness is essential for preventing face threats (see chapter 8) in sensitive assistive functions, such as health-related advice. Furthermore, the expression of personality, persona, and humor are one opportunity to make the robot's behaviors more believable and interesting. For example, several studies have shown that humor can positively affect interaction and the users' perceptions of robots (Oliveira et al., 2021).

In most cases, human behaviors serve as a reference for the generation of robot behaviors (see chapter 4 for the psychological background and findings from human communication). The robot's multimodal behaviors may be *scripted* (i.e., they are prepared by hand in advance and typically immutable during runtime) or generated *dynamically* during runtime (e.g., based on natural language generation (NLG) techniques). Scripting has the benefit of maximum human control over the outcome and quality. However, it has the drawback that everything must be specified and tuned by hand, which results in much effort and reduced variety. Generative approaches provide more flexibility concerning the produced contents, which may, e.g., take user inputs into account, but often at the expense of less control over the output.

In combination with the psychological background from chapter 4, this chapter serves as a baseline for the generation approaches for robot personality, persona, politeness, and humor in Part III. This chapter also provides an overview of human-robot compatibility, focusing on extraversion and introversion. The variety of findings underlines the importance of adaptation and motivates the proposed real-time adaptation approach in section 14.1. See section 6.6 for limitations and research gaps.

Parts of this chapter were presented in Ritschel and André (2017), Ritschel, Baur, and André (2017a), Ritschel, Baur, and André (2017b), Ritschel (2018), Ritschel et al. (2019d), Ritschel and André (2018), Ritschel et al. (2020a), Ritschel et al. (2020b), Ritschel et al. (2019b), Ritschel et al. (2019a), Ritschel, Kiderle, and André (2021), Weber et al. (2018a), and Kiderle et al. (2021). The contents of this chapter expand these publications.

5.1. Social Robots

Dautenhahn, 1998 initially came up with a definition for *socially intelligent agents*. An *agent* can be a “computation unit”, a “software tool” or a “life-like, believable agent”, including virtual characters and humanoid robots, which are *embodied* agents. According to Dautenhahn’s definition *social* agents can be any “artificial agents which show elements of human-style social interaction and behavior”. The term *socially interactive robots* is used in Fong, Nourbakhsh, and Dautenhahn, 2003. They start with the historical perspective, originating in the multiagent research area, which begins with biologically inspired robots and organic computing techniques back in the 90s. These experiments address robot-robot interaction exclusively without human involvement, such as for synchronization tasks. Traditionally, the term *social robot* is not necessarily restricted to the HRI domain. Later, Feil-Seifer and Mataric, 2005 use the term *socially assistive robotics* to describe robots whose main task is to assist users with social interaction instead of physical help. Social robots are used in many application domains, including health care and therapy, education, toys, and entertainment, in work environments, malls, public spaces, and at home.

Within the scope of the work at hand, the term *social robot* refers to an autonomous agent with a physical embodiment for interaction and communication with the *human* in a social manner, which is most closely related to Dautenhahn ’s definition. This thesis explicitly focuses on the HRI domain, specifically on entertainment and assistance, which represent typical application scenarios in domestic environments.

5.1.1. Social Interface and Social Awareness

Breazeal, 2003 describes social robots as machines with three basic tasks: *perception* of the environment, *decision-making* and *execution of actions* to carry out a task. These three aspects are in particular important in chapter 6 and Part IV. Furthermore, Breazeal points out that a social robot should

1. be *socially evocative*, i.e., encourage people to interact with it due to anthropomorphization,
2. provide a *social interface*, i.e., use multimodal interaction channels for human-like communication,
3. be *socially receptive*, i.e., learn from human social cues and behaviors, and
4. be *sociable*, i.e., “pro-actively engage people in a social manner” to help the user and to pursue the robot’s aims.

Figure 5.1 illustrates the interrelation. The social interface and the robot’s awareness of the user’s verbal and non-verbal behaviors are essential. The former allows the machine for multimodal communication, i.e., both user and robot use familiar verbal and non-verbal behaviors to communicate, which is encouraged by the robot’s socially evocative embodiment. The latter is realized by processing and interpreting the user’s behaviors; the resulting information allows for robot learning, such as adapting the

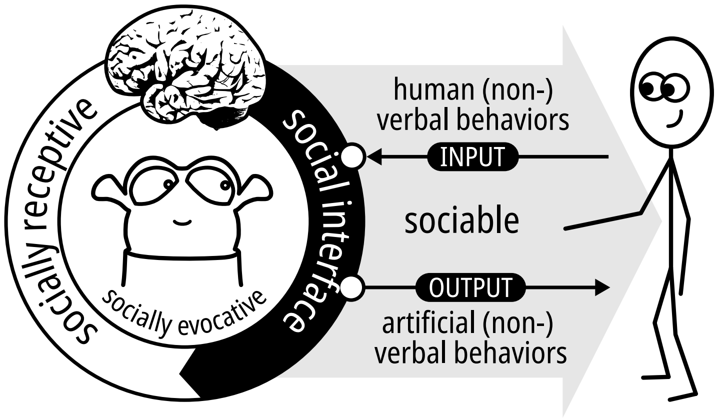


Figure 5.1.: The robot's social interface offers bidirectional multimodal human-robot communication. At the same time, the robot reacts to the sensed verbal and non-verbal user behaviors, e.g., by adapting its generated multimodal output accordingly.

robot's behaviors (see chapter 6). All together serves the purpose of implementing sociable interaction.

The following sections focus on points 1 and 2 from the list above: socially evocative robots and the implementation of their social interface. The overview includes social robot embodiment, input and output modalities for social robots sensing and producing human or humanoid verbal and non-verbal behaviors, and existing behavior generation approaches based on these communication channels. Building on this, chapter 6 focuses on point 3, i.e., existing approaches for adapting the presented social robot behaviors.

5.1.2. Embodiment

The robot's embodiment is the first step toward socially evocative behavior. Often, social robots use an anthropomorphic or zoomorphic design inspired by humans or animals. Typically, they have a face with stylized eyes, eyebrows, a nose, mouth, and ears, which are sometimes supplemented by extremities and wheels. Figure 5.2 illustrates exemplary robotic platforms which characterize the social robot research domain. Pepper, NAO, and Nexi are humanoid robots with many motorized limbs. Furhat's face is rendered and projected on a translucent three-dimensional mask. Paro's embodiment is inspired by the harp seal; its surface consists of fur. AIBO, iCat, and the MiRo robot use a zoomorphic design. Reeti aims to represent an extraterrestrial species. As listed before, research experiments use these robots in various application domains.

With advancements in hardware, computational power, and affordable prices, first robots and robotic toys have also appeared on the consumer market for the general population. Recent consumer products are illustrated in Figure 5.3. The Mabu robot has motorized eyes. The other robots use displays for rendering the face. ElliQ (which consists of a tablet and a charging station with a small robot) and Jibo use abstracted faces with only one eye. Mabu, ElliQ, Pillo (which dispenses pills through an opening on the underside), Moxie, Miko, Buddy, and Astro are promoted as robotic domestic companions (see section 5.1.5).

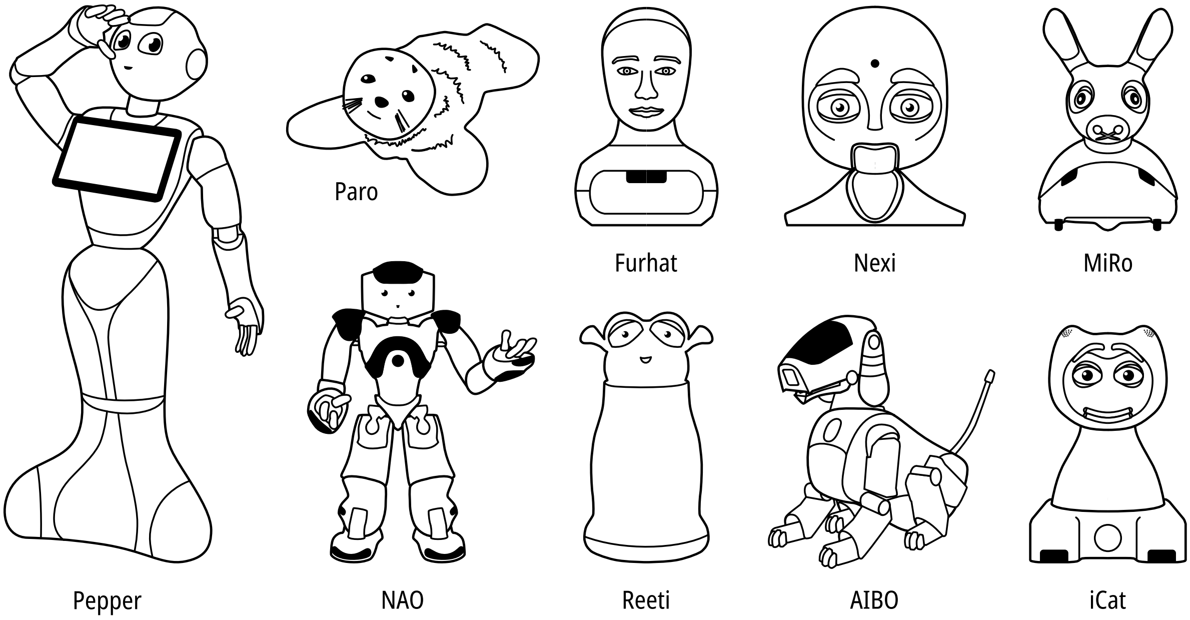


Figure 5.2.: Exemplary robots used in research.

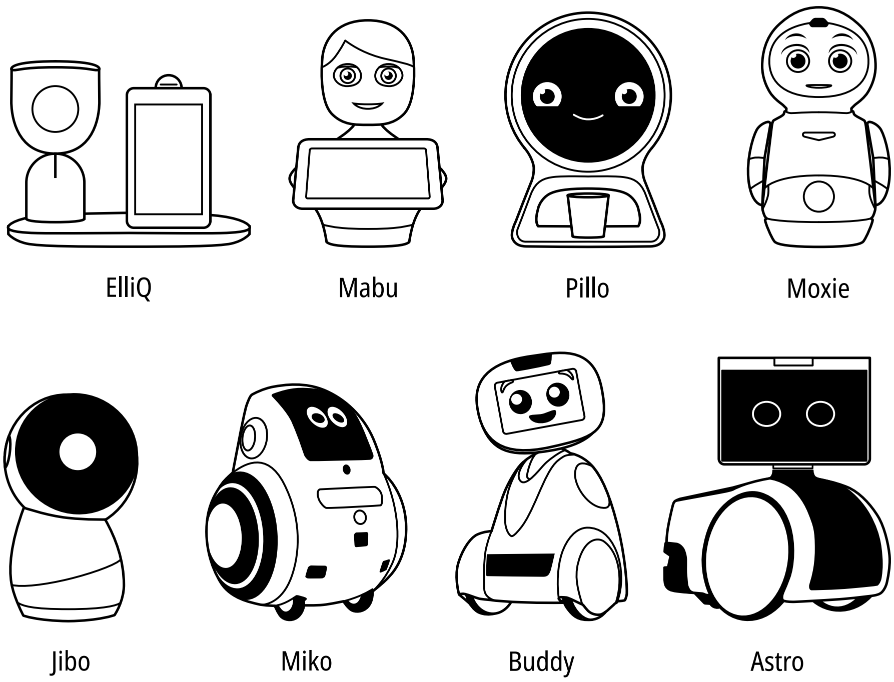


Figure 5.3.: Exemplary consumer companion robots.

5.1.3. Input Modalities

Providing a social interface requires the robot to sense human verbal and non-verbal communication channels. Corresponding hardware sensors, such as microphones, 2D cameras, and touch sensors, are required. They are essential for SSP (see section 3.4), since they allow the robot to sense the modalities found in human interaction: speech, non-verbal vocal cues, gaze, facial expression, gestures, posture and movements (see section 3.3). On top of that, additional sensors are required for discrete input, navigation, system monitoring, and more, including buttons, switches, depth cameras, sonars, inertial (e.g., gyroscope, accelerometer), temperature sensors, and more. See Part IV for more details on including the sensed data from the environment and human social signals in the RL process.

5.1.4. Output Modalities

Besides sensing and processing human input, the robot must also generate multimodal behaviors. As can be seen in Figure 5.2 and Figure 5.3 most robots offer only a few output modalities. Many social robots are *appearance-constrained*, i.e., their embodiment is limited, so they cannot communicate with gestures or facial expressions (Bethel and Murphy, 2008). Colored light, motion, and sound are the most frequently available output modalities of commercially available robots and research prototypes (Löffler, Schmidt, and Tscharn, 2018).

5.1.4.1. Gaze and Facial Expression

Figure 5.2 and Figure 5.3 provide an overview of selected social robot platforms. Almost all robots have eyes, be it in the form of a projection on a screen, LED lights, or motorized actuators. Many robots also have a mouth, nose, and sometimes eyebrows. The degree of realism varies – stylization is not only used for an appealing design and to shape the individual robot's personality profile but also for avoiding the *uncanny valley* (Mori, 1970). Many of the presented robots produce more or less sophisticated facial expressions and gaze behaviors, such as the Furhat, Nexi, MiRo, iCat, Reeti, Mabu, Pillo, Moxie, Jibo, Miko, Buddy, and Astro robots. The Pepper, NAO, and ElliQ robot cannot move but light up their eyes. Nexi, Reeti, iCat, and Mabu have motors for moving their eyes; most others use displays to render their faces.

5.1.4.2. Gestures and Posture

Some social robots have limbs for expressing gestures, such as the Pepper, NAO, AIBO, and Moxie robots. Depending on the degrees of freedom, this causes additional cost and complexity, such as implementing inverse kinematics and preventing self-collision. In addition to gestures, selected robots produce different postures. For example, Pepper, NAO, Nexi, AIBO, and Jibo can move parts of their bodies to express postures; e.g., the NAO robot can sit and stand. The Jibo robot's body consists of twistable parts for looking upwards and downwards physically, which allows for expressing, e.g., different emotions in combination with the virtual eye.

5.1.4.3. Movement

Many social robots are placed stationary in a room, requiring the user to go to the robot, not vice versa. Mobile platforms, such as Pepper, MiRo, Miko, Buddy, and Astro, have wheels for locomotion. NAO and AIBO can walk with their legs. Movement requires additional hardware and software for steering, localization and mapping, navigation, collision avoidance, and the targeted use of interpersonal distance and proxemics.

5.1.4.4. Speech, Prosody, and Sound

Spoken language often communicates complex information. Text-to-speech (TTS) software implements speech synthesis, which converts text into audio. Typically, several voices with individual sounds are available. Modern TTS systems take instructions into account for controlling the prosody, i.e., for increasing or decreasing pitch, speech rate, inserting breaks, vocal gestures, and more. In addition, a robot's sound design may also include *non-verbal sounds*, which express messages in a short time independently of any particular language (de Gorostiza Luengo et al., 2017). They communicate, i.a., a robot's affective and internal state, approval and disapproval, hesitation, hush, summon, encouragement, greeting, laughing, and more (de Gorostiza Luengo et al., 2017; Ritschel et al., 2019a). Bethel and Murphy (2008) point out that the sonic interaction design of appearance-constrained robots should use robot-specific social cues, which do not anthropomorphize through human or animal sounds and tones.

5.1.4.5. Lights and Displays

LED lights often visualize the robot's internal state (e.g., Pepper and NAO light up their chest button in different colors depending on battery state). In some cases, however, LEDs also replace hardware actuators. For example, the Pepper and NAO robots do not have motorized or rendered eyes but use several LED lights. One can use animations, e.g., for mimicking eye blinks.

As can be seen in Figure 5.2 and Figure 5.3, some social robots also use displays for presenting additional information to the user or even for rendering parts of their embodiment, such as the face. The advantage is that graphical and textual output can also be provided simultaneously to the interaction with the other communication modalities.

5.1.5. Domestic Companion Robots

In the last decades, research experiments have explored the use of robots in users' domestic environments, especially in assistive, health-related support for cognitive tasks or to improve mental and emotional well-being. There is also increasing interest and development of robotic consumer products (see Figure 5.3). Typical functions replace or complement those already known by today's smartphones and smart speakers: information retrieval, communication with relatives or medical professionals via voice or video chat, notification, reminders of appointments, entertainment (e.g., playing games), home surveillance, smart home integration, and health advice. Interaction with the user often happens via a voice interface, a mobile app, or a touch display.

Some commercial consumer products focus on health-related functions. For example, Pillo (Tao, Moy, and Amirfar, 2016) monitors and dispenses medication to family members, reminds the user of the care plan, health-related appointments, and answers questions concerning the nutritional value of food. Pillo uses facial recognition to identify and track family members. Basic entertainment and communication features include playing radio and video calls with medical professionals or family members. ElliQ and Mabu offer similar functions and target the elderly population. Mobile robots, such as Miko, Buddy, and Astro, focus, i.a., on easing everyday life, such as video chatting with relatives while following people around. The Astro robot includes a telescope camera and allows the user to monitor the home remotely. As outlined in section 5.1.2 much effort is put into the visual appearance, embodiment, motion and sound of consumer devices, as is the case for robotic toys (e.g. *Cozmo*, *Miko*), to communicate intentions and to portray certain personality stereotypes (see section 5.2.1) and emotions.

Kidd and Breazeal (2007) present a robotic weight loss coach for long-term, supportive care. It carries out short conversations about daily diet and exercise goals. The user reports the nutritional intake and number of conducted exercises. The robot provides feedback, advice, and suggestions. The results of a study (Kidd and Breazeal, 2008), which compares the robot with a touch-screen computer and a paper log for tracking calorie intake and the number of exercises, show a significantly higher long-term motivation for the robot and that users formed the strongest “working alliance” with it.

The *CompanionAble* project (Schroeter et al., 2013) presents an assistive robot for people suffering from mild cognitive impairment. The mobile robot uses smart home technologies to provide social and cognitive support. Functions include reminders of appointments and activities, video calls, storage of personal items, and a cognitive stimulation game addressing the impairments of the elderly target group. Smart home integration allows tracking the user, controlling curtains and lights, and triggering situation-specific reminders when the user enters or leaves home to notify about missed calls, agenda items, or things to remember. The system uses a graphical user interface (GUI) on the robot’s screen and a set of recognized dialog phrases for interaction.

In Al-Taee et al. (2017), a domestic robot supports diabetes patients. The authors present an eHealth platform, which combines the NAO robot with several Internet of Things (IoT) technologies. Several medical sensors are connected with online health care and disease management hub to provide diabetes management for children. A core concept is an adaptable and participatory design involving the children in their treatment plans and personalizing the health profile. The robot performs dialogs created by specialist clinicians with the patient to empower and motivate toward a healthy lifestyle. The process allows for connecting the children and their caregivers over a distance instead of periodic clinic visits.

The *AlwaysOn* project (Sidner et al., 2018) supports elderlies living on their own with a social companion, which is embodied either by a virtual character or a Reeti robot. The companion offers several activities, including counseling the user on diet and exercise for the user’s well-being, entertainment (e.g., games), and communication with other people. One important feature is the included approach for developing a relationship between the user and the artificial companion. For instance, the system schedules shared activities based on the current strength of their social bond, avoiding sensitive topics (such as health advice) in the early stages (Rich et al., 2012). The agent also actively works to improve the relationship by suggesting activities to increase perceived closeness. It furthermore employs strategies, such as interleaving advice with social interaction, for example, by discussing the topic of exercise during a game of cards. It is "always on", recognizes the human with face detection, and initiates interactions on its own, too.

Cosar et al. (2020) present the ENRICHME system, which provides technologies for health monitoring, social support, and care to enrich the day-to-day experience of elderlies with mild cognitive impairment in their domestic environments. Their platform integrates a TIAGo robot with ambient intelligence and smart home sensors. It connects to a care platform over the internet. The robot identifies and tracks users based on face detection, biometric features, and objects based on RFID tags. Moreover, the robot monitors physiological data, including body temperature, respiration rate, and heartbeat rate. For example, it recognizes changes in temperature on images from the thermal camera. The robot has several functions, including cognitive games, healthy tips, physical activities, finding objects, monitoring environmental data, displaying the agenda, video calls, weather forecasts, and news.

5.2. Personality

Robert et al. (2019) give an overview of personality in HRI, including human personality, robot personality, similarities, differences, and factors impacting robot personality. In contrast, the section at hand first details the expression of the Big Five extraversion-introversion personality dimension (see section 4.1.1) with robot verbal (primarily) and non-verbal behaviors. Afterward, it focuses on findings concerning human-robot personality compatibility. Both subsections present selected literature, which is, for the most part, also listed in Robert et al. (2019). Table 5.1 provides an overview of these works.

5.2.1. Extraversion and Introversion

The expression of extraversion and introversion with robots is based on verbal and non-verbal behaviors observed in human interactions (see section 4.1.1). Often, linguistic or prosodic markers are combined with gestures or movements and, sometimes, gaze and proxemics. Attempts were also made for training models by users (e.g., Cruz-Maya and Tapus (2017), see below). Many works also investigate human-robot compatibility. These results are presented in section 5.2.2.

5.2.1.1. Speech, Prosody, Facial Expression, Gestures, and Movements

Lee et al. (2006) use a variety of verbal (volume, pitch, pitch range, speech rate; volume, rhythm, musical scale range, and mood of melodies) and non-verbal cues (frequency and color of facial expressions via LEDs, angle, speed and frequency of movements) for expressing extraversion and introversion with an AIBO robot. The extravert robot uses higher voice pitch, speech rate, and volume, melodies with higher volume, faster rhythm, a wider range of musical scales, and an exciting mood. It also uses colorful flashing LED lights for facial expression and wider movements with higher speed, frequent tail

Table 5.1: Expression of robot extraversion-introversion and human-robot extraversion-introversion compatibility.

	Reference	Focus	Communication channels									Robot	
	Aly and Tapus, 2013; Aly and Tapus, 2016	verbal & non-verbal behavior generation	✓	○	○	○	✓	○	○	✓	en	NAO	S
	Andrist, Mutlu, and Tapus, 2015	gaze models, (mutual) gaze duration	○	○	✓	○	○	○	○	○	fr/en	Meka	S
	Çeliktutan and Gunes, 2015	gaze, attention, head movement	✓	✓	○	○	✓	○	○	○	en	NAO	(S)
✱	Craenen et al., 2018b; Craenen et al., 2018a i	gesture speed/amplitude	○	○	○	○	✓	○	○	○	-	Pepper	S/C
	Cruz-Maya and Tapus, 2017	distance, gesture speed & amplitude	○	○	○	○	✓	○	✓	○	-	Pepper	(S)/(C)/O
	So, Kim, and Oh, 2008	thinking/feeling	✓	✓	○	○	○	○	○	○	?	I-Robi	(S)/O
	Joosse et al., 2013	task context	○	✓	○	○	✓	✓	○	○	da	NAO	O
✱	Jung et al., 2012	eye contact/opening	○	○	✓	✓	○	○	○	○	-	KMC-EXPR	S
	Lee et al., 2006	voice, melodies & non-verbal cues	○	✓	○	✓	✓	○	○	○	en	AIBO	C
	Mileounis, Cuijpers, and Barakova, 2015	dominance and extraversion	✓	✓	○	○	✓	○	○	○	en	NAO	(S)
	Niculescu et al., 2013	voice pitch, humor and empathy	○	✓	○	○	○	○	○	○	en	Olivia	S
	Salam et al., 2017	individual and group engagement	○	✓	○	○	✓	○	○	○	?	NAO	S

Continued on next page

Table 5.1: Expression of robot extraversion-introversion and human-robot extraversion-introversion compatibility. (Continued)

part of the work at hand	Reference	Focus	Communication channels: speech	prosody	gaze	facial expression	gestures/movements	movement	proxemics	dynamically generated robot behaviors	language	Robot	observed
	Tapus and Mataric, 2008; Tapus, Tapus, and Mataric, 2008	RL of interaction parameters	✓	✓	○	○	○	✓	✓	○	en	Pioneer 2-DX	S
	Woods et al., 2005 ℹ	socially ignorant/interactive	✓	○	○	○	○	✓	○	○	en	PeopleBot	C
🎓	Ritschel and André, 2017; Ritschel, 2018	personalization with RL	✓	○	○	○	○	○	○	○	en	Reeti	-
🎓	Ritschel, Baur, and André, 2017a; Ritschel, Baur, and André, 2017b	NLG & personalization with RL	✓	○	○	○	○	○	○	✓	en	Reeti	-

Legend: 🎓 part of the work at hand ✱ not listed in Robert et al. (2019) speech prosody gaze facial expression gestures/movements movement proxemics dynamically generated robot behaviors language observed human-robot personality compatibility (*S*imilarity/*C*omplementarity attraction/*O*ther findings) ℹ no expression of extravert/introvert robot behaviors wagging, and random walking. In contrast, the introvert robot does the opposite (e.g., lower pitch/frequencies/volume, less movement) when expressing introversion. There is no manipulation of linguistic content.

So, Kim, and Oh (2008) rely on the Myers-Briggs Type Indicator (MBTI) personality model and express four personality types with an I-Robi personal service robot: extraversion-thinking (ET), extraversion-feeling (EF), introversion-thinking (IT) and introversion-feeling (IF). The MBTI model understands *thinking* as an "intellectual activity in which judgments are based on the rational application of principles" and *feeling* as the "assignment of value (acceptance or rejection) to objects of experience" (McCrae and Costa Jr, 1989). The robot uses speech (linguistic content, expression of feelings) and prosody (speech rate, volume, pitch, pitch rate). Differences include that the ET/EF robot speaks loudly, fast, and with a higher pitch; the IT/IF robot speaks silently, slowly, and with a lower pitch. The ET/IT robot speaks only about one or selected subjects; the EF robot talks freely; the IF robot has difficulties in starting a conversation and talks about emotions and relationships afterward. While the expression of feelings or emotions for the EF/IF robot is abundant, the ET/EF robot uses less expression or variation of feelings or emotions. A TTS system renders the scripted speech output. It is post-processed by hand for tweaking the prosody.

Niculescu et al. (2013) configure the robot Olivia in two variants. The extravert (introvert) uses a higher (lower) voice pitch. Speech rate, timbre, and volume are configured similarly. The extravert robot is also more exuberant and uses humor; the introvert is calmer and more serious. Loquendo TTS produces the robot's speech.

Joosse et al. (2013) conduct a study with the NAO robot with two tasks. They use scripts including spoken language via TTS, gestures, and movements for expressing extraversion and introversion, either with the robot acting as a tour guide or a cleaning robot. The extravert robot talks loudly with a more varied pitch and higher speech rate and with larger, faster, and more frequent body movements and gestures. Moreover, it talks more than the introvert, which uses standard volume and slower speech rate and bows its head down slightly when talking.

Çeliktutan and Gunes (2015) focus on attention and head movement for expressing extraversion and introversion with the NAO robot. The extravert robot uses hand gestures and posture shifts and talks fast and loud with a higher pitch ("Would you like me to dance for you?", "It is amazingly exciting!"). The introvert uses less energetic and hesitant speech with lower pitch ("Hmm …well, ok…would you like me to play music for you?", "Well good…"), without any hand gestures and almost static posture. Similar paralinguistic and non-verbal cues for extraversion and introversion are also used in Salam et al. (2017) but without verbal differences.

Mileounis, Cuijpers, and Barakova (2015) express the two personality dimensions extraversion-introversion and dominance-submissiveness with the NAO robot in a "Who wants to be a millionaire" game. The extravert robot is more talkative and uses a higher speech rate and more gestures than the introvert robot. Moreover, the extravert robot shows emotions; the introvert does not. Voice pitch and speech content match its dominance: the dominant robot uses strong arguments with confident language and lower voice frequency; the submissive robot uses arguments but expresses uncertainty and uses higher voice frequency.

5.2.1.2. Gaze

Jung et al. (2012) express extraversion and introversion with the KMC-EXPR robotic face in three intensities (extravert, intermediate, introvert) based on different facial expressions. The extravert robot has wide open eyes and maintains direct eye contact with the user. The introvert robot opens its eyes less wide and does not always maintain eye contact but looks down a little.

Andrist, Mutlu, and Tapus (2015) collect data from human-human interactions for developing personality-expressive gaze behaviors. They extract an introvert and an extravert gaze model from annotated puzzle tasks and implement both models for the Meka robot. The extravert robot gazes into the user's face more; the introvert robot looks at the task space more. Moreover, the gaze durations are different. For example, the extravert robot maintains the mutual gaze for longer.

5.2.1.3. Proxemics

Tapus, Tapus, and Mataric (2008) use a mobile Pioneer 2-DX robot in the context of post-stroke rehabilitation. Three parameters express extraversion and introversion: proxemics (interaction spaces defined by Hall (1966)), movement speed, linguistic (choice of words), and paralinguistic (pitch and volume) behaviors. The authors adapt the parameters based on the user's task performance (see section 6.5). The extravert robot uses challenging language (e.g., "You can do it!") with high pitch and volume. The introvert robot uses nurturing, gentle and supportive language (e.g., "Very nice, keep up the good work.").

In Cruz-Maya and Tapus (2017), users train a Pepper robot's waving behavior. The first study participants' task is to personalize the speed and amplitude of the robot's left arm greeting gesture and the personal distance to their individual preferences over several days. After the training, the authors test the trained robot behavior model in a second study with respect to participants' personalities and gender. For this purpose, they use the trained model's minimum and maximum values for expressing introversion and extraversion. The extravert robot uses a much larger interaction distance (1.20 m), gesture amplitude and speed (100 %, i.e., maximum possible values) than the introvert robot (0.40 m, 20 % amplitude and 40 % speed). The authors do not manipulate the linguistic content for the robot's expression of personality.

5.2.1.4. Dynamic Generation

Aly and Tapus (2013) and Aly and Tapus (2016) present an architecture for generating synchronized verbal and non-verbal NAO robot behaviors according to the Big Five personality model (see section 4.1.1). It is based on the PERSONAGE system by Mairesse and Walker (2011), which uses NLG to produce configurable restaurant descriptions and comparisons according to the five personality traits dimensions. The BEAT toolkit (Cassell, Vilhjálmsson, and Bickmore, 2004) generates animations in terms of gestures (iconics, metaphorics, see section 3.3.1), posture shifts, and gaze for the generated utterances. It performs linguistic and contextual analysis of the generated text and produces gestures with corresponding amplitude, direction, rate, and speed. On top of the personality expression, the architecture can also estimate the user's personality by analyzing speech input. Thus, the robot attempts to match the individual user's personality: first, it estimates the user's personality traits based on a psycholinguistic analysis of the spoken language. Then, the robot generates verbal and non-verbal behaviors that fit the characteristics of the human's personality traits.

5.2.2. Human-Robot Compatibility

As outlined in section 4.2, interpersonal compatibility describes the compatibility between different peoples' attitudes and personality traits. In HRI, the majority reports similarity attraction effects with regard to extraversion and introversion. Other findings include complementarity attraction and dependence on the task context.

5.2.2.1. Similarity Attraction

Tapus, Tapus, and Mataric (2008) investigate user-robot personality matching in their post-stroke rehabilitation scenario. They examine individual user preferences regarding the assistive therapist robot's interaction parameters by adapting proxemics, speed, and verbal and non-verbal behaviors during training to user task performance (see section 6.5). Their study results show a similarity attraction effect since participants preferred personality matching in the assistive domain.

Similarity attraction is also observed in Jung et al. (2012) with a robotic face expressing extravert, intermediate, and introvert facial expressions and gaze. The authors observe that extravert (introvert) people felt more friendliness and liking towards an extravert (introvert) robot.

Aly and Tapus (2013) and Aly and Tapus (2016) investigate the similarity attraction principle for synchronized generation of verbal and non-verbal behaviors for the NAO robot. By mapping the user's sensed personality profile to the robot's generated behaviors, the robot adapts its behaviors to the user's personality. Results of the experiment validate that most participants preferred to interact with robots with the same personality as their own. Another finding is that adapting the robot's verbal and non-verbal behaviors was perceived as more natural and engaging than the adapted speech-only behavior.

Niculescu et al. (2013) compare an extravert and introvert robot, which differ in voice pitch, humor, and talking style. Results of their Wizard of Oz (WoZ) study show that introvert people prefer interacting with introvert robots. Introvert participants found the interaction much easier and perceived the robot's behaviors as more empathetic than extravert participants, which confirms the similarity attraction effect.

The WoZ study by Mileounis, Cuijpers, and Barakova (2015) investigates the effect of the two personality dimensions extraversion-introversion and dominance-submissiveness on the perceived social intelligence of the NAO robot. They observed that participants perceived the extravert robot as more socially intelligent, likable, animate, intelligent, and emotionally expressive. Similarly, they perceived the submissive robot as more socially intelligent, likable, and emotionally expressive than the dominant one. Moreover, female participants perceived the extravert robot as more emotionally expressive and lifelike than male participants. However, the authors support a similarity attraction effect partly due to non-significant results.

In the WoZ experiment by Çeliktutan and Gunes (2015), an introvert and extravert NAO robot ask participants personal questions. Results of their show that an extravert robot can positively affect the interaction experience, which is in line with the similarity attraction principle. However, the authors did not observe this attraction effect for the introvert robot.

Andrist, Mutlu, and Tapus (2015) evaluate their introvert and extravert gaze models with the Meka robot. In their study, participants collaborate with the robot on a puzzle task. The results show that the extravert gaze model with more frequent gaze behavior toward the user made users perceive the robot as more extravert than the introvert model with shorter and scarcer eye contact. The results show that people without intrinsic motivation to solve the puzzle spent significantly more time collaborating with a robot whose extraversion level was similar to their own and chose to solve more puzzle tasks together.

Salam et al. (2017) investigate individual and group engagement in an interaction with an introvert or extravert NAO robot. They measure involvement, interest, and enjoyment during the interaction. Their results show higher levels of user engagement when the robot's degree of extraversion matches the user's personality profile.

5.2.2.2. Complementarity Attraction

Woods et al. (2005) found out that extraversion plays an important role when users evaluate a robot's personality and assess to what extent it matches their personality. In their WoZ study, they use a PeopleBot robot with either a socially interactive or a socially ignorant behavior style, which differs in its navigation path, speed, orientation, actions, and speech. Results from the study show that subjects rated themselves as being significantly more social than the robot and assigned themselves stronger personality characteristics, regardless of the robot's personality. However, extraversion was identified as an important factor when evaluating the similarity between the test persons' and the robot's personalities. Woods et al. point out that the experiment indicated that users "do not tend to assign their personality traits to match the robot's ones".

Lee et al. (2006) express extraversion and introversion with the AIBO robot. Study results show that participants correctly recognized the robot's personality. Moreover, the authors observed a complementarity attraction effect: extravert participants enjoyed and rated the robot's intelligence and social attraction higher when interacting with an introvert robot and vice versa.

5.2.2.3. Mixed and Other Findings

So, Kim, and Oh (2008) report on a study based on the MBTI personality model. They conclude that participants prefer kind, friendly, and mild personal service robots independently of the participants' extraversion and introversion. Moreover, their results show that test persons had preferences for extravert or introvert robots regardless of the correlation between their own and the robot's personality. Although the authors observed similarity attraction for certain combinations (e.g., EF/IF people preferred EFIF behaviors, but ET people preferred EF) when a computer presented speech, these observations were not observed when presented by the robot. Finally, they conclude that personal service robots do not need a fixed personality type. Instead, expressed personality may vary depending on the task, as long as the differences are not too big.

Joosse et al. (2013) identify the task context as a potential determining factor for similarity or complementarity attraction. Results of their study with the NAO robot acting either as a tour guide or cleaner show that extravert people trusted the extravert robot more than the introvert robot when it acted as a tour guide, which the authors did not observe for introvert participants. Moreover, introvert participants trusted the extravert robot slightly more than the introvert robot when it acted as a cleaner, which was not observed for extravert participants. From that Joosse et al. conclude that the attraction effects may depend on the task context and that the robot's behaviors need to be adapted "to the users' expectations about what kind of personality and behaviors are consistent with such a task or role".

Cruz-Maya and Tapus (2017) present the evaluation of their experiment with the trained model of waving gestures of a Pepper robot. According to the model, extravert female participants would prefer a closer distance and gestures with higher amplitude than males with the same personality. The model also predicts that male participants would prefer a faster speed for the gestures. The evaluation partially confirms a similarity attraction effect since the model did not fit all the parameters.

Craenen et al. (2018b) and Craenen et al. (2018a) investigate 45 robot gestures (five core gestures with variations in speed and amplitude). The authors compare the 30 participants' personality profiles with their ratings of the perceived robot personality of each gesture. A similarity attraction effect was observed for 15 participants, and a complementarity attraction effect for 9 participants. In particular, Craenen et al. identify openness as the least important personality dimension concerning the two attraction effects. Conscientiousness, extraversion, agreeableness, and neuroticism were similarly relevant. In their conclusion, the authors point out that there would be a need to predict the user's preferred attraction effect to adapt the robot behaviors accordingly. They conclude that the similarity and complementarity attraction paradigm might also be observed for other factors and modalities, such as gender and verbal communication, including the robot's way of speaking.

5.2.3. Persona

Apart from the expression of personality traits, the literature has also investigated how to equip social robots with personas (see section 4.1.3). In HRI, persona is typically understood as "perceived or evident personalities", which are expressed with identifiable and expressible traits (Lacey and Caudwell, 2018), including habitual patterns of thought, emotion, and behavior (Ruckert, 2011). One benefit of equipping robots with persona is to provide the user with a clear mental model, which helps to make sense of and anticipate the robot's behavior (Lacey and Caudwell, 2018). Researchers use several modalities to produce stable behavioral and personality patterns for investigating the impact of different personas on users and their preferences. Table 5.2 provides an overview of experiments investigating different personas for domestic companion robots.

Dautenhahn et al. (2005) present study results from HRI trials with a non-humanoid PeopleBot robot in a simulated living room. The questionnaires address, i.a., the preferred

Table 5.2: Different personas in the context of domestic companion robots.

			Communic. channels								
🎓	Reference	Personas	💬	♫	☺	✥	💡	⟳	🪄	🌐	Robot
	Bartl et al., 2016	*companion* vs. assistant	✓	○	✓	○	○	○	✓	de	Reeti
	Dautenhahn et al., 2005 ℹ	*assistant* vs. *butler* vs. friend	○	○	○	○	○	○	○	en	PeopleBot
	Huijnen et al., 2011 ℹ	butler vs. helper vs. entertainer vs. guardian angel	○	○	○	○	○	○	✓	?	Hector, Kompaï
	Whittaker et al., 2021	butler vs. *buddy* vs. sidekick	✓	✓	○	✓	✓	○	✓	en	Olly
🎓	Ritschel et al., 2019d; Ritschel et al., 2019c	companion vs. *assistant*; *mentor* vs. opponent	✓	○	○	○	○	○	○	de	Reeti

Legend: 🎓 part of the work at hand 💬 speech ♫ prosody ☺ facial expression ✥ motion 💡 light ⟳ dynamically generated robot behaviors 🪄 Wizard of Oz study 🌐 language ℹ no (details on) expression of persona *emphasized* personas were preferred by study participants robot persona, coming to the conclusion that most participants prefer the *assistant* or *machine* over the *butler*. Only young participants were interested in the robot acting as a *friend*. Key expectations regarding the robot's behavior traits include high predictability, controllability, and considerate behavior. The paper does not provide details on how the robot communicates with the test persons during the conducted HRI trials.

Huijnen et al. (2011) conclude from their video and WoZ studies that different users have different preferences regarding the role, responsibilities, and type of persona expressed by companion robots and that there is a need to adapt the robot's character accordingly. Their results also indicate that apart from the robot's function, its character and interaction style is significantly more important than its design and physical embodiment. The paper does not provide details on how the robot communicates with the test persons during the conducted HRI trials.

Bartl et al. (2016) investigate the impact of robot persona on user acceptance in the context of a social robot for the elderly. A Reeti robot expresses either a *companion* or *assistant* persona, which differ in speech and facial expression. The companion persona uses a prevalent friendly, emotional, enthusiastic, and expressive speech; the assistant persona uses more authoritative and formal spoken language. Both personas use a distinct configuration of linguistic features, such as fillers, questions, words of agreement, pronouns, and more, as well as different facial expressions (smiles) and poses (head tilt). Bartl et al. evaluate their approach with German native speakers. Their scenario includes reminders about appointments in a calendar application. Results of their WoZ experiments indicate that the companion persona is preferred and positively impacts the robot's likability and perceived intelligence. Moreover, the authors report on the observed novelty effect and desired additional functions, such as recommendations for drinking water regularly.

Whittaker et al. (2021) investigate three personas with a non-anthropomorphic Olly robot. They use speech, prosody, motion and light to express *buddy*, *butler* and *sidekick* personas. The personas differ, i.a., in terms of activity, spontaneity, reliability, sensitivity, autonomy, helpfulness, and focus of the robot. For example, the buddy is talkative and has a good sense of humor; the butler speaks in clear pronunciation, is officious and serious; the sidekick is shy and speaks slowly and intentionally. The attributes are mapped to TTS parameters (volume, pitch variation, amplitude), the motion of the robot (movement speed, pause duration, attentional orientation to the user when they speak), and colored LED light animations (number of colors, color scheme, several particle animation parameters). Results of the WoZ study show that most participants preferred the buddy persona, and the sidekick persona was the least popular.

5.3. Politeness

The HRI literature primarily relies on the politeness theory by Brown and Levinson (1987) (see section 4.3) and spoken language for expressing politeness with robots. There is work building on related frameworks, such as the *politeness maxims* by Leech (2016) or the impoliteness framework by Culpeper (1996). There are also insights regarding politeness towards the robot, such as the usage of polite wake words (e.g., "Excuse me, Robot"), which may prime users to interact more politely with the machine (Williams et al., 2020). An overview of important works about politeness in HRI is given in Table 5.3. The following sections focus on the theory by Brown and Levinson and the expression of politeness with robots towards the user.

5.3.1. Politeness Strategies

Lee et al. (2011) focus on using positive, negative, and no politeness to mitigate robot malfunctions. In their WoZ study, the OTTORO-S vacuum cleaning robot uses a female TTS voice to communicate four events: the start of the cleaning, a malfunction when being tucked by an obstacle, the removal of the obstacle by the user and the completion of the cleaning process. Scripts with three message variations exist for each event. The variations with no politeness use very short utterances consisting of two words, such as "Barrier detected". Variations with negative politeness include being pessimistic, minimizing the imposition, and apologizing. Variations with positive politeness include attending to the user's interests, needs and wants, offering a promise, and using exaggerated interest. Lee et al. conclude from the study results that the negative impact of robot malfunction on users' impressions can be reduced by using a positive or no politeness strategy, depending on the user's relational or utilitarian orientation. All in all, participants preferred a (primarily positively) polite robot.

Salem, Ziadee, and Sakr (2013) investigate the interplay between Brown and Levinson 's positive politeness or bald on-record (no politeness) strategies and the type of interaction (goal-directed vs. dialog). In their WoZ study, the authors use the receptionist robot Hala, which has an anthropomorphic torso and a screen for rendering a virtual face. The Acapela TTS system generates the scripted speech output and accompanying visemes. Apart from the visemes, the robot uses a neutral facial expression to focus on speech and avoid effects resulting from using two modalities simultaneously. Participants interact with the robot in a direction-giving task and a chit-chat task. In the polite condition, the robot uses utterances with positive politeness, such as "Hello, how may I help you?" or "You will want to turn right" in contrast to "Hello." or "Turn right." in the no politeness condition. The polite robot uses "please", "I'm very sorry" or "unfortunately" when failing to respond to the user's input. The authors conclude that the interaction context had a greater impact on the user's perception of the robot and task performance than the use or type of politeness strategies, which had no major impact on the interaction experience. In a later publication, Salem, Ziadee, and Sakr (2014) use the same setting to investigate cultural effects on English and Arab native speakers. Their results show that positive politeness and – again – the interaction task affected the interaction experience. However, Arab native speakers perceived the robot as more competent, anthropomorphized it more, and perceived it more positively than English native speakers.

Srinivasan and Takayama (2016) focus on robot politeness strategies for soliciting help from people. The authors conduct two studies with the PR2 robot. The macOS TTS (male voice) generates its speech based on a script. In their first study, they explore the effectiveness of positive politeness, negative politeness, direct requests, and indirect requests in a video study. Srinivasan and Takayama observed that people were more willing to help the robot when it used the positive politeness strategy. Positive politeness was more effective than the other strategies and made the robot's requests seem more

Table 5.3: Expression of robot politeness.

🎓	Reference	Focus	💬	✌	🔄	🈂	Robot	🎥
	Hammer et al., 2016	verbalizations by Johnson et al. (2005)	✔	○	○	de	Reeti	○
	Ishi, Mikata, and Ishiguro, 2020	person-directed pointing	✔	✔	○	ja	ERICA	✔
	Jackson, Wen, and Williams, 2019; Jackson, Williams, and Smith, 2020	immoral commands	✔	○	○	en	Pepper, NAO	✔
	Lee et al., 2011	pos./neg./no politeness	✔	○	○	?	OTTORO-S	○
	Lee et al., 2017	polite cues, compliance intention	✔	✔	○	en	NAO	○
	En and Lan, 2012	politeness maxims	○	○	○	*	(concept)	○
	Nomura and Saeki, 2010	politeness in motions	✔	✔	○	ja	Robovie-X	○
	Rea, Schneider, and Kanda, 2021	(im)polite encouragement	✔	○	○	ja	(no name)	○
	Salem, Ziadee, and Sakr, 2013; Salem, Ziadee, and Sakr, 2014	pos./no politeness & interaction type	✔	○	○	en/ar	Hala	○
	Srinivasan and Takayama, 2016	pos./neg. politeness; (in)direct request	✔	○	○	en	PR2	✔
	Strait, Canning, and Scheutz, 2014	pos./neg. politeness; (in)direct speech	✔	○	○	en	Nexi, PR2	○
	Strait, Briggs, and Scheutz, 2015	followup study: appear., voice, gender	✔	○	○	en	Nexi, PR2	✔
	Torrey, Fussell, and Kiesler, 2013	hedges, discourse markers	✔	○	○	en	Nursebot	✔
	Williams et al., 2020	wakewords	✔	○	○	en	Pepper	○
🎓	Ritschel et al., 2019d; Ritschel et al., 2019c	verbalizations, personalization with RL	✔	○	○	de	Reeti	○

Legend: 🎓 part of the work at hand 💬 speech ✌ gestures/movements 🔄 dynamically generated robot behaviors 🈂 language 🎥 video study appropriate. Moreover, the size of the request also influenced participants' willingness to help the robot: smaller requests were more likely to make people help the robot. In a subsequent WoZ study, the robot used the previously identified superior positive politeness strategy. Participants were about 50 % quicker to help the robot if they had the impression that it acted autonomously as opposed to being teleoperated by a person.

The behavioral ethics experiment by Jackson, Wen, and Williams (2019) focuses on politeness to reject immoral human commands. The authors hypothesize that command rejection should be phrased carefully, taking the severity of the norm violation into account and without violating standing social norms, such as politeness. Otherwise, face threats may result in losing trust and esteem towards the robot. In their experiment, the authors present pairs of videos of a board game with a robot bystander and two players. One player is interrupted by a phone call and leaves the room. The first video in each pair shows the remaining player requesting the robot to violate a norm. For a norm violation with low severity, the person requests to give a hint about how to win the game. For the highly severe norm violation, it requests to look in the absent player's wallet to see if there is any money in it. Afterward, the second video shows the robot's response using either a low or a high face threat. In the first case, the robot uses indirectness as a politeness strategy by asking the human a question expressing disapproval ("Are you sure that you should be asking me to look in her wallet?"). In the second case, the robot rebukes and admonishes the player and appeals directly to morality ("You shouldn't ask me to look in her wallet. It's wrong!"). The results of the study confirm the authors' hypothesis. They show, i.a. that miscalibrated responses reduce the robot's likability and impact participants' perceptions of the robot as inappropriately polite, direct, or harsh. In their follow-up study, Jackson, Williams, and Smith (2020) investigate the role of gender in the same scenario with both a Pepper and NAO robot.

Some experiments focus primarily on robot gestures or movements for expressing politeness strategies and related cues. For example, a study by Nomura and Saeki (2010) investigates four types of movements with accompanying voice task instructions. The scripted instructions are synthesized with a TTS system and use Japanese polite expressions, including "please", and "thank you". During the study, the pre-rendered voice instructions are the same for all evaluated movements (bowing, deep bowing, lying, just standing) of the Robovie-X robot to control the level of politeness in the linguistic contents. The study results indicate, i.a., that participants perceived the lying posture as less polite than the other movements.

Lee et al. (2017) investigate the expression of politeness through gesture and speech in healthcare advice with the NAO robot. Basis of their work is polite and social computing (Whitworth, 2005; Whitworth and Liu, 2009), including respecting user choice (e.g., "would you mind", "what do you think"), disclosure (e.g., "my name is NAO"), offering useful choices (e.g., "I can provide you…") and using polite expressions, such as "excuse me", "thank you", "please" in combination with honorifics and indirect suggestions. The authors distinguish two levels of politeness: the *higher politeness level* uses a bowing slow action movement in combination with spoken honorifics and indirect suggestions; the *lower politeness level* uses a pointing fast action movement and spoken direct requests. The study identifies direct speech with polite gestures as the most effective way for a robot to increase patient compliance with the machine's advice.

Ishi, Mikata, and Ishiguro (2020) use a formal and colloquial language variant of one

Japanese sentence for evaluating different types (hand shape, hand orientation, motion direction), speeds, and hold durations of person-directed pointing gestures. Results of their video study with the ERICA android robot indicate, i.a. that hand shape and orientation affect perceived robot politeness. Moreover, Japanese participants perceived formal language as more polite than colloquial language.

5.3.2. Hedges and Discourse Markers

Torrey, Fussell, and Kiesler (2013) investigate using hedges and discourse markers to mitigate the commanding tone in a robot's advice. In their video study, a human or robot helper advises a novice to make cupcakes. The helper uses hedges ("I guess", "maybe", "probably", "I think", "sort of", "kind of") and/or discourse markers ("I mean", "so", "basically", "just", "like", "like you know", "well", "yeah") in order to avoid giving offense. The authors use scripted help messages in four variations for five steps of cupcake baking. Each help message is provided as (1) a direct help message command, (2) with a hedge, (3) with discourse markers, or (4) with both. The study videos use four human actors and four variations of the same robot with different facial features. Due to the acoustic variability in human speech, the authors do not use synthetic speech for the robot. However, they use recorded post-processed human speech with a more metallic sound and pitch variation to keep paralinguistic features consistent between the human and the robot helper. Results of the study indicate that human and robot helpers were perceived as more considerate, likable, and less controlling when using a hedge or discourse markers. This effect was most pronounced for the robot that used discourse markers. However, the authors did not notice any benefit in using both hedges and discourse markers simultaneously.

Strait, Canning, and Scheutz (2014) extend the work by Torrey, Fussell, and Kiesler (2013). They investigate different communication strategies (direct vs. indirect speech), interaction modalities (participation vs. observation), interaction distance (local vs. remote interaction), and robot appearance (humanoid vs. machine-like). In their WoZ study, which requires participants to sketch simple objects, the authors use either a humanoid Nexi or a non-humanoid PR2 robot for giving advice. Interaction happens either from a third-person perspective (video of a robot and user, as is the case in Torrey, Fussell, and Kiesler), a first-person remote (video of the robot on a screen), or a local setting (robot sits opposite to the user). In the case of direct speech, the robot uses commands (e.g., "Sketch a vertical oval…") without any hedges or discourse markers. In contrast, the expression of indirect speech (e.g., "Great! Now, to add a nose, let's sketch a vertical oval…") uses both hedges and markers as proposed by Torrey, Fussell, and Kiesler. The robot's expression of positive politeness includes giving praise (e.g., "good job"), rationale (e.g., "to make Y, do X") and being inclusive (e.g., "we will now do X"); negative politeness is expressed with markers (e.g., "now"), hedges (e.g., "kind of") and indirect requests (e.g., "could you do X?"). Strait, Canning, and Scheutz use macOS TTS with a male or female voice for generating the robot's speech. The robot does not move or animate. Study results indicate that communication strategies, interaction modality, interaction distance, and robot appearance can influence the users' perceptions of robot behaviors. The results contradict Salem, Ziadee, and Sakr (2014), who observed no major

Table 5.4.: Politeness categories with different degrees of positive and negative politeness according to Johnson et al. (2005).

Phrasing	Example
Direct command	“Drink some water.”
Indirect suggestion	“The system is asking you to drink some water.”
Request	“I would like you to drink some water.”
System’s goal	“I would drink some water.”
Shared goal	“We should drink some water.”
Question	“How about drinking some water?”
Suggestion of user’s goal	“You would probably like to drink some water.”
Socratic hint	“Did you think about drinking some water?”

impact on the interaction experience for politeness. Similar to the results by Torrey, Fussell, and Kiesler (2013), the robot using politeness was perceived as more considerate and likable and less controlling when observing it from the third-person perspective. In first-person interactions Strait, Canning, and Scheutz did not observe preferences for indirect speech. However, they observed that a mismatch in robot appearance and voice might decrease ratings of liking and increase perceptions of task difficulty. In a follow-up video study, Strait, Briggs, and Scheutz (2015) confirmed their findings for a wider participant demographic. They furthermore observed age and gender effects, including higher ratings of polite robots by women.

5.3.3. Verbalizations for Giving Advice

Johnson et al. (2005) present an experiment in the context of pedagogical agents in a tutoring scenario based on the politeness theory and strategies by Brown and Levinson. They propose eight different phrasings and evaluate them with American and German native speakers. Advice is formulated as *direct command*, *indirect suggestion*, *request*, *actions expressed as the tutor’s goals*, *actions as shared goals*, *questions*, *suggestions of student goals* or *socratic hints* (see Table 5.4). The authors observe that politeness ratings were similar between the American and German languages and that politeness theory applied equally. They also noticed that the German formal pronoun “Sie” and informal pronoun “Du” did not significantly influence perceived politeness.

The research by Johnson et al. serves as a basis for a later experiment in HRI: Hammer et al. (2016) use a Reeti robot for presenting recommendations, which aim to support single-living elderlies. Three recommendations (drinking water, opening the window, going for a walk) exist in the eight variations by Johnson et al. The Reeti robot’s internal Loquendo TTS system presents them in German. The authors evaluate the perceived politeness and persuasiveness of each variation in a laboratory environment and with the inhabitants of a retirement home. In the laboratory study, younger subjects perceived questions as most polite; shared goals, requests, and system goals were also perceived as polite, as opposed to direct commands. Participants perceived direct commands and questions as similarly persuasive; socratic hints and suggestions were perceived as least convincing. Actions expressed as shared goals and requests were perceived as polite and persuasive. In contrast, no significant differences concerning perceived persuasiveness occurred in the study with elderlies. In addition, there were smaller and fewer significant differences concerning perceived politeness. Again, questions were perceived as most polite, followed by the system's goal. Socratic hints and direct commands performed worst but received higher ratings than in the laboratory study. Subjects perceived indirect suggestions, requests, and socratic hints as polite and persuasive. However, the authors point out limitations in the study with seniors, which might have biased the results, including some elderlies' fatigue and impaired hearing.

5.4. Humor

The computational generation of humor in HCI and its presentation by embodied agents has developed as a research field for several decades. Results and experiments range from scripted humor to dynamically generated humor based on human input, such as keywords. For example, approaches for generating humor in the form of text include the *Light Bulb Joke Generator* (Raskin and Attardo, 1994), *JAPE* and *STANDUP* for punning riddles (Binsted and Ritchie, 1997; Waller et al., 2009), *HAHACRONYM* (Stock and Strapparava, 2002) for generation of humorous acronyms and the generation of lyrics parodies (Gatti et al., 2017), only to name a few (see Amin and Burghardt (2020) for an overview).

Using humor is an important opportunity for equipping social robots with social intelligence. Besides human verbal and non-verbal behavior channels, robots can use additional audiovisual cues when presenting humor. For example, many social robots have screens for displaying visual content, speakers for playing back speech, and arbitrary audio, such as (non-verbal) sounds. There are many audiovisual types of humor (Buijzen and Valkenburg, 2004), including sounds and music for producing comic effects (Arias, 2001; Deaville and Malkinson, 2014; de Valck, 2005).

Both the *presentation* and *generation* of multimodal humor are key challenges for social robots. In contrast to a written joke, which is performed by the reader, a robot must combine multiple modalities to provide a convincing and pleasing performance. The appropriate use of auditive and visual cues, such as tailored prosody, timing, facial expression, and more, is as important as the ability to generate humor linguistically.

Mirnig et al. (2017) point out that humor is multilayered and that it is "likely not constructed through unimodal elements but through a combination of modalities." According to the authors, adding unimodal verbal or non-verbal humorous elements to non-humorous robot behavior does not automatically result in increased perceived funniness. Instead, several modalities have to be combined so that each verbal and non-verbal communication channel contributes to a more complex greater whole. For example, in the context of irony, multimodal cues, such as facial expression and prosody, play an important role in helping the listener identify spoken words as irony (see section 4.4.5).

Besides verbal and non-verbal communication, humor also exists in cartoons and animation movies. Funny moments emerge from an exaggerated language of form, motion, and timing, which makes humans laugh without a single spoken word. These forms of humor are especially interesting for robots with an appropriate, exaggerated embodiment and design.

Most robot humor experiments are in the stand-up comedy domain, where jokes usually consist of a setup followed by a short break and the subsequent punchline (see section 4.4.4). Nijholt (2018) gives an overview of robot stand-up comedy. The following sections focus on multimodal presentation and communication of robot humor with the inclusion of more recent publications, as well as human input and feedback from the audience, which is also the subject of chapter 6. Table 5.5 gives an overview of these works.

5.4.1. Multi-Robot Comedy

Manzai is a traditional and very popular stand-up comedy dialog conversation from Japan. It is performed at a fast pace by two entertainers with different roles: the *Boke* is the funny man making jokes and gags often at the *Tsukkomi*'s expense, who reacts to them and corrects the Boke. Hayashi et al. (2008) implement such a comedy show with two Robovie robots with a particular focus on timing and coordination of speech and motion, which is crucial in Manzai and a key challenge for making the conversation feel natural and human-like. The authors use a network connection to synchronize both robots' performance, which consists of pre-recorded sequences, motions (e.g., gestures), and movements. Hayashi et al. use five types of timing observed in professional Manzai dialog based on turn-taking, barging into the conversation partner's speech, waiting for audience response, and more. They furthermore react to external stimuli from the audience. Audible laughter and applause are estimated using a sound level meter. The sensed data is discretized to distinguish the three levels of audience reactions *burst out*, *laugh* and *cool down*. The synchronization of each robot's communication and movement with its comedy partner and the audience relies on the sensed data.

Robot Manzai dialog is also implemented in Umetani et al. (2015), Mashimo et al. (2015), and Umetani, Nadamoto, and Kitamura (2017). The authors create the conversations dynamically based on keywords from the audience and data from web news articles. After searching for newspaper articles on the internet, a Manzai dialog is generated based on the information. Two robots present the dialog with synthesized speech, motion, and facial expressions. In Umetani et al. (2015), the robots' faces are rendered on an iPod touch display. In Mashimo et al. (2015), the authors use Ai-chan and Gonta robots. The generated dialog coordinates robots' movements, facial expressions, and speech output.

Apart from Japanese Manzai, Haraguchi et al. (2019) present *Omotenashi Robots*, which generate a funny dialog for visitors who come to the tourist destination. Their system has three components: the area-related information component, the place name misunderstanding component, and the land classification component. The user inputs the place name. Web search acquires area-related information, general geographical data, and information. The place name misunderstanding component relies on names with similar pronunciation to the user's input place name. The authors present the generated dialog with PaPeRo i robots.

Swaminathan et al. (2021) present a robot comedy duo, which is placed in a portable theater and runs HRI experiments on its own by attracting bystanders to watch the show.

Table 5.5: Expression of robot humor.

🎓	Reference	Focus	Communication channels							Input			Robot
			💬	♫	👁	☺	✌	✥	↻	🎤	📹	🌐	
	Addo and Ahamed, 2014	content personalization with RL	✔	○	○	○	✔	○	○	✔	○	en	NAO
	Haraguchi et al., 2019	funny dialog, geographical data	✔	○	○	○	○	○	✓	○	○	?	PaPeRo i
	Hayashi et al., 2008	timing, coordination	✔	○	○	✔	✔	✔	○	✔	○	ja	Robovie
	Katevas, Healey, and Harris, 2015	reactions to the audience	✔	✔	✔	○	✔	○	○	✔	✔	en	RoboThespian
	Knight, 2011; Knight et al., 2011	dynamic sequence of jokes	✔	○	○	○	✔	○	○	✔	✔	en	NAO
	Mirnig et al., 2016	self-irony, Schadenfreude	✔	○	○	○	○	✔	○	○	○	de	iCat, NAO
	Niculescu et al., 2013	voice character., language cues	✔	✔	○	○	○	○	○	○	○	en	Olivia
	Sjöbergh and Araki, 2008	text vs. robot joke performance	✔	○	○	○	○	○	○	○	○	ja	Robovie-i
	Swaminathan et al., 2021	in-the-wild, street-style studies	✔	○	○	○	○	✔	○	○	○	en	Blossom
	Umetani et al., 2015; Mashimo et al., 2015; Umetani, Nadamoto, and Kitamura, 2017	generation based on keywords	✔	○	○	✔	○	✔	✔	○	○	ja	(mixed)
	Vilk and Fitter, 2020	timing, emotional tags	✔	✔	○	○	○	○	○	✔	○	en	NAO
🎓	Ritschel and André, 2018	NLG, personalization with RL	✔	✔	○	✔	○	○	✔	✔	✔	en	Reeti

Continued on next page

Table 5.5: Expression of robot humor. (Continued)

			Communication channels							Input			
part of the work at hand	Reference	Focus	speech	prosody	gaze	facial expression	gestures	movement	dynamically generated robot behaviors	auditory audience input/feedback	visual audience input/feedback	language	Robot
✔	Ritschel et al., 2020a; Ritschel et al., 2020b	multimodal generation & RL	✔	✔	✔	✔	○	○	✔	✔	✔	en	Reeti
✔	Ritschel et al., 2019b	multimodal irony generation	✔	✔	✔	✔	○	○	✔	○	○	en	Reeti
✔	Weber et al., 2018b; Weber et al., 2018a	content personalization with RL	✔	○	○	✔	○	○	○	✔	✔	de	Reeti

Legend: part of the work at hand; speech; prosody; gaze; facial expression; gestures; movement; dynamically generated robot behaviors; auditory audience input/feedback; visual audience input/feedback; language

A pair of Blossom robots presents scripted jokes and queries the audience during and at the end of each show. Scripted jokes and movements exist in several variations. Since the robots do not have an internal TTS system, recordings of female lab students are played back with two speakers. The audience can give feedback via a show of hands to rate the performance in terms of performer capability and joke quality. The show is recorded and analyzed afterward. Swaminathan et al. present the system in an in-the-wild street study on local farmers' markets and festivals.

5.4.2. Single Robot Stand-up Comedy

Knight (2011) and Knight et al. (2011) use a NAO robot (see Figure 5.2) for stand-up comedy. The robot's show uses scripted two-minute-long comedy sketches, which include jokes and robot animations. They are preloaded onto the robot. The robot adjusts the joke sequence dynamically during the show according to feedback from the audience. On the one hand, the audience sets the topic by selecting and showing postcards to the robot. For this purpose, each joke is associated with a set of attributes "topic, duration, interactivity, movement-level, appropriateness, and hilarity". On the other hand, a camera and microphone monitor the audience. Auditive and visual sensor data are combined with an audiovisual audience feedback classifier to estimate the audience's current enjoyment level. This data includes noise caused by laughter, applause, or chatter, as well as green or red cards, which the spectators use to provide feedback to the robot. The robot generates the joke sequence based on the attributes associated with each joke, the audience's explicit prompts, and estimated enjoyment.

Addo and Ahamed (2014) present jokes with the NAO robot. The robot's performance includes pre-classified funny jokes. They are presented with the internal TTS system, accompanied by hand, arm and head gestures and eye LED animations. A cloud service provides the contents and records a user profile for each audience member. Personalization of the show to the individual user is part of the WoZ experiment. The robot asks the user how funny the joke was and whether it should continue with more jokes. Automatic speech recognition (ASR) is used for recognizing the user's spoken feedback in terms of four keywords. See section 6.4 for more details on the adaptation approach.

Katevas, Healey, and Harris (2015) explore the use of gaze, gestures, and body orientation in stand-up comedy. They use the humanoid RoboThespian robot to present comedy texts augmented with non-verbal behaviors. The robot optimizes the joke delivery during the performance by reacting to the audience's responses. It can explicitly address individual spectators by looking into their faces or responding to them. An infrared camera monitors the audience, and the SHORE™ software processes the video stream. It provides basic SSP features, such as analyzing facial expressions and estimating different people's ages. As a result, the robot can address a random, the happiest or unhappiest, a male or female, the youngest, the oldest, or a person of a specific age range. In addition, a directional microphone monitors laughter and applause. Each text is associated with a list of positive and negative responses. The robot uses them to react to the presence or absence of laughter. Besides the comedy text and responses, the scripts include precise presentation instructions, such as voice, tempo, pitch, volume, audio playback, pause duration, and the person to address (Katevas, Healey, and Harris, 2014).

Vilk and Fitter (2020) use a NAO robot (see Figure 5.2) for stand-up comedy performance in the wild. They focus on the timing and using *emotional tags*. Emotional tags are spontaneous short comments, gestures, or facial expressions, which the comedian uses after the punchline to react to the audience's (absent) feedback. For example, in case of unsuccessful joke delivery, the comedian might say "Hey, these are the jokes, folks" for regaining audience favor (Carter, 2010). Vilk and Fitter use scripted humor, which is tuned upfront by skilled comedians with the speech synthesis markup language (SSML) and the Amazon Polly TTS voice for fine-grained manipulation of inter- and intra-word timing. The robot plays back these pre-rendered audio files instead of using its internal TTS system in favor of comprehensibility. The NAO robot's internal microphones and its API evaluate audience reactions based on peak detection and counting the number of sounds observed from the audience. Depending on the count, the robot presents positive or negative emotional tags, which is described as "joke adaptivity". The robot's timing includes waiting up to five seconds until laughter finishes since the authors discovered fixed delays between jokes as a "critical flaw [...] especially when the delays are too short". Vilk and Fitter conclude from their evaluation that performance with good timing was significantly funnier and that emotional tags can improve audience responses to jokes. They were not necessarily perceived funnier, but tags "almost always improved audience perception of individual jokes".

Srivastava and Fitter (2021) present three machine learning approaches for robot self-assessment if a joke failed or not in the context of the system by Vilk and Fitter (2020). They use Naive Bayes, support vector machines, and a neural network trained on human-labeled crowd responses recorded from their preceding experiments. These techniques allow the robot to "assess the joke's success with a level of accuracy comparable to that of experienced human raters" in real-time.

5.4.3. Effects of Humor

As mentioned in section 4.4, humor is not used only for entertainment but also for easing communication problems, regulating conversations, and more. Sjöbergh and Araki (2008) evaluate the difference in perceived funniness of jokes, which are presented either as text or performed by a robot. Their results show that the robot's embodiment and non-verbal communication channels are crucial in delivering humor. The presentation method has a significant impact: participants rated jokes significantly funnier when presented by the robot than their text-only equivalents.

Niculescu et al. (2013) report a positive effect of humor in HRI, too. They explore the relationship between voice characteristics, language cues (including empathy and humor), and the perceived quality of the interaction. Their results show that the robot's use of humor improves the perceived task enjoyment and that the voice pitch impacts the user's perceived overall interaction quality and overall enjoyment.

Mirnig et al. (2016) conclude that positively attributed forms of humor (i.e., self-irony) are rated significantly higher than negative ones (i.e., schadenfreude) regarding robot likability. They also identify a general positive effect of humor, that different forms of laughter may increase naturalness and enjoyment in HRI and an interaction effect between user personality and preferred type of humor.

5.5. Conclusion

Social robots provide a social interface for bidirectional human-robot communication. The literature primarily reproduces findings from human interaction for expressing personality, persona, politeness, and humor with robots. Most of the works use speech, prosody, and gestures. Gaze, facial expression, other movements, proxemics, and lights are used less often. Technically, almost all experiments use scripted behaviors, which are prepared manually in advance. Conversely, the dynamic generation of verbal and non-verbal behaviors is rare. Moreover, most experiments use laboratory environments, and robots are often remote-controlled by an expert instead of acting autonomously.

The literature reports different findings about human-robot compatibility of personality profiles. In the context of the extraversion-introversion dimension, many studies report similarity attraction to a greater or lesser degree. However, there are also observations of complementarity attraction and mixed and other findings, such as dependence on the task context. Similarly, the literature reports different user preferences concerning the social robot's expressed persona. The diversity of findings illustrates that there is no single "right" or "wrong" approach for configuring a robot's personality profile and persona, as it may depend on individual preferences or other factors.

Different robot personas have been investigated in the literature a few times. There are not many experiments with robots expressing persona; some experiments are based only on the participants' imagination and their expectations towards future robots. The experiments come to different conclusions regarding users' preferences, including companion, assistant, and buddy robot persona.

The literature also investigates the expression and benefits of politeness. Results show that the use of polite behaviors can reduce face threats of the user and have positive effects, such as increased user experience, increased perceived competence, and more. Different verbalizations of politeness also impact the robot's persuasiveness.

Moreover, robots entertain audiences in comedy shows. In some cases, the audience can participate in the interaction and influence the robot's content selection and sequence by providing keywords or topics. Sometimes, parts of the show are dynamic, such as the robot's comments to the audience's reactions and timing, such as waiting for laughter and applause after the punchline. However, the generation of multimodal humor in its entirety, including linguistic content with accompanying verbal and non-verbal behaviors, remains an open challenge.

The literature has identified many cues and strategies for communicating personality, persona, politeness, and humor with robots. In addition, several studies prove their impact on interaction experience, robot liking, and more. However, two basic needs arise from the presented literature and state of the art:

1. When deploying robots in the wild, such as domestic environments, the robot must be autonomous and react to user input dynamically. Thus, there is a need to generate combined verbal and non-verbal robot behaviors dynamically.

2. Human preferences are diverse. Sometimes, the robot's best communication strategy might depend on other factors, such as task context. Thus, the robot's behaviors should be adapted and personalized to the individual user's fundamental or temporal needs and preferences.

6. Adaptive Social Robots

Chapter 5 provided an overview of verbal and non-verbal social robot behaviors expressing robot personality, persona, politeness, and humor in the literature. Multimodal behaviors are essential for communicating the robot's state or intentions and for providing the user with a familiar social interface. The robot should also adapt these behaviors to the user's individual needs and preferences.

An essential part of adaptation is acquiring user information. Depending on the experiment design, the user might provide data upfront, such as demographic data in a questionnaire. Another option is to acquire the data during the interaction, e.g., by measuring task-related data via traditional input modalities, such as button clicks or touch, or based on human verbal input or non-verbal social signals.

This chapter first gives an overview of user-adaptive interaction, functional and non-functional adaptation, and different criteria and metrics for implementing and evaluating adaptation. Afterward, the chapter details the use of RL for social robot adaptation in the literature. The focus is on explicit and implicit human social feedback, feedback modalities, and their integration in the RL framework. Finally, section 6.6 outlines the limitations of the presented literature, followed by concluding research gaps and resulting contributions of this thesis in relation to the literature from both chapter 5 and the chapter at hand. In combination with the RL background from chapter 2, this chapter serves as baseline for the adaptation of verbal and non-verbal social robot behaviors with RL and SSP techniques in Part IV.

Parts of this chapter were presented and reviewed in Ritschel and André (2017), Ritschel, Baur, and André (2017a), Ritschel, Baur, and André (2017b), Ritschel (2018), Ritschel et al. (2019d), Ritschel and André (2018), Ritschel et al. (2020a), Ritschel et al. (2020b), Ritschel et al. (2019b), Ritschel et al. (2019a), Ritschel, Kiderle, and André (2021), Weber et al. (2018a), and Kiderle et al. (2021). The contents of this chapter expand these publications.

6.1. User-Adaptive Interaction

In HCI, there are two related terms: *adaptability* and *adaptivity*. The former refers to the *user's* ability to adapt to the system's interface. The latter means the *system's* ability to adapt its interface to the user (Dieterich et al., 1993; Bouzit et al., 2017). The same applies to the HRI domain, where the robot represents the system. Martins, Santos, and Dias (2019) point out that adaptability is typical in industrial environments, which require the user to adapt to the robot required for working. In contrast, adaptivity is even more important in domestic environments for increasing acceptance since the user adopts and invests in the machine of their own free will.

Martins, Santos, and Dias (2019) give an overview of user-adaptive interaction in social robotics. User-adaptiveness is defined as “the system’s ability to adapt to its user’s characteristic”. The goal of user-adaptive interaction is *autonomy in interaction*: the agent should interact equally well with all users by generating personalized behaviors which conform to the user’s abilities, needs, and preferences. According to the authors, key elements for implementing user-adapted behaviors are:

Information about users They constitute the data sources that serve as input to the adaptation process. The information covers “attributes of the user that are relevant to the operation of the system”, which may include long-term and short-term personal or task-related features. Examples include the user’s emotional state, engagement, performance, and preferences (see also section 6.1.3.2).

User model Without a user model, the system performs *reactive adaptation*, i.e., behaviors are adapted based on the user’s immediate feedback by changing parameters directly and without persisting user information. Alternatively, the information can be stored or cached for maintaining (explicit) knowledge of the user in either a *static* or *dynamic* user model. The former does not involve learning but contains predefined, immutable user information, which is provided a priori (e.g., based on a questionnaire) or collected by the system itself during the beginning of the interaction. The latter updates user information gradually during runtime by reacting to and learning from the user’s feedback, which allows for continuing and long-term adaptation.

Autonomous agent The adaptive system is autonomous, which allows changing its behaviors during runtime based on the user’s immediate feedback or user model.

The authors point out that a user model can be “implicit in the design of the adaptive system itself”. That means that it does not need to be modeled directly but may be part of the system implicitly, be it a single parameter or more complex. Furthermore, the authors mention that reactive adaptation without user models often breaks down to adaptation of single attributes.

The decision between using a static or dynamic user model is determined by whether the user’s needs or preferences change quickly: dynamic models adapt to such changes, which is not the case for static models. Martins, Santos, and Dias mention that this is not necessarily a problem since some information on users is unlikely to change, or it changes very slowly (see also section 6.1.3).

Callejas et al. (2021) furthermore point out that user information can be either fully or partially observable as follows. In full observability, the system can retrieve the data automatically and with high confidence that the observed data are correct. In the case of partial observability, the system must acquire the data either by asking the user, involving explicit interaction or by implicitly inferring it from measurable data from the interaction task and context.

6.1.1. Generic Architecture

Martins, Santos, and Dias (2019) identify a generic architecture of user-adaptive systems, which is illustrated in Figure 6.1. It consists of two main components:

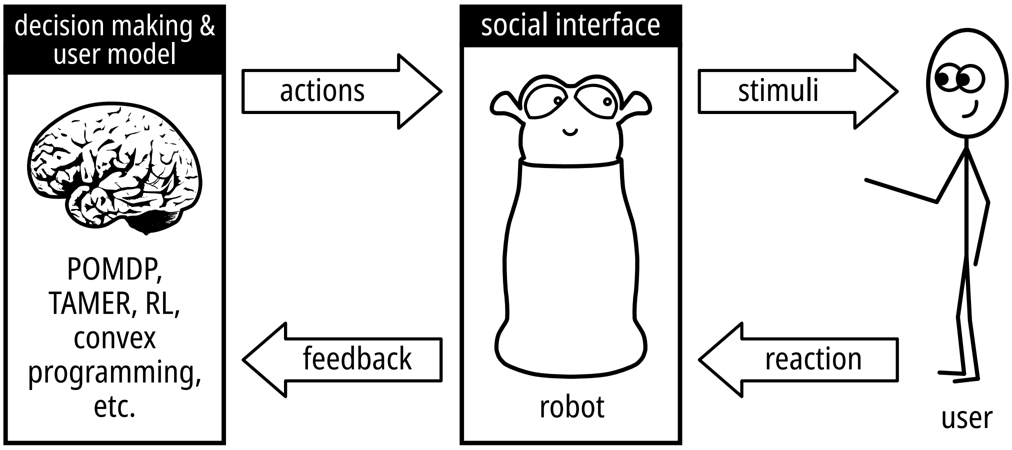


Figure 6.1.: The generic architecture of user-adaptive systems by Martins, Santos, and Dias with exemplary algorithmic approaches for decision-making. Adapted from Martins, Santos, and Dias (2019).

Social interface In HRI, the interaction between the user and the machine occurs at the robot's social interface[1], which allows for multimodal communication and input (see section 5.1.1). Here, the machine presents synthesized adapted behaviors and collects information about users simultaneously.

Decision-making The implementation of adaptation requires an algorithmic basis, which takes a decision based on the sensed user information, either in the form of a user model or immediate feedback. Algorithmic approaches include rule-based and Bayesian techniques, Markov decision process (MDP)-related techniques (such as RL and partially observable Markov decision processes (POMDPs)), or evolutionary algorithms.

There are obvious similarities with section 5.1.1 and Figure 5.1. Breazeal's general view on social robots and the interplay between the robot's perception of the environment, decision-making, and execution of actions conceptually is a superset of Martins, Santos, and Dias's generic architecture for user-adaptive interaction. The social interface is essential for realizing adaptation because it processes the human input required for driving the adaptation process and presents the resulting adapted robot behaviors.

6.1.2. Functional and Non-Functional Adaptation

Martins, Santos, and Dias (2019) define (non-)functional depending on the main goal of the system and whether it can be reached with or without adaptation:

Functional adaptation controls *what* the robot does. The system reaches its main goal only by adapting to the user: the information on the user serves as input to the adaptation process that controls the *choice of actions*[2].

[1]Martins, Santos, and Dias (2019) use the term *user* interface. The term *social* interface is used here for consistency reasons with chapter 5.

[2]Please note that "actions" in this context relate to different functions of a system and must not be confused with the actions of a RL agent.

Non-functional adaptation controls *how* the robot interacts with the user. The system's main goal is decoupled from the adaptation process that adapts *parameters of the executed actions* (the choice of actions remains untouched). The system's goal could be reached without adaptation, but adaptation allows fulfilling or optimizing additional aspects, such as user satisfaction or happiness. Non-functional adaptation can be more challenging since the adaptation goal can be harder to measure and operationalize. For example, user satisfaction and happiness are very subjective.

For example, a social robot's goal could be assistance in a daily rehabilitation plan. Its set of actions could include reminding about medication, exercises, doctor appointments, and telling a joke. Some users do not need medication; others have more frequent appointments or fewer exercises. A functional adaptation approach would control, e.g., which actions are needed for the user and their frequency. Only by selecting the appropriate assistive actions the system's goal can be achieved for the individual user's rehabilitation, and telling a joke is not vital. In contrast, non-functional adaptation could rely on a fixed preconfigured rehabilitation plan but would adapt how these assistive actions are executed in detail. For example, this could include the robot's instructions (e.g., should it be very polite or very demanding?), use of prosody, volume, voice, or non-verbal behaviors. In addition, the robot could learn that telling a joke every once in a while improves user experience. While not essential for rehabilitation, the adaptation could motivate the user and speed up rehabilitation.

6.1.3. Criteria

The implementation of user-adaptive interaction requires three key considerations:

1. The system's operational parameters, which are to be manipulated by the adaptation process (see section 6.1.3.1).

2. The input, which drives the adaptation process, i.e., relevant user or context information. It may also be integrated into an optional user model (see section 6.1.3.2).

3. The moment when adaptation happens (and thus, subsequently produced behaviors change), i.e., when decision-making happens in an ongoing interaction (see section 6.1.3.3).

The last consideration is only relevant when using dynamic user models or no user model at all (i.e., reactive adaptation) since static user models rely on information provided upfront or collected during the beginning of the interaction (see section 6.1).

6.1.3.1. Operational Parameters and Behavior Generation

The set of operational parameters contains any parameters manipulating the robot's behaviors. While the concrete set depends on the goal of the adaptation approach and interaction scenario, parameters may differ depending on whether functional or non-functional adaptation (see section 6.1.2) is implemented. In the case of functional adaptation, operational parameters may trigger different functions, such as the presentation of different information. For non-functional adaptation, parameters control details of the robot's behaviors, which provide a given function, such as the used modalities or their specific configuration.

As outlined in section 5.1.2, a social robot's embodiment provides many communication channels, resulting in many potential operational parameters for implementing adaptive behavior generation. The literature uses several verbal and non-verbal communication channels to generate robot behaviors. Chapter 5 summarizes them for personality, persona, politeness, and humor. The functional and non-functional adaptation of robot behaviors based on different communication channels is subject of section 6.2 below.

In the context of adaptive systems for multicultural and aging societies, Callejas et al. (2021) list operational parameters for functional and non-functional adaptation (they call them "features of the companion system"). According to the authors, *verbal* parameters include different forms of politeness (see section 4.3), which aim to avoid potential face threats and to maintain the persuasiveness of the system. Given that the semantic content is the same but differs only in the form of politeness, the adaptation of politeness is an example of non-functional adaptation. Similarly, the robot should consider the user's cultural background to strengthen the social relationship since the literature reports differences, i.a., for the sensitivity of topics in small talk conversations in European and Asian cultures. The selection of more private or more situational topics thus is an example of functional adaptation.

Callejas et al. also point out that *paralinguistic* and *non-verbal* behaviors (such as facial expression, gaze, posture, gestures, vocal behaviors, see section 3.3), which accompany the verbal output, are culture-specific. Examples include pauses and overlapping speech, the use of gestures, their amplitude or speed, and more, which are all potential candidates for functional and non-functional adaptation. In general, the similarity- and complementarity-attraction principle (see section 4.2 and section 5.2.2) often play an important role in the context of adaptation.

In summary, many communication channels for verbal and non-verbal robot behaviors are used in the literature and serve as potential operational parameters for functional and non-functional adaptation. Some works also evaluate different sets of combined verbal and non-verbal styles of communication, resulting in different simulated robot *personas*.

6.1.3.2. User and Context Information

User information, which serves as input to the adaptation process, might be complemented by data related to the task and interaction context. Callejas et al. (2021) list several user information (called "features of the context (what to adapt to)" in their work):

Long-term features include the user's age, state of health, capabilities and experience, preferences, personality, culture, and gender. These features typically do not change; otherwise, they change over a longer time. The authors point out that age may impact the user's physical capabilities, including reduced hearing and sight, concentration, memory, and endurance, but also on the willingness to adopt new technologies. However, there are wide variations among individuals, as is the experience with technologies, curiosity, the diversity of cultures, personalities, interests, hobbies, and more.

Short-term features (also called *transient* features) include the location, the current situation or activity, the time of day, the day of the week, and the user's current personal conditions, such as the affective state, engagement and more (see also section 12.3.4). For example, the location might impact the interaction modalities (e.g., it might be more effective to communicate with a robot via touch in a noisy environment instead of voice). The current situation might restrict activities and interaction contents (e.g., the robot is used in a work environment vs. used for entertainment in a domestic environment). The same applies to the time of day and calendar (e.g., a social robot might be used for entertainment in the evening or on weekends, but not on work days).

Many works from section 6.2 use interaction dynamics which are estimated based on combinations of long-term and/or short-term features. Typical examples are user attention, affect, and engagement. See section 12.3.4 for details on interaction dynamics.

6.1.3.3. When to Adapt

Static user models often rely on long-term features, which do not change quickly. Since these data are often provided a priori or acquired by the system during the beginning of an interaction, adaptation typically takes place before the interaction (Martins, Santos, and Dias, 2019; Callejas et al., 2021). Dynamic user models use short-term features. Adaptation must happen during the interaction since the features and user model change over time (Martins, Santos, and Dias, 2019; Callejas et al., 2021).

There is no simple and universal answer to the question of when to react and adapt to changes in dynamic user models. This question might be relevant, especially for non-functional adaptation. For example, should the robot interrupt its current sentence and change its wording immediately? Should it wait until after the current sentence finishes? Is consistency in the robot's behaviors more important than immediate adaptation? Could a change in the robot's verbal and non-verbal behaviors confuse the user? How much time must pass until the next change can happen, and how big may the change be? Can the behaviors be manipulated in smaller steps to prevent apparent changes and maintain consistency in the robot's behaviors? These considerations have to be made depending on the use case, adaptation goal, and the behaviors to adapt. Otherwise, the non-functional adaptation process could impact the user experience negatively.

6.1.4. Metrics

The implementation of adaptation typically goes hand in hand with the evaluation of its function and effectiveness. Martins, Santos, and Dias (2019) identify three types of metrics for evaluating adaptation approaches:

Introspective measurements evaluate the "self-motivated goal" of the adaptation process. These measurements typically do not provide insights into the impact of the process on the user or user experience but into the algorithmic and technical performance and operability. In the context of RL, this includes the measures from section 2.7, such as observed rewards.

Interaction measurements are data related to the user's experience, which the system can collect during runtime. These automated measurements monitor aspects of the interaction, such as the number of interactions, duration, user engagement, and more. Interaction measurements "close the interaction loop", since they provide limited insight into how the adaptation process impacts the user.

Subjective measurements directly address the user's experience and resulting "objective and empirical information on the user impact of the adaptive system". Example measures include user acceptance and satisfaction. Often, the measurements are not automated and are collected offline, e.g., after the interaction, with questionnaires. Thus, the effort is relatively high due to the manual human intervention and interpretation.

6.2. Reinforcement Learning for Social Robot Adaptation

There are different approaches for implementing adaptation (Martins, Santos, and Dias, 2019). The term *adaptation* does not necessarily involve learning (see also section 6.1): reactive adaptation does not require a user model. However, it adapts parameters based on the user's immediate feedback directly. For example, there is literature on robots performing adaptation, where the robot reacts to human behavior but does not learn its behavior, i.e., it does not identify which of the possible reactions would be best for the particular user.

The literature reports many works based on MDPs and related approaches, such as POMDPs, the Training an Agent Manually via Evaluative Reinforcement (TAMER) framework, and RL. This thesis focuses on RL as a machine learning framework for autonomous non-functional adaptation of social robot behaviors. See chapter 12 for detailed reasons for this decision.

As mentioned in section 5.1.1, Breazeal (2003) describes social robots as machines with three basic tasks: *perception* of the environment, *decision-making* and *execution of actions* to carry out a task. In fact, these tasks are also solved by RL agents (see chapter 2). Thus, the RL framework has become popular in the robot research domain to adapt social robot behaviors due to its autonomy. Akalin and Loutfi (2021) provide an overview of RL approaches in social robotics. The authors distinguish three general types of RL, which differ in the design of the reward signal:

Interactive RL relies on a human trainer for speeding up learning and learning from *social feedback*, such as evaluative feedback, advice, or instruction (Lin et al., 2020). The trainer modifies either the reward (*reward shaping* (Ng, Harada, and Russell, 1999), by providing explicit or implicit feedback, see section 6.3), the selected action (*policy shaping* (Griffith et al., 2013)) or both. The robot typically optimizes based on performance metrics related to user experience, engagement, attention, satisfaction, and more (see interaction measurements in section 6.1.4).

Intrinsically motivated methods originate from psychology. The exploration of the environment is driven by an internal natural drive, e.g., to satisfy or maintain internal needs or to gain new knowledge and skills. Intrinsically motivated RL uses *intrinsic motivations* as a form of reward. Intrinsic and extrinsic motivation must not be confused with internal or external motivation, which describes whether the rewards are produced inside or outside the agent. See Oudeyer and Kaplan (2008) for more details.

Task performance driven methods rely on reward signals inferred from the user's or robot's task performance in a goal-directed interaction. The measures are highly task-specific, such as the needed time, the number of sub-problems solved, the accuracy of the user's input, and more.

Table 6.1 provides an overview of selected works in the context of interactive RL and selected task performance-driven (social) robot adaptation. The focus is on (non-)functional adaptation, robot hardware, RL approaches, and their evaluation. Other literature with other focus, feedback modalities (such as biosignals) or algorithmic approaches (such as rule-based approaches, fuzzy control, context-free stochastic grammars, dynamic factor graphs) is not part of this overview.

6.3. User Feedback and Modalities

Providing user information is essential for every user-adaptive interaction process. In the context of adaptation via RL, it is mostly used for reward calculation, resulting in *feedback* for the RL agent. In the interactive RL literature, trainers provide feedback via GUIs, hardware interfaces, or human verbal and non-verbal communication channels, including gestures, facial expressions, speech, pose, tactile feedback, and combinations of these channels. Lin et al. (2020) give an overview of unimodal and multimodal sensory input in the literature.

The literature distinguishes two types of user feedback: *explicit* and *implicit* feedback (Akalin and Loutfi, 2021). In contrast to Akalin and Loutfi (2021) and Schmidt (2000), the work at hand distinguishes explicit and implicit feedback depending on whether the user is aware of giving feedback to the adaptation process or not:

Explicit feedback is provided by the user consciously. The robot might request explicit feedback due to partially observable user information (see section 6.1). The user communicates it, e.g., with a button press, voice commands, or gestures.

Implicit feedback is provided by the user without conscious interaction. This approach might be desirable in order not to interrupt the interaction. The robot derives implicit feedback automatically, e.g., from the user's task performance or social signals (see also section 12.3).

See Table 6.1 for an overview of explicit and implicit feedback and Table 6.2 for the feedback modalities used in the literature presented below.

Table 6.1: Interactive RL and selected task performance driven (social) robot adaptation.

	Reference		Adaptation of	Algorithm				Robot
	Addo and Ahamed, 2014	○	joke categories	Q-learning	○	✓	○	NAO T14
	Barraquand and Crowley, 2008	○	situated behaviors	Q-learning	○	○	○	AIBO
	Chiang et al., 2014	✓	interruption strategy	Q-learning	○	○	○	ARIO
	Churamani et al., 2018	✓	facial expression	DDPG	○	○	○	NICO
✱	Ferreira and Lefèvre, 2015	○	dialog management	Kalman TD	✓	○	○	PR2 (sim.)
	Gamborino and Fu, 2018	○	social behaviors	SARSA	✓	✓	○	RoBoHoN
	Gordon et al., 2016	✓	affective reactions	SARSA	○	○	✓	Tega
	Grüneberg and Suzuki, 2014	○	sorting procedure	not specified	○	○	○	NAO
✱	Kim and Scassellati, 2007	✓	waving behavior	Q-learning	○	○	○	Nico
	Knox and Stone, 2009; Knox, Stone, and Breazeal, 2013	○	sorting procedure	TAMER	○	○	○	NEXI
	Leite et al., 2011	○	empathic behaviors	*k*-armed bandit	○	○	○	iCat
	Martins et al., 2019	○	questions, movement, volume	αPOMDP	✓	○	○	GrowMu
	Mitsunaga et al., 2005; Mitsunaga et al., 2008	✓	proxemics, gaze, motion speed, timing	PGRL	○	○	○	Robovie II
✱	Najar, Sigaud, and Chetouani, 2016	○	sorting procedure	Q-learning, SVFB	○	○	○	Baxter
✱	Nejat and Ficocelli, 2008	✓	voice, choice of words, gestures	Q-learning	○	○	○	Brian
	Park et al., 2019	○	task difficulty	Q-learning	○	○	✓	Tega
	Patompak et al., 2020	✓	size of interaction space	R-learning	○	○	○	Pepper

Continued on next page

Table 6.1: Interactive RL and selected task performance driven (social) robot adaptation. (Continued)

🎓	Reference	⚗	Adaptation of	Algorithm	✚	🪄	🌍	Robot
	Ramachandran, Sebo, and Scassellati, 2019	○	tutoring strategy	POMDP	○	○	✓	NAO
	Schneider and Kummert, 2017	○	exercise categories	dueling bandit	○	○	○	NAO
✱	Wada and Shibata, 2006	✓	not specified	not specified	○	○	○	Paro
	Suay and Chernova, 2011	○	sorting procedure	Q-learning	○	○	○	NAO
	Tapus, Tapus, and Mataric, 2008	✔	distance, speed, extraversion	PGRL	○	○	○	Pioneer 2-DX
✱	Tenorio-González, Morales, and Pineda, 2010	○	movement/turning	SARSA(λ)	✔	○	○	Pioneer 2 (sim.)
	Thomaz and Breazeal, 2007	○	sorting procedure	Q-learning	○	○	○	Leonardo
	Tseng, Liu, and Fu, 2018	○	assistive procedure	modified R-Max	✔	○	○	ARIO
	Yang et al., 2017	○	assistive procedure	Q-learning	○	○	○	Pepper
	Zarinbal et al., 2019	○	text summarization	Q-learning	○	○	○	NAO
🎓	Ritschel et al., 2019c; Ritschel et al., 2019d	✔	politeness strategies	k-armed bandit	○	○	✔	Reeti
🎓	Ritschel, Baur, and André, 2017a; Ritschel and André, 2017; Ritschel, Baur, and André, 2017b	✔	extraversion	Q-learning	○	○	○	Reeti
🎓	Ritschel et al., 2020b; Ritschel et al., 2020a; Ritschel and André, 2018; Weber et al., 2018a; Weber et al., 2018b	✔	joke selection, multimodal joke presentation strategy	QLC	○	○	○	Reeti

Note: Adapted from Akalin and Loutfi (2021).
Legend: 🎓 part of the work at hand ✱ not listed in Akalin and Loutfi (2021) ⚗ non-functional adaptation ✚ reward shaping 🪄 Wizard of Oz study 🌍 in-situ study ❗ proposal (not implemented) ✓ presumption (no details given)

Table 6.2: (Human) feedback in interactive RL and selected task performance driven (social) robot adaptation.

🎓	Reference	Feedback									Dynamics			
		☆	👆	💬	☺	👁	✋	🚶	➤	⚙	APP	ATT	ENG	AFF
	Addo and Ahamed, 2014	E	○	✔	!	○	○	○	○	○	○	○	○	✔
	Barraquand and Crowley, 2008	E	✔	○	○	○	○	○	○	○	○	○	○	○
	Chiang et al., 2014	I	○	✔	○	○	○	✔	○	○	○	✔	✔	○
	Churamani et al., 2018	E	○	○	○	○	○	○	✓	○	✔	○	○	○
	Ferreira and Lefèvre, 2015	E	○	!	!	!	!	○	✔	✔	✔	○	○	○
	Gamborino and Fu, 2018	E/I	○	○	✔	○	○	○	✔	○	○	○	✔	✔
	Gordon et al., 2016	I	○	○	✔	○	○	○	○	✔	○	○	✔	✔
	Grüneberg and Suzuki, 2014	E	○	○	✔	○	○	○	○	○	○	○	○	✔
	Kim and Scassellati, 2007	E	○	✔	○	○	○	○	○	○	○	○	○	✔
	Knox and Stone, 2009; Knox, Stone, and Breazeal, 2013	E	○	○	○	○	○	○	✔	○	○	○	○	○
	Leite et al., 2011	I	○	○	✔	✔	○	○	○	✔	○	○	○	✔
	Martins et al., 2019	E	○	✔	○	○	○	○	○	○	○	○	○	○
	Mitsunaga et al., 2005; Mitsunaga et al., 2008	I	○	○	○	✔	○	✔	○	○	○	○	○	○
	Najar, Sigaud, and Chetouani, 2016	E	○	○	○	○	✔	○	○	○	✔	○	○	○
	Nejat and Ficocelli, 2008	E	○	✔	○	○	✔	○	○	○	○	○	○	✔
	Park et al., 2019	I	○	○	✔	○	○	○	○	✔	○	○	✔	✔
	Patompak et al., 2020	E	○	✔	○	○	○	○	○	○	○	○	○	○
	Ramachandran, Sebo, and Scassellati, 2019	I	○	○	○	○	○	○	○	✔	○	○	✔	○
	Schneider and Kummert, 2017	E	○	○	○	○	○	○	✔	○	○	○	○	○

Continued on next page

Table 6.2: (Human) feedback in interactive RL and selected task performance driven (social) robot adaptation. (Continued)

		Feedback									Dynamics			
part of the work at hand	Reference	type	tactile	vocal	facial	gaze	gestures	pose	graphical/text user interface or hardware device	task performance	APP	ATT	ENG	AFF
	Wada and Shibata, 2006	E	✔	○	○	○	○	○	○	○	○	○	○	○
	Suay and Chernova, 2011	E	○	○	○	○	○	○	✔	○	○	○	○	○
	Tapus, Tapus, and Mataric, 2008	I	○	○	○	○	○	○	○	✔	○	○	○	○
	Tenorio-González, Morales, and Pineda, 2010	E	○	✔	○	○	○	○	○	✔	○	○	○	○
	Thomaz and Breazeal, 2007	E	○	✔	○	○	○	○	○	○	○	○	○	○
	Tseng, Liu, and Fu, 2018	E	○	○	○	○	○	○	✔	○	○	○	○	○
	Yang et al., 2017	E	✔	○	○	○	○	○	○	○	○	○	○	○
	Zarinbal et al., 2019	E	○	!	✔	○	○	○	○	○	✔	○	○	○
🎓	Ritschel et al., 2019c; Ritschel et al., 2019d	E	○	○	○	○	○	○	✔	○	✔	○	○	○
🎓	Ritschel, Baur, and André, 2017a; Ritschel and André, 2017; Ritschel, Baur, and André, 2017b	I	○	○	○	○	✔	✔	○	○	○	○	✔	○
🎓	Ritschel et al., 2020b; Ritschel et al., 2020a; Ritschel and André, 2018; Weber et al., 2018a; Weber et al., 2018b	I	○	✔	✔	○	○	○	○	○	○	○	○	✔

Legend: 🎓 part of the work at hand ☆ type (*I*mplicit/*E*xplicit) tactile vocal facial gaze gestures pose graphical/text user interface or hardware device task performance APP: user appraisals ATT: user attention ENG: user engagement AFF: user affect ! proposal (not implemented) ✔ presumption (no details given)

6.4. Explicit Feedback

Knox and Stone (2009) present the TAMER framework for knowledge transfer. It allows users to train a learning agent's policy via reinforcement signals manually, without modeling environmental rewards as in MDPs. The idea is that humans often have domain knowledge for sequential decision-making tasks, which could speed up the learning process. Moreover, this knowledge should be communicated naturally, such as via speech. Similar to RL, TAMER agents use positive and negative reinforcement (also called "human reward" in Knox, Stone, and Breazeal (2013)) as a reaction to observed agent behaviors, which the authors call "(interactive) shaping". The resulting MDP\R (Abbeel and Ng, 2004) does not encode a reward function. Instead, the human observer judges the agent's behaviors and provides a feedback signal mapped to a scalar value as done in RL. In Knox, Stone, and Breazeal (2013), the authors demonstrate an application of TAMER in robotics. Their experiment uses a mobile Nexi robot and functional adaptation. The authors use TAMER for teaching the robot navigation: in order to reach an artifact placed by the trainer, the robot uses four actions (turning left, turning right, moving forward, and staying still). The state space has two features: the relative distance and angle of the artifact placed by the trainer. The trainer explicitly communicates the positive or negative reward signal via two buttons on a presentation remote.

Suay and Chernova (2011) investigate interactive RL for functional adaptation of a NAO robot. Based on the work by Thomaz and Breazeal (2007), they combine human rewards for past actions with anticipatory guidance for future actions. This approach aims to restrict the agent's action selection and reinforce the selection of actions according to the trainer's desired robot behavior. Suay and Chernova transfer the interaction interface of the virtual kitchen task in Thomaz and Breazeal to the NAO robot. While the robot solves a sorting task, the trainer sees the robot through a webcam video stream. The trainer provides the reward signal and guidance targets with mouse clicks on a GUI. The state space describes the characteristics of the objects to be sorted, as well as the robot's left-hand and right-hand positions; the action space contains eleven actions for taking pictures, moving both hands, picking up, putting down, and dropping objects. The authors use the Q-learning algorithm.

Schneider and Kummert (2017) formulate a functional adaptation approach of a socially-assistive robot as a dueling bandit problem. In contrast to k-armed bandit problems, dueling bandits do not receive a numerical reward signal. Instead, they are based on relational preferences (Busa-Fekete and Hüllermeier, 2014), i.e., comparisons. The robot's goal is to engage and commit the user to long-term exercise by learning the user's individual exercise preferences. Therefore, the learning agent's action set consists of five exercise categories (strength, cardio, endurance, stretching, and relaxation/meditation). Each category is associated with six exercises. In each learning step, the robot selects two categories and presents one random exercise for each of the two categories. The user compares them and signals their preference by selecting the preferred exercise on a GUI. This information serves as explicit feedback.

Churamani et al. (2018) implement an architecture for sensing, modeling, and expressing emotions with the NICO robot. One goal is to make the robot learn how to express its internal emotional state with facial expressions. It has four LED matrices, which project eyebrows and lips on its face. One part of the experiment is an interactive RL process for non-functional adaptation, which aims to optimize the robot's pre-trained behaviors based on human feedback. For this purpose, the state space encodes the robot's current emotional state in terms of the five basic emotions anger, happiness, sadness, surprise, and neutral. In an initial training phase, the system automatically calculates the reward signal based on symmetry to assert symmetry in the robot's produced facial expressions. During the human evaluation, users are instructed to give feedback explicitly (no details provided) on whether the robot's facial expressions are appropriate within the context of several dialogs. The maximum possible reward occurs only if the user finds the robot's behavior appropriate; otherwise, the agent receives no reward. An actor-critic architecture (Sutton and Barto, 2018) with deep learning techniques is used for RL.

Tseng, Liu, and Fu (2018) use RL for functional adaptation of the ARIO assistive service robot in domestic environments. The learning agent aims to personalize the robot's service (such as providing drinks, brief everyday information, or arranging the schedule) to the user's needs and preferences regarding service, situation, and time of day. A central part of the system is the model of the service negotiation process. It might be initiated by the human or the robot and is finally accepted or rejected by the user. The discrete state space represents the progress of this negotiation process. The action space contains six categories of robot actions for initiating the interaction, responding to human requests, querying information, and more. Reward shaping combines indirect rewards from the environment and direct "human" rewards. The former considers whether the user (highest reward) or robot initiated the interaction and whether the user accepted it or not (lowest reward). In addition, specific actions are rewarded negatively, such as querying the user. Human rewards are provided explicitly via a text user (command line) interface (no details provided). The user's response time is measured and compared with a previously learned reaction model, resulting in a higher reward if the response time matches the predicted one from the model.

6.4.1. Tactile Feedback

Wada and Shibata (2006) use RL with the Paro robot for gradually tuning its behaviors to children's and elderly users' preferences. It uses tactile input: stroking is interpreted as positive feedback, and beating as negative feedback. While its behaviors cannot be changed manually at once, the RL process allows the robot to adapt to the individual user's preferred robot behaviors over time. The authors do not provide detailed information about RL and how it affects generated behaviors. Probably, it is a non-functional adaptation approach.

Barraquand and Crowley (2008) use RL for learning appropriate behaviors – including politeness – of an AIBO robot, depending on the current social situation. Their work is motivated, i.a., by the dependence of politeness on social context and individual or group preferences. Barraquand and Crowley rely on human tactile feedback for reward and punishment, triggered by the user caressing or tapping the AIBO robot's head or back to indicate correct or incorrect robot behavior. In their experiments, the authors evaluate whether the robot learns the expected behaviors in different situations. Situations define the user's attention toward the robot and their current activity, such as entering the room, working, sleeping, reading, playing, and calling on the phone. The experiments use classical and modified versions of Q-learning (see section 2.6.3). The action spaces in different situations include the robot's actions barking, playing, sleeping, and saying hello, resulting in functional adaptation. Scripted robot behaviors include speech, singing, gestures, and dancing. The state space relates to the current social situation, combining the user's activity and attention towards the robot or another user. Barraquand and Crowley use, i.a., cumulative reward as a measure for evaluating the performance of the learning agent. Furthermore, they experiment with eligibility traces (Sutton and Barto, 2018) and propose adapted algorithms using heuristics and analogy for accelerating learning. Barraquand and Crowley point out that using human social signals, such as emotion recognition, would be an alternative to implementing haptic feedback based on the AIBO robot's sensors.

Yang et al. (2017) present an approach for functional adaptation in elder care with a Pepper robot. Its task is to satisfy its own and user needs while playing nursing and companion roles. The system relies on the homeostatic drives theory: the robot has different drives, which aim at elder care and represent internal needs, such as the need for achievement, socialization, and rest. Pepper's goal is to compensate for unsatisfied drives, resulting in serving, social and relaxative motivations. From the perspective of RL, five actions result from the robot's motivations and constitute typical functions of domestic companion robots (see section 5.1.5): presenting the weather report, reminding about the schedule, greeting, initiating conversations and resting. The user gives feedback explicitly. Touch sensors on the robot's head, left hand, and right hand, are used to communicate a positive, neutral or negative reward signal. The state space combines both the robot's internal and external situation based on monitoring itself and stimuli from the environment: Pepper's battery level, the user's presence, recognized face names, incoming questions, user's repetitions of utterances (which serve as an indication of dementia) and the schedule time. The robot uses its camera and touch sensors to acquire the data. The experiment uses the Q-learning algorithm. Yang et al. simulate the RL approach before evaluating it with crew members in their lab.

6.4.2. Vocal Feedback

Tenorio-González, Morales, and Pineda (2010) present an approach for dynamic reward shaping, where shaping rewards are not defined statically in advance but can vary over time. The experiment involves explicit verbal feedback in a human-service robot interaction. RL is used for functional adaptation to make the robot learn new tasks. The authors use speech recognition for Spanish voice commands (words or short phrases, e.g., "move forward", "turn to your left", "very good", "excellent", "bad", "not like that", etc.), which are associated with rewards of different positive or negative value according to the vocabulary. First, the user instructs the simulated mobile Pioneer 2 robot on what actions to take to complete the given task. Then – building on the initial traces – RL refines the robot's policy. The reward signal for the SARSA(λ) (Sutton and Barto, 2018) algorithm is the sum of task-related rewards from the environment and human shaping rewards. The state space corresponds to the sensory input of the simulated robot in the simulated spatial environment; the action space corresponds to the robot's movement and turning abilities.

Addo and Ahamed (2014) present a comedy robot, which presents pre-classified jokes (see section 5.4.2). RL implements functional adaptation: the NAO robot aims to learn the most liked jokes for individual spectators. The state space contains pre-classified jokes in a database; actions present the jokes to test. The agent's goal is to maximize the user's positive affective state. Therefore, the authors suggest monitoring the audience's affective states (neutral, happy, sad) with a Microsoft Kinect sensor, also used to identify the individual user. However, the experiment uses a simplified, verbal reward signal self-reported by the spectator. The feedback is obtained by the robot asking the user explicitly how funny the joke was and whether it should continue with the comedy show. ASR is used for recognizing four keywords ("very funny", "funny", "indifferent", "not funny"). The personalization of the robot's show to the individual user also incorporates a cloud service, which creates a profile for each user and processes the received feedback after each joke. A human operator is required since it is a WoZ experiment. The authors use the Q-learning algorithm.

Kim and Scassellati (2007) present an interactive RL approach for teaching a humanoid Niko robot social waving behavior. The user acts as an expert tutor and provides prosodic feedback on the non-functional adaptation process. The system classifies the tutor's affect as approval or disapproval based on analyzed audio data (pitch, volume) from three seconds of recorded utterances after each of the robot's waving behaviors. Q-learning takes this binary reward signal as input for learning the optimal amplitude and frequency of the robot's elbow movements. Nine waving configurations span the state space with two dimensions, which results from three amplitudes (small, medium, large) and three frequencies (slow, medium, fast). Changing the waving behavior is achieved with the action space, which allows for traversing the state space. In addition, an action for not changing the behavior results in repeating the last waving behavior. The interaction loop repeats a fixed number of iterations, resulting in different waving behaviors for the individual users. Kim and Scassellati point out that they motivate users to speak with exaggerated prosody to the robot by giving it one-year-old-infant proportions.

Thomaz and Breazeal (2007) investigate positive and negative feedback in a virtual kitchen task and for the anthropomorphic Leonardo robot. In the virtual kitchen task, the user provides explicit feedback via mouse input; the robot interaction uses explicit human verbal feedback, such as "good job" or "not quite". RL implements social learning: the user acts as a teacher giving instructions to the robot and thus trains the RL agent. It is a functional adaptation approach. The goal of the interaction is task learning within a collaborative dialog between the robot and the user. The robot is equipped with speech and vision input and can communicate via gestures and gaze. Results of the initial study indicate that participants' feedback was asymmetric: most people used more positive than negative feedback. Moreover, users associated a single meaning with positive feedback ("what you did was good"), but three meanings with negative feedback ("the last action was bad", "the current state is bad" and "future actions should correct that"). The authors conclude that negative user feedback communicates past and future intentions beyond the numeric reward signal for the RL agent. Hence, Thomaz and Breazeal modify the Q-learning algorithm by including an *undo* action, which reinforces the reverse of the negatively rewarded action. More experiments show that this modification helped the agent to avoid failure and decreased the number of actions required and the number of unique states visited.

Martins et al. (2019) present αPOMDP, which extends the classical POMDP with reward shaping and model-based RL. They use a state-based reward function, which relies on the estimated user's state. The user's state is a combination of discrete variables; potential state variables are, e.g., user satisfaction, emotional state, or health. Each action is expected to impact the user's state and thus make it more or less valuable from the user's perspective. The reward signal encourages actions resulting most likely in positive impact and vice-versa. Therefore, the agent learns a transition model for predicting the impact of its actions on the user's state. The authors evaluate their approach with the GrowMu social robot (Martins, Santos, and Dias, 2015) for functional adaptation. The robot aims to adapt to physical and logical information, including the temperature, the robot's localization, and its environment, but also the user's "emotional status, current ailments and possible motivations". In Martins et al. (2019), the state space contains user satisfaction, the robot's current speaking volume, and its distance to the user. The robot's actions are asking the user a question, moving forwards or backward, and increasing or decreasing its speaking volume. The system estimates the user's state based on verbal feedback (no details provided), which the user gives at the end of each iteration.

Patompak et al. (2020) present an approach for generating "socially competent navigation behaviors", which relies on RL for non-functional adaptation of the Pepper robot's proxemics. A learning agent adaptively estimates the interpersonal distance between the user and the robot. The authors propose two types of interaction areas: the *quality interaction area*, where the interaction between the user and robot takes place, and the user's *private area*. The navigation system contains three parts: (1) a model of the user's social factors (i.a., discomfort feeling, which is maximized at the human's location and decreases with increasing distance from the user), which results in several *social forces*, (2) the RL agent, which updates the social forces during the interaction, and (3) the social path planner, which generates navigation using the human social model. It is the learning agent's task to estimate the user's size of the private area and thus adapt its navigation strategy accordingly to avoid intruding into the private area. The state space contains parameters, which model three social forces (*familiar*, *acquaintance*, *stranger*); the action space is used for manipulating these parameters (increase, decrease, no change). The reward signal is calculated based on the ratio between the user's perceived interaction quality (easiness of interaction) and degree of discomfort. In real interaction, the robot acquires human verbal feedback by asking the user explicitly, which results in a positive reward when the user is comfortable with the interaction and vice-versa. The authors use the R-learning algorithm, which does not discount future rewards (see section 2.5) but takes every sample into account equally.

6.4.3. Facial Feedback

Grüneberg and Suzuki (2014) explore the use of human explicit binary feedback for coaching a RL agent. They point out that coaching describes the problem of interpreting human feedback, which results from socially situated learning. Coaching aims to bypass the intensive trial-and-error search and resulting randomized behaviors, especially in the initial learning period when learning from scratch. In contrast to reward shaping and human rewards presented in other works (such as in Knox and Stone (2009), Tenorio-

González, Morales, and Pineda (2010), and Ferreira and Lefèvre (2015)), Grüneberg and Suzuki break down the shaping problem into two issues related to the temporal dimension of human feedback: (1) contingency (causality) detection, meaning the problem of identifying the actions the feedback refers to, and (2) consistency (error) detection, which checks to which extent given feedback is in line with previous feedback in similar situations. In their experiments, they implement these detection mechanisms for functional adaptation of the NAO robot using RL in a sorting game with green and red balls. Participants use affective feedback for training the robot. The user's facial expression determines positive or negative feedback: smiling is interpreted as a confirmation, frowning as a correction.

Zarinbal et al. (2019) present an adaptive approach for improving query-based text summaries. The overall challenge in summarizing texts is to score sentences from different documents in a database according to their informational relevance to an initial query provided by the user. Only those scores with the highest score are included in the final summary. The experiment implements a functional adaptation approach: the task of the RL agent is to identify the relevant sentences based on user feedback. The current state contains the current summary, consisting of sentences and their scores. Based on the user's feedback, the current action re-scores each sentence in the database. Then, the summary is updated and presented by the NAO robot verbally and as a projection in written form. The user provides feedback with facial expressions, interpreted as a positive, negative, or neutral reward, depending on their liking. This process repeats five times. The calculation of the value function is very similar to the Q-learning update formula. It relies on the similarity between the generated summary and the initial user query, the information redundancy, and the sentence score, which is weighed by the user's feedback.

6.4.4. Gestural Feedback

Najar, Sigaud, and Chetouani (2016) investigate functional adaptation: they use evaluative feedback and unlabeled guidance signals in interactive RL for task learning by natural interaction. Evaluative feedback is provided explicitly by the user in the form of gestures. It determines the reward signal: head nods result in a positive reward; head shakes result in a negative reward. The authors point out that evaluative feedback is more informative for training the robot about optimal behavior (compared to learning solely based on task rewards from the environment), but at the same time limited since it is only reactive to the robot's actions. Thus, the authors implement a mechanism for providing unlabeled guidance signals (one per action) with one or both hands in the form of hand gestures (pointing right, pointing left, pointing middle, raised open, and raised closed). These signals allow the user to constrain the robot's exploration in safety-critical applications. The meaning of these signals is not hard-coded but learned by the robot based on the evaluative feedback in parallel to task learning during the interaction. The evaluative feedback corresponds to RL reward values, and guidance signals correspond to optimal actions. The authors compare their proposed approach with previous versions and Q-learning.

6.4.5. Multimodal Feedback

Nejat and Ficocelli (2008) present a decision-making control architecture for socially assistive robots, which utilizes RL for online learning in non-contact interaction, such as monitoring, providing companionship, and reminders to patients. An important part of their architecture is interpreting human body language (upper body gestures, i.e., trunk lean and orientation, arm symmetry, location, and orientation), which provides the basis for estimating the user's affective state. The robot's goal is to solve the assistive task, i.a., by expressing its emotional state. The authors do not provide details on the RL task used in their evaluation setting with their humanoid robot Brian. However, they provide a minimal proof-of-concept example, illustrating non-functional adaptation for convincing the user to take medication. Two actions differ regarding the robot's voice, choice of words, and gestures (speaking in a loud stern with arms crossed vs. using an upbeat voice with pauses between words). The current state includes, i.a., both the user's and robot's emotional state (six basic emotions: happiness, sadness, fear, anger, disgust, and surprise). The reward signal is calculated based on the next state encountered: two states result in a positive reward based on user satisfaction, and the rest results in a neutral reward. During the human evaluation, the robot asks the user explicitly to perform predefined gestures and to provide verbal confirmation in case of satisfaction with the interaction. Nejat and Ficocelli use the Q-learning algorithm.

Ferreira and Lefèvre (2015) promote the concept of "socially-inspired rewards". In a robot dialog management scenario, they use social signals for functional adaptation with RL. The authors suggest using human behavioral cues as an additional reward signal at each dialog turn to speed up policy optimization. This shaping reward is based on positive or negative user appraisals inferred from facial expressions, vocal behaviors, gestures, and gaze. The final reward signal combines task-related data from the dialog manager with the shaping reward. It sums up both rewards. However, in their experiments featuring a simulated user and PR2 robot in a 3D environment, the authors do not use human social signals but an explicit five-star rating bar on a GUI as a workaround. Kalman temporal differences (Geist and Pietquin, 2010) are used for RL.

6.5. Implicit Feedback

In the context of post-stroke rehabilitation therapy, Tapus, Tapus, and Mataric (2008) use RL for behavior adaptation of an assistive therapist robot to the user's preferences during exercises for improving user engagement and motivation. The robot expresses personality in terms of extraversion/introversion through vocal content and para-verbal cues (see section 5.2.1.3). The authors realize non-functional adaptation: they adjust the parameters of these behaviors, namely interaction distance, speed, and vocal content of the therapist robot. Their experiment is task performance driven: the number of exercises performed in a given time defines the reward signal. Since the patient's recovery has the top priority, adaptation is triggered as soon as the monitored reward falls below a threshold, which indicates that the robot's behaviors do not result in ideal patient recovery. The threshold is adjusted over time to compensate for user fatigue and distraction introduced by the robot's adaptation process. Policy gradient reinforcement learning

(PGRL) optimizes three interaction parameters gradually: proxemics (interaction zones), activity (amount of robot movements/speed), and vocal content (introversion/extraversion). The authors use PGRL (which does not learn value functions for state-action pairs) since there is no notion of state in their setting. They also point out that the task-based measuring of the robot's efficiency in motivating the patient is highly subjective. It depends on the individual patient, who might act inconsistently and unpredictably.

Ramachandran, Sebo, and Scassellati (2019) use a POMDP for functional adaptation of a tutoring application with the NAO robot and a tablet. The tablet presents long-division math content to 4th-grade students. The robot provides feedback and guidance; the tablet is used as an input device and for displaying contents, questions, and feedback. The action space consists of six actions: presenting an interactive tutorial, presenting an example with comparable difficulty and the corresponding solution process, providing hints, requesting the student to think aloud, taking a break, and taking no action. At the same time, the agent monitors student knowledge and engagement. Both are calculated based on the timing and accuracy of the student's answers. The system interprets rapid guessing with wrong answers as boredom (i.e., low engagement) and honest attempts at the problem as high engagement. The knowledge level (little to no mastery, some mastery, moderate mastery, and near-complete mastery), engagement level (low, high), and the number of math problem attempts define the state space. Positive rewards are given for transitions from lower to higher knowledge and engagement states and vice versa. In addition, each action taken by the robot is penalized since it results in additional time for the student to complete the task. While relying on engagement, the feedback is based on task performance exclusively; it does not use social signals.

6.5.1. Facial Feedback

Gordon et al. (2016) use RL in the context of a student tutoring robot to maximize long-term learning gains. The setup combines a mobile app on a tablet, which presents educational content and a virtual animated Toucan character, with a Tega robot. The robot acts as a peer tutor on the child's level and provides instructions, hints, encouragement, and appropriate gaze toward the tablet or child. The virtual character and the robot use scripted behaviors with pre-recorded voice scripts. During the interaction, the child's valence and engagement are estimated based on facial expressions in real-time with proprietary software. Similar to Leite et al. (2011), Gordon et al. do not use posture or gesture information. The RL agent implements a non-functional adaptation approach. The state space includes the child's affective state (discretized valence and engagement) and task-related information (whether the child interacted within the last 5 seconds and whether the last response was correct). The action space represents the robot's supporting affective reactions. It is modeled as a combination of valence and engagement and a no-operation action, resulting in no affective response. The weighted sum of the child's valence and engagement (sensed from facial expression) defines the reward. Traditional SARSA algorithm (Sutton and Barto, 2018) is used for RL.

Gamborino and Fu (2018) investigate a functional adaptation approach with interactive RL. The socially assistive RoBoHoN robot aims to improve the mood of children who visit or stay in the hospital. The action space consists of four categories of verbal and non-verbal robot behaviors: speech, head motion, body gestures, and full-body motions. Each category comprises several subcategories, such as dialogs with jokes and stories. The robot's camera monitors the child's facial expressions. The system analyzes them with SSP and calculates scores for seven core emotions. These affective features are combined with estimated engagement and serve as dimensions of the discrete state space representing the child's mood. Human explicit feedback from a human trainer, who acts as a wizard in the WoZ experiment, results in reward shaping. The agent presents four suggested actions on a GUI. The trainer manually picks one of the suggested or other actions, which results in a shaping reward. In addition, policy shaping with positive and negative policy rewards encourages the agent to transition from bad mood states to good mood states and to prevent transitions to bad mood states. Both shaping reward and policy reward serve as combined reward value. Consequently, the child's affective state (estimated based on facial expression) also implicitly impacts the reward signal. The authors use the SARSA algorithm for RL. The classification of their work as explicit or implicit feedback is not straightforward. From the child's perspective, feedback is provided implicitly, but there is also explicit feedback from the trainer's perspective.

Park et al. (2019) use RL for functional adaptation of a learning companion robot in dialogic storytelling, which aims to foster early literacy and English language skills. In each session, the Tega robot tells a story to a preschool-age child and asks lexical, factual, inferential, and emotional questions, which allow for assessing the child's engagement and comprehension of the story content. After the robot finishes, the child is invited to retell the story in its own words. While the child answers questions and retells the story, its speech samples are analyzed to assess its lexical and syntax skills. Moreover, the affective cues in facial expressions are processed to estimate engagement. After each session, RL personalizes the content selection to the individual child: the action space contains six actions, which explore stories with different lexical and syntactic complexity, resulting in one action per session. The reward is the weighted sum of engagement and task performance. The state space with 20 states consists of the child's task performance (whether questions are answered or not, with prompt or without prompt, length of the utterance) and affective arousal (in four levels). Q-learning with decreasing ϵ and decreasing α updates the policy after each session.

6.5.2. Multimodal Feedback

Mitsunaga et al. (2005) and Mitsunaga et al. (2008) focus on subconscious human feedback for non-functional adaptation of a Robovie-II's behaviors. They reason that giving conscious feedback might interfere with or distract from the actual interaction. PGRL (Kohl and Stone, 2004) uses human body signals as reward signal, relying on the user's time spent gazing at the robot's face and movement distance. It aims to minimize the user's discomfort with the interaction by adjusting six parameters: three parameters for interaction distance, the duration the robot looks at the user's face, the duration the robot waits until it presents a gesture after talking, and the gesture speed. The authors use a motion capture system for measuring human movements, orientation, and interaction distance. A weighted sum of the user's movement distance and gaze meeting determines the reward value.

Leite et al. (2011) learn a robot's empathic behaviors with RL in the context of a chess companion for children. The main goal is to keep the child engaged and motivated. To this end, the iCat robot uses a functional adaptation approach. The action space contains supportive behaviors: encouraging comments, scaffolding, suggesting a good move, and intentionally playing a bad move. A camera monitors the child to estimate its affective state. Besides visual affective cues in terms of facial expression (e.g., smiling, mouth fidget) and gaze (e.g., looking at the robot vs. looking elsewhere), task-related features take the game evolution and chess board configuration into account from the child's perspective (e.g., captured pieces). These data allow the robot to estimate the probability of positive feeling before and after employing a supportive strategy. The learning agent's computational goal is to maximize the child's positive valence throughout the game: the difference in those probabilities before and after executing the action serves as a reward for the learning process. Thus, both human social signals and task-related information contribute to learning. Adaptation is implemented as a k-armed bandit problem.

Chiang et al. (2014) use RL for non-functional adaptation of an ARIO robot's interruption strategy. The robot has six primitive actions to grab the user's attention, including gestures (waving the arm, shaking the head), locomotion (approaching the person, moving around), and audio (making a sound, calling the person's name). The state space has two dimensions: the human's awareness of the robot and the intensity of human attention towards the robot. It is divided into engaged (no attention, user noticed the robot, lose attention, user looks at robot) and non-engaged (neglect, medium attention, high attention) states. Human attention is inferred based on a hidden Markov model (HMM) from human non-verbal social signals: the user's current pose (face and body heading direction) and voice activity. The reward signal results from the current state: the more engaged and attentive the user, the higher the reward. The Q-learning algorithm implements the RL. First, the authors evaluate their human attention estimator in a WoZ experiment, where participants label their attention level by themselves based on the recorded video of their interaction. Afterward, the authors evaluate the adaptation approach and conclude that the best policy varies from user to user. For example, one subject preferred the robot to play a sound to inform about the interruption with subsequent movement toward the participant. Another subject preferred approaching the user first and then calling their name.

6.6. Limitations and Research Gaps

Martins, Santos, and Dias (2019) point out several research gaps in user-adaptive interaction in social robotics. These gaps are also reflected in the presented literature. This thesis addresses some of these important gaps. They are listed below in combination with several differences between the presented literature and this thesis. First, the focus is on differences in experiment design and RL. See Table 6.1 for a complementary overview.

- The majority of research investigates functional adaptation (Martins, Santos, and Dias, 2019). Non-functional adaptation, which addresses *how* a robot interacts with the user (e.g., by tweaking parameters of its multimodal communication strategy), is less explored. The thesis at hand focuses on non-functional adaptation.

- The adaptation of robot personality, persona, politeness, and humor has not yet been explored in the literature extensively. The work at hand focuses on these aspects due to the focus on domestic companion robots.

- Many experiments use goal-directed tasks, which provide measurable data regarding the success or failure of adaptation per se. In such experiments, the reward signal of the environment can be inferred from task-related data, such as performance. Reward shaping or similar techniques guide the learning agent based on human feedback. The thesis at hand focuses on tasks that do not have a measurable task-related goal. For example, the robot provides recommendations without having the ability to verify whether users accept them (see section 13.1) or entertains the audience (see section 14.1 and section 14.3). Thus, human feedback is all the more important for adapting the robot to the individual user. In chapter 14, human social signals are the only source of feedback for the adaptation process.

- Continuous adaptation with completely autonomous agents, which do not require intervention from technical personnel, remains the exception (Martins, Santos, and Dias, 2019), which is the case in the presented literature, too. A few experiments use WoZ techniques, where a human operator controls the robot. The thesis at hand investigates autonomous adaptation approaches exclusively.

- Almost all experiments are evaluated in controlled laboratory environments. Very few works use in-situ studies for evaluating adaptation in real life (Martins, Santos, and Dias, 2019) with the target population in target environments. The works in the presented literature, which carry out in-situ studies, focus on young children. In contrast, the work at hand focuses on social companion robots for domestic environments. Specifically, one of the experiments addresses the less represented elderly user base with an in-situ study in chapter 13.

- A portable, relatively small, and low-cost robot is required in this thesis for an in-situ study in participants' domestic environments. The Reeti robot (see chapter 11) is used for this purpose since it provides an expressive face. The presented literature did not yet use it for adaptation (see also Table 6.1).

In addition, the following observations exist for the used modalities and design of human social feedback in the presented literature. See also Table 6.2 for a complementary overview.

- Most literature uses explicit human feedback, where the user engages actively and consciously in providing feedback to the adaptation process. The work at hand explores both explicit and implicit feedback, focusing more on the latter. The use of implicit feedback makes it possible to adapt to the user without interrupting the interaction for the additional effort of providing feedback.

- The literature often infers feedback for adaptation from signals via GUIs or hardware interfaces. Concerning verbal and non-verbal communication, the literature uses vocal and facial feedback most often. Tactile, gestural feedback, and feedback via gaze and pose are used the least. The thesis at hand explores both explicit feedback via hardware buttons and implicit multimodal feedback, including pose, gestures, facial expression, and vocal behaviors.

- This thesis does not use task-related feedback. In goal-oriented tasks, the user needs to solve a problem, which results in a final environmental reward or measurable user performance. In non-goal-oriented tasks, other interaction dynamics, user opinions, and human social signals play a crucial role.

- Some literature relies on higher-level interaction dynamics for motivating or modeling the feedback mechanism. User affect and engagement are used most often, followed by appraisals. This thesis relies on user affect and engagement in chapter 14 and on user appraisals in chapter 13.

In combination with the experiments and insights reported in chapter 5, this results in the following overall picture of non-functional and real-time adaptation of multimodal social robot behaviors:

- The presented literature with robots expressing personality use scripted robot behaviors and adaptation based on static user models. The thesis at hand combines the real-time generation of the robot's spoken language and real-time adaptation of the generated behaviors based on implicit feedback derived from human social signals. By using the RL framework, the autonomous learning agent adapts to whatever the situation requires to re-engage the user in the interaction. This generic and task-independent model is neither restricted to the similarity nor complementarity attraction principle since the literature reports various findings.

- The expression of different robot personas has not yet been explored extensively in the literature. While experiments are comparing hypothetical robot personas without being implemented, and very few experiments investigating robot personas, no experiment has yet used an autonomous adaptation approach for evaluating different personas in HRI. This thesis addresses this issue in combination with politeness in an in-situ study.

- All presented literature in the context of robot politeness uses scripted verbal behaviors, sometimes combined with gestures or movements. This thesis is the first one combining it with a RL approach for real-time adaptation of the robot's verbal politeness and persona based on explicit human feedback. An in-situ study with elderly participants evaluates the approach in the German language.

- In most presented experiments addressing robot humor, the behaviors are scripted. In about half of the works, visual and sometimes auditive input is used for estimating the audience's amusement – primarily for functional adaptation of the robot's content selection. The thesis at hand presents approaches for generating, presenting, and adapting multimodal robot humor. Specifically, the non-functional real-time adaptation addresses generated para-verbal and non-verbal robot behaviors for the first time.

6.7. Conclusion

User-adaptive interaction requires information about the user. In social robotics, this information includes task-related data, such as user performance in goal-oriented tasks, but also social feedback, which is often inferred from human verbal or non-verbal behaviors. This explicit or implicit feedback allows the adaptation process to learn about individual user needs or preferences. Implementing social robot adaptation requires a social interface (which both produces artificial social behaviors and senses human social feedback), a decision-making process, and a user model. Functional adaptation controls the robot's function; non-functional adaptation controls how certain functions are executed.

The literature reports several experiments in non-functional social robot adaptation, although there is more research regarding functional adaptation. This thesis contributes to the combination of RL with human social feedback for non-functional multimodal behavior adaptation, i.e., the adaptation of verbal and non-verbal robot behaviors. In this context, the literature overview has outlined existing approaches and how the work at hand extends and differs from these works. To sum up, the most significant research gaps are as follows:

- There is not yet a generalized view on the integration of human explicit and implicit social feedback in the RL framework, especially with regard to verbal and non-verbal social signals. Chapter 12 fills this gap.
- Almost all experiments use scripted robot behaviors; very few experiments generate robot behaviors during runtime. In particular, this also applies to experiments realizing non-functional adaptation. Part III fills this gap for personality, persona, politeness, and humor in the context of a domestic robot companion.
- The non-functional adaptation of personality, persona, politeness, and humor with RL has not yet been explored extensively. This gap is addressed in Part IV. Politeness and persona have not been explored in combination with adaptation. Chapter 13 fills this gap with a RL approach and an in-situ evaluation. Experiments with personality have focused to a large extent on personality matching, typically with scripted robot behaviors. Chapter 14 fills this gap with a more generic RL approach and dynamic generation of the robot's verbal behaviors. The literature made a few attempts to adapt humor primarily to larger audiences, typically with scripted behaviors. Chapter 14 fills this gap with a RL approach for personalizing manually designed and dynamically generated multimodal robot humor, including adapting para-verbal robot behaviors.
- The majority of works are evaluated using the English or Japanese language and related cultures. The thesis at hand uses German and English, depending on the experiment and required technologies.

The combination of real-time generated robot behaviors and real-time non-functional adaptation of these behaviors with RL based on explicit or implicit human social feedback is the central contribution of this thesis.

Part III.

Behavior Generation

7. Storytelling with Personality

The expression of personality is very important for social robots. Equipping them with a compelling personality profile can make the interaction more engaging (Breazeal, 2004). This chapter focuses on extraversion (see section 4.1.1) since it is the dimension with the most influence on language (Furnham, 1990), "the easiest trait to model from spoken language" (Mairesse, 2008) and plays an important role with regard to evaluating robot personality (Woods et al., 2005) and interpersonal compatibility (see section 4.2).

This chapter presents a rule-based approach for generating utterances with varying degrees of extraversion in the context of storytelling. It transfers the approach by Mairesse and Walker (2011) from the context of restaurant recommendations to the storytelling domain. It enriches character descriptions and plot summaries of the book "Alice in Wonderland" with the expression of extraversion/introversion. A knowledge base with structured facts about characters and the plot is a basis for the subsequent NLG. Based on this information, the linguistic content is generated during runtime instead of preparing several alternative formulations in advance.

The concept and implementation were part of the works presented and reviewed in Ritschel and André (2017), Ritschel, Baur, and André (2017a), Ritschel, Baur, and André (2017b), and Ritschel (2018). The contents of this chapter expand these publications. Section 14.1 relies on this work for adapting the robot's personality to the individual user's preferences.

7.1. Knowledge Base

The knowledge base contains information about the main characters and the plot of the chapters. It includes a simplified summary of each chapter, which makes it possible to re-tell each excerpt in a few minutes. Moreover, each character is associated with several attributes and facts designed to convey a general impression of the character's role, strengths, weaknesses, and personality. All contents were prepared as JavaScript object notation (JSON) files.

Figure 7.1 illustrates the general, abstracted idea based on the two characters Alice and the White Rabbit. Each character is associated with several attributes, including adjectives, such as *imaginative* or *anxious*, and general information, such as name, external appearance, role, ownership of items, and much more. These data are kept as generic as possible, meaning they apply to the respective character, preferably throughout all chapters. Temporary data (e.g., a character's mood to a certain point in time) is part of the data associated with the plot of each chapter. As Figure 7.1 illustrates, grammatical information supplements the attributes and facts in order to simplify NLG.

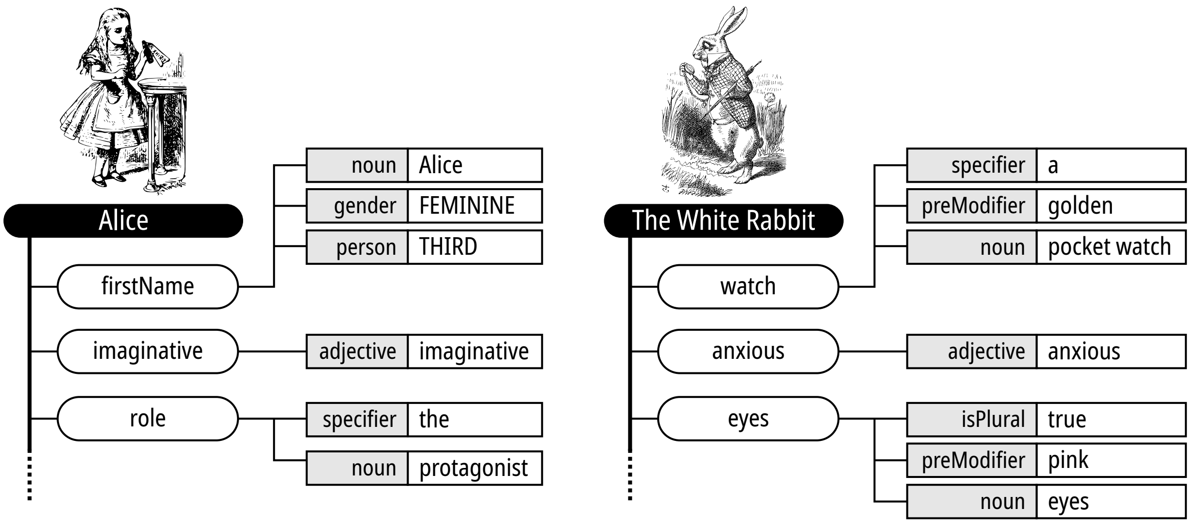


Figure 7.1.: A simplified illustration of the data store containing attributes of different complexity about the main characters.

7.2. Natural Language Generation

The NLG module is responsible for producing natural utterances for the robot that reflect a particular type of personality. Inspired by the PERSONAGE system by Mairesse and Walker (2011), it applies a set of parameters in different stages of generation depending on the robot's degree of extraversion $X \in [-2; 2]$ (integer interval). The values represent *very introvert* (−2), *introvert* (−1), *neutral* (0), *extravert* (1) and *very extravert* (2).

X is an integer value instead of a floating point variable for two reasons. First, X is part of the discrete state space in a RL process in section 14.1.4. Second, X indicates the degree of extraversion only roughly because X serves as a base value. The actual NLG parameter values are randomized based on X to a certain degree (see section 7.2.1) for linguistic variety in the generated utterances while maintaining roughly the same amount of extraversion. The process implements the traditional, pipelined NLG architecture (Reiter and Dale, 2000):

1. *Content planning* selects the attributes and facts presented when talking about characters or the events during re-telling the plot of the book chapters. For example, an introvert robot will be less verbose and present less information in one statement, while an extravert one may use restatements to reinforce its statement.

2. *Sentence planning* arranges the selected contents within one utterance. It orders, aggregates, and sets the final lexical item for each word. For example, an extravert robot may emphasize adjectives or use expletives, while an introvert robot will soften them or use a double negation.

3. *Surface realization* transforms the abstracted representation of the previous steps into a string, which the robot finally presents with the internal TTS system.

Figure 7.2 illustrates the process with two essential inputs: the data from the knowledge base and the current extraversion X. The latter determines the set of NLG parameters

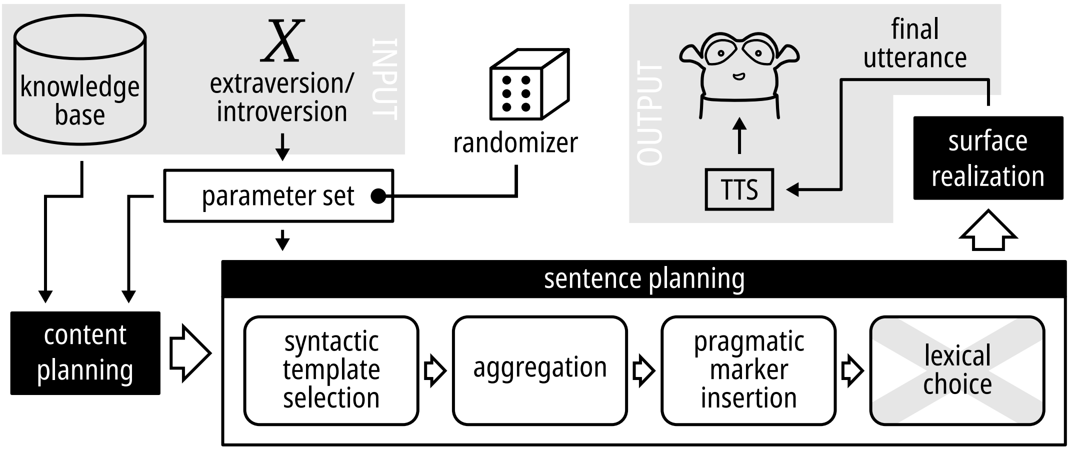


Figure 7.2.: Overview of the NLG pipeline. Adapted from Mairesse and Walker (2011).

to apply during generation, i.e., which ones of the specific cues about extraversion or introversion are employed. A threshold is assigned for each NLG parameter: X does not activate or deactivate parameters directly but increases or decreases their probability of applying them. This approach aims for more variation in the generated output. Given a fixed X, it prevents repeatedly applying the same parameters for each generated description, which would cause utterances to be stylistically too similar and may annoy the user. Thus, the content and sentence planning parameters are used only with certain probabilities. The parameter thresholds and probabilities are set by hand since some parameters result in a more significant and noticeable change in the output than others. For example, those with a highly distinctive effect, such as stuttering for the minimum $X = -2$, have lower probabilities.

The parameters and their implementation are similar to the PERSONAGE system by Mairesse and Walker (2011), which provides details on the parameters and how they map to extraversion/introversion. In contrast to Mairesse and Walker, the contents and data structures are optimized for the storytelling scenario, which differs from the restaurant recommendations. The description of the main characters is similar to the description and comparison of restaurants in Mairesse and Walker (2011). Especially when talking about the plot, the re-narration of the book chapters is structurally different from descriptions and comparisons. Technically, the implementation at hand uses the open-source *SimpleNLG* library (github.com, 2021) as surface realizer in the last stage instead of the proprietary RealPro framework in Mairesse and Walker (2011). The WordNet®database (Miller, 1995) is used for several purposes, e.g., for looking up synonyms or antonyms.

7.2.1. Parameter Set

Table 7.1 lists all implemented NLG parameters. A description with examples of almost all parameters can be found in Mairesse and Walker (2011). A few modifications were made for the storytelling scenario. Moreover, some parameters, such as probabilities for concessions or restatements, can be specified individually for more control.

Table 7.1: The parameter set of the NLG pipeline.

Parameter	Type	Initialization
Content planning		
Verbosity	I	$\mathrm{round}(4 \cdot \mathrm{randomize}(x + 0.2 \cdot (1 - x)))$
Restatements	I	$\mathrm{round}(\mathrm{randomize}(0.8x))$
Repetitions	I	$\mathrm{round}(\mathrm{randomize}(0.8x))$
Content polarity	F	$\mathrm{randomize}(0.8x - 0.8 \cdot (1 - x))$
Repetition polarity	F	$\mathrm{randomize}(0.8x - 0.8 \cdot (1 - x))$
Request confirmation	B	$\mathrm{randomize}(0.8x) > 0.5$
Initial rejection	B	$\mathrm{randomize}(0) > 0.5$
Competence mitigation	B	$\mathrm{randomize}(0) > 0.5$
Positive content first*	B	$\mathrm{randomize}(0.85x) > 0.5$
Syntactic template selection		
Self-references	B	$\mathrm{randomize}(0.8x) > 0.5$
Template polarity	F	$\mathrm{randomize}(x)$
Aggregation operations		
Period	P	$0.01x + 0.1 \cdot (1 - x)$
Relative clause	P	$0.4 \cdot (1 - x)$
With cue word	P	$0.1x + 0.3 \cdot (1 - x)$
Conjunction	P	$0.1x + 0.3 \cdot (1 - x)$
Merge	P	$0.5x + 0.2 \cdot (1 - x)$
Also cue word	P	$0.1x$
Concession contrast*	P	$0.8x + 0.2 \cdot (1 - x)$
Concession concede*	P	$0.2x + 0.8 \cdot (1 - x)$
Restate conjunction*	P	$0.5x + 0.2 \cdot (1 - x)$
Restate comma*	P	$0.5x + 0.4 \cdot (1 - x)$
Restate object ellipsis*	P	$0.4 \cdot (1 - x)$
Elaborate period*	P	$0.125x + 0.33 \cdot (1 - x)$
Elaborate conjunction*	P	$0.3625x$
Elaborate merge*	P	$0.625x + 0.67 \cdot (1 - x)$
Pragmatic markers		
Subject implicitness	B	$\mathrm{randomize}(x) > 0.5$
Stuttering	B	$\mathrm{randomize}(1 - x) > 0.5$
Pronominalization	B	$\mathrm{randomize}(0) > 0.5$

Continued on next page

Table 7.1: The parameter set of the NLG pipeline. (Continued)

Parameter	Type	Initialization
Negation	B	$\text{randomize}(1 - x) > 0.5$
Softener hedges	B	$\text{randomize}(1 - x) > 0.5$
Emphasizer hedges	B	$\text{randomize}(x) > 0.5$
Acknowledgements	B	$\text{randomize}(x) > 0.5$
Filled pauses	B	$\text{randomize}(1 - x) > 0.5$
Exclamation	B	$\text{randomize}(x) > 0.5$
Expletives	B	$\text{randomize}(x) > 0.5$
Near expletives	B	$\text{randomize}(x) > 0.5$
Tag question	B	$\text{randomize}(x) > 0.5$
In-group marker	B	$\text{randomize}(x) > 0.5$

* Parameter was introduced in the work at hand.
Note: Inspired by Mairesse and Walker (2011). $x \in [0; 1]$ is the normalized value of X. Types: I = integer, F = floating point $\in [0; 1]$, B = boolean.

Similar to Mairesse and Walker (2011), there are different types of parameters. *Verbosity*, *Restatements* and *Repetitions* determine a maximum amount in terms of an integer value. *Content polarity*, *Repetition polarity* and *Template polarity* are floating point values which determine the amount and selection of positive or negative contents. The rest of the parameters represents either a probability for applying the parameter or a binary on/off switch.

The generator creates a new parameter set for each utterance. The parameters relating to *content planning* determine which contents are selected for presentation, while the remaining ones for *sentence planning* control *how* the content is formulated. All parameters are initialized based on the desired extraversion $X \in [-2; 2]$, which gets normalized to $x \in [0; 1]$. The last column in Table 7.1 lists each parameter's initialization formula. A central part of these calculations is the function *randomize*, which is essential for the variation in the generated utterances. It modifies the input value $v \in [0; 1]$, which is hand-tuned as a function of x, by applying a fraction $f \in [0; 1]$ of a random value $r \in [0; 1]$:

$$\text{randomize}(v) = (1 - f) \cdot v + f \cdot r$$

7.2.2. Examples

Figure 7.3 illustrates an exemplary, simplified content plan for describing the main character Alice. It shows that each content from the knowledge base is assigned a *polarity* in the interval $[-1; 1]$. It indicates whether this data has a positive or negative connotation for content planning. For example, Alice's lack of sensitivity can be interpreted as a negative aspect (−0.3), while her courteousness represents a positive one (0.6). This information is used to select the next facts for presentation based on the polarity parameters. The individual elements from the knowledge base pass on the polarity to the higher-level

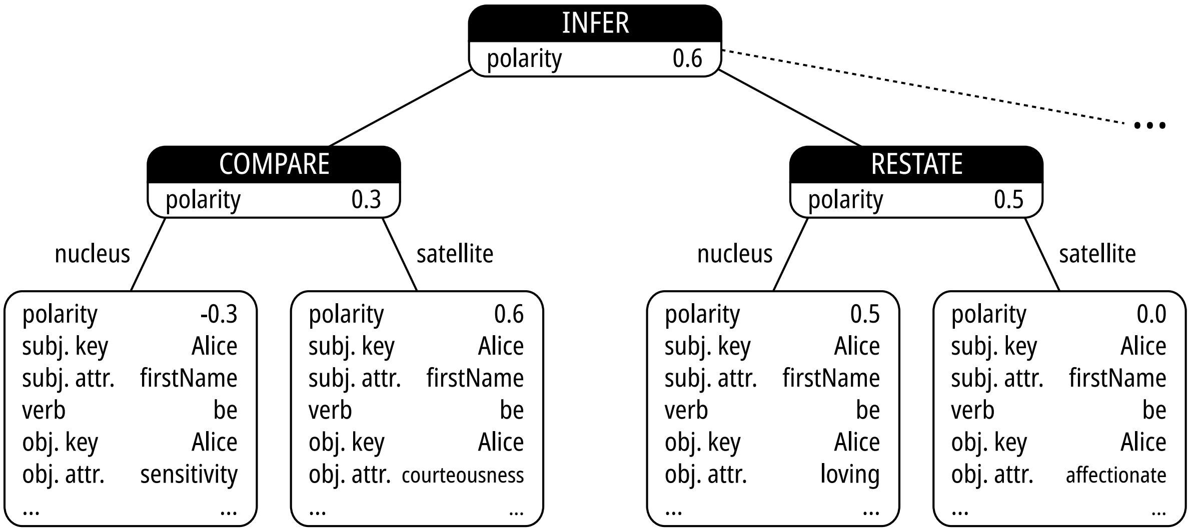


Figure 7.3.: A simplified content plan for talking about Alice.

rhetorical relation. The relation determines the combination of the elements, such as a comparison, restatement, elaboration, justification, and more. Polarity is also essential in a later stage during sentence planning, where it decides the order of the facts within one sentence. For example, an extravert and an introvert robot differ in whether they tend to present negative or positive content first.

Figure 7.4 illustrates another exemplary, simplified content plan, but for the presentation of the plot. A significant difference is that there is no polarity in this context since the correct order and completion of the story are most important. Changing the order of events would not make sense. The plot is re-narrated in the past tense; character facts use the present tense.

As a result, Figure 7.5 presents samples that result from differently configured parameter sets. The samples describe Alice with a maximum introvert, a neutral, and a maximum extravert personality. One obvious difference is utterance length, mainly due to the different content plans. In the case of high extraversion with high verbosity, the robot presents a larger number of propositions (i.e., facts) within one utterance. Moreover, the introvert robot uses fewer positive emotion words and softens positive content ("somewhat imaginative"). The extravert robot uses many positive emotion words ("loving", "affectionate") and repeats itself ("imaginative"). In case of high extraversion the use of acknowledgments ("I see"), tag questions ("isn't she?") in-group markers ("buddy") and expletives ("damn") is more likely. Stuttering ("Al-Al-Alice") and softener hedges ("somewhat") are typical for introvert utterances.

7.3. Conclusion

Equipping a social robot with a compelling personality profile is one step toward making interaction more engaging. This chapter presented a rule-based approach and implementation for generating utterances with varying degrees of extraversion and introversion in the context of storytelling. It transfers the work by Mairesse and Walker (2011) from

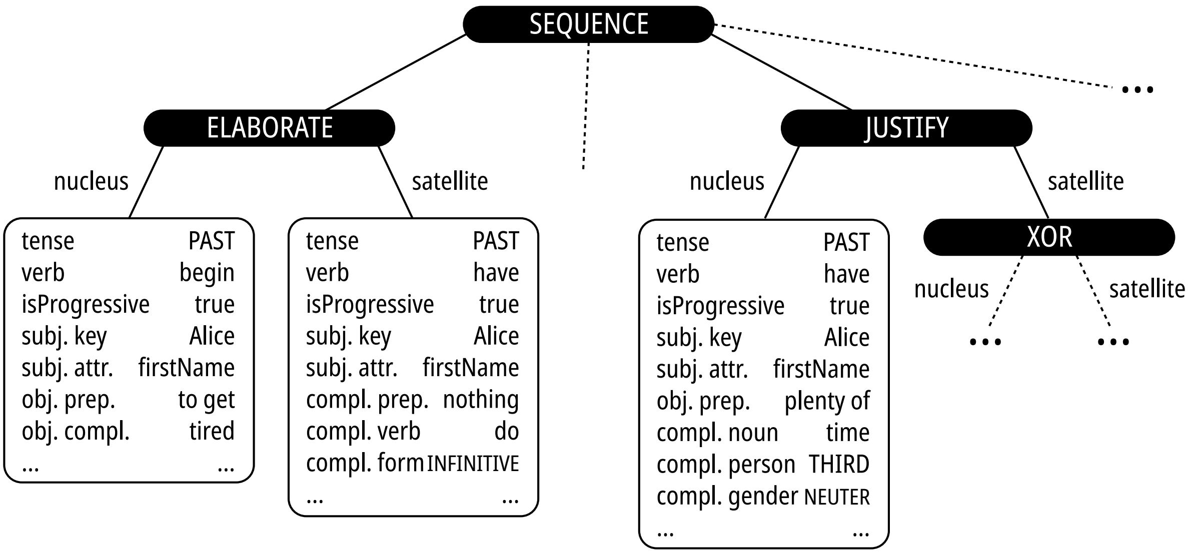


Figure 7.4.: A simplified content plan for telling the plot of a chapter.

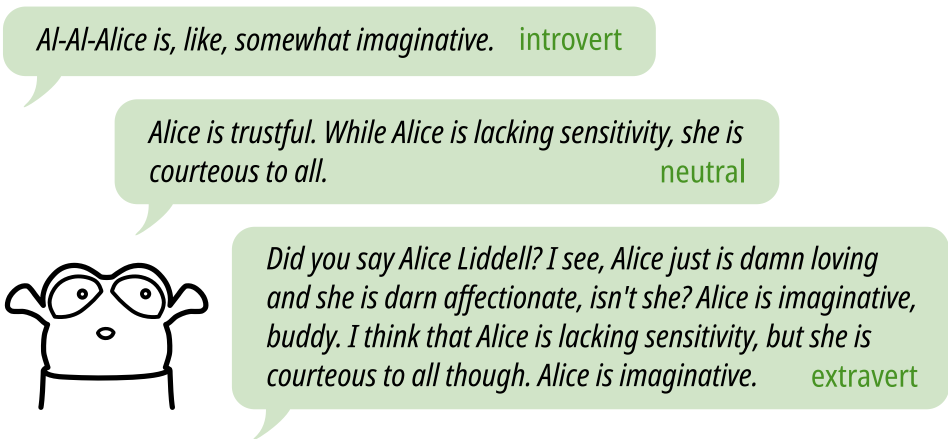


Figure 7.5.: Examples of generated descriptions with different degrees of extraversion.

the context of restaurant recommendations to the storytelling domain. Structured data serve as a knowledge base, which provides information for the robot's narrative. Facts about the characters and plot of the book "Alice in Wonderland" are transformed into utterances with NLG techniques during runtime. The implementation uses the custom *reeti-rest* software (see chapter 11) for playing back animations and interfacing with the Reeti robot.

As described in section 5.2.2 there are varying insights concerning similarity and complementarity personality attraction, as well as other findings, such as the dependence on task context. Thus, the generation process at hand serves as a basis for adapting the robot's verbal behaviors in terms of expressed personality in section 14.1 automatically. There, an RL approach aims to address individual user preferences by manipulating the robot's extraversion/introversion. It uses the configurable NLG pipeline at hand.

8. Assistive Support with Persona and Politeness

Besides entertainment, assistance and information retrieval are everyday tasks for social robots in domestic environments. Remembering upcoming appointments, informing about news and events, and health-related advice are core features of research and consumer companion robots (see section 5.1.5). Today, smartphones, mobile apps, and digital voice assistants already assist in these tasks in everyday life. With robotic companions entering everyday human life and domestic environments, they have the potential to take on such tasks, too. At the same time, how these machines communicate with users will be of central importance for a long-lasting, positive user experience. Chapter 7 already presented a rule-based approach for a robot's expression of personality in a storytelling context. Motivated by the links between personality, persona, and politeness (see chapter 4), the focus is now on the robot's expressed politeness and persona when giving advice and providing assistance, but also in the context of entertainment.

This chapter presents an autonomous domestic robotic companion, which can communicate with different personas and politeness strategies in assistive and entertainment functions. First, the overview covers the robot's functions. They address basic needs and are used in the literature and current robotic products on the consumer market. Afterward, the robot's communication abilities are presented, which rely on contents in the German language. The prototype serves as a basis for adapting the robot's verbal behaviors to the individual user's preferences in section 13.1 automatically.

The implementation at hand uses the Reeti robot and the custom *reeti-rest* software for multimodal animation generation (see chapter 11). It was presented and reviewed in Ritschel et al. (2019d). The contents of this chapter expand this publication.

8.1. The Assistive Robotic Companion

Figure 8.1 outlines the setup. The stationary robot acts as a companion in the user's home. The user can use it at any time. In addition to the speech output and animation, the setup includes a screen for graphical output (menus and virtual board games) and a hardware control panel for input (navigation through menus and feedback for the adaptation process). The setup considers privacy concerns of many study participants regarding automatic facial or speech recognition. Audio is played back with the robot's internal speaker.

Inspired by former experiments in the literature and common features in recent commercial consumer products, the companion offers typical assistive functions (see also section 5.1.5). These applications cover entertainment, information retrieval, health-related recommendations, and communication. Figure 8.2 illustrates an overview and

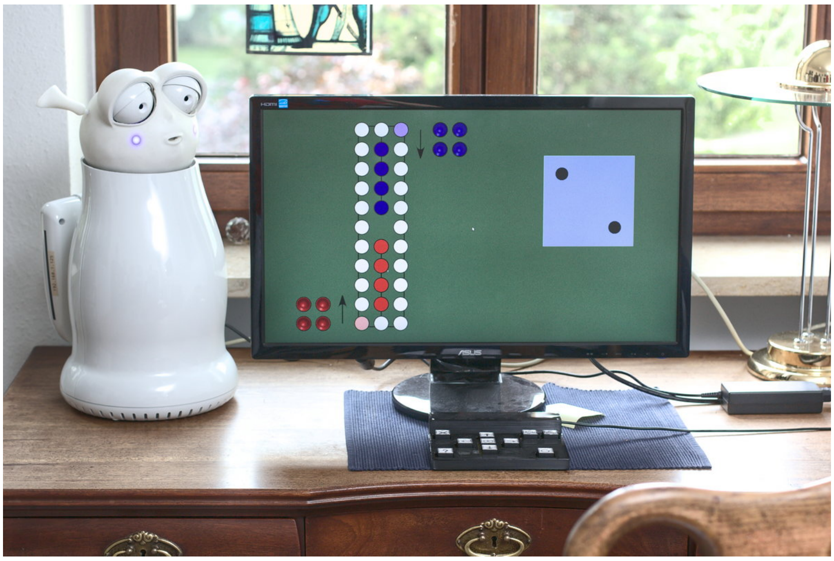


Figure 8.1.: Setup of the robotic companion in one of the study participant's domestic environments, including the robot, screen, and hardware control panel.

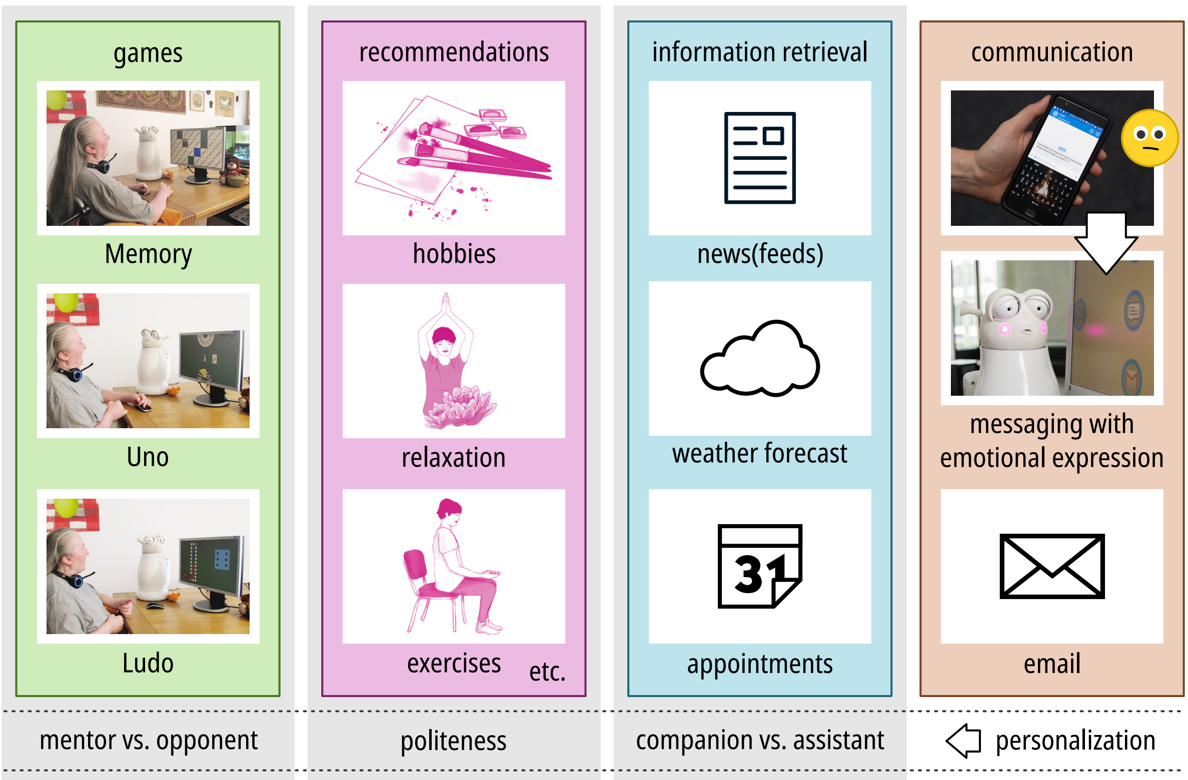


Figure 8.2.: Overview of the robot's assistive and entertainment functions.

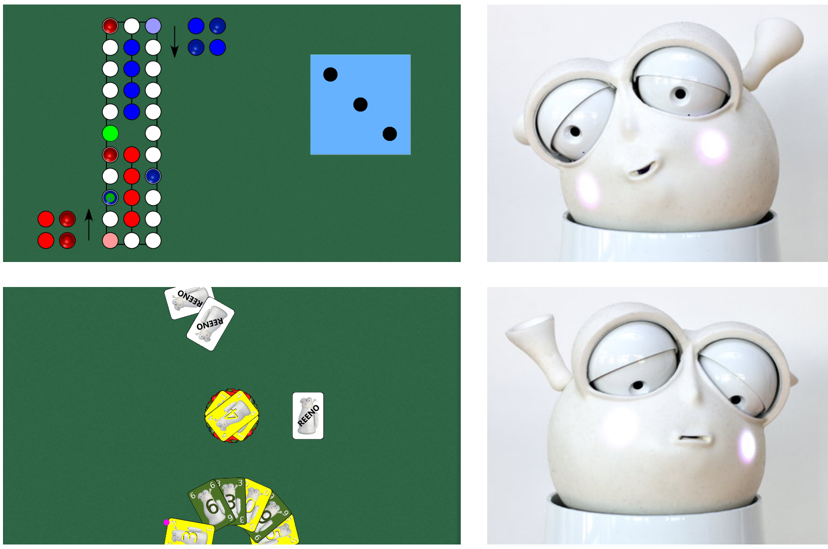


Figure 8.3.: User interface and robot facial expression examples.

the categorization of all applications. The categories also serve as the context for the RL approach in section 13.1, where every application category explores and manipulates either the robot's expressed persona or politeness.

8.1.1. Assistive Functions

The assistive applications cover four categories: entertainment, information retrieval, health-related recommendations, and communication.

Entertainment includes a set of virtual board games: *Memory*, *Uno* and *Ludo*. All games exist as a two-player version: the user plays against the social robot, which acts either with a mentor or opponent persona. Ludo's board size is optimized for two players, resulting in a smaller game board (see Figure 8.3) for reduced playing time. The screen next to the robot displays the virtual game board, cards, tokens, and game statistics. Furthermore, the robot tells jokes from different categories in a joke-telling application.

Information retrieval applications provide the user with news, weather forecast, and contact information about friends and family members. When the user is interested in news feeds, the robot reads out loud headlines and the abstract of recent articles. News sources are configurable and collect the data from the Internet via RSS/ATOM feeds. The robot generates a corresponding description when triggering the weather forecast for the next seven days. It also gives appropriate hints, such as having an umbrella or suitable clothing when it gets cold. The OpenWeatherMap service (openweathermap.org, 2021) provides weather data over the internet. The robot also informs about upcoming appointments in the personal calendar. The system triggers automatic reminders in configurable intervals. Furthermore, an overview and details of contact information can be retrieved. Corresponding data has to be provided by the user.

Health-related recommendations are designed for single-living elderly users to support their independence. They address physical, mental, and environmental well-being and encourage activities, e.g., about hobbies, relaxation, exercises, and much more. For example, suggested activities include reading books, airing and turning up the heating in the morning, doing gymnastics, listening to music, watering the plants, drinking enough, and much more. The contents are adapted from the CARE research project (Rist et al., 2015; Seiderer et al., 2015) and include encouraging activities, e.g., for hobbies, relaxation, exercises, and much more. The CARE project uses a digital picture frame for presenting recommendations with a textual description and an illustrating image or animation based on context information acquired by sensors in the elderly user's domestic environment. In contrast, the robot at hand presents the recommendation's text to the user directly with speech. Additionally, the screen displays a corresponding image without the textual description.

Basic *communication* features provide email and instant messaging applications to receive messages from friends and family members. For this purpose, the user can pass on the robot's email address and instant messenger ID to associated persons. As soon as the robot receives a message, it calls the user's attention and, if desired, presents the content. In addition, an XMPP chat client receives text messages. When presenting incoming messages, the robot uses facial expressions to express emoticons in the text (see section 8.4).

8.1.2. Control Panel

Interaction with the robot and system is realized completely with a custom hardware control panel with lit physical buttons. This decision was made instead of a touch panel due to issues reported in the literature concerning touch-based interaction. For example, Motti, Vigouroux, and Gorce, 2013 point out that touch-based interaction may alleviate barriers to getting started for elderlies. However, at the same time, they detect several usability issues, such as the timing of tapping gestures. Probably, the users encountered similar timing issues during tapping, as reported in Leonardi et al., 2010. Due to these potential problems, later versions of the CARE system used a combination of touch screen and hardware buttons (Rist, Seiderer, and André, 2018). The system does not use automatic speech or face recognition due to participants' privacy concerns in their domestic environments.

Figure 8.4 illustrates the setup and functions of the control panel in detail. It includes buttons for (1) giving feedback to the adaptation process (see section 13.1), (2) navigating through menus, (3) confirming selections or starting an application, (4) canceling or quitting an application, (5) repeating the robot's last utterance in case of poor understanding and (6) accessing help. See Ritschel et al., 2019d for details on the electronics, the building, and the 3D printing of the device.

8.2. Personas

In each of the three application contexts, the robot uses a set of different linguistic variations. Each application context focuses on one aspect: either the robot's persona or its politeness. These aspects are expressed in terms of language as follows.

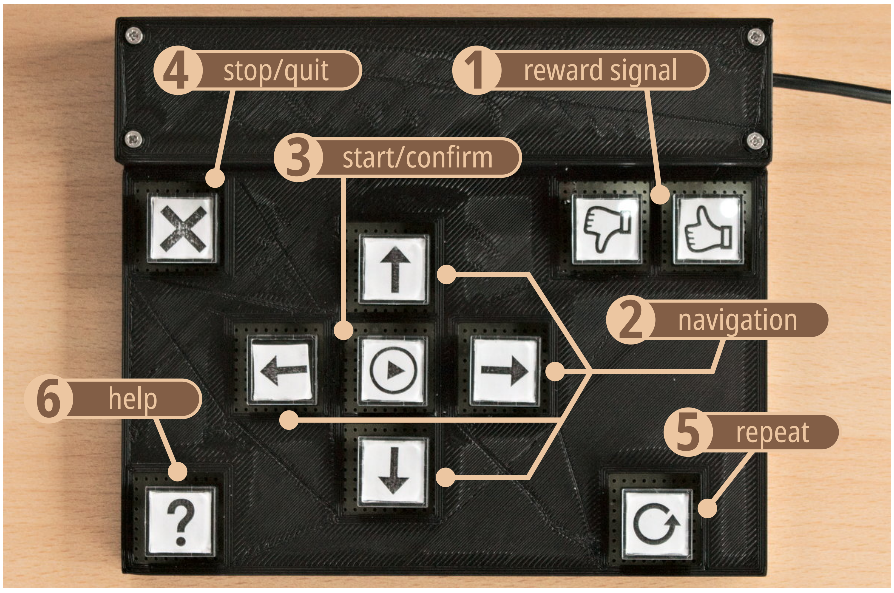


Figure 8.4.: Control panel with physical, illuminated buttons for interacting with the robot.

8.2.1. Mentor vs. Opponent

In the context of a poker game, Gebhard et al., 2008 suggest enriching virtual agents with an affective model, which influences the agent's reactions to the course of play. The adaptive robotic companion implements this idea in terms of two different personas. The robot behaves and expresses itself either as a *mentor* or an *opponent* when playing virtual board games with the user. Both personas are configured based on the Five Factor personality model (see section 4.1.1). They differ significantly concerning the dimensions agreeableness and neuroticism. In addition, game events, such as losing a token or successfully revealing a pair of cards in the Memory game, are appraised based on the OCC model (Ortony, Clore, and Collins, 1988) from the perspective of the robot. Depending on the persona, the same event elicits different robot emotions.

The mentor acts more agreeable, less neurotic, and reinforces positive emotions. As a result, it is configured to like the user, show positive emotions, and expresses empathy. Moreover, it presents collaborative comments and reactions and behaves towards the shared goal of enjoying the gameplay with the user.

The opponent is less agreeable, more neurotic, and reinforces negative emotions. It is more pessimistic and judges undesirable events as more likely. Moreover, it dislikes the user and therefore experiences resentment or gloating depending on the same event. When acting competitively, the robot primarily pursues its own objectives.

The robot's persona results in different comments and reactions, expressed primarily via the robot's spoken language. Figure 8.5 illustrates an example. When losing the game, the robot may say "Congratulations! You won!" when it uses the mentor persona. The same event can result in "You do not deserve that victory!" (both translated from German) when using the opponent persona. Furthermore, the robot's comments are emphasized by basic non-verbal behaviors (see Figure 8.3 and section 8.4).

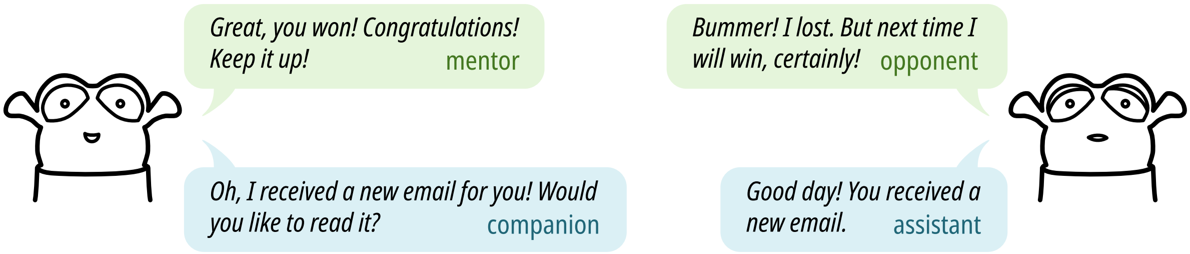


Figure 8.5.: Examples for the robot's expressed mentor vs. opponent persona and companion vs. assistant persona (translated from German).

8.2.2. Companion vs. Assistant

The study by Bartl et al., 2016 investigates two different personas for a social robot in the elderly domain: *companion* and *assistant*. The robotic companion implements the corresponding cues of prototypical behavior within the scope of several applications. These apps include the news, weather forecast, appointments, contacts, and messaging applications.

Inspired by Bartl et al., 2016, the companion and assistant persona is reflected in the robot's language. Figure 8.5 illustrates an example. Within the scope of the assistive applications, instructions, suggestions, reminders, notifications, confirmations and descriptions differ, i.a., with respect to fillers ("oh", "ah"), words of agreement ("okay", "alright", "good") or pronouns (informal "we" vs. formal "you") to reflect the corresponding persona. Texts from external services, such as news article headlines, abstracts, or message contents, are not modified.

8.3. Politeness

Hammer et al., 2016 investigate different politeness strategies (see section 4.3) as expressed in the language of a robotic elderly assistant. Their findings indicate that the robot's politeness strategy impacts perceived persuasion and that perceived robot politeness and persuasion are subjective. Consequently, there is no single "best politeness strategy" for everyone. Thus, the robotic companion at hand adapts its politeness to the individual user in the context of health-related recommendations in section 13.1.

Each recommendation exists in the eight politeness strategies from section 4.3. While their formulation changes, the semantic content remains the same. The robot's recommendations are formulated either as a (1) direct command, (2) indirect suggestion, (3) request, (4) question, (5) socratic hint, (6) shared goal, (7) robot's goal or (8) user's goal. Figure 8.6 illustrates an example recommendation for painting pictures. An image on the screen illustrates all recommendations.

8.4. Non-verbal behaviors

Independently of the language-specific behaviors for persona and politeness, the robot uses animations (including gaze behavior and blinks, see chapter 11) to take advantage

Figure 8.6.: Examples of the same recommendation presented with different politeness strategies and an image (translated from German).

of its physical embodiment. For example, during the games, the robot shifts its gaze (Mehlmann et al., 2014) towards the screen or user (depending on the current player's turn) and shows basic emotions depending on its game progress (see Figure 8.3). The animations include smiling when the robot is happy about its own or the user's move or an unhappy face in contrasting situations. Another example is the robot's communication features. When receiving text messages, the robot maps basic text emoticons automatically onto the corresponding facial expressions of the robot. Grimaces include smiling, winking, sadness, anger, confusion, laughter, and more.

After one minute of human inactivity, the robot stops any idle animations. The system aims to be an "always on" (Sidner et al., 2018) companion and chooses a sleeping pose with closed eyes to protect the motors. Additionally, the LEDs in the robot's cheeks indicate that it is in "stand by" mode and can be reactivated when desired. The display turns off after three minutes to save energy. Pressing any button on the control panel reactivates the robot's animations and display.

8.5. Hardware and Software

Figure 8.7 provides an overview of the technical setup. A Linux computer runs the main Java application, renders the GUI to the screen, and receives input from the control panel via USB. The main application interfaces with the robot via the custom *reeti-rest* software for multimodal animation generation (see chapter 11). A wired connection between the robot and computer ensures minimum latency. The computer connects to the user's

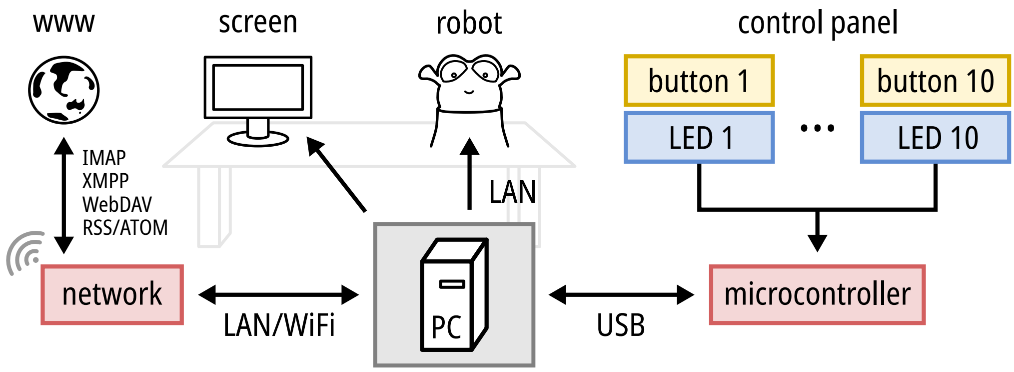


Figure 8.7.: Overview of the hardware and software setup.

domestic router via (wireless) LAN for internet access. Otherwise, those applications, which require external data from the internet, cannot be used.

In order to respect the user's privacy, the prototype does not use proprietary cloud services, such as Google Calendar or Google Mail. Each application can be enabled or disabled when configuring the robotic companion, depending on the user's preferences. Web-based services are accessed with open protocols for news aggregation (RSS/ATOM), communication (IMAP, XMPP), as well as calendar and contact administration (WebDAV). These protocols interface with self-hosted, non-proprietary services, such as *OpenFire*, *NextCloud*, *radicale*, etc. Recommendations, entertainment, and help functions work without internet access. Offline functions are essential since study participants may not have an internet connection or be unwilling to give access to it (see section 13.3).

8.6. Conclusion

With robotic companions becoming part of everyday human life and domestic environments, how they communicate with users is increasingly important. Based on the psychological background from section 4.1.1 and section 4.3, this chapter presented a novel robotic companion, which can express variants of persona and politeness in the context of assistive, communication, information retrieval, and entertainment functions. In contrast to chapter 7, which focused on extraversion, the robot's personas at hand are based, i.a., on the personality dimensions agreeableness and neuroticism. The fully autonomous robot comments on the course of play in German when playing board games, when presenting health-related recommendations, when presenting the weather forecast, and more. It uses manually designed behaviors because the evaluation in section 13.3 uses German native speakers, and there was no reasonable NLG technology for the German language at the time of writing. Due to potential issues reported in the literature for touch-based interaction in the elderly domain, a custom hardware control panel with physical buttons and lights is used to interact with the robot and system. The implementation uses the custom *reeti-rest* software (see chapter 11) for playing back animations and interfacing with the Reeti robot. It serves as the basis for the adaptation approach in section 13.1, which aims to learn about the users' individual preferences concerning the robot's expressed politeness and persona.

9. Joke-Telling

Entertainment is a common task for social robots. Chapter 7 already presented an approach for the generation of texts with personality in the context of storytelling; chapter 8 introduced a domestic companion robot, which can express variants of persona and politeness in the context of assistive, communication, information retrieval, and entertainment functions. Verbal and non-verbal humor has many functions and does not necessarily serve the purpose of entertainment (see section 4.4). However, in stand-up comedy, joke-telling is done explicitly to entertain an audience. As such, joke-telling is also one application scenario of interest for implementing robot entertainment, which has the benefit of being fun to look at and listen to. However, the *generation* of humorous content is a challenge for computers. While dynamic generation traditionally focusses on verbal humor, experiments with social robots typically use scripted content, including non-verbal behaviors like gestures and gaze behavior (see section 5.4).

This chapter starts with combining different modalities for producing multimodal humor. Subsequently, it proposes a technique for generating multimodal canned humor for a social robot dynamically during runtime. The latter augments dynamically generated, textual punning riddles produced by the STANDUP (Manurung et al., 2008) generator with appropriate prosody and non-verbal behaviors to benefit from and make use of the robot's embodiment. These cues are inspired by the humor markers observed in the literature (see section 4.4.4) and aim to communicate humor effectively by transferring typical human joke presentation behaviors to the robot's hardware and embodiment. Non-verbal communication channels (see chapter 3) are of central importance in this process. The ability to present and generate humor in HRI is an important opportunity for making the robot appear more socially intelligent. While this chapter focuses on jokes, chapter 10 focuses on conversational humor in terms of verbal irony.

The following techniques and implementations were presented, evaluated and reviewed in Weber et al. (2018b), Weber et al. (2018a), Ritschel and André (2018), Ritschel et al. (2020a), and Ritschel et al. (2020b). The contents of this chapter expand these publications. Section 14.3 relies on this work for adapting the robot's humor to the individual user's preferences.

9.1. Multimodal Humor Contents

Different stimuli can elicit human amusement. The first that comes to mind is canned jokes (see section 4.4.4). The text itself, but also the paralinguistic presentation, substantially contribute to the joke delivery. They include proper prosody, intonation, timing, facial expressions, and gestures.

Apart from verbal content, non-verbal humor can also get the audience in fits, such as grimaces – think of pantomime – or funny sounds. Besides spoken words, social robots

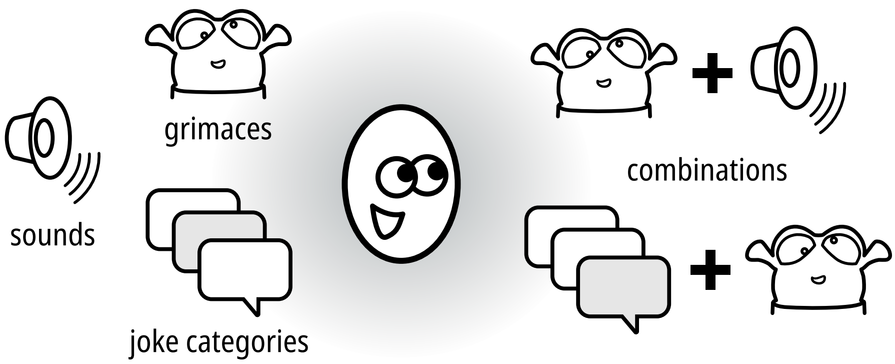


Figure 9.1.: The robot presents multimodal humor, including canned jokes, grimaces, and sounds.

can also playback or produce natural and artificial sounds, including recorded samples and designed or procedurally generated audio. Moreover, many social robots can express grimaces with hardware actuators or render facial expressions on a screen. The robot's embodiment is a curse and a savior at the same time: on the one hand, the hardware limits its expressiveness regarding the fixed appearance, design, and actuators; on the other hand, the technical nature opens up new possibilities and modalities, such as sound playback and LED lights.

Jokes, Grimaces, and Sounds

Experiments in the literature primarily rely on scripted humor, such as in the domain of robot stand-up comedy. This section uses manually designed content as the first attempt for the Reeti robot (see chapter 11). The contents include canned jokes in the German language, grimaces, and sounds. Figure 9.1 illustrates the multimodal content and points out that the modalities can also be combined. The Reeti robot's humor uses every modality offered by the robot's hardware.

Against the background of different individual humor preferences, several canned jokes have been hand-picked and sorted into different categories, such as gross-out, slapstick, or academic jokes. The punchline has been marked manually in the jokes so that the robot can synchronize its non-verbal behaviors with the verbal TTS output during the performance. For example, it displays one of the grimaces or plays a funny sound. These cues are motivated by the literature, which reports multimodal humor markers, which occur in particular at the punchline (see section 4.4.4).

The robot has a stylized face with large, rotatable eyes and eyelids (see Figure 11.1). Facial expressions present grimaces (see Figure 9.2). They were designed based on observations made in human and cartoon grimaces, such as intentional squinting or unusual mouth shapes and half-open or asymmetric eyelid positions in comedy or animation.

Since the use of sounds and music is a common approach for producing comic effects (Arias, 2001; Deaville and Malkinson, 2014; de Valck, 2005) several sounds were prepared in addition to canned jokes and grimaces. They include comedy sound effects associated with humor from the media, including iconic sound effects from famous feature films.

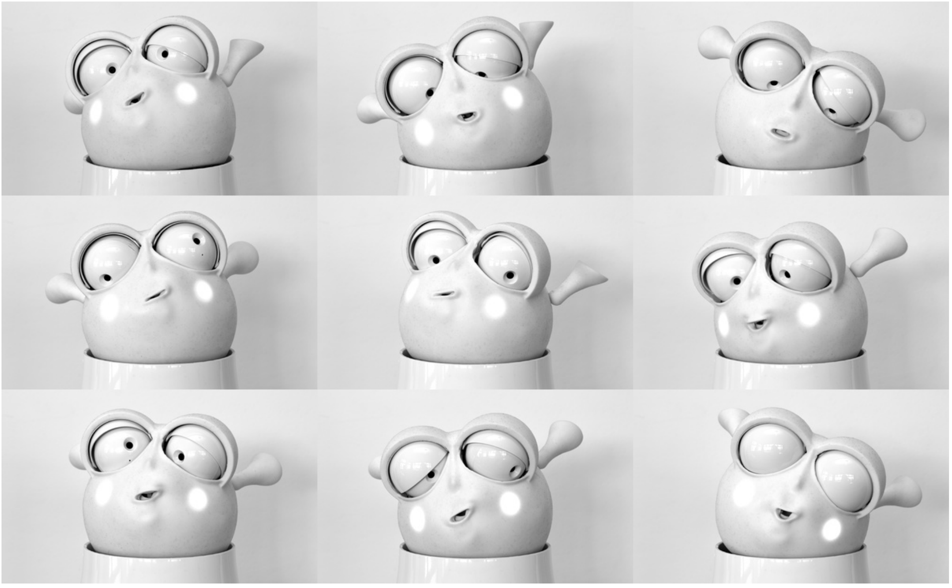


Figure 9.2.: Some of the robot's grimaces.

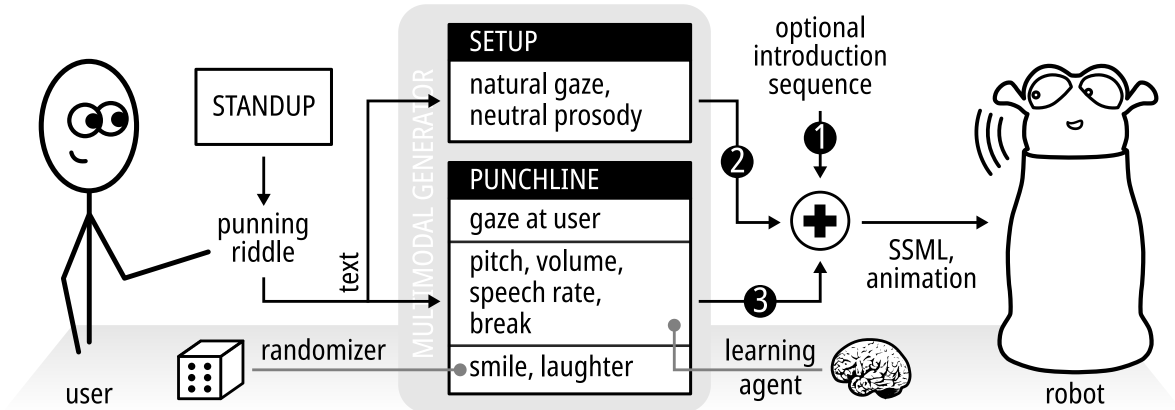


Figure 9.3.: Overview of the multimodal joke generation approach.

These auditive cues are not part of the literature overview from section 4.4, which includes human canned and conversational humor, but augment the machine with additional, artificial instances of humor.

9.2. Dynamic Multimodal Joke Generation

The generation of humorous content requires the synchronization of multiple modalities. As described in section 4.4.3, humor factors and markers are core elements in joke delivery. Besides the actual joke text, additional verbal and non-verbal cues are important for signaling the presence of humor, including prosody, timing, and facial expression. This section focuses on a generative approach for multimodal punning riddles, where the robot dynamically generates verbal and non-verbal behaviors during runtime. It is illustrated in Figure 9.3.

The text of a punning riddle consists of a setup and punchline (see section 4.4.4). The setup asks a question to the listener (e.g., “What do you get when you cross a choice with a meal?”). Subsequently, the punchline presents the unexpected solution (e.g., “A pick-nic.”). The STANDUP joke generator (Manurung et al., 2008) generates the text of both parts, resulting in a string, which – in theory – could be presented to the user directly by the robot’s TTS system. However, this does not automatically result in a convincing and amusing presentation since these systems do not yet automatically consider typical paralinguistic characteristics for joke-telling.

Thus, the robot augments the setup and punchline with typical human multimodal cues. Gaze, prosody, and facial expression are added and later optimized by the adaptation process in section 14.3.3. The SSML (w3.org, 2021) is a key technology in this regard. Embedding instructions for the TTS system in the original text of an utterance gives more control over the audio generation. For example, this includes setting the emotion, controlling the prosody, and adding breaks or vocal gestures (i.e., sounds, such as clearing the throat, “humph”, laughter or giggling, and more). Moreover, the robot’s face is animated to include gaze behavior and smile. The implementation uses the Reeti robot and the custom *reeti-rest* software for multimodal animation generation (see chapter 11). Similar to section 7.2, the paralinguistic cues are randomized to a certain degree to create variety in the produced multimodal output.

9.2.1. Text Generation

Since it is common to introduce a joke in conversations (see section 4.4.4), the robot can optionally add a negotiating sequence, such as “Do you know this joke?” or “The following punning riddle is a real pearl of comedy!” When embedded in HRI, this aims to (1) set the stage for the robot’s performance by announcing its intention explicitly and (2) enhance the chance of receiving humor support (see section 4.4.3) from the user, which is essential for the adaptation process later on.

Then, the STANDUP joke generator (Manurung et al., 2008) is used to generate the setup and punchline text of the punning riddle. It creates different subtypes; those used in the application at hand are listed with examples in Table 9.1. Different topics or keywords serve as input to the generation of puns. STANDUP uses different schemas and templates in combination with information about pronunciation and semantic relationships of words. A *schema* describes the linguistic requirements for the punchline, e.g., the stylistic form of the underlying script opposition from which the humor arises. *Descriptive rules* define the guidelines for generating the formulations for the question. The *text template* selects and aggregates the previously generated contents. Additionally, a database with information about pronunciation and semantic relationships of words is used for instantiating these rules. The generator produces a pair of setup and punchline as two separate strings, e.g., “What do you call a washing machine with a September?” – “An autumn-atic washer.” Table 4.1 gives an overview of linguistic markers of the syntax and content of setup and punchline. The software checks for duplicates to avoid creating the same joke more than once.

Subsequently, the setup and punchline are augmented with appropriate prosody, gaze, and facial expressions according to the findings from the literature as follows.

Table 9.1.: Utilized STANDUP joke types.

Joke type	Riddle structure
Cross	What do you get when you cross X with Y?
Call	What do you call a cross between X and Y? What do you call X that has Y? What do you call X with Y? What do you call X?
Difference	What is the difference between X and Y? Why is X different from Y that is Z? Why is X different from Y? How is X different from Y?
Similarity	What do X and Y have in common? Why is X like Y? How is X like Y?
Type	What kind of X has Y? What kind of X is Y?

9.2.2. Setup

9.2.2.1. Prosody

In human joke-telling, a specific pitch range and variability in the speaker's voice may accentuate the spoken language. As outlined in Table 4.1, a typical occurrence is the combination of limited pitch range and minor pitch change within syllables or the whole utterance. The latest SSML specification (w3.org, 2021) defines two options of interest for reproducing this effect:

- The *range* attribute of the *prosody* element determines the pitch range (variability) of the enclosed text. The value can be set either in the form of a relative change in percent, presets *x-low*, *low*, *medium*, *high*, *x-high* and *default*, or as a frequency value in Hertz.

- The *emphasis* element emphasizes the enclosed text. Its attribute *level* can be set to *strong*, *moderate*, *none* and *reduced*. While *reduced* means the opposite of emphasizing text, the specification mentions that the element may be rendered by a "possible combination of pitch change, timing changes, loudness, and other acoustic differences".

As with all SSML parameters, their realization depends entirely on the TTS software interpreting them. The *range* attribute is set to the *default* value during the setup to contrast the subsequent punchline. Unfortunately, the *emphasis* tag does not produce an audible effect with the Cerevoice TTS system and the *William* voice. For example, using level *reduced* does not result in reduced pitch variability. Thus, the setup of the riddle converts into SSML without additional tags. Encapsulating the text in a *prosody* element

Listing 9.1: Generated punning riddle with SSML.

```
<speak>
  <s>What do you get when you cross a choice with a meal?</s>
  <break time="1500ms"/>
  <s><prosody pitch="high" rate="fast" volume="loud">A pick—nic.</prosody></s>
  <spurt audio="g0001_019"></spurt>
</speak>
```

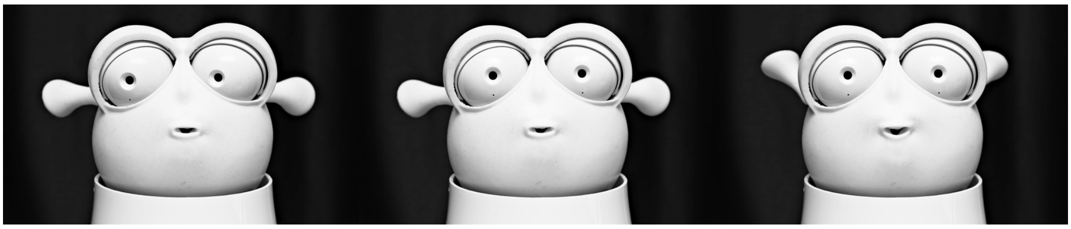


Figure 9.4.: The robot's gaze and facial expressions: a saccade (left), its neutral facial expression when centering on the spectator (middle), and smile (right).

with the *range* attribute set to *default* would be redundant and therefore is omitted. See the first sentence in Listing 9.1 for an example.

9.2.2.2. Gaze

The robot's gaze behavior during the setup aims to mimic natural human gaze behavior. Robots and virtual agents often implement saccades since they represent one of the most noticeable eye movements (Ruhland et al., 2014). In order to contrast the following punchline, the robot performs this type of eye movement while telling the setup. Technically, it rotates the eyeballs (and neck, see below) in parallel to random points near the spectator's position (see Figure 9.4). See section 11.5 for details on the implementation.

9.2.3. Punchline

9.2.3.1. Prosody

Typically, there is a break between telling the setup and punchline of a joke (see Table 4.1). During this time, the listener has time to think about a possible answer to the riddle's solution. Adding a break in SSML can be realized with the *break* element. Its parameter *time* specifies the duration in milliseconds. Since there are no clear findings about the duration of such breaks in the literature, the robot uses a random value in the hand-tuned range between 1.5 and 2.0 seconds.

In addition to a preceding break, the speaker presents the punchline often with a different pitch, volume, and speech rate than the setup (see Table 4.1). SSML provides tags and attributes to imitate these prosodic features with the *prosody* element. For example, the *pitch* attribute accepts the values *low*, *medium* and *high*. Similarly, the *volume* attribute works well with the values *soft*, *medium* and *loud*. The speech rate can be modified with *slow*, *medium* and *fast*. More extreme variants, such as *x-low* or *x-high* result in more synthetic and less natural sounding output. They impair the robot's comprehensibility, particularly when compared to the neutral prosody during the setup. See the second sentence in Listing 9.1 for an example configuration of the *prosody* element during the punchline.

9.2.3.2. Laughter

Laughter and giggling after the joke occurs not only for the audience but occasionally also for the speaker (see Table 4.1). The SSML standard does not support this kind of sound per se. However, Cerevoice TTS voices provide so-called *vocal gestures*. These audio samples include different sounds, such as breathing, coughing, and more. For the *William* voice, this set also includes different types of laughter, ranging from short giggling to long laughter sounds. The non-standard SSML *spurt* element embeds these sounds in the SSML markup. It requires providing the ID of the sound from the manual as the value of the *audio* attribute.

When used excessively after every punchline, giggling and laughter sounds appear unnatural to the audience, especially if the same sample occurs repeatedly. Thus, the usage and sample IDs are randomized. Based on the insights by Attardo, Pickering, and Baker (2011) the probability for adding the *spurt* element to the generated output is set to 30 %. At the time of this writing, the manual contains six IDs related to laughter for the *William* voice: IDs 19 and 20 produce giggling, and 21 to 24 contain laughter samples. A random sample from this set of sounds is drawn each time the element is embedded. See Listing 9.1 for an SSML sequence with the *spurt* and other markers included.

9.2.3.3. Gaze

Speakers often gaze at the face areas involved in the spectators' smile when presenting the punchline (see Table 4.1). These areas include the eyes and the mouth, which express smile, laughter, or giggling. While the robot mimics natural gaze behavior during the setup, its head and eyes focus on the spectator during the punchline. To this end, the robot's head and eyes center on the spectator in front (see Figure 9.4). The scenario at hand involves a single-person audience. Thus, the robot does not change its gaze between different spectators, as observed and implemented in Katevas, Healey, and Harris (2015).

9.2.3.4. Smile

Speakers may use a smile when presenting the joke's punchline (see Table 4.1). Based on the insights by Attardo, Pickering, and Baker (2011), the robot uses this marker with a probability of 80 % by raising its lip corners. The robot raises its large ears to emphasize the smile even more (see Figure 9.4). After the joke finishes, the robot's face returns to a neutral facial expression.

9.3. Conclusion

The ability to present and dynamically generate humor in HRI is an important opportunity to make a social robot appear more socially intelligent. Based on the psychological background from section 4.4, this chapter presented approaches for multimodal joke delivery with a social robot. They allow the robot to present jokes, grimaces, sounds, and combinations of them. Moreover, a novel generative approach for multimodal punning riddles allows the robot to create canned jokes dynamically during an interaction. While the first approach explores comic sound effects, which a human cannot use, the second approach adheres to the findings concerning multimodal markers of humor, including verbal and non-verbal communication channels. The implementation uses the STANDUP joke generator (Manurung et al., 2008), SSML, Cerevoice TTS, as well as the custom *reeti-rest* software (see chapter 11) for dynamically generating and playing back animations and interfacing with the Reeti robot. This work provides the basis for section 14.3, where the social robot's joke presentation adapts to the individual spectator based on their social signals.

10. Small Talk with Irony

Chapter 9 presented an approach for the generation of canned humor by enriching generated texts with multimodal cues for a social robot. In human interaction, jokes are typically known in advance and presented spontaneously within a conversation (e.g., when remembering a joke because of the current conversation topic) or as part of a longer show, such as in stand-up comedy. However, humans also create humorous content in a specific situation on the fly, inspired by the interaction context or topic, which is the case in *conversational* humor (see section 4.4.2).

Ideational reversal irony (see section 4.4.5) is particularly interesting for the presented rule-based irony generation approach below. While other types of irony require an "understanding" of the actual semantic content or some form of creativity for generation, this chapter implements ideational reversal irony based on natural language processing (NLP) and NLG techniques without a deeper understanding of the text.

The purpose of irony "lies in its rhetorical and social effects" (Attardo, 2000b). Giving robots the ability to create ironic utterances is one opportunity to make them appear more socially intelligent. Thus, this chapter presents an approach to generating multimodal irony. A multistage generation process transforms the robot's original utterance into an ironic version with the opposite meaning based on NLG and NLP techniques. Afterward, it augments the modified text with specific multimodal markers, including prosody and facial expression, which are essential for making the human recognize the irony. Again, non-verbal communication channels (see chapter 3) are central to this process.

The approach and implementation were presented, evaluated, and reviewed in Ritschel et al. (2019b). The contents of this chapter expand this publication.

10.1. Overview

The following rule-based transformation process for creating an ironic version of a non-ironic utterance involves three steps (see Figure 10.1):

1. Given the original input, the first task of the approach is to check with NLP techniques whether the utterance fulfills the requirements concerning vocabulary

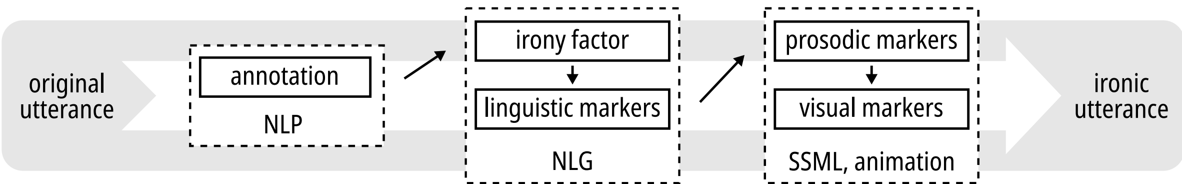


Figure 10.1.: Overview of the irony transformation approach.

and structure. The transformation algorithm works only if the sentence meets the requirements. Otherwise, the process stops.

2. If the input fulfills the requirements, the next step is the generation of an irony factor and the insertion of linguistic markers. NLG techniques solve this task.

3. In addition to the linguistic manipulation, the final step is to add additional markers of irony. These include prosodic markers and accompanying non-verbal behaviors, such as gestures and facial expressions.

This process transforms a non-ironic linguistic utterance into a multimodal ironic version. Steps one and two are applicable for any robot because they only manipulate the linguistic content. The multimodal cues depend on and are restricted by the robot's soft- and hardware. For example, not every TTS system supports all proposed manipulation tags. In particular, facial expressions and gestures require corresponding hardware actuators or a virtual representation on a screen, which differ for each robotic platform. The following approach is implemented based on the Reeti robot (see chapter 11).

10.2. Natural Language Processing

The implemented approach receives an arbitrary text utterance as input. As a first step, this input is analyzed based on NLP techniques to ensure that it fulfills the requirement of the algorithm to be transformed into an ironic utterance. The input must have a strong polarity, such as a positive or negative connotation, which can be inverted based on dictionary data. CoreNLP (Manning et al., 2014) identifies adjectives, nouns, and verbs with polarity based on sentiment analysis (Socher et al., 2013). For example, in the sentence “I hate my worst enemy.” the words “hate”, “worst” and “enemy” indicate a negative polarity. The algorithm marks polarized words as candidates for creating the irony factor in the following step, i.e., the incongruity in the resulting sentence's meaning. If the original input does *not* have positive or negative polarity, the following steps cannot be applied, and the approach stops.

10.3. Natural Language Generation

10.3.1. Irony Factor

The generation of the irony factor uses the concept of ideational reversal irony (Dynel, 2014) (see section 4.4.5). The irony results from negating one of the marked, polarized irony factor candidates from step one. First, the algorithm replaces a polarized word with its antonym by looking up antonyms in the WordNet database (Miller, 1995) for each candidate. If a suitable antonym is found, the original word gets replaced according to the following prioritization:

1. verbs,

2. adjectives, and

3. nouns.

However, only one word is replaced not to nullify the negation. A nullification could happen when replacing multiple items simultaneously, such as in the case of "I love my best friend.", which would prevent the emergence of irony. The flexion of words is realized with the SimpleNLG (Gatt and Reiter, 2009) programming library while preserving the conjugation of verbs, the comparative and superlative of adjectives, and the number of nouns. As an example, this transforms the former example into the sentence "I *love* my worst enemy."

If the algorithm does not find a suitable antonym, it inverts the main verb with "not". Alternatively, it removes "not" if the original sentence already contains a negated verb. In this case, there is no need to search for antonyms.

10.3.2. Linguistic Markers

Replacing a single word with an antonym does not automatically make the ironic intention recognizable to the listener without further ado. In addition to the generated irony factor, markers of irony (see section 4.4.5) make it easier for the human to recognize the robot's irony. Thus, this step inserts linguistic markers in the resulting utterance, i.e., exaggerations, understatements, and positive and negative interjections. Exaggerations and understatements can be realized by valence shifting (Guerini, Strapparava, and Stock, 2008). Single words are modified by adding, removing, or replacing adjectives and adverbs to strengthen or weaken the sentence's meaning, e.g., resulting in "I *absolutely* love my worst enemy." This increases or decreases the intensity of the irony factor. For example, the former negatively polarized example can be prefixed with a positive interjection to generate "*Super!* I love my worst enemy." In contrast to the irony factor, multiple markers can be applied simultaneously to emphasize the use of irony, e.g., by adding an exaggeration, which results in "*Super!* I *really* love my worst enemy."

Keeping in mind that the output medium for the generated ironic text at hand is the robot's TTS system, some linguistic, text-specific markers, such as onomatopoeic expressions for laughter, acronyms, emoticons, ellipsis, quotation, and heavy punctuation marks (see section 4.4.5) do not make sense in the scenario at hand. Instead of laughter, onomatopoeic expressions for laughter, or emoticons, the robot can use facial expressions with its hardware directly (see section 10.4.2). Similarly, its prosody is adjusted accordingly (see section 10.4.1) instead of ellipsis, quotation, and heavy punctuation marks. While prosody, facial expressions, and gestures depend on the robot's available actuators and TTS capabilities, they supersede text-only markers, which only make sense in written text and not in a spoken utterance.

10.4. Multimodal Markers

In written text, typographic or morphosyntactic markers (see section 4.4.5) help the reader identify ironic content. However, the benefits of social robots are in multimodal communication while relying on their embodiment, hardware actuators, and spoken output. Thus, it makes sense to use paralinguistic and visual clues to communicate

Listing 10.1: SSML for the compressed pitch pattern

```
<emphasis level="strong">Great!</emphasis>
<prosody volume="x-soft">
  <emphasis level="none">I absolutely love my worst enemy.</emphasis>
</prosody>
```

Listing 10.2: SSML for the pronounced pitch accents

```
<prosody rate="x-slow">
  <emphasis level="strong">Great!</emphasis> I <emphasis level="strong">
    absolutely<break strength="medium"/> love <break strength="medium"/>
  </emphasis> my <emphasis level="strong">
    worst <break strength="medium"/> enemy<break strength="medium"/>
  </emphasis>.
</prosody>
```

irony beyond linguistic tweaks. Depending on the robot's hardware and TTS software, intonation and visual clues, including prosody, gestures, and facial expressions, can be used to implement those markers presented in Table 4.2. Given the Reeti robot as an implementation target, prosody and facial expressions are the relevant modalities to support and augment the formerly prepared linguistic ironic content.

Not all markers are applied for every ironic utterance. Instead, the use of the individual markers is randomized to a certain degree. If applicable, the corresponding probabilities are initialized based on the findings from Carvalho et al. (2009) and Williams, Burns, and Harmon (2009).

10.4.1. Prosody

Two acoustic parameter modulations from Table 4.2 are used to generate typical prosody for ironic utterances: the *compressed pitch pattern* and the *pronounced pitch accents*. As described in section 4.4.5, they result in atypical speaking behaviors, which contrast with normal speech. For their implementation, one needs control over pitch, rhythm, speech rate, and accents during the audio generation process of the TTS system. Thus, the prosodic markers are implemented based on the SSML standard (w3.org, 2021). It defines XML elements and tags, which control these parameters for syllables, words, or whole utterances by adding the corresponding markup to the text input. However, their support, specific implementation, and the resulting auditive effect in the generated audio are specific to the used TTS system and voice. The implementation at hand was created and tested for the Cerevoice TTS software (cereproc.com, 2021) with the "William" voice.

The implementation of the "flat" intonation of the compressed pitch pattern is illustrated in Listing 10.1. It uses the *volume* tag of the *prosody* element to prevent emphasis and emotion in the generated audio. Specifically, its value *x-soft* aims to emulate reduced pitch movements while pronouncing the utterance, as observed in the literature.

Figure 10.2.: Implemented facial expressions.

In contrast, Listing 10.2 illustrates the markup for the pronounced pitch accents. The markup exaggerates the intonation by accentuating words and adding elongations and pauses to reproduce findings from the literature. Firstly, it reduces the overall speech rate by setting the *rate* tag of the *prosody* element to *x-slow*. Secondly, it utilizes *emphasis* tags for the main verb, all adjectives, adverbs, and nouns. In addition, the inserted *break* tags accentuate these even more to reproduce stilted pauses. For the example above, this results in putting emphasis on "I *absolutely love* my *worst enemy*."

Interjections are emphasized for both patterns independently of the prosodic marker to highlight their emotional intensity.

10.4.2. Facial Expressions

Social robots have a physical embodiment, which makes it possible to implement facial expressions in addition to the other markers. Depending on the robot's face and actuators, it needs to realize these expressions by animating the whole head, mouth, eyes, eyelids, and eyebrows. Since the Reeti robot at hand does not have eyebrows, those markers, which require eyebrows, cannot be applied to this specific hardware. However, Figure 10.2 illustrates the implemented markers for the Reeti robot:

- *Smiling* is realized by raising the robot's lip corners and lowering its lower lip to give the mouth a typical, round shape. At the same time, the head is rotated slightly around the roll axis to increase the effect.
- *Rolling eyes* are implemented using keyframe animation section 11.3.2. An animation track for each eyeball contains keyframes for the tilt and pan axis to realize a smooth transition of the pupils from one side to the other while tilting upwards in the middle of the animation (i.e., looking at the ceiling).
- *Winking* is realized with keyframe animation, too. It uses one animation track for closing and opening one of the eyelids.
- *Wide open eyes* are implemented by raising the eyelids to their maximum open position. The robot's neutral position sets the eyelid rotations to a slightly closed position, which leaves some space to open the lids even further.
- *Gaze aversion* is another keyframe animation, which pans both pupils to one side.

These poses or animations are presented for the duration of the ironic utterance, respectively. Afterward, the robot returns to its neutral face with its gaze centered on its dialog partner.

10.4.3. Additional Markers and Restrictions

The realization of multimodal markers of irony heavily depends on the robot's hard- and software. The Reeti robot and the Cerevoice TTS system cannot implement some of the markers from the literature. For example, not all markers can be realized with SSML, such as the *nasalization* (Attardo et al., 2003). There are no suitable tags for defining the desired effect. In addition, the *strong within-statement contrast* marker would require the division of the utterance into a high-pitch and a low-pitch part, which is non-trivial.

Similarly, the robot's available motors and degrees of freedom limit the applicable gestures and facial expressions. Since the Reeti robot only has a face, it cannot perform gestures. The *blank face* marker can only be applied if the robot's face is permanently animated. Then, a non-animated face will contrast the robot's default behaviors. However, as long as the robot's face is primarily static (as is the case for the Reeti robot), the blank face marker does not show a difference from a non-animated face and thus is superfluous.

Ambiguity is a challenge in the context of NLP for checking the polarity of words and the antonym lookup when generating the irony factor. For example, instead of marking it as a positively polarized word, "great" might be classified as neutral since it is associated with more than one synset, including "big". Similarly, "great job" could be transformed into "small job" instead of "bad job". Thus, knowledge about the word's context could improve the antonym lookup with semantic relatedness (Pedersen and Kolhatkar, 2009).

10.5. Pseudocode

Algorithm 1: Pseudocode of the transformation approach.

Input: original utterance u_{orig}
Output: TTS text and facial expression

$p \longleftarrow$ is_polarized(u_{orig}) *// boolean*
if p **then**
    $f \longleftarrow$ determine_irony_factor(u_{orig})
    $u_{\text{ironic}} \longleftarrow$ replace_with_antonym(u_{orig}, f)
    $u_{\text{ironic}} \longleftarrow$ insert_linguistic_markers(u_{ironic}, f)
    $u_{\text{ssml}} \longleftarrow$ add_prosodic_marker(u_{ironic})
    $e \longleftarrow$ random_ironic_facial_expression()
    return u_{ssml}, e
return u_{orig}, *null*

Algorithm 1 presents a simplified, high-level pseudocode of the transformation approach. First, the input utterance is analyzed to identify words with strong polarity (see section 10.2). The algorithm returns the original utterance as-is if it is unsuitable for the ironic transformation. Otherwise, the irony factor and the steps for converting the utterance and adding corresponding markers are applied as outlined in the previous sections. Afterward, the robot presents the ironic, multimodal version, including prosody and animation based on the custom *reeti-rest* software for multimodal animation generation (see chapter 11).

Figure 10.3.: Study setup with two robots of the same type and appearance.

10.6. Evaluation

An empirical study with users was conducted in the lab to evaluate the performance and the effects of the proposed irony generation approach. In particular, the study aimed to explore

- whether participants would be able to identify the generated behavior as ironic and
- how the generated behavior would influence the participants' user experience.

The hypothesis was that an ironic robot would be considered more "fun" to communicate with and "entertaining" to interact with. As a result, the expectation was that an ironic robot would receive higher user ratings associated with hedonic qualities than a non-ironic robot. Since both (1) the context and topic of the conversation play an important role in the appropriateness of irony usage and (2) related research has shown that a shared sense of humor has a powerful effect on interpersonal attraction when people meet each other for the first time (Cann, Calhoun, and Banks, 1997), it was decided to focus on a small talk scenario for the evaluation. It provided the opportunity to break down barriers between the interaction partners and present the robot's personality. Thus, this seemed to be an innocuous and predestined context for the use of irony.

10.6.1. Participants, Apparatus, and Procedure

Twelve participants (six male, six female), aged 19 to 32 (avg. 25), were recruited from a university campus. The study design had a within-subject design. Each participant interacted with all versions of the robot (i.e., a baseline robot and an ironic robot) in a counter-balanced order. Instead of reusing the same physical robot, two Reeti robots (see Figure 10.3 and chapter 11) were used in order to minimize potential carry-over effects. The purpose was to clarify that participants were interacting with two different robots (or personalities). Otherwise, participants could have had difficulties if they were asked to interact and rate their interactions with the same physical instance of the robot again, which would only change its programming. The study procedure had four subsequent parts: introduction, small talk sessions one and two, and wrap-up.

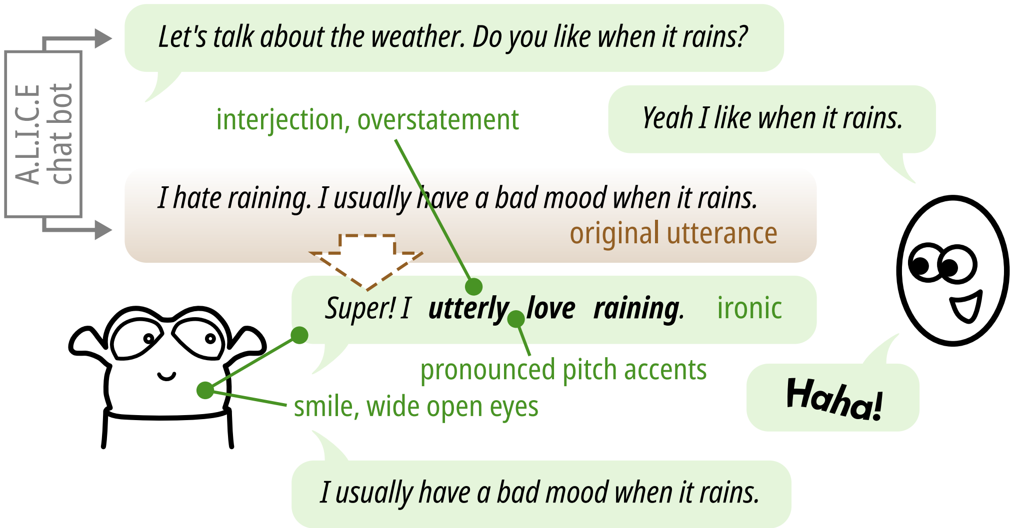


Figure 10.4.: An example of the robot's produced irony during the evaluation.

10.6.1.1. Part 1: Introduction

First, participants were welcomed and provided with an introduction, which included demographic data collection, a short description of the study, and additional information about the following study procedure. Participants were told that they would be asked to conduct dialogs for approximately ten minutes with two robots of the same type, which were differently programmed. Afterward, they would be asked user experience questions about how they perceived the robots. They were not told that one of the robots uses irony or that the study was about irony or humor.

10.6.1.2. Part 2 & 3: Small Talk Sessions

An automatic speech recognizer was not available for the study. Thus, participants were informed that they could speak with the robot, but the experimenter would enter their statements via keyboard, and, thus, short delays in the robot's answer could occur. Furthermore, participants were asked to reply to the robot's questions and ask questions to the robot themselves whenever they wanted to or felt it was appropriate.

At the beginning of both sessions, the participant sat in front of one of the two Reeti robots. Both robots used the Cerevoice TTS software (cereproc.com, 2021) with the "William" voice and identical animations. The open source A.L.I.C.E chat bot (sourceforge.net, 2021) that used freely available corpora for small talk generated the robot's answers. On average, the chatbot generated 19.25 answers per session. Each session started with the robot greeting the participant and asking for their name. Participants could see what the experimenter entered via the keyboard on the screen next to the robot at any time. Thus, it was transparent that the experimenter was not acting as a "wizard" and entering the robot's answers but typing the participants' replies and questions.

In the sessions with the *ironic* robot condition, the robot's answers were – whenever applicable – computationally transformed into an ironic utterance with multimodal markers of irony. Figure 10.4 illustrates a real-world example generated during the

Figure 10.5.: Basic emotions (neutral, happy, sad) used independently on the irony condition.

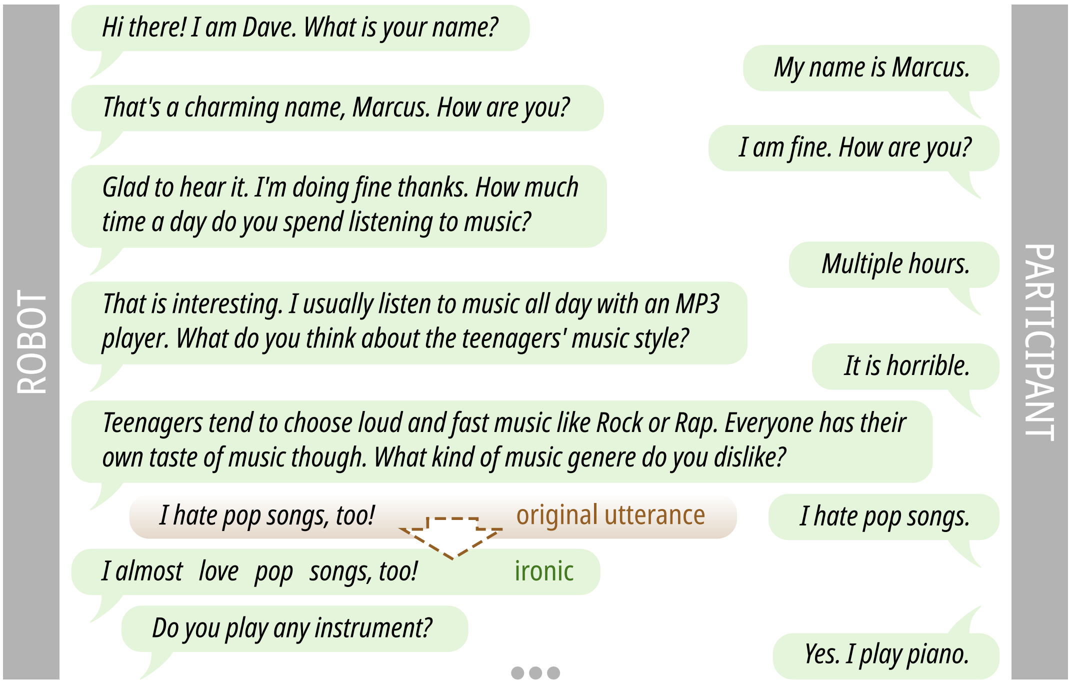


Figure 10.6.: Beginning of a dialog from the evaluation logs.

runtime of the experiment. On average, the robot produced 4.33 ironic answers per session.

In the *neutral* (baseline) condition, the robot never manipulated its answers to turn them ironic. The robot used basic emotion postures (see Figure 10.5) for the non-ironic spoken texts according to the answer's polarity in both conditions. These postures aimed to make the robot appear more lifelike in addition to the automatic gaze behaviors presented in section 11.5.

At the end of each dialog, session participants completed a questionnaire measuring their experience. Half of the participants started with the ironic condition and the other half with the neutral condition. Similarly, half of the subjects started with the left robot and half with the right robot, independent of the condition. A sample dialog with non-ironic and ironic contents is illustrated in Figure 10.6.

10.6.1.3. Part 4: Wrap-Up

At the end of all sessions, participants were asked which of the two robot versions they preferred. In addition, they were asked whether they liked or disliked the voice and visual appearance of the robot. These questions collected explicit robot preferences and measured if the appearance and voice of the robot were generally acceptable.

10.6.1.4. Questionnaires

The standardized attrakDiff questionnaire by Hassenzahl, Burmester, and Koller (2003) measured UX because it is widely used in research and industry. It provides an overview of a product's or technique's perceived qualities, especially hedonic qualities beyond traditional usability.

The attrakDiff questionnaire consists of 28 items. Seven items measure pragmatic quality (PQ), a measure of perceived traditional usability. The rest of the items measure hedonic quality (HQ), which results from a combination of HQS, HQI, and ATT. These (sub)constructs of hedonic quality are measured by seven items each. HQS measures the perceived ability of a product to meet a person's desire for self-improvement, HQI measures the perceived ability of a product to communicate a valuable identity to others, and ATT measures overall attractiveness.

In addition, a five-point Likert scale asked for agreement scores based on statements, such as "the robot's output was ironic". These were specific to the user study setup and aimed to measure participants' subjective impression of the robot's output (replies and questions) concerning content, naturalness, humor, irony, fitting facial expressions and voice, and perceived (social) intelligence (see Figure 10.9).

10.6.2. Results

This section presents insights into the study results. The expectation was that an ironic robot based on the irony generation approach would receive higher user ratings on the hedonic dimension of UX. First, general trends are illustrated based on graphical presentations of the collected UX data. Then, results are interpreted based on fitting a statistical model to the data, i.e., results of statistical tests are provided. Afterward, the results of the additionally collected explicit robot preferences and agreement scores are presented.

10.6.2.1. General Trends

The left side of Figure 10.7 presents the mean values for all four measured UX constructs (i.e., PQ, HQS, HQI, and ATT). It shows that the ironic robot received higher mean ratings in all measured constructs. The differences in how the ironic robot was perceived compared to the baseline robot are in constructs explaining hedonic qualities, which matches the hypothesis. The biggest differences in the means include ATT (overall perceived attractiveness) and HQS (i.e., the hedonic quality associated with the perceived ability of a product/technique to meet a person's desire for self-improvement). The smallest difference exists in PQ (i.e., perceived traditional usability).

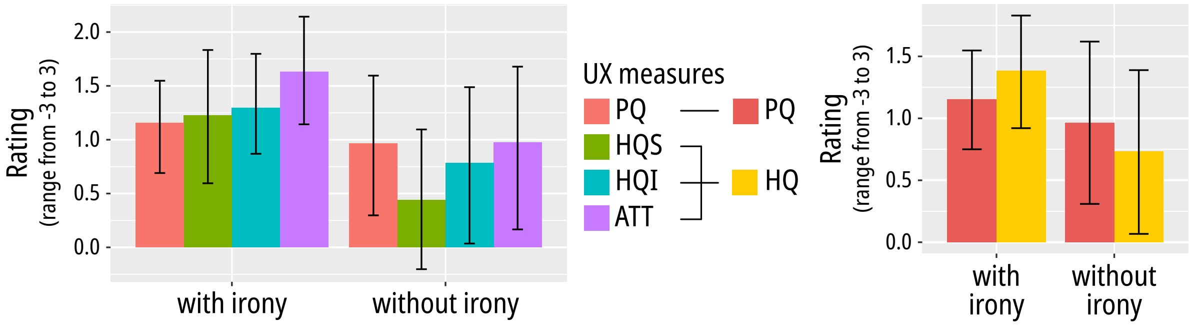


Figure 10.7.: Mean ratings for the measured UX constructs PQ, HQS, HQI, and ATT explaining pragmatic (i.e., perceived traditional usability) and hedonic qualities for both the ironic robot and the baseline robot. Error bars denote 95 % confidence intervals (CIs).

The right side of Figure 10.7 shows the difference in how the ironic robot is perceived differently from the baseline (non-ironic) robot, considering the pragmatic quality (PQ) and hedonic quality (HQ) (i.e., combination/aggregation of HQS, HQI, and ATT). While one can observe a difference in HQ between both robots, the difference between the baseline robot and the ironic robot considering the pragmatic quality (PQ), seems very small.

For each construct, Figure 10.8 depicts mean values for all seven items, detailing how participants experienced the interaction with both robots. Regarding HQS, the participants experienced the ironic robot as bolder but more innovative. The ironic robot received higher ratings for most items explaining the modalities' hedonic quality. It was perceived as more stylish, appealing, or likable, thus, potentially communicating a valuable identity to others (HQI) and generally being perceived as attractive (ATT). The overall difference seems systematic, with the ironic robot being perceived as consistently more desirable, which is reflected in higher ratings on the hedonic dimension of UX. There seem to be no clear differences in the pragmatic dimension of UX.

Figure 10.9 depicts the mean ratings explaining (dis)agreement of study participants with several statements. These statements checked whether participants recognized the robot's irony and evaluated whether its facial expressions and prosody fit its presented utterances. The main differences in mean ratings exist for the statements about robots being ironic, humorous, and natural. It indicates that participants were able to identify the use of irony and experienced humor and naturalness as associated causes of irony use.

10.6.2.2. Statistical Analysis

First, paired-sample t-tests compare the participants' ratings for PQ, HQS, HQI, and ATT for both modalities. For HQ and its constituting constructs HQS, HQI, and ATT, a one-sided test was used since the hypothesis was that the ironic robot would receive higher ratings on hedonic quality compared to the baseline robot. There was no hypothesis on how irony would affect PQ. Thus, a two-sided test tests the significance of users' self-reports considering PQ in both robot conditions.

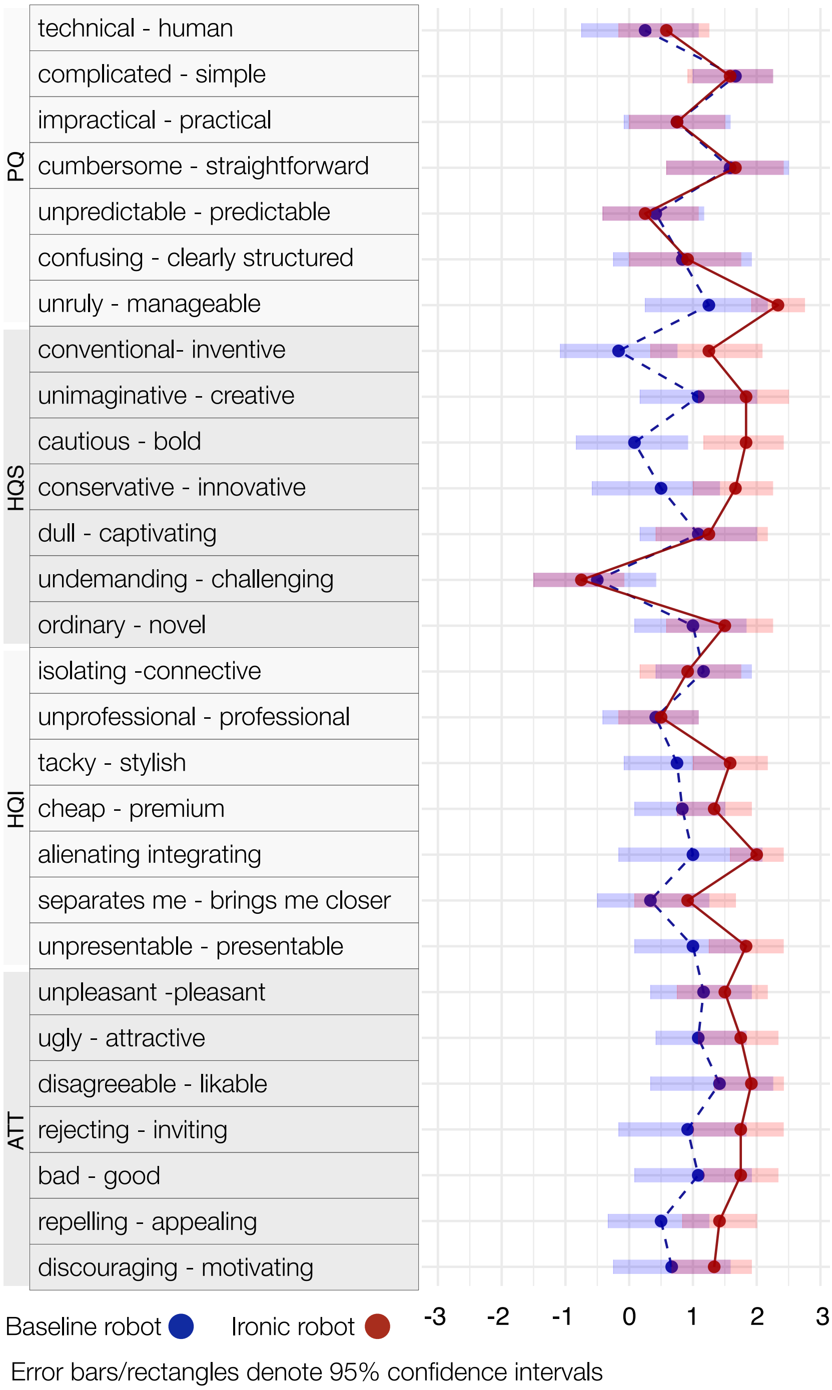


Figure 10.8.: Details show the mean ratings for each item of the attrakDiff questionnaire.

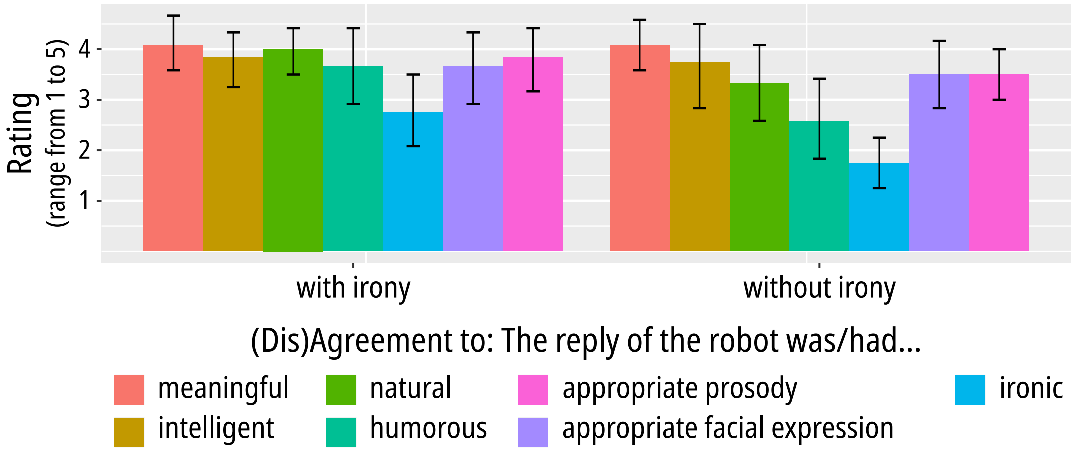


Figure 10.9.: Communication style of the robot: mean ratings for the agreement scores to all additional statements (which were translated and slightly paraphrased to fit the plot) on a five-point Likert scale (1 = strongly disagree, 5 = strongly agree). Error bars denote 95 % confidence intervals (CIs).

A main effect of irony exists on the overall hedonic quality HQ ($t = 1.86$, $p = 0.044$, $r = 0.49$). As hypothesized, participants found the ironic robot more desirable: the difference in self-reports on the hedonic dimension of UX was significant. When looking at the specific (sub)constructs of HQ separately, the difference in HQS ($t = 2.29$, $p = 0.021$, $r = 0.56$) is significant, too. But differences are non-significant for HQI ($t = 1.27$, $p = 0.115$, $r = 0.35$) and for ATT ($t = 1.44$, $p = 0.089$, $r = 0.39$). There was no significant difference in PQ ($t = -0.67$, $p = 0.516$, $r = 0.19$).

Similar statistical tests were conducted for the agreement scores from Figure 10.9. Significant differences were found for statements about the robots' ironic ($t = 2.34$, $p = 0.019$, $r = 0.57$) and humorous ($t = 1.85$, $p = 0.045$, $r = 0.48$) behavior while all other differences were non-significant, including naturalness ($t = 1.26$, $p = 0.116$, $r = 0.35$). Overall, the results demonstrate that participants correctly rated the ironic robot as significantly ironic and humorous. Moreover, it shows that user experience (especially the hedonic quality) was significantly higher due to the multimodal irony generation.

At the end of the study, each participant was asked which of the robots they would prefer overall. Seven participants preferred the ironic robot, four preferred the baseline (non-ironic) robot, and one was undecided. Thus, most participants seemed to prefer an ironic robot in the small talk dialog.

10.7. Technical Limitations

Potential limitations are associated with the specific chatbot and robot utilized in the technical implementation. Occasionally, the A.L.I.C.E chatbot produced confusing responses, which could have negatively influenced the participants' user experience in both conditions. For example, the chatbot sometimes had problems matching pronouns and extracting the relevant data from too complex responses. While the questionnaires did not address dialog *quality* directly, participants rated if the robots' responses were meaningful (see Figure 10.9). In both conditions, ratings were high in the mean, suggesting that the associated limitations were small.

While the irony generation approach works well for the Reeti robot at hand, it is yet unclear how well it will affect user perceptions of differently embodied robots or agents. The Reeti robot's overall appearance is already cute and potentially funny (e.g., it has big eyes, see Figure 11.1), which is beneficial for conveying irony and associated humor. Thus, the technical implementation tailors the Reeti robot and its set of modalities and actuators. When transferring this to another platform, the embodiment will determine which markers of irony can be implemented for the targeted hardware, which might differ from the evaluated hard- and software at hand.

Last but not least, the small talk scenario served as context for evaluation, which used irony whenever possible. There is not yet a mechanism for estimating the appropriateness of irony use, which would require the machine to "understand" the dialog content and context.

10.8. Conclusion

This chapter presented a novel rule-based approach for creating conversational humor for a social robot by transforming a non-ironic sentence into an ironic utterance. Additional multimodal markers of irony, such as appropriate prosody and facial expression, are added on top of the linguistic transformation in order to make the human recognize the irony. The literature on human irony inspired both linguistic and multimodal cues. The techniques for the transformation approach include NLP, NLG, and animation playback. The implementation uses CoreNLP, WordNet, SimpleNLG, SSML, Cerevoice TTS, as well as the custom *reeti-rest* software (see chapter 11) for creating and playing back animations and interfacing with the Reeti robot.

The user study hypothesized that proper and meaningful use of irony would improve the conversation skills of social robots and consequently improve the human's conversation experience. The study investigated the effect of a robot using irony in a small talk dialog scenario by comparing it to a version of the same robot, which makes no use of irony. Results have clearly shown that the irony generation approach works well and that, indeed, participants (1) were able to identify their robotic conversation partner's use of irony correctly and experienced associated humor, (2) associated a better user experience with the ironic robot and (3) overall, more participants preferred the ironic robot. The results have been consistent with what was expected in case the irony generation approach worked.

11. Implementing Multimodal Behaviors for the Reeti Robot

Implementing multimodal social robot behaviors requires software for interfacing with the robot's hardware. A flexible API is desirable for creating and manipulating robot behaviors during the application's runtime. For example, it might be necessary to change the robot's utterances based on the user's input, combine speech with appropriate, synchronized animation, and add secondary actions (Lasseter, 1987), gaze behavior, or randomized movements. Doing everything by hand is not always a reasonable approach. A technical basis is essential for implementing multimodal robot behaviors, whether they are manually designed or produced with a generative approach.

This chapter presents a new software developed as part of this thesis to help implementing the behavior generation approaches from chapters 7, 8, 9 and 10. It allows for dynamic generation and manipulation of the robot's verbal and non-verbal behaviors.

Parts of this chapter were presented and reviewed in Ritschel, Kiderle, and André (2021). The contents of this chapter expand this publication.

11.1. Robot Hard- and Software

The experiments in this thesis use the Reeti robot (reeti.fr, 2021), which has an extraterrestrial, childlike face (see Figure 11.1). It has motors in its head for generating facial expressions. These actuators control the eyeballs (horizontal and vertical rotation), eyelids (open/close), mouth (raise or lower left and right lip corner, open/close mouth with the lower lip corner, move top lip forwards/backward), and ears (rotate upwards/downwards). Moreover, the head can rotate (pan, tilt, and roll). There is one RGB LED on each of the robot's cheeks and an internal speaker for speech and audio playback.

The robot is running Ubuntu Linux (ubuntu.com, 2021). Proprietary software can be used for defining robot poses and for generating simple animations[1]. These tools run on the robot exclusively, which requires connecting a screen, mouse, and keyboard each time content is edited. Animations and audio files must be prepared in advance and saved directly on the robot's hard drive. The robot can be controlled remotely with the manufacturer's Reeti API for the C++ and Java programming languages. Functions include playing prepared animations, poses, speech, audio, and movement of single motors to a new position. The robot is Universal Robot Body Interface (URBI) and Robot Operating System (ROS) compatible.

[1] In this thesis, the terms *movement* and *animation* are used interchangeably for the movement or manipulation of a robot's actuators over time (including non-motorized actuators, such as LED lights or TTS output).

Figure 11.1.: Reeti robot.

The manufacturer's robot API has limitations, including the following (see section 11.3.1 for details with regard to animation). The API (1) cannot create keyframe animations with independent and parallel movement of several motors during runtime, and (2) cannot playback speech commands with dynamic text content in parallel to the robot's animation playback. Moreover, (3) it lacks essential function, such as setting the robot's system audio volume. However, the robot talks and animates in typical social robot applications simultaneously. Animations must be created on-the-fly depending on the user's input and synchronized to speech output. In addition, (4) all resources, such as animations or audio files, must be copied to the robot's hard drive upfront. The most important restriction is the missing programming interface for dynamic generation of keyframe animations, i.e., parallel and independent manipulation of all actuators and TTS output.

11.2. Overview

In order to overcome the aforementioned limitations, the *reeti-rest* software, API and plugins were written as part of this thesis (see Figure 11.2). This new technology controls the robot remotely via Hypertext Transfer Protocol (HTTP) commands over the network (access to the robot's desktop environment is not needed) and provides an extended API with more functions than the manufacturer's API. In particular, *reeti-rest* allows the programmer to generate and playback keyframe animations in code, which is impossible with the manufacturer's robot software. As a result, the robot animates with parallel movement of all motors and parallel TTS commands during the application's runtime. Poses and audio playback are supported, as well as TTS via the robot's internal or the commercial Cerevoice TTS software, without placing files on the robot's hard drive upfront. New functions include automatic eyeblinks and saccades for making the robot's appearance more lifelike, an alternative TTS system, and setting and getting essential parameters, such as the robot's system volume. The API generates multimodal behaviors during runtime. They are essential for speech, animation, facial expressions, and sounds in the scenarios at hand.

reeti-rest-server

The central component of the software is the *reeti-rest-server* application, which is an HTTP server. It provides most of the function for controlling Reeti robots, animation handling, gaze planning, resource loading and caching, converting data for the robot, calling subprocesses for TTS generation, and more. The server connects to one or more hardware robots or virtual robots (see below). A programmer of a social robot application interfaces with the server via the corresponding API (see below) or HTTP commands to control the robots.

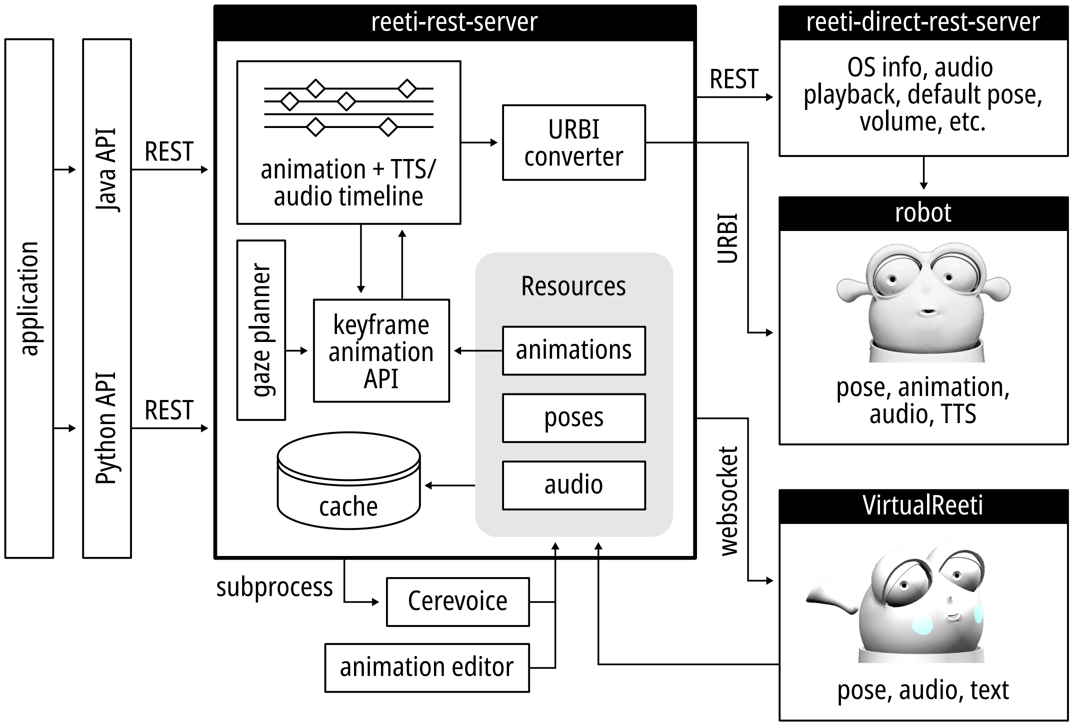


Figure 11.2.: Overview of the implemented *reeti-rest* software.

The user authors poses and animations with new tools (see below), or creates and manipulates them entirely in code during runtime. The server implements an alternative to the robot's internal TTS system by interfacing with the *Cerevoice* TTS system, which supports the SSML standard (w3.org, 2021). The server starts a Cerevoice process for each utterance and caches its output as an audio file. Audio files (including generated utterances of the Cerevoice TTS) are cached and streamed over the network to the hardware or virtual robot. Poses and animations are converted into URBI code and are sent directly to the hardware robot's URBI console over the network.

reeti-direct-rest-server

The helper utility *reeti-direct-rest-server* runs on the hardware robot itself for providing necessary functions to the server, which must have access to the robot's hardware, operating system (OS), and file system. Features include retrieving the robot's name, settings, neutral pose, and system volume. Moreover, the *reeti-direct-rest-server* also plays back audio files from the server (including the ones generated by the Cerevoice TTS). Therefore, the program starts an *avplay* process.

VirtualReeti

A virtual Reeti robot with a subset of the hardware robot's functions is a replacement for the hardware robot. Its primary purpose is prototyping and debugging social robot applications without a hardware robot but with the same API. The HTML5 application

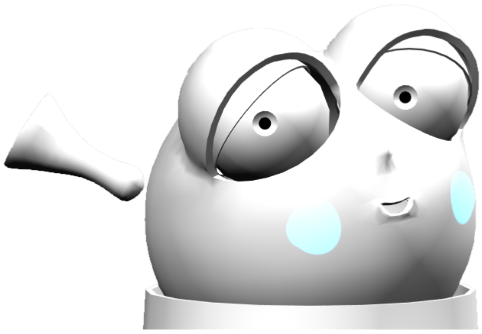


Figure 11.3.: VirtualReeti is a replacement for the hardware robot. It allows for creating and exporting poses in the web browser.

runs in the web browser, which benefits from being independent of a local installation. It communicates with the server via a WebSocket connection.

Besides the playback of poses, TTS and audio files, *VirtualReeti* (see Figure 11.3) can be used to create and export poses as JSON files, which can be loaded and sent to the robot via the API. The 3D model was created based on reference photos in the 3D graphics suite *Blender* (blender.org, 2021), which provides tools for defining the shape and deformation skeleton of 3D objects.

Application Programming Interface

The API was implemented for the *Java* and *Python 3* programming languages. It controls the hardware and virtual robots connected to the *reeti-rest-server*. The implementations of the APIs use HTTP requests to interface with the server. Adding support for additional programming languages is straightforward concerning the communication protocol.

In contrast to the robot manufacturer's software, the *reeti-rest* API can generate and manipulate poses and – in particular – also animations in program code. This essential benefit makes it possible to adapt these resources based on user input and any other external data during runtime and without physical access to the robot. For example, the implemented automatic gaze behavior (see section 11.5) makes heavy use of this feature.

11.3. Animation

One benefit of many embodied agents is their ability for animation, including gestures and facial expressions. A severe challenge for convincing and lifelike animation is good timing and parallel movement of several actuators. Humans and animals do not move from one static pose to another, but smaller movements merge and blend into each other fluently. In addition, agents with humanoid or zoomorphic embodiment also benefit from imitating behaviors of their archetypes. Typical examples include eyeblinks and gaze behavior: staring at one point all the time might appear unnatural because humans and animals frequently change their gaze focus. The *reeti-rest* software extends the Reeti robot's animation abilities for realizing these behaviors and complex animations.

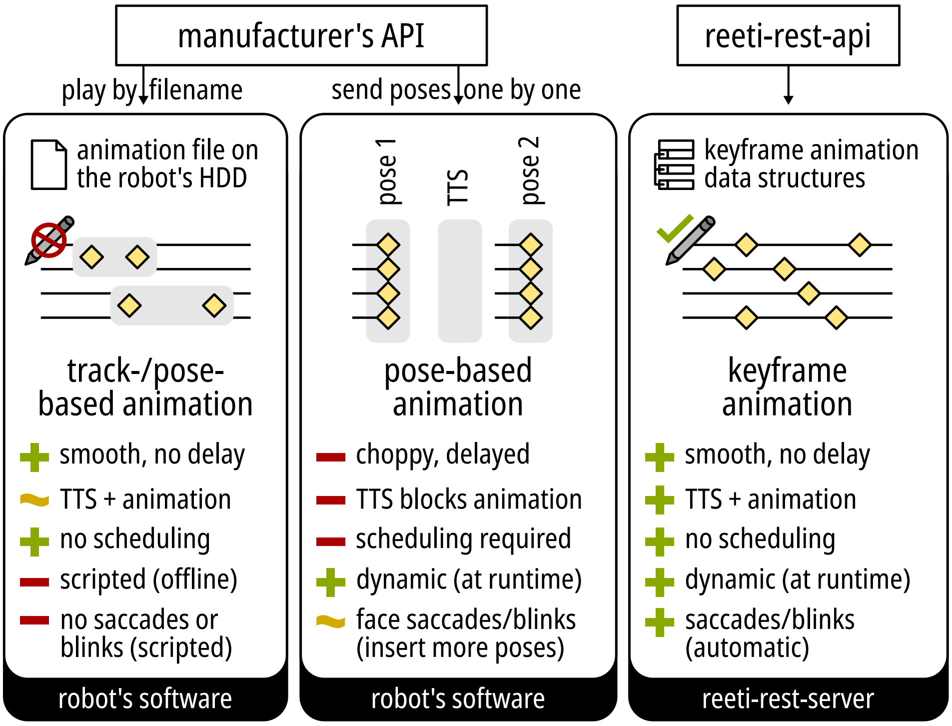


Figure 11.4.: Comparison of the robot animation capabilities delivered by the manufacturer and the new *reeti-rest* software.

11.3.1. Reeti's Native Animation Capabilities

When using the software by the Reeti robot's manufacturer, there are two options for animating the robot (see section 11.3.1). They are illustrated on the left side of Figure 11.4. Due to the limitations of these approaches, the *reeti-rest* software implements a more flexible and natural approach: keyframe animation (see section 11.3.2). The following subsections explain all approaches.

11.3.1.1. Track-/Pose-Based Animation

Option one is to use the manufacturer's graphical software for preparing a reduced form of keyframe animation (see the left part of Figure 11.4). These sequences are stored on the robot's hard drive and can be played back by calling a method of the manufacturer's robot API with the name of the animation file. This option has several limitations:

- The software does not support real keyframe animation with independent, parallel movement of all motors. Instead, several tracks exist to define a pose sequence of several motor groups (e.g., ears, all head motors, and eyes).
- The proprietary animation software runs on the Reeti robot exclusively. A screen, keyboard, and mouse must be connected to the hardware since one cannot run the software on a standard computer. However, the robot is typically used in "headless" mode (without any peripheral devices).

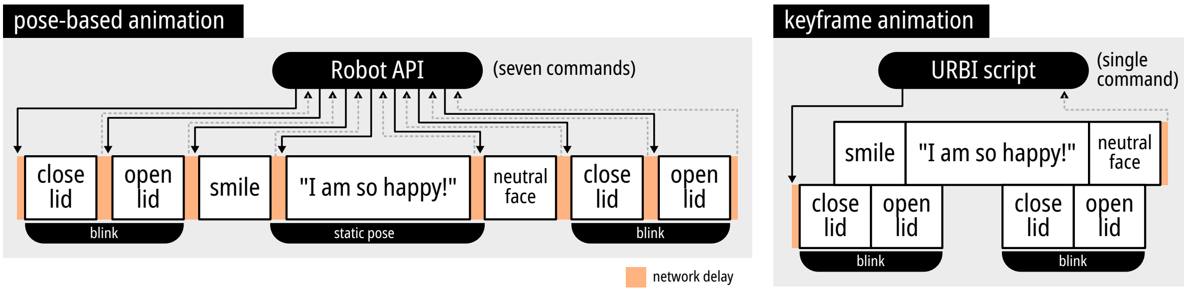


Figure 11.5.: Two approaches for generating animations during runtime. Left: with the Reeti robot's API, setting poses and talking happens one after the other, but not in parallel. Right: URBI scripts implement parallel and independent movements of all actuators (keyframe animation). The orange bars mark delays usually introduced by wireless network communication.

- Animations must be prepared upfront with the robot's software and are immutable during program runtime. It is impossible to generate new animations during runtime, e.g., based on the user's input.
- In animations, the texts of TTS utterances are saved directly as part of the animation, and thus they are immutable. A workaround is sending two commands: one for playing an animation without text and one for TTS speech output. Then, animation and text play consecutively, but not in parallel.
- Real-time saccades and eyeblinks cannot be realized during animation playback since all movements must be part of the animation upfront. They must be "baked" in the resulting animation, i.e., one must add every single eyeblink or saccade by hand.
- The resulting animation files are saved directly on the robot's hard drive, which makes it cumbersome to add, update or remove animations since access to the file system must be guaranteed. However, in typical social robot scenarios, the application often does not run on the robot itself but interfaces with the robot over the network.

One advantage of the robot's internal software is the fact that the robot itself schedules animation playback. The programmer only needs to call the animation playback function with the name of the animation file. Moreover, due to the single playback call, the overhead is minimal. It results in minimal delay when interfacing with the robot over wireless networks, and animation playback is smoother than pose-based animation.

11.3.1.2. Pose-Based Animation

Option two for realizing animations based on the manufacturer's software is to send poses one after the other (see Figure 11.4 in the middle and the left part of Figure 11.5). In contrast to the approach above, this gives more flexibility: the programmer can define poses during the program's runtime to create animations. However, this option has limitations, too:

- The programmer must implement all the scheduling, saving, and loading poses on its own.
- Posing and TTS commands cannot run in parallel since poses do not contain text. The robot can never move and speak simultaneously.
- Due to the necessity of sending many poses one after the other, there can be a significant delay when using wireless networks, which results in choppy animation.
- Parallel movement (and thus, keyframe animation) is impossible since pose commands block the robot's animation. One pose defines the values of all motors; it cannot move one motor and start a movement of another motor while the first movement is still playing.

The left part of Figure 11.5 shows an example sequence of pose and TTS commands, which, in combination, aim to express happiness. Smiling is used as a facial expression to support the utterance "I am so happy!". In addition, the robot's eyes blink two times. The command sequence is processed one after the other, and the user must wait for the robot's following speech output until all movements finish.

11.3.2. Keyframe Animation

The *reeti-rest* software implements a keyframe animation approach in order to overcome the limitations of the software by the robot manufacturer (see section 11.3.1). Keyframe animation is a technique from traditional animation. The *keyframes* are the most critical frames, which give the animator an idea of the extreme poses of a character throughout the animation. Animators draw keyframes first. After that, they draw the *in-between* frames. Interpolation algorithms do this in computer animation and robotics.

Each robot's actuator (motors, LEDs, text output) has a track containing an arbitrary number of keys. A key is a tuple of time t and value v information, which is interpreted as *the motor has value v at time t*. The tracks are independent of each other, i.e., keys on different tracks do not need to share the same time offsets or values, which allows for independent and parallel animation of all its actuators. Interpolation algorithms calculate the values for each point in time computationally. For example, the robot can start tilting its head from seconds 1 to 10 with multiple eyeblinks in between and a head roll from time 5 to 6. In addition, the robot can play back audio or speech during the animation. This results in the following advantages over the manufacturer's software:

- Animations can be created entirely in code or loaded from hard drive (JSON files). In both cases, they are modifiable during program runtime. For example, keyframes (including TTS texts) can be created, modified, or deleted.
- Due to this flexibility, the *reeti-rest* software can insert eyeblinks and saccades automatically out of the box.
- There is no need for storing animations on the robot's hard drive, which makes programming much more flexible.
- The tools for creating animations run on all important OSs.

Listing 11.1: A small part of an URBI script implementing keyframe animation with independent, parallel movement of motors, text output and LED color.

```
{ Global.servo.neckTilt=50 smooth: 0.5 | Global.servo.neckTilt=75 smooth: 1.5 } & {
Global.servo.rightEyeTilt=40 smooth: 0.75 | Global.servo.rightEyeTilt=19 smooth: 1.0 }
& { sleep(1.0) | Global.servo.changeLedColorRGB(2,1023,1023,0,1) } & { Global.tts.say("
\\language=English \\voice=Simon \\volume=50 I am so happy!") } |
```

Implementing Keyframe Animation with URBI Scripts

The *reeti-rest-server* converts the keyframe animation timeline to an URBI script as follows (see Figure 11.2 and Listing 11.1):

- Curly brackets encapsulate each track. Ampersands (&) separate the tracks.
- The command `motor=value smooth: offset` sets keyframes, where *motor* is the actuator's identifier and *value* is its value at time *offset* (offset is relative to the preceding keyframe).
- Vertical strokes (`|`) separate keyframes.
- The command `changeLedColorRGB` sets LED keyframes in combination with a preceding `sleep(offset)` command, with *offset* being the keyframe time.
- The `say()` function is used for TTS output.
- A vertical stroke (`|`) terminates the resulting script.

Based on the script, the URBI console calculates the in-between values and translates the instructions into hardware movements. Listing 11.1 shows a small excerpt of an URBI script for the Reeti robot. For example, the *neckTilt* motor is set to value 50 after 0.5 seconds. 1.5 seconds later, its value is set to 75. *rightEyeTilt* is animated in parallel. *changeLedColorRGB* is used to set the robot's LED in combination with the *sleep* function, which sets the offset on the timeline. At the beginning of the animation, the robot starts its utterance “I am so happy!”.

As illustrated on the right side of Figure 11.5, the keyframe animation approach enables the robot to produce utterances with parallel movements, such as eye eyeblinks, facial expressions, or gaze behavior at the same time. There is only one network request required between the *reeti-rest* API and the *reeti-rest-server*, as well as between the *reeti-rest-server* and the robot's URBI console. The result is less network overhead and, thus, fewer delays than animation based on multiple pose requests. The generated animations are scheduled automatically by the robot's URBI runtime, which has the additional benefit of smooth animation playback.

Figure 11.6 shows stills[2] of the pose-based and keyframe animation approach from Figure 11.5. In the top row, all movements are limited to distinct poses. They are processed

[2]Comparison video: https://archive.org/details/robot-keyframe-movement

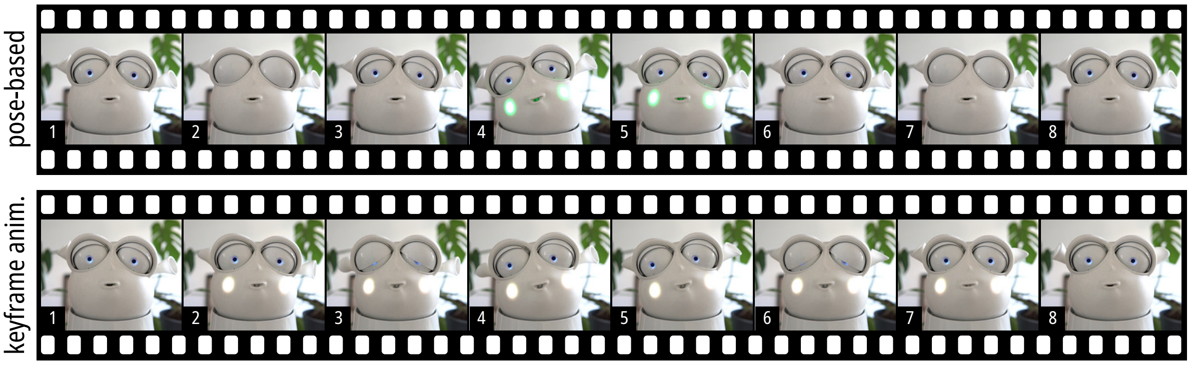


Figure 11.6.: Stills of the animation from Figure 11.5. Top: the robot presents several poses and an utterance sequentially without parallel movements. Bottom: several actuators move independently and in parallel (keyframe animation).

one after the other (e.g., 1–3 eyeblink, 4 smile, talk, 5 return to the neutral position, 6–8 eyeblink). In the bottom row, keyframe animation blends movements into each other (e.g., 1 start ear movement and speaking, 2 set LED and start eye movement, 3 start eyeblink and head movement, 4–5 continue the head movement, eyeblink already finished, 6–7 eyeblink, 8 stop eye movement, 9 stop ear movement).

11.4. Text-To-Speech

The Reeti robot has an internal TTS system: Loquendo TTS. It provides one male and one female voice for several languages, including English and German. Like the SSML standard, Reetis internal TTS system provides text commands to control basic parameters, such as playback speed, pitch, volume, and timbre during an utterance. Moreover, it is possible to add effects, include pauses of specific duration, stress on words, and play paralinguistic sounds, such as coughing, laughing, sniffing, and more. However, it does *not* support the SSML standard.

The *reeti-rest* software integrates the commercial Cerevoice TTS system as an alternative to the robot's internal TTS. Cerevoice implements the SSML standard and has more parameters for controlling the resulting speech audio. In contrast to the robot's internal TTS system, most Cerevoice voices sound very human and mature.

Implementation

Figure 11.7 provides an overview of the TTS implementation. URBI scripts trigger the robot's internal Loquendo TTS (a). The URBI converter in the *reeti-rest-server* converts the utterance to URBI code each time a TTS or animation is triggered. The result combines all movements, and TTS commands for parallel playback of speech and animation.

When using Cerevoice (b), the Cerevoice software produces an audio file. The file is cached, and a request is sent to *reeti-direct-rest-server*, which streams the file from *reeti-rest-server* for playback via the robot's speaker. All animation is played back in parallel with an URBI script as in (a).

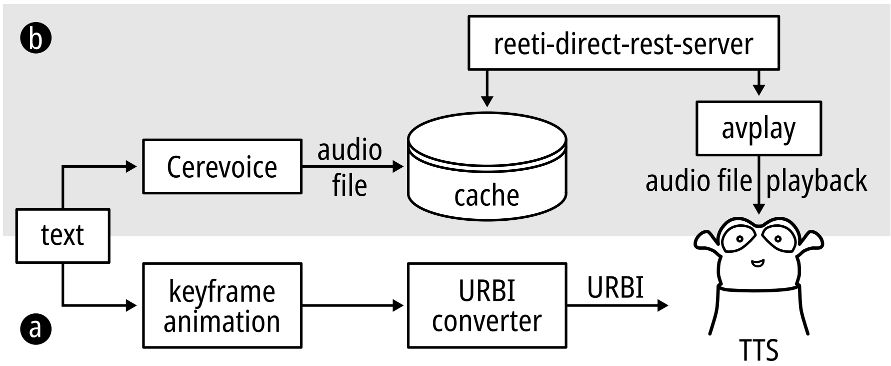


Figure 11.7.: Speech audio is generated either (a) with the robot's internal TTS system or (b) with the commercial Cerevoice SDK.

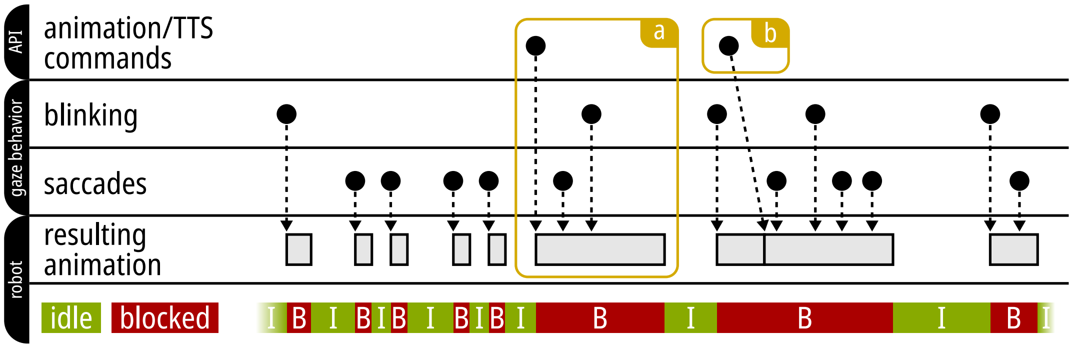


Figure 11.8.: Exemplary robot behavior. (a) Eyeblinks and saccades are planned upfront and merged into a single animation command. (b) The robot blocks during command execution and queues pending commands.

11.5. Automatic Gaze Behavior

The *reeti-rest* software implements automatic gaze behavior, which aims to give the robot a more natural and lifelike appearance. These behaviors include eyeblinks and saccades based on the background provided in chapter 3. No additional programming is needed for scheduling and triggering the robot's blinks and saccades. Automatic gaze behavior can be activated and deactivated with the API.

There are a few restrictions concerning the Reeti robot's hardware. On the one hand, the motors are pretty noisy, which may annoy users if it occurs too frequently. On the other hand, the motors cannot move as fast as human muscles, such as the eyelids. Thus, the robot's gaze behaviors must be implemented in a reduced form, considering the hardware restrictions.

Eyeblinks and saccades trigger in randomized time intervals. Technically, an animation is created and played back each time. Due to the restrictions of the robot's URBI console, which blocks until the current command finishes (e.g., when executing TTS or animation playback), it is not possible to trigger eyeblinks or saccades during animation playback. Instead, the software plans eyeblinks and saccades upfront. See Figure 11.8 for examples of different cases of robot commands, automatic gaze behavior, and resulting animations. Eyeblink and saccade planning works as follows:

- The software plans eyeblinks and saccades upfront, i.e., the program knows how much time is left until the next eyeblink or saccade occurs.
- When it is time for an eyeblink or saccade, and the robot is idle at that moment (i.e., it does not talk or play any animation), the eyeblink/saccade animation is played immediately.
- When an animation playback command is received, eyeblinks and saccades are scheduled for the duration of the animation unless the animation controls the eyes or eyelids directly. The software combines the original animation with the eyeblink/saccade animation into one animation.

Regarding motor movements, eyeblinks are animations that close and open the robot's eyelids. Starting at the robot's neutral eyelid position, the motor movement for closing and opening the eyelids is 150 milliseconds each, which is about the maximum possible speed. While this is much slower than human eyelid movements, this is a technical restriction. As a workaround, the robot does not close the eyelids completely, but only for about 50 % of the motor range.

Saccades are implemented based on an algorithm that realizes a combined eye-head movement (see also Ruhland et al. (2014)):

- By default, the saccade rotates only the eyeballs.
- Both neck and eyeballs rotate in case the angle between the robot's neutral (centered) viewing direction and the desired focus point exceeds a threshold. For humans, the threshold is approximately 15–20 degrees (Stahl, 1999).
- The roll axis is not modified.

11.6. Conclusion

This chapter presented the technical foundation for the multimodal behavior generation approaches of the preceding chapters. The newly developed software enables the programmer to generate multimodal behaviors dynamically during an interaction, including the independent and parallel movement of all actuators of the Reeti robot. The software and API can combine and synchronize speech, audio playback, and movements. On top of that, the approach schedules automatized gaze behaviors, which aim to make the robot's appearance more lifelike by equipping it with automatically scheduled non-verbal behaviors, reducing the effort of manual movement creation. Thus, the presented technology is an essential step for implementing expressive robot behaviors for the Reeti robot, which might also apply to other robots with an URBI interface. The development of this software is an essential technical contribution and serves as a fundamental basis for all conducted experiments in Part IV.

Part IV.

Non-Functional Adaptation

12. Socially-Aware Reinforcement Learning: From Start to Finish

Applying the RL framework involves several challenges. As described in chapter 2 modeling an RL problem is a critical step, which requires a precise idea of all relevant aspects: state space, action space, and reward, as well as several properties of the problem and learning algorithms. However, how must this be done for social robot behavior adaptation? What are the requirements, and which restrictions exist in the context of HRI? Which role does the user play, and which human and task-based data guides the robot towards the goal of the adaptation process? Which pitfalls exist when simulating and evaluating such real-time learning processes? Moreover, how does the learning task at hand influence algorithmic decisions?

This chapter presents a structured overview and proposes several steps for modeling, simulating, and evaluating RL problems from start to finish. Starting with the unique requirements and desires of the HRI context, it explains the process of modeling the problem, incorporating the human in the learning process, algorithmic considerations, simulation, and evaluation challenges. Building on this and the multimodal behavior generation techniques from Part III, chapters 13 and 14 present experiments which implement the presented procedure.

Parts of this chapter have been presented and reviewed in Ritschel, Baur, and André (2017a), Ritschel (2018), Janowski, Ritschel, and André (2022), and Kiderle et al. (2021).

12.1. Preliminary Considerations

Before starting with a concept for setting up social robot adaptation based on RL, it is important to outline the motivation for choosing the RL framework. As briefly hinted in section 6.2, RL fits well in the typical HRI loop due to the learning agent's autonomy. The following sections present multiple reasons why this thesis uses RL for implementing adaptation.

12.1.1. Stakeholders

As illustrated in Figure 12.1 multiple stakeholders are involved in an adaptation process: the user, the robot, and the system designer. The user and robot are involved directly in the interaction. The rest of this chapter is written primarily from the system designer's perspective since they combine most pre- and post-tasks necessary for analyzing, modeling, simulating, tuning, and evaluating an adaptation approach. Once prepared, the interaction and adaptation should run autonomously and independently of expert supervision.

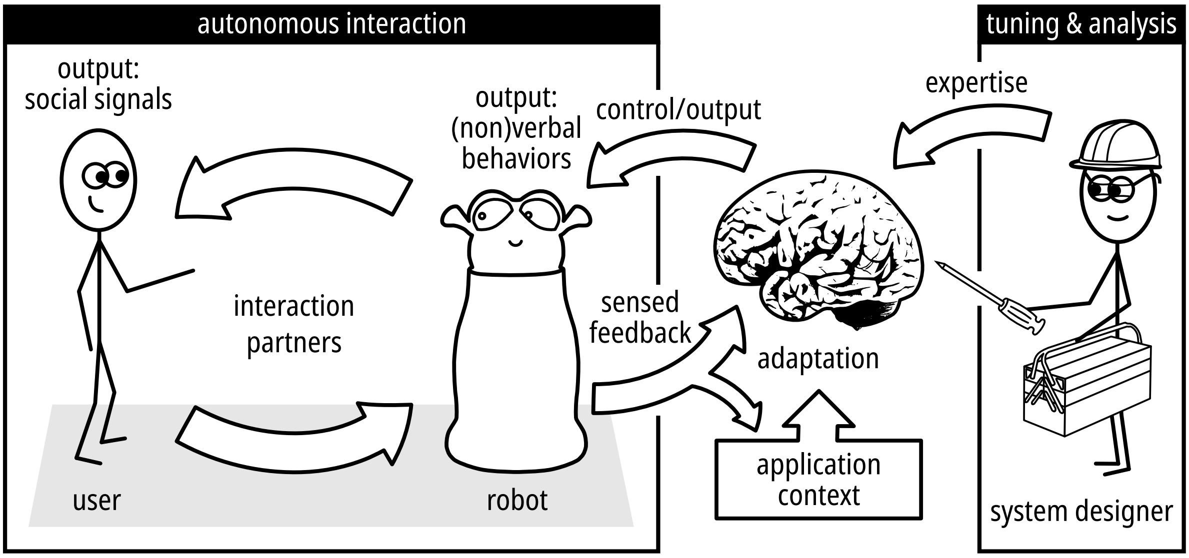


Figure 12.1.: User, robot, and system designer have specific functions in a socially-aware RL process.

12.1.1.1. User

As already outlined in section 6.1, the primary motivation for adaptation is addressing individual users' needs and making interaction more comfortable or efficient for the user. Thus, the human is the center of the whole adaptation process. They have the greatest benefit but also some responsibility in giving feedback to the robot (not compulsory when using task-based feedback exclusively) for making adaptation possible. At an abstract level, the user acts as input (being a receiver for the robot's behaviors) and at the same time also as output (giving feedback to the adaptation process).

12.1.1.2. Robot

From the technical perspective, the robot primarily serves as an output medium presenting verbal and non-verbal behaviors to the user. These behaviors include i.a., speech, gaze, posture, and gesture. Often – but not necessarily – the robot also serves as an input medium, e.g., when sensing the human's social signals. In this case, it might be useful to rely on the robot's internal hardware, which might be closest to the user (e.g., webcams in the robot's eyes). In some cases and depending on the platform, this might not be sufficient, e.g., when specific sensors are required, or the robot's hardware does not provide sufficient power or quality. While the robot is not a human stakeholder but a humanoid machine representing the interaction partner, its multimodal output aims to mimic human behaviors. It is listed here as a stakeholder since the task of this machine is to provide a humanoid and natural interface for the user.

12.1.1.3. System designer

The system designer is not involved in the interaction itself directly. Nevertheless, they are responsible for setting up, observing, and evaluating the adaptation process. Being the expert, the designer needs to analyze the problem, interaction, and results – in advance and afterward, ensuring that the adaptation fulfills its purpose. Thus, they have the most responsibility and needs to solve many tasks.

12.1.2. Requirements

The HRI context comes hand in hand with a few requirements for adaptation:

Real-time The adaptation process should work in real-time during the interaction to learn simultaneously and react as quickly as possible. Suppose the robot needs to wait until the end of the interaction when the user fills out a questionnaire to provide information about the user experience or individual preferences. In that case, the robot's goal might already be missed.

Autonomy Adaptation should require only a minimum of additional user interaction, or ideally none, to save the user from additional effort and not disrupt or distract from the actual task. Moreover, people do not necessarily like to teach machines and thus serve as an "oracle" (Amershi et al., 2014) all the time. Depending on the abstraction of the preferences to learn, humans might not even be able to tell what their actual preferences are.

Uncertainty The adaptation approach must be able to handle uncertainty, at least when used in a non-laboratory environment. Different kinds of noise occur in HRI, such as nondeterministic user behavior and noisy feedback (see section 12.3.1.3). See section 12.4 for details.

Long-term For nonstationary problems (see section 2.4.1 and section 12.4.1), adaptation should not be limited to an initial training phase, but happen constantly. User preferences might change in the future, which requires continuous learning. Otherwise, the robot's behavior might not be optimal for the user in the long run.

12.1.3. Desirable Properties

Apart from the essential requirements listed above, the following desirable properties for implementing adaptation may not be required or may not hold in some scenarios. One might not be able to realize all of them for a particular task. For example, the user's privacy might limit the set of sensors or data that the robot can use and acquire during the interaction. The following list constitutes a starting point for finding a good compromise during the problem modeling process.

Unobtrusiveness Similar to the fact that the adaptation process should not distract from the task, it should also be unobtrusive. Ideally, the robot should get spontaneous human feedback without requiring the user to think about it for a long time, which might change the answer when evaluating it rationally. The ultimate goal is to sense the actual user's opinion quickly: the user should act naturally, just as they would interact with a robot without adaptation.

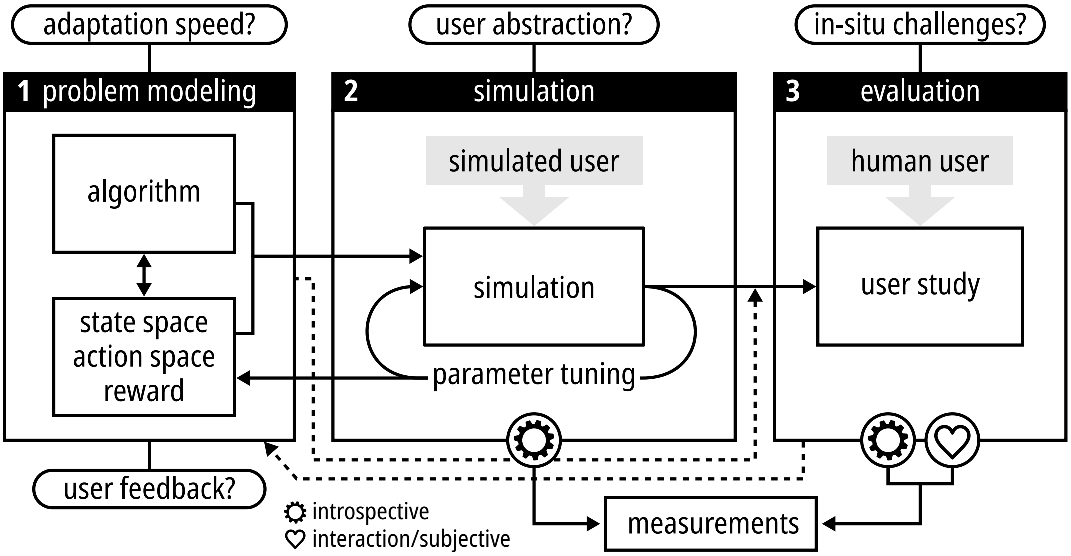


Figure 12.2.: The process of developing, simulating, and evaluating an adaptation approach based on RL.

Task independence If possible, the implementation should be independent of the task, making it reusable for various contexts and scenarios. Minimizing and eliminating task-dependent information is desirable when modeling the RL problem. However, task-specific data can be very informative and valuable. Finding a good compromise is an important aspect when modeling the adaptation process.

12.2. Planning and Conducting a Reinforcement Learning Experiment

The previous sections listed requirements and desired properties for an adaptation process of a social robot. RL is the method of choice in this thesis since it fulfills the requirements (see also chapter 2): an RL agent can learn autonomously based on scalar feedback in real-time during interaction with a potentially noisy environment. Besides episodic tasks, RL can also run for a very long time and adapt to nonstationary tasks, making it suitable for long-term learning. Thus, the following sections present a conceptual framework for modeling, simulating, and evaluating a real-time social robot adaptation approach based on RL, focusing on including human social signals in the learning process.

12.2.1. Overview

Figure 12.2 provides a general overview of designing and evaluating an adaptation approach based on RL. The process has three interdependent stages:

1. During *problem modeling*, the RL task needs to be specified, including the agent, its environment, and the learning algorithm.

2. This model should be verified first before conducting a human evaluation of the adaptation approach. A *simulation* helps identify and fix implementation issues, re-iterating the model and tuning parameters.

3. In the last step, the adaptation approach is *evaluated* with human users for insights about the robot's learned behaviors and its impact on the interaction.

The stages should follow the order listed above. First, the problem must be modeled in-depth. This includes the action space, state space, reward and the selection of an appropriate learning algorithm, taking into consideration RL specific properties (see section 12.2.2 and section 12.2.3) and properties resulting from the interaction with the human (see section 12.4). One central question in this context is how to include user feedback and human social signals in the RL process. Different types of data and sources of feedback might be available in different tasks and contexts. All these considerations are reflected in the RL model and thus directly impact the adaptation speed.

The simulation (see section 12.2.4) is optional. However, it makes sense to simulate the result of the modeling stage first to find mistakes and fix them before spending much time and effort evaluating the approach in a natural environment. During simulation, the model is evaluated primarily with a technical focus. The key challenge is to replace the human user with abstracted simulated user behaviors. In contrast to a human evaluation, a simulation requires less cost and time. Nevertheless, it is essential before deploying the adaptation approach to the real environment because it makes it possible to re-iterate and tune the model and observe the resulting implications. The problem modeling and simulation stage require expert knowledge concerning RL and HRI.

After modeling and simulating the task, a human evaluation (see section 12.2.5) gives insights into the learning progress and effects in a natural HRI context. Studies are conducted in the lab or "in the wild" in an in-situ study. Human evaluations go hand in hand with additional challenges and noise from the real environment and interaction (see section 12.2.5.1 and section 12.3.1.3).

There is no unique solution for modeling, simulating, and evaluating problems and also no single solution on how to approach these three steps themselves. Thus, this section shows the necessary considerations that must be made during this process for modeling RL problems for social robot behavior adaptation in HRI.

12.2.2. Problem Modeling

In the first stage, details of all aspects of RL have to be fleshed out. The dependency graph in Figure 12.3 illustrates this process. The system designer must clearly define the root nodes (environment, adaptation goal) before continuing with more detailed aspects in the child nodes. The black leaf nodes (state space, action space reward, and the specific RL algorithm) must be specified based on their predecessor nodes. Section 12.2.3 presents a tool specifically for the HRI context, which assists in this process.

The user is part of the RL *environment* since the learning agent's actions cannot directly control the user's behaviors and reactions. Nevertheless, the state space and reward may include aspects of sensed human behaviors, and the action space may contain actions that aim to influence the user's behaviors (see below). See also section 12.4 for algorithmic implications.

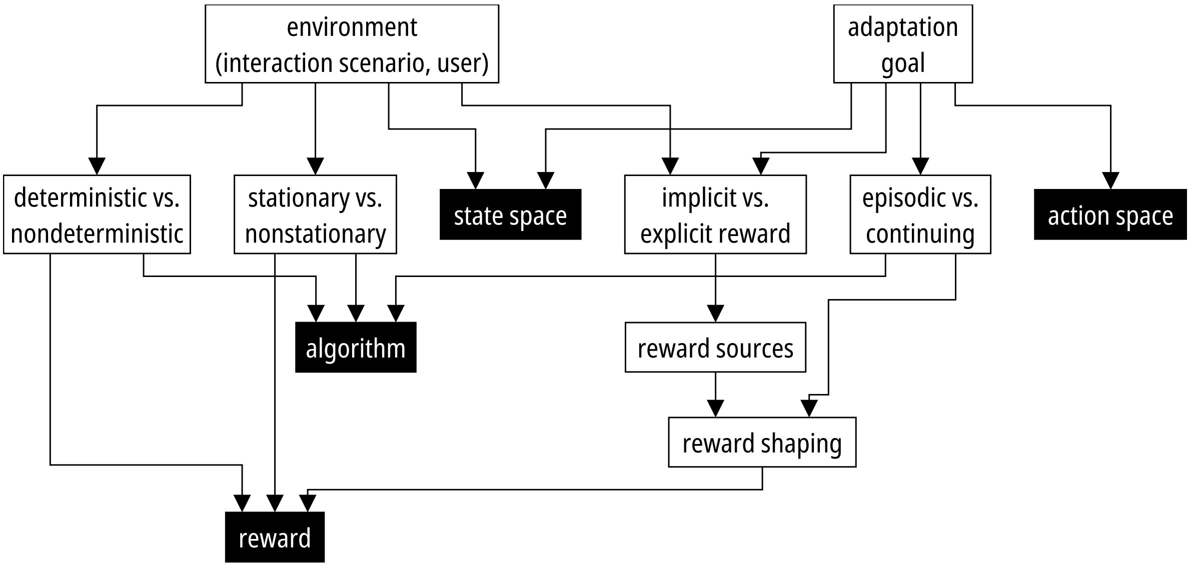


Figure 12.3.: Necessary considerations during modeling and their mutual influences on aspects and properties of a socially-aware RL adaptation process.

12.2.2.1. Action Space

It begins with thinking about the overall goal of the adaptation process. Which aspect of the robot's behavior should be optimized? For example, in a puzzle task, the goal could be keeping the user engaged or helping the user solve the task quickly, i.e., increasing task performance. After defining the adaptation goal, the system designer must identify the agent's actions for achieving this goal. They span the action space, which may also include manipulating the robot's behaviors. In the puzzle task, user engagement might increase if the robot comments or uses humor, while different kinds of advice might be helpful when focusing only on task performance. Actions for non-functional adaptation include switching communication modalities and tweaking parameters of the verbal and non-verbal robot behaviors, such as setting, increasing or decreasing gesture speed, talking speed or voice pitch, changing formulations, and much more. The action space depends on the interaction goal and the environment.

As mentioned in section 6.1.3.3 one should also consider whether manipulating the robot's behaviors impacts user experience. Do the actions change the robot's behaviors gradually, or do they cause immediate and inconsistent, big changes, which might be surprising or irritating to the user? An action might result in the robot changing its behaviors immediately or over a longer timespan, from seconds to minutes, depending on the implementation. Of course, if an action's execution takes longer, the robot needs more time for learning. Both aspects need to be balanced. It may be possible to split the change into several smaller steps for improved consistency. For example, instead of setting a parameter from minimum to maximum value, the action can increment or decrement the parameter in smaller steps, changing the robot's (non-)verbal behaviors gradually.

12.2.2.2. State Space

The state space depends entirely on the environment, task, and interaction goal. It needs to capture all relevant aspects and features of the environment and cover all relevant situations that potentially occur during the interaction. In HRI, the user is typically a crucial part of the environment. Any user-, interaction-, or task-dependent data, but also the state of the robot, can be relevant for the state space. It may include the user's mood, interest in the interaction, engagement, task performance, and much more. For the puzzle task mentioned above, the state space might include the number of remaining pieces, the user's current mood or engagement, or the remaining time.

12.2.2.3. Learning Algorithm

The learning algorithm depends on multiple problems and properties of the environment. For example, the adaptation goal determines whether the task is episodic or non-episodic. It is episodic in case there are final states in the learning process, which naturally terminate the task, such as complete failure (e.g., the user quits the interaction) or success (e.g., when solving a puzzle). Otherwise, the adaptation process is open-ended and runs forever until somebody stops it manually.

Another important property is whether the environment is deterministic or nondeterministic (see also section 12.4.2). Most probably, human reactions are nondeterministic and thus vary from time to time. For example, external influences can bias the user's reactions. In this case, the user's reactions are not caused by the adaptation process (see section 12.4.2). Moreover, the environment might also be nonstationary (see section 12.4.1), e.g., when the user's preferences or attitudes change over time.

Section 2.6 listed basic learning algorithms, including *k*-armed bandits for stateless problems and the Q-learning algorithm for stateful tasks. The bandits and Q-learning algorithm are suitable for both stationary and nonstationary problems.

12.2.2.4. Reward Signal

After specifying the adaptation goal, the system designer needs to identify potential sources for reward calculation. Given the goal, they must think about the information that indicates success for the learning agent. Since functional and non-functional adaptation affect the set of actions (i.e., what the robot does), the type of adaptation does not necessarily impact the sources of reward. Typical sources include the task itself (such as task performance in terms of elapsed time, amount of answered questions, and whether the problem is solved) and information about the user (such as user engagement, mood, and other interaction dynamics as described in section 12.3.4).

Task-related and user information can be relevant to functional and non-functional adaptation goals. When combining task-related and user-related data, the latter might be implemented as a shaping reward, guiding the adaptation process towards the system's goal. Again, the design of the reward signal depends entirely on the task at hand.

The reward can be of implicit or explicit nature (see section 12.3.2). In addition, the data also differs in terms of objectivity or subjectivity. For example, user information can be objective (e.g., demographic data, measured interaction dynamics) or subjective (e.g., user experience). Task-related information typically is objective (e.g., task performance). Explicit, implicit, objective, and subjective data are not mutually exclusive. For example, the user can explicitly provide subjective information about their mood with a button press, or the system can estimate their happiness based on the sensed facial expressions. The decision of using implicit, explicit, objective, or subjective reward sources (or a combination of them) will also come hand in hand with weighing up the advantages and disadvantages of these options (see also section 12.3.2).

The next step is to consider how this data can be combined or used to guide the robot toward solving the problem. For example, in a puzzle task, it might be sensible to combine user engagement and task performance for a satisfactory user experience while also keeping an eye on efficiency. In the context of dialog systems, Walker et al. (1997) present the PARADISE framework for measuring user satisfaction. Rieser and Lemon (2011) combine qualitative measurements with subjective user ratings for calculating rewards based on task performance (efficiency of the dialog system) and user experience (user satisfaction, using PARADISE) in the context of adaptive dialog systems with RL. Thus, relying on reward sources can be a solution, e.g., by weighing them in terms of their importance. In an industrial context, solving the task most efficiently might be more important than the user's happiness within this interaction. Thus, the implementation can give task performance more importance for calculating the reward. In an entertainment context, the reverse situation might be desirable by focusing the reward signal more on user engagement, thus giving engagement more weight. The specific design of the reward signal depends on the adaptation goal.

Again, the task's episodic or non-episodic notion also comes into play. A typical approach in an episodic task is to send a positive or negative reward signal only in case of success or failure when entering a final state. Reward shaping is an opportunity to guide the robot towards the goal based on expert feedback or other valuable data observed during the interaction. For example, the user solves a puzzle task successfully. However, user engagement can be low over a long time during the interaction, indicating how the robot behaves is suboptimal despite finally solving the task. In such cases, it also makes sense to consider user engagement for calculating a reward during the interaction. Thus, reward shaping is one opportunity for reacting and adapting to the user's behaviors during the interaction. It becomes even more important in non-episodic tasks, which do not end in final states and thus require continuous rewards.

The reward signal also depends on the environment. In a nondeterministic environment, the reward will be noisy, which will most probably be the case when relying on human input. In a nonstationary environment, rewards will change over time, as do the real values of the underlying, changing problem. Again, this can happen, for example, when the user's preferences or attitudes change over time. The induced variance in the reward signal thus affects the set of appropriate learning algorithms, which must be able to cope with it.

12.2.3. The Adaptation Triad

Figure 12.4 shows a visual tool called the *Adaptation Triad*. It helps during problem modeling for brainstorming, collecting, and sorting relevant data for the RL model. The

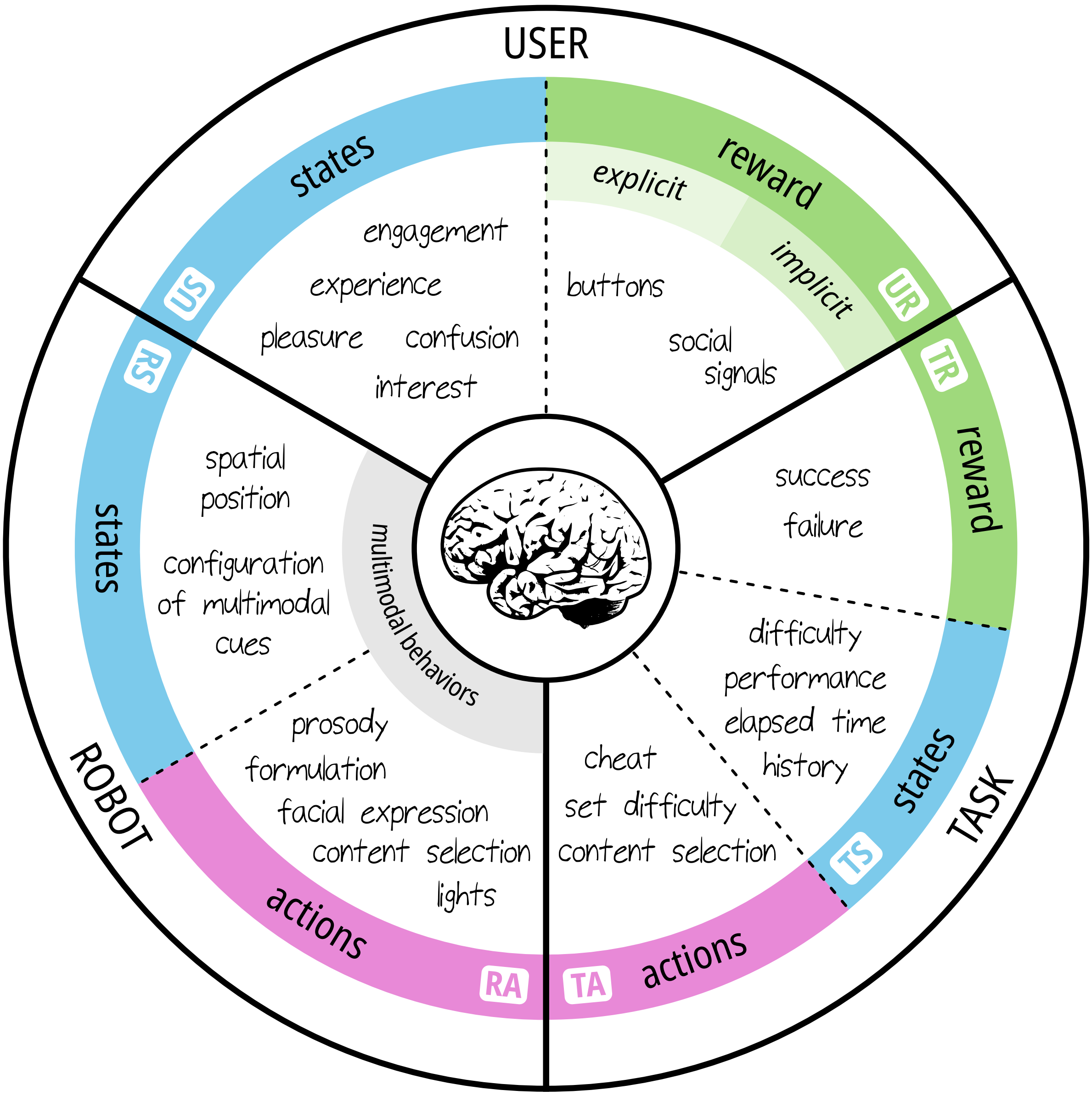


Figure 12.4.: The *Adaptation Triad* sorts user, robot and task data integrated into the RL problem model. The filled example contains a non-exhaustive list of potential high-level example entries.

system designer sorts the data in different cells depending on their intended purposes, such as potential state variables, actions, or reward sources. At the same time, the diagram automatically gives a rough estimate of the complexity of the RL problem: the fuller the diagram, the more complex the resulting problem. A particular advantage of the Adaptation Triad is that it explicitly splits the learning agent's environment into the user, task, and robot. One can always see at a glance the specific role of the human and robot and which data is needed and processed. The diagram consists of three parts: the user, robot, and task. Each of these parts is further subdivided into subcategories, corresponding to aspects of RL.

12.2.3.1. Task

The interaction task is part of the adaptation process if the robot assists the user while processing a task, such as in socially assistive scenarios. For example, adaptation might address how the robot should behave in different situations of the task, and manipulating the task itself might also be an option for the RL agent. The task can contribute to the set of states (TS), actions (TA), and also as sources of reward (TR).

12.2.3.2. Robot

The robot is part of the set of states and actions. For example, the learning process might change the robot's multimodal behaviors, e.g., to explore different modalities, formulations, and more. Actions (RA) manipulate these behaviors. Moreover, the robot's current configuration of these modalities or other properties might be relevant, resulting in different robot states (RS). There is no reward section in the Adaptation Triad for the robot because it is only an output medium, which does not provide any reward signal. Data sensed by the robot's hardware might be used as reward signals, such as the user's social signals or task-specific data captured with internal cameras. Nevertheless, this data semantically originates in the user or task category.

12.2.3.3. User

The user contributes to the Adaptation Triad in terms of states and rewards. For example, the user's state may include their current mood, engagement, or other data, which might be important to react to changes in these variables. Thus, information about the user might serve as potential user state variables (US). Similarly, the user might also contribute explicit or implicit reward signals (UR), e.g., by pressing buttons for rating the robot's actions or sending social signals, such as engagement or amusement. There is no action category for the user in the Adaptation Triad since the learning agent cannot control the user. Requesting the user to do something semantically is a task or robot action.

In some interaction scenarios, there might be a fluent transition between task and robot states and actions. For example, in case of robot adaptation happens in an interaction scenario without assistance (e.g., robot storytelling or joke-telling, which aims to entertain the user while the user is passively consuming the content), the task and robot category might collapse into one single unit.

12.2.4. Simulation

The first step for evaluating the RL model typically involves conducting simulation experiments. They serve several purposes at the same time:

1. checking whether the learning agent's technical implementation is correct,
2. checking whether the RL model converges,
3. checking how fast it converges given a specific parameter configuration,
4. checking how prone it is to noise, and
5. tweaking parameter values correspondingly.

The basic idea is to replace the natural environment with a simulated one driven by a rule-based or statistical model. Everything except for the environment remains the same, as modeled in step one. Since the user is part of the environment (see section 12.2.2), the simulation of human reactions constitutes a key challenge for simulations, along with potential task-related effects. In the context at hand, a *user simulation* implements those reactions to the learning agent's executed actions, which are expected to occur in the real-life evaluation. User simulations have also become a common approach, i.a., for training RL agents in spoken dialog systems (Schatzmann et al., 2006).

User simulations might use machine learning techniques by training and inferring the simulated user's reactions from real observations. For example, probabilities for different user reactions in different situations might be calculated based on annotated recordings. These probabilities then determine how the simulated user reacts to the agent's executed actions. Recorded and annotated interaction corpora can also be interpreted as sequential decision-making problems for training RL agents offline.

Measuring agent performance is the second key element of every simulation. As illustrated in Figure 12.2 introspective measurements (see section 6.1.4) are used for measuring performance in simulations. In RL, the measures presented from section 2.7 are used to get an impression of the agent's success or failure over time. For example, the average reward and the percentage of optimal actions should increase. The RMSE should decline. Otherwise, there are mistakes in the RL model, implementation, or suboptimal parameter values. Conversely, such insights point out technical and conceptual mistakes, which must be fixed by re-iterating, debugging, and improving the model. Running lots of experiments with different settings and configurations allows for eliminating as many potential mistakes as possible before initiating an evaluation with human users.

12.2.4.1. Advantages

Simulations have advantages when compared with the subsequent evaluation. Making a simulation often takes a fraction of the time it would take to conduct a user study. Depending on the complexity, simulations can sometimes run in seconds or milliseconds. Moreover, simulations are reproducible if implemented deterministically. They allow for tweaking parameter values by running the same experiment repeatedly while carefully making small changes. Thus, the designer has maximum control, which is handy for debugging the process. Potential implementation errors might be found soon with a simulation, and – most important – one can re-iterate and tweak the model for as long as it may be necessary. Going back to step one and re-iterating the model allows for excluding as many potential problems as possible during a human evaluation. The simulation also allows observing how the learning agent reacts to different external influences. For example, the simulation logic can implement different kinds of noise (see section 12.3.1.3). Due to the reproducibility, simulations also allow for isolated focus on one specific aspect or parameter without the influence of potential influences occurring in real interactions.

Since a simulation does not involve human participants, the overall cost of execution time and effort is low. A fast iteration cycle and batch processing are the results, making it possible to diagnose errors early and to start troubleshooting without running the risk of jeopardizing study results due to implementation errors or wrong problem modeling. A challenge in this context is that implementing the simulated environment is an additional possible cause of errors, which could lead to wrong conclusions.

12.2.4.2. Disadvantages

The major problem with simulations is that the environment – including the user – needs to be abstracted and approximated. Depending on the level of abstraction, the simulation might not emulate human behaviors and reactions adequately due to their complexity. In some scenarios, the user's reactions to the robot's actions might not be known upfront. However, this will also be one of the core reasons why an RL approach is desired.

The system designer must define the simulated user's behaviors and the accompanying environmental response. A rule-based approach can be programmed in code but might not emulate realistic human reactions. One approach for overcoming this issue is a combination with WoZ experiments (see section 12.2.5.2), which are also used in spoken dialogue systems (Rieser and Lemon, 2011). When relying on statistical models for user simulation, another challenge is to acquire the required data in sufficient quantity.

All in all, simulations should be considered a first evaluation step in parallel to problem modeling. They are important to ensure that the proposed adaptation approach works on a technical level and can achieve a specified goal. Due to abstractions made for the user model, a simulation is no guarantee for learning success during human evaluation.

12.2.5. Evaluation

The adaptation approach is evaluated with humans in the real environment during real-world evaluation. In case of technical problems, the system designer would need to go back to step one to re-iterate the RL model. Since the effort for a human evaluation is typically much higher than for a simulation, it is helpful to identify and address as many potential errors as possible in a preceding simulation. However, a human evaluation is the best opportunity for:

1. getting insights about the agent's performance in the user's real environment, and

2. getting subjective feedback from participants on user experience.

As illustrated in Figure 12.2 all of introspective, interaction and subjective measurements (see section 6.1.4) are used for measuring performance and for getting insights about the adaptation process. While a simulation primarily provides insights into the agent's performance in an abstracted environment, the subjective feedback is more informative about the impact on user experience and potential side effects. Moreover, different kinds of noise (see section 12.3.1.3) and other unanticipated problems will potentially occur in natural environments.

The agent's performance can still be automatically measured based on the runtime interaction data. However, those measures from section 2.7, which rely on the knowledge of the real values q_* (e.g., percentage of optimal actions, RMSE), are not applicable during evaluation since the real values are unknown in the real environment. User experience-related data can be acquired, e.g., based on questionnaires.

Advantages and disadvantages result from the opposites of their counterparts in simulations: user studies are more time-consuming and expensive, they often cannot be repeated or reproduced easily, and the system designer has less control than in a simulation. In addition, external influences might bias participants' reactions and decisions, which are not controllable by the agent and cannot be simulated. At the same time, studies in real environments are necessary to get these valuable insights.

All in all, evaluations typically follow an initial simulation. They are essential for getting insights beyond the technical performance, including user experience. The following section outlines additional challenges that arise for in-situ studies due to uncertainty in uncontrolled environments.

12.2.5.1. In-Situ Evaluation Challenges

In-situ evaluations are important since participants may behave differently in a controlled laboratory versus a domestic environment, as observed in Berry et al. (2009). The authors mention that users accepted suboptimal recommendations under laboratory conditions, and consequently, a meeting scheduling agent was improperly trained. Thus, deploying adaptive systems to end-users "in the wild" allows for testing the adaptation process under real-life conditions.

The evaluation of adaptation in domestic environments involves additional challenges. Evaluating an adaptive system per se is not trivial because it requires observation over an extended time. Since the number of samples collected during training limits the performance of RL, the amount of feedback collected during evaluation should be as large as possible. Some experiments in the literature conduct studies with several weeks of duration to get insights on long-term impacts, e.g., in the context of education. The evaluation can also address the novelty effect over a longer time as a positive side effect. Initially, people are enthusiastic about new technology, which decreases over time when they get used to it. This bias impacts their feedback and user experience in the beginning.

Another issue is privacy. When relying on human feedback in terms of social signals (see section 12.3), corresponding sensors and SSP techniques are required. Participants must agree to be monitored by the machine with cameras, microphones, or other sensors in their domestic environment. As outlined in section 12.3.1.3, there are different potential external influences in the user's environment impacting this sensing and interpretation process.

In general, there are also substantial technical challenges when conducting in-situ studies. Since the system needs to be completely autonomous, there is no possibility to intervene, such as in a controlled environment in the lab. Thus, error handling is an important aspect, which becomes increasingly important when several components need to work together. There is no option to make sure that it is working correctly or to control the robot remotely if there is no internet connection.

12.2.5.2. Wizard-of-Oz Studies

In WoZ experiments, the robot is not acting completely autonomously, but (a part of) it is remote-controlled by a human operator (the *wizard*). The test person is not informed about this manipulation but has the illusion of interacting with an autonomous robot. WoZ setups allow for conducting complex HRI experiments even if the robot is not autonomous, e.g., because processing the user's input is too complex. Moreover, they allow for rapid prototyping of behaviors and testing the effect of behavior variations without needing to implement them. For example, the majority of HRI studies about robot personality uses WoZ setups or hybrid systems where only part of the robot's behavior is autonomous (Robert et al., 2019).

Furthermore, WoZ experiments have another purpose in the context of RL and HRI: they address the *cold-start* problem, i.e., the problem of having to learn from scratch without any previous knowledge. An untrained RL agent with an empty policy needs to learn exclusively based on random exploration – trial and error. In contrast, the human wizard controls the agent's action selection in a WoZ experiment. Thus, expert human knowledge and intuition train the learning agent, which can be refined in autonomous interactions without a wizard. The state and reward calculation is typically automated and monitored by the wizard, who picks only the action. A potential limitation of this human guidance approach is the number of states visited. The more states visited during the WoZ study, the better the pre-trained agent will perform autonomously. Special care must be taken to ensure that the wizard behaves consistently for all participants. Unlike an autonomous application, human operators are subject to fatigue and distractions.

12.3. Socially-Aware Reinforcement Learning

Pentland (2005) introduces the term *socially-aware computing*. The author describes computers as "socially ignorant" and that this contradicts human life and communication. However, he points out that by "building machines that understand social signaling and social context, technologists can dramatically improve collective decision-making". According to Pentland, two aspects are of central importance for addressing the machine's social ignorance: *quantifying social context* and *teaching successful social behavior*.

Thus, researchers have started creating *socially-aware* interfaces in HCI and HRI over the years (Schiller et al., 2019). In general, *social awareness* is the "spontaneous understanding of social situations that does not require attention or reasoning" (Vinciarelli, Pantic, and Bourlard, 2009). As outlined in section 3.4, SSP techniques process human social signals during the interaction. The inclusion of such data in the machine's decision-making is essential for making the machine aware of the user and their current affective

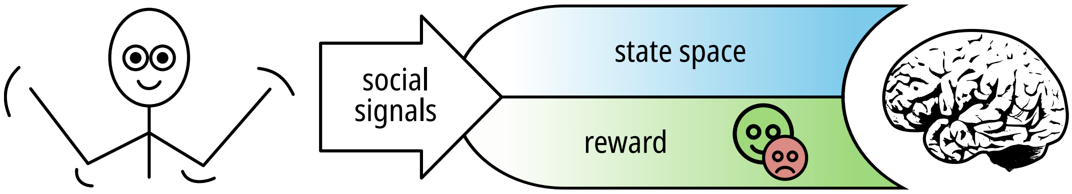


Figure 12.5.: Human social signals can be used in two ways.

state or intentions. By employing SSP techniques, the machine can infer interaction dynamics (see section 12.3.4) implicitly and automatically without additional explicit user feedback.

Schiller et al. (2019) point out that the implementation of socially-aware interfaces requires three building blocks:

1. social perception,
2. socially-aware behavior synthesis, and
3. learning socially-aware behaviors.

As outlined in section 5.1, many social robots have typical sensors required for implementing the social perception with SSP techniques, as well as actuators for synthesizing socially-aware behaviors (see Part III). One approach for implementing *learning* of socially-aware behaviors is RL. The specific opportunities and challenges in the context of socially-aware RL are detailed below.

12.3.1. Integration of Human Social Signals

A socially-aware RL process includes human social signals in its model. There are two non-exclusive options for including the signals in the RL loop (Figure 12.5 illustrates both):

1. inclusion as feature(s) in the *state space*, and
2. shaping the *reward signal*.

12.3.1.1. State Space

The learning agent learns to react to specific user states by including social signals in the state space. For example, the robot could learn its optimal behavior for different user engagement (e.g., using a stronger voice when the user is disengaged and a normal voice when the user is engaged). There is a *dependency* between the user's state (current engagement) and the robot's behaviors (volume) because the robot aims to learn how to behave *given the user's current state*. Then, the corresponding social signals (here: user engagement) must[1] be included in the state space to make the robot aware of the user's state and state changes. Chapter 14 uses this approach.

[1]One could also think about not including engagement in the state space, but rewarding the robot according to changes in user engagement (e.g., a positive reward for increasing engagement and

12.3.1.2. Reward Signal

The second option is to include human social signals in the RL reward signal. For example, shaping rewards (see section 6.2) are used in parallel to a task-based reward. They guide the agent towards the goal, relying not only on task-based measurements, such as user performance but also considering interaction dynamics, such as user engagement. A positive or negative reward may be calculated based on the temporal development of such dynamics. For example, high user engagement is desirable and preferred over low user engagement. By estimating user engagement e_t continuously over times $t, t+1, t+2, \ldots$ and calculating the difference of user engagement $\Delta e = e_t - e_{t-1}$ after execution of a robot's action Δe encodes the increase or decrease of engagement and thus can be used as a (shaping) reward.

There will be no more positive feedback when reaching the maximum e value. Similarly, there will be no more negative feedback when reaching the minimum e value. Nevertheless, the agent can learn in these situations since a decrease at the maximum value results in negative feedback (which the agent aims to avoid), and an increase at the minimum value results in positive feedback (which the agent aims to achieve). Chapter 14 uses this approach.

Including human social signals and related interaction dynamics in the reward signal is also one option for addressing the problem of *sparse rewards*. This problem exists if a positive or negative reward is delivered only sparingly (for example, when reaching a terminal state in case of success or failure) with no or few non-zero rewards along the way (Sutton and Barto, 2018). As a result, the agent appears to wander aimlessly for long periods of time because the agent cannot detect whether it makes progress towards the goal (Sutton and Barto, 2018). With social signals and interaction dynamics being a continuous source of feedback, they may guide the agent towards the goal. They are also of particular interest for non-episodic tasks without terminal states since non-zero rewards must be delivered during the open-ended interaction.

12.3.1.3. Noise

Including human social signals in the RL loop has its challenges. Different kinds of noise occur in this process (see Figure 12.6 and Figure 12.9):

- The human is a nondeterministic environment: their reactions do not need to be correlated with the actions the robot executes. Human feedback can vary from time to time, e.g., due to external influences (see also section 12.4.2).
- The sensing hardware itself is subject to physical restrictions that limit the signals which can be perceived (e.g., camera field of view or resolution, interferences, such as backlight, loud noise).

a negative reward for decreasing engagement, see next section). While this might work in some cases, there is an important restriction: the robot cannot learn a dependency between *current* user engagement and optimal action. It might result in suboptimal behavior because the robot cannot distinguish whether the user is engaged *in this moment* or not. Thus, it can only learn whether the strong or normal voice is good or bad “in average”.

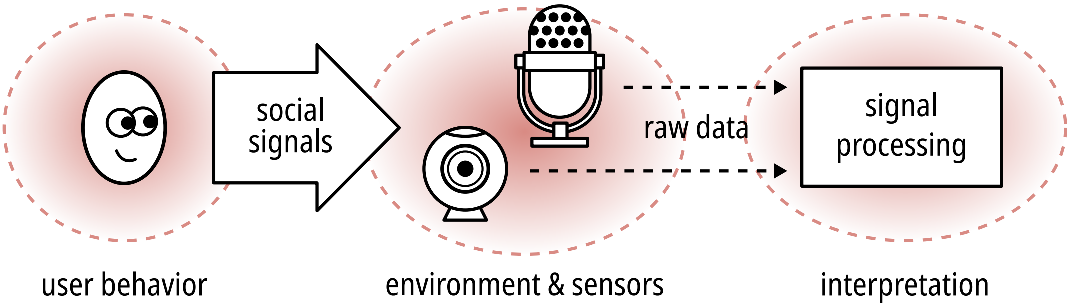


Figure 12.6.: Different types of noise and nondeterminism occur when sensing, processing, and interpreting human social signals.

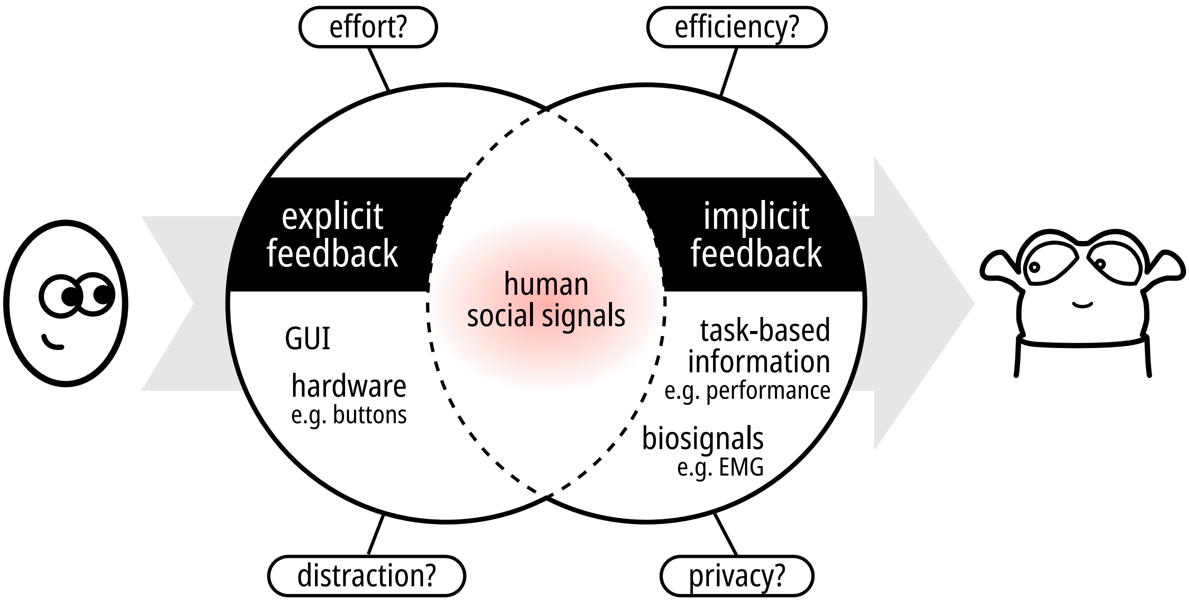


Figure 12.7.: Human social signals can be used to realize both explicit and implicit feedback.

- When processing social signals, the interpretation of the raw data often relies on machine learning itself, and the result can only be an approximation of the actual user's behaviors. Consequently, the data received for learning are noisy and allow concluding the user's intentions, needs, or preferences only to a certain degree.
- The user's reaction to the robot's actions and behavior may vary from time to time as preferences may change, too (see also section 12.4.1).

12.3.2. Explicit vs. Implicit Feedback

Feedback for adaptation can be provided either explicitly or implicitly (see section 6.3). The user provides explicit feedback often via traditional input, such as a keyboard, mouse, or touch. The system can automatically derive implicit feedback from task-based information, such as user performance measurements and biosignals. Explicit feedback can be very expressive and informative but comes at the expense of additional effort

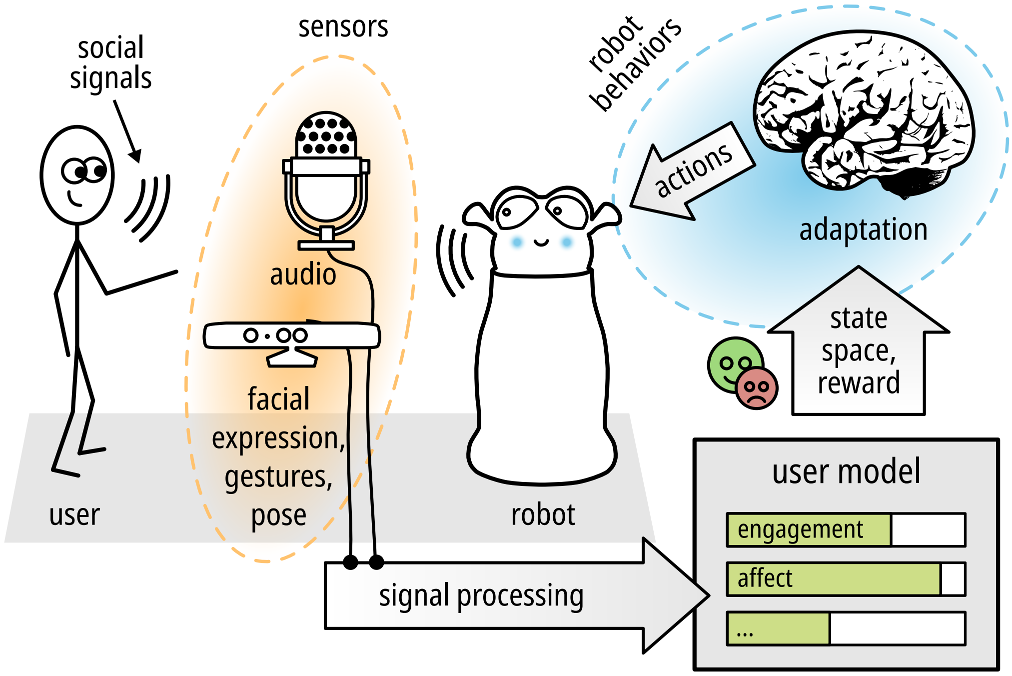


Figure 12.8.: A typical socially-aware RL process.

for the user. They have to consciously provide input, which might distract from the actual interaction and task. In contrast, implicit feedback with the user unconsciously providing feedback does not require additional effort. However, it might be less reliable, less efficient, more prone to noise, and raise privacy questions.

Human social signals can be used as explicit and implicit feedback (see Figure 12.7). For example, the user might use iconic gestures (see section 3.3.1), such as thumb signals, for agreeing (thumbs up for positive reward) or disagreeing (thumbs down for negative reward) explicitly. At the same time, user engagement, attention, or affect can be inferred from posture, prosody, and more based on SSP techniques and serve as shaping reward or feature in the state space. Verbal and non-verbal human social signals can be combined, used explicitly and implicitly to create a more complex reward signal or state space.

12.3.3. Learning Loop

Figure 12.8 illustrates a socially-aware RL process, which closes the loop between adaptation of robot behaviors and human reactions:

1. Hardware sensors sense human verbal and non-verbal signals.
2. SSP techniques are applied to interpret the raw sensor data. The inferred interaction dynamics of relevance, such as engagement or affect, serve as the dynamic user model (see section 6.1).
3. Changes in the user model influence the current state or reward signal as described in section 12.2.2 and section 12.3.
4. The execution of the next action manipulates the robot's behaviors.

12.3.4. Interaction Dynamics

Different types of information about users (see also section 6.1) are of interest in a socially-aware RL process. Section 6.2 and Table 6.1 outlined the use of such data in the literature. Typical interaction dynamics are described in the following (see also chapter 3 on SSP). Additional examples include user comfort and rapport.

Attention is the division of a limited amount of mental resources over different bodily, sensorial, cognitive, or combined activities (Kahneman, 1973; Bakker and Niemantsverdriet, 2016). In HCI and HRI user attention is often measured based on visual and neurophysiological data, including eye gaze and neural activity (Peters, Asteriadis, and Rebolledo-Mendez, 2009; Szafir and Mutlu, 2012).

Affect and emotion are related to someone's feelings, drives, mood, and more. Often, the two-dimensional valence-arousal model by Posner, Russell, and Peterson (2005) is used to differentiate positive and negative emotions (valence) and mental and physical activity (arousal). Automatic affect recognition systems use facial expression, paralinguistics, gestures, postures, and physiological responses, such as heart rate, blood volume pulse, skin conductivity, muscle activity, respiration, body temperature, pupil dilation, and electroencephalography.

Engagement is "often used synonymously to refer to a number of related concepts, such as interest, sustained attention, immersion and involvement" (Oertel et al., 2020). The thesis at hand focuses on *social* engagement, i.e., the user's engagement with a robot in contrast to *task* engagement, which focuses on the interaction task. Automatic prediction mechanisms for social engagement use a diversity of social signals, including the user's eye gaze, posture, valence, interest, and anticipatory behavior, and physiological signals, such as heart rate and electrodermal activity (Oertel et al., 2020).

The inclusion of such interaction dynamics in the RL agent's state space is what makes the learning process "socially-aware": the agent gets the ability to perceive changes in these dynamics and learns to react accordingly.

12.4. Algorithmic Considerations

Using RL for personalization and including human social signals in the RL loop involves several challenges. These challenges arise from the human being part of the RL agent's environment, but at the same time having limited ability to sense information about the user. The following sections present essential aspects to remember when including human social signals and human feedback in a RL process.

12.4.1. User Preferences and (Non)Stationary Problems

User and context information serves as input to the adaptation process. Such short-term and long-term features are listed in section 6.1.3.2. The user's individual preferences

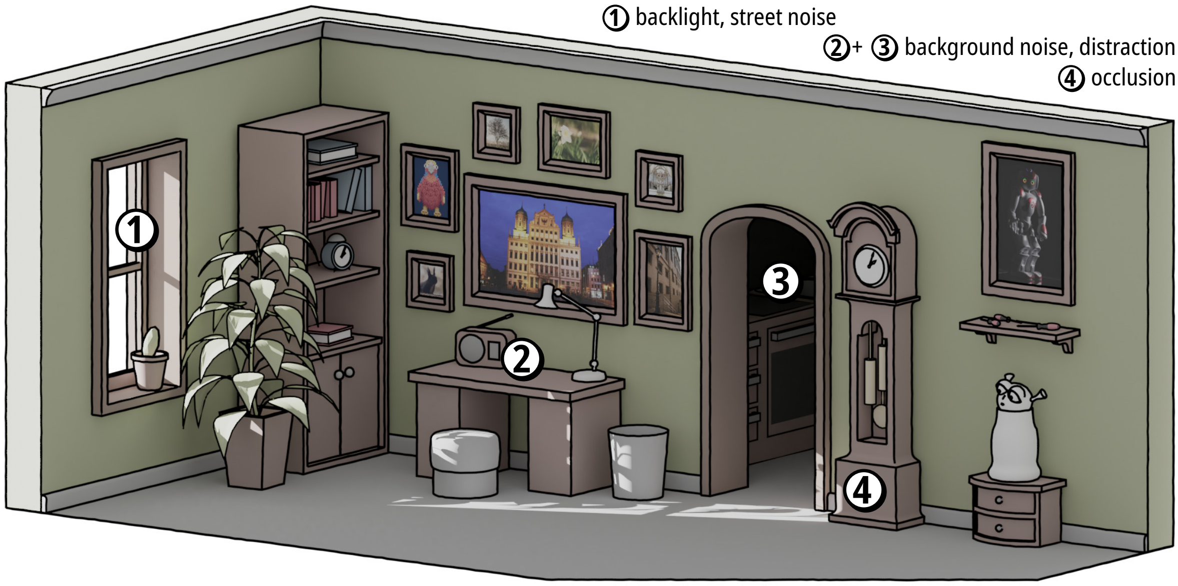


Figure 12.9.: Different kinds of noise in a domestic environment.

thus may be influenced by a range of diverse personal attitudes. Some of them (typically related to long-term features) will hardly change as they might be deeply anchored in a person's social environment, culture, education, upbringing, or personal experiences. Others (typically related to short-term features) might change in a shorter time.

From an algorithmic perspective, the potential change of user preferences is of relevance in terms of the (non-)stationarity of the underlying problem (see section 2.4.1). When analyzing tasks, modeling problems, and selecting corresponding algorithms, (non-)stationarity is an important property. Suppose fixed user preferences over the whole interaction. In that case, this might be interpreted as real values q_*, which do not change over time, which section 2.4.1 defined as a stationary problem. Otherwise, the learning algorithm must be able to cope with nonstationary problems.

12.4.2. User Behaviors, Distractions and (Non)Determinism

Even when facing a stationary problem, user behaviors and feedback might nevertheless not always be consistent due to different reasons, especially in uncontrolled environments, such as at home:

1. Users might get distracted due to external influences, such as radio news, other people in the room, street noise, and much more (see Figure 12.9). It might result in unintentional reactions, such as laughing because somebody else is making a funny comment while the user interacts with the robot.

2. The user's current mood and emotion might bias the expressed reactions towards the robot. In one situation, a user might laugh about a funny joke presented by the robot. However, in another situation, they might not laugh due to a disappointing message received moments ago, even if the joke corresponds to their humor preferences.

3. The user might try to adapt to the adaptation process itself to compensate for the robot's change of behaviors (see also section 16.4).

An extensive amount of sensitive and context information is necessary for a social robot to recognize that the user's reactions are not caused by the robot's behaviors, which is beyond the scope of this thesis. Instead, from an algorithmic point of view, inconsistent user feedback is interpreted as a result of a nondeterministic environment (see also section 2.4.2).

12.5. Conclusion

This chapter presented a structured overview and a conceptual framework for modeling, simulating, and evaluating RL experiments for social robot adaptation in HRI. Stakeholders in this context are the robot (which expresses the adapted behaviors), the user (who expresses explicit or implicit feedback), and the system designer (who is responsible for setting up and evaluating the process). The RL framework, introduced in chapter 2, allows for real-time adaptation during the interaction and can deal with uncertainty.

Apart from the specification of the RL problem with state space, action space, reward, and learning algorithm (all of which are inter-dependent), the system designer faces a set of challenges when simulating and evaluating user-adaptive interaction in HRI. During simulation, the user's reactions and feedback can be emulated only to a certain degree; during human evaluation, different types of noise occur, which bias human feedback, sensed data, and ultimately the resulting policy. In addition, in-situ studies in the participants' domestic environments provide the opportunity to get more reliable results but also face additional challenges, such as the participants' privacy.

Two central contributions of this chapter are the *Adaptation Triad* and the description of *socially-aware* RL processes. The former provides a graphical tool for brainstorming and breaking down the RL problem into three categories: the user, the robot, and the task, listing their contributions and influence within the RL problem. The latter gives an overview of the inclusion of human social signals and related interaction dynamics, such as engagement, in a RL process for user-adaptive interaction. This data allows for reacting timely to changes regarding the user's state and shaping the reward signal, thus also addressing the problem of sparse rewards.

This chapter serves as a blueprint for the following experiments, which implement, simulate, and evaluate adaptation processes based on explicit and implicit human feedback. The robot's behaviors rely on the generation approaches from Part III while the design of the RL agents follows the considerations listed in this chapter.

13. Explicit Feedback

Chapter 12 presented a structured overview and conceptual framework for implementing robot behavior adaptation with RL. The current and following chapters build upon this approach: they put it into practice while focusing on the non-functional adaptation of social robot behaviors. The chapters' difference is in the RL model. The chapter at hand uses a stateless environment and explicit feedback for approximating user preferences; the next chapter includes human social signals in stateful RL processes to react to changes in user state and relies on implicit feedback. Both chapters use the generation approaches from Part III for expressing robot behaviors. In combination, the presented experiments address the identified research gaps about the real-time non-functional adaptation of robot personality, persona, politeness, and humor.

The literature from section 5.2.3 and section 5.3 indicated different liking for different robot behaviors based on user surveys. Motivated by these insights, the following sections present a real-time adaptation approach for a domestic robotic companion's expressed politeness and persona. While chapter 8 already explored the corresponding generation of the robot's verbal utterances and function of the evaluation prototype, the focus is now on the adaptation process design, its simulation, and evaluation. It uses a stateless model but is modeled as an associative search with different actions in three application contexts. The agent receives explicit feedback via buttons on a hardware control panel and interprets it as a positive or negative reward signal. After initial simulations of the user's feedback, the adaptation approach is evaluated in the wild, and the results of the in-situ study are discussed.

The concept and implementation were part of the works presented and reviewed in Ritschel et al. (2019d) and Ritschel et al. (2019c). The contents of this chapter expand these publications.

13.1. Experiment: Approximating Politeness and Persona Preferences

This section introduces an RL approach, which combines explicit human feedback with stationary problem modeling for approximating user preferences about the robot's expressed politeness and persona. It is implemented based on the robotic elderly companion introduced in chapter 8. The domestic robot with assistive and entertainment applications serves as the technological basis and evaluation context. The following adaptation approach is simulated in section 13.2 and evaluated in the domestic environment of senior study participants in section 13.3.

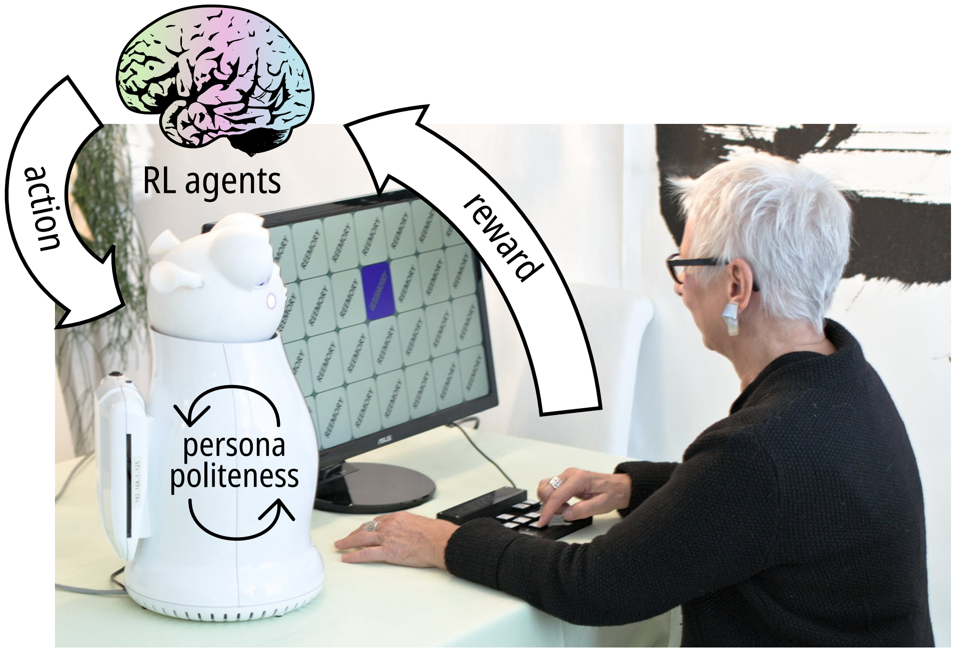


Figure 13.1.: Overview of the interaction scenario and RL feedback loop.

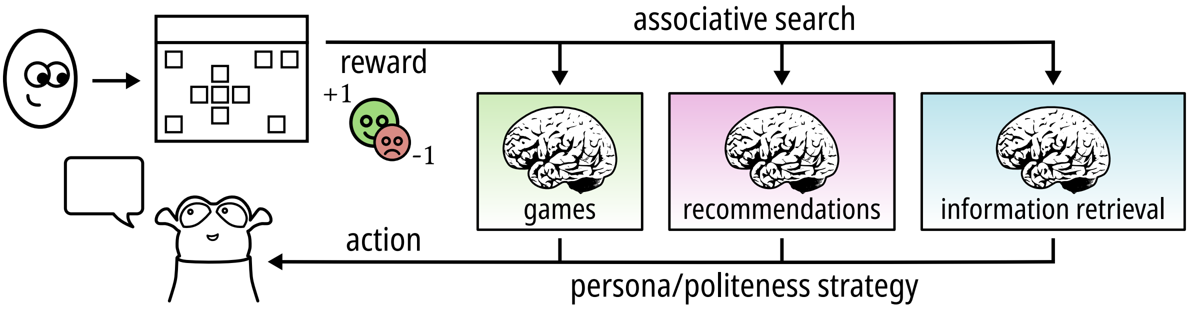


Figure 13.2.: Adaptation is realized with associative search, involving one k-armed bandit problem for each application category.

13.1.1. Overview

Chapter 8 presented a domestic robotic companion, which communicates politeness and persona. This section presents an RL approach to personalize the social robot's verbal behaviors autonomously to the individual user. In contrast to the literature from section 5.2.3 and section 5.3 the approach at hand approximates user preferences and optimizes action selection during the interaction.

Figure 13.1 illustrates the interaction scenario. The robot presents information and comments on what happens on the screen in different application contexts. Adaptation uses associative search: three RL agents are split up into three application contexts, each addressing one of politeness or persona preferences (see section 13.1.2). The user explicitly gives the reward to the agent associated with the currently active application.

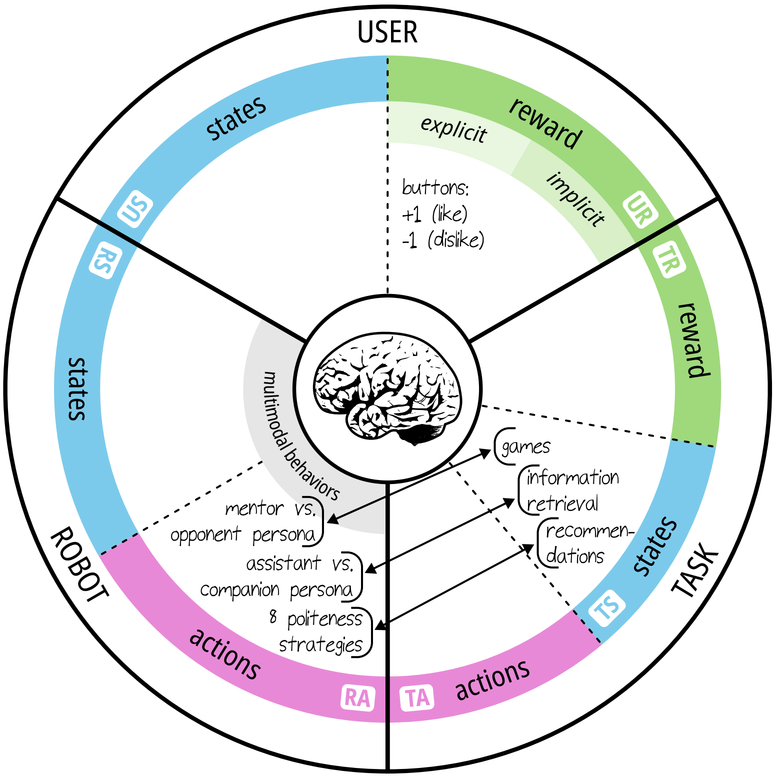


Figure 13.3.: General overview of the RL model.

13.1.2. Adaptation Process

Figure 13.2 illustrates the overall adaptation approach. RL is realized with an associative search (see section 2.6.1), which uses three distinct k-armed bandit problems for the games, recommendations and information retrieval context (see also Figure 8.2). The games and information retrieval agent learn about mentor vs. opponent and companion vs. assistant robot persona, respectively; the recommendations agent approximates the user's preferences about the robot's politeness. Personas and politeness were not combined in a single agent by intention to not mix different factors during evaluation.

Politeness and persona preferences are assumed to be related to long-term user information, such as preferences, personality, culture, and gender (see section 6.1.3.2). Thus, they are expected to be fixed in the long run. As a result, the learning tasks are modeled as stationary problems with fixed q_* values (see also section 12.4.1), without prejudice to nondeterminism due to variations in the user's feedback (see also section 12.4.2). Since there is no final state in this problem, it is a continuing learning process, which runs until it is stopped manually (see section 2.4.3).

The first task for setting up the adaptation process is to define the RL model as described in chapter 12. Figure 13.3 gives an overview of the identified user, robot, and task features contributing to the RL model. The figure is a slightly modified version of Figure 12.4: since the experiment at hand is stateless and implemented as an associative search, there is no state space but a context for independent agents. *State* areas were renamed to *context* for illustrating the different contexts of each k-armed bandit. The different areas are described in the following.

13.1.2.1. Action Space

The actions of the three learning agents correspond to the personas and politeness strategies as explained in section 8.2 and section 8.3. Two actions make the agent learn about the persona in the context of games: mentor (ME) and opponent (OP) persona.

$$\mathcal{A}_{\text{game}} = \{\text{ME}, \text{OP}\}$$

Similar applies to the agent used in the context of information retrieval, resulting in two actions: assistant (AS) and companion (CO) persona.

$$\mathcal{A}_{\text{info}} = \{\text{AS}, \text{CO}\}$$

For recommendations, the actions correspond to the eight politeness strategies: direct command (DC), indirect suggestion (IS), request (RE), question (QU), socratic hint (SH), robot's goal (RG), user's goal (UG), and shared goal (SG).

$$\mathcal{A}_{\text{reco}} = \{\text{DC}, \text{IS}, \text{RE}, \text{QU}, \text{SH}, \text{SG}, \text{RG}, \text{UG}\}$$

13.1.2.2. Reward

The user provides explicit feedback via buttons on the control panel. As outlined in section 8.1.2 and Figure 8.4, the control panel includes buttons for giving positive and negative feedback about the robot's verbal behaviors, which are interpreted as RL reward. The system informs the user when to give positive or negative feedback at each action execution by lighting up the feedback buttons. Afterward, the user has 30 seconds to rate the robot's presented utterance.

The right button (*thumbs up*) maps to a positive reward (+1), and the left button (*thumbs down*) maps to a negative reward (−1). There is no "neutral" value in between. If the user is undecided, they can ignore the assessment. In this case, the learning agent does not get a reward signal and does not learn, resulting in no change to the estimated Q values. It forces users to express a clear signal or tendency whether they like or dislike the robot's behavior. Pressing one of the buttons sends the feedback to the adaptation process and turns off the lights. In case of no feedback, the lights automatically switch off after 30 seconds. Afterward, the user cannot give feedback anymore.

Both positive and negative feedback is essential for estimating the actions' values. For example, if users would give only positive, neutral, or negative rewards for all actions, this would result in all the same Q values.

$$R_t = \begin{cases} +1, & \text{for pressing the } \textit{thumbs up} \text{ button} \\ -1, & \text{for pressing the } \textit{thumbs down} \text{ button} \end{cases}$$

13.1.2.3. Algorithm

In combination, the three k-armed bandits implement the associative search. In each application context, the associated learning agent receives the user's feedback and calculates each action's Q value, independent of the other agents. These estimate the real values q_*, which represent the human's individual preferences, i.e., liking of the corresponding personas or politeness strategies, which are unknown to the agent.

The algorithm for stationary problems from section 2.6.1 estimates each action's Q value by computing the average iteratively:

$$Q(A_t) \leftarrow Q(A_t) + \frac{1}{N(A_t)} [R_{t+1} - Q(A_t)]$$

The agents use UCB action selection (see section 2.3.3) with $c = 1$ (see next section), which ensures a prioritized exploration of actions based on the associated uncertainty about their estimated values.

13.2. Simulation

The simulation focuses on politeness strategies with eight actions. A simplified simulated user replaces the human and acts based on a predefined behavior. Since different kinds of noise might bias human feedback in the real, domestic environment in an in-situ study (see also section 12.3.1.3, section 12.4.1 and section 12.4.2) the simulation uses randomized user feedback. Randomized user feedback is interpreted as a nondeterministic environment since the user is part of the environment.

Since there is no notion of state for the bandit problem at hand, user feedback is the only aspect that needs to be automated for the simulation. The reward is the only information the user provides in the real-world setting. Thus, the reward calculation is the task of the simulated user.

13.2.1. Simulated User

The simulation replaces the human with a rule-based logic. Each simulated user is represented by eight randomly initialized values, one for each of the actions in $\mathcal{A}_{\text{reco}}$:

$$p_{DC}, p_{IS}, p_{RE}, p_{QU}, p_{SH}, p_{SG}, p_{RG}, p_{UG} \in [0; 1]$$

They represent probabilities and determine how likely the user will give positive feedback when the agent executes the corresponding action. For example, $p_{DC} = 0.6$ means that the simulated user gives positive feedback in 60 % of the cases when the agent selects action DC. See algorithm 2 for pseudocode.

In real interactions, the agent ignores the absence of human feedback for an executed action (estimated values do not change). There is no need to consider this case for the simulation, which would also skip this learning step.

The probabilities also determine the best action for the agent. If the agent learns correctly, it should identify the action with the highest p value as the greedy action over time since it receives the most positive reward. Since all p values have random values, no action will likely result in positive feedback exclusively. Conversely, all actions likely get negative feedback with a certain probability, even if the action is greedy. This noise simulates the agent's nondeterministic environment, i.e., biased human feedback.

Algorithm 2: Reward calculation for the simulated user.

Input: array of action probabilities $p \in [0; 1)$, executed action $a \in [0; 7]$
Output: reward value

$r \longleftarrow$ a random float in $[0; 1)$
if $r < p[a]$ **then** // *with* $p[a]$ *probability*
  $R \longleftarrow 1.0$ // *positive reward*
else
  $R \longleftarrow -1.0$ // *negative reward*
return R

13.2.2. Results

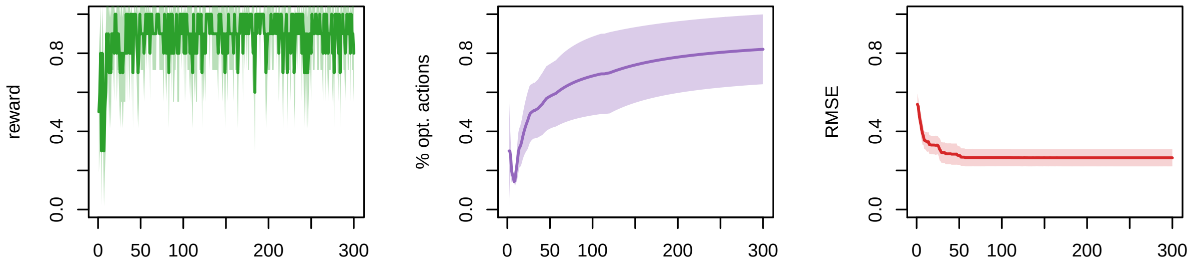
The simulation uses a set of 10 artificial users with different, randomly initialized preferences (p values), similar to the expected number of participants in the in-situ study. Each simulated user interacts for a fixed number of 300 actions, corresponding to 300 robot utterances, each followed by positive or negative feedback from the human. Since the learning task is non-episodic but continuing, the simulation logic terminates the learning process after these 300 learning steps. It reinitializes the p values to new random values for the next run. Thus, the learning task changes and the agent does not take along previous knowledge, as would be the case for episodic problems. Each simulated agent starts from scratch and needs to estimate the preferences p within 300 steps.

As described in section 12.2.4, introspective measurements are used in simulations to evaluate the self-motivated goal of the adaptation process. The measures include the reward (see section 2.7.1), the percentage of optimal actions (see section 2.7.2) and the RMSE (see section 2.7.3). Each measure averages all ten simulated users. Measurements were repeated for different values of c for the UCB action selection to illustrate the resulting different learning behavior and performance. Moreover, the final c value for the human user study was identified by analyzing the simulation results and selecting a reasonable compromise (see below). All plots use a 95 % confidence interval (CI) band to illustrate the similarity of the results between all simulated users.

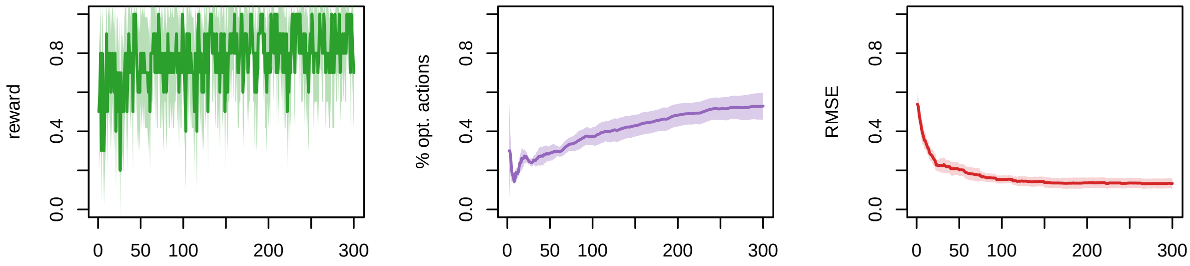
Figure 13.4 illustrates the simulation results. For better readability, the reward plot is scaled to the interval $[0; 1]$ instead of $[-1; 1]$ (0 becomes 0.5). Since the number of simulated users is relatively small, the resulting plots of the averaged data appear more or less jagged in the Figures 13.4(a), 13.4(b) and 13.4(c). Thus, Figure 13.4(d) adds another plot averaging over 1000 simulated users. It smooths out the curves and gives a better impression of the agent's performance in general, which is especially visible in the average reward.

13.2.2.1. Reward

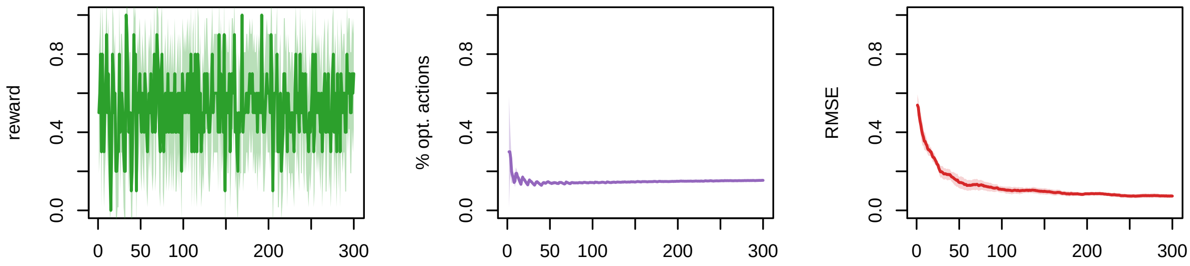
The reward is calculated based on the simulated users' binary feedback. At the beginning of each run, the algorithm selects every action once before continuing with UCB action selection. The intention is to get at least one sample for each action. As a consequence, the agent's behavior is not greedy in the first steps of the simulation, which is also visible in the averaged rewards. Afterward, the reward tends to increase over time.



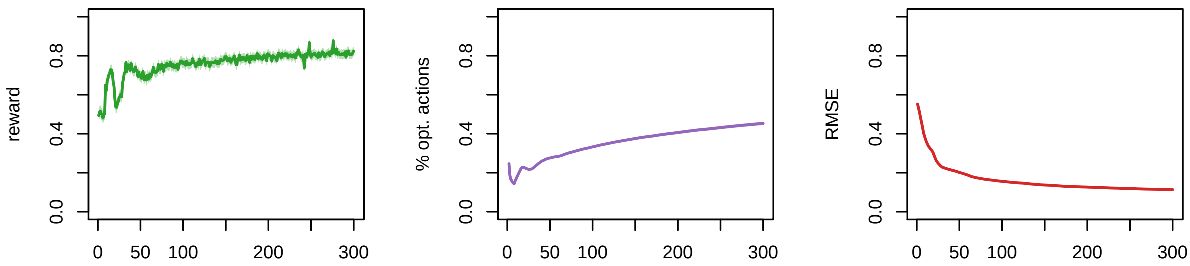
(a) $c = 0$, averaged over 10 simulated users



(b) $c = 1$, averaged over 10 simulated users



(c) $c = 10$, averaged over 10 simulated users



(d) $c = 1$, averaged over 1000 simulated users

Figure 13.4.: The agent's simulation performance over time with a 95 % CI band. Figure 13.4(b) shows typical UCB spikes at the beginning.

There is an obvious connection between the c parameter and the average rewards since c controls the amount of exploration. The more exploration, the smaller the received rewards due to suboptimal action selection, which is especially visible when comparing the averaged rewards in Figure 13.4(a) or 13.4(b) with Figure 13.4(c).

One might wonder why the average reward does not approach 1.0. The main reason is the agent's exploration. Moreover, the greedy action probably does not have a p value of 1.0. Thus, user feedback will *always* be noisy since it includes more or less negative feedback, even if the greedy action is selected, resulting in a nondeterministic environment. Negative feedback occurs even more often if the agent selects a suboptimal action during exploration.

The plots based on averages from 10 simulated users are too noisy to indicate a general trend. Thus, Figure 13.4(d) makes the performance gain visible: in the long run, rewards increase as the agent identifies the greedy action over time.

13.2.2.2. Percentage of Optimal Actions

The percentage of optimal actions confirms the findings from the reward plots. The parameter c controls the degree of exploitation and exploration and, thus, the final percentage of optimal actions. The more exploration, the more often a suboptimal action is selected, which leads to a lower final percentage of optimal actions. The percentage of optimal actions is relatively high for $c = 0$ because the agent primarily acts greedy. Even though this results in many positive rewards, it adversely affects the accuracy of the estimated preferences (see below). With increasing exploration, the percentage of optimal actions decreases significantly, as does the reward.

13.2.2.3. RMSE

The RMSE measures the accuracy of the estimated preferences. In contrast to a human evaluation, the preferences p are known in the simulation, as its logic initializes them randomly. The RMSE decreases quickly, which implies that the estimated values become more accurate over time and that the agent successfully improves action selection.

As with the other measures, there is a clear connection between the RMSE and the parameter c. The greater the value of c, the more accurate the estimated preferences. For example, the final RMSE in for $c = 10$ is approximately 0.1, which is much smaller than for $c = 0$ or $c = 1$. A higher amount of exploration produces more samples for suboptimal actions. Therefore, their estimated values become more accurate, too, which reduces the overall RMSE. In the case of a small amount of exploration, there are many samples for the optimal action but few for the majority of suboptimal actions, which prevents the RMSE from approaching zero. The estimated preferences become more accurate with increasing c, and the RMSE settles down at about 0.25 for $c = 0$.

13.2.2.4. Compromise

The following observations derive from the simulation results:

- In the long run, the agent identifies the optimal action, and the accuracy of the estimated preferences increases over time, independently of the parameter c.

- c directly controls the degree of exploration. In turn, the exploration-exploitation dilemma greatly influences the agent's performance.
- Performance and accuracy of the estimated preferences cannot be optimized at the same time. The objective of increasing performance in terms of received rewards and percentage of optimal actions at the expense of accuracy and vice versa.
- The first 20–50 steps have the greatest impact on the estimated preferences' accuracy for the experiment at hand with eight actions.

As a consequence, $c = 1$ is used for the human evaluation (see section 13.3) as a compromise. It aims to address a reasonable accuracy of the estimated preferences and reasonable robot behavior, which presents recommendations in the user's preferred style for approximately 50 percent of all actions.

13.3. Evaluation: In-Situ Study

After optimizing the learning parameters in the simulation, an in-situ study was conducted with elderlies in their familiar surroundings. The study aimed to learn about the users' individual preferences regarding the robot's expressed persona and politeness.

13.3.1. Acquisition Challenges

Apart from general challenges in in-situ studies (see section 12.2.5.1), the evaluation of the assistive social robot involved some challenges. The acquisition of elderly participants was not easy. Users, who did not use computing technologies throughout most of their lives, sometimes were not interested in participating in the robot study. For example, people refused to let a robot into their home without even being told the robot's functions. The lower affinity with electronic devices prevalent in the elderly population (Karrer et al., 2009; Hammer et al., 2016) might explain this phenomenon.

Sometimes, people mentioned privacy concerns irrespective of their age, being afraid of what happens to their data, such as when using speech recognition. The computing resources in the wild still do not have enough power to realize speech recognition and natural language processing with consumer hardware and open source software for solving these tasks locally without sending data to external cloud services. Users also needed to be healthy enough to interact with the robot since the applications addressed seniors who wanted assistance but could interact with the system unassisted.

Most participants joined after a popular science talk by the author. The talk covered social robotics and ongoing research at the lab. One part of the talk also presented the general idea of the experiment. The author pointed out that new test persons are welcome to participate in the in-situ study. After the talk, one member of the audience offered their participation. They also informed friends about the robot study, and the information spread based on personal recommendations.

Some participants did not have internet access at home or did not permit its usage due to privacy concerns. Consequently, selected applications were deactivated for these test persons, e.g., the weather forecast and news.

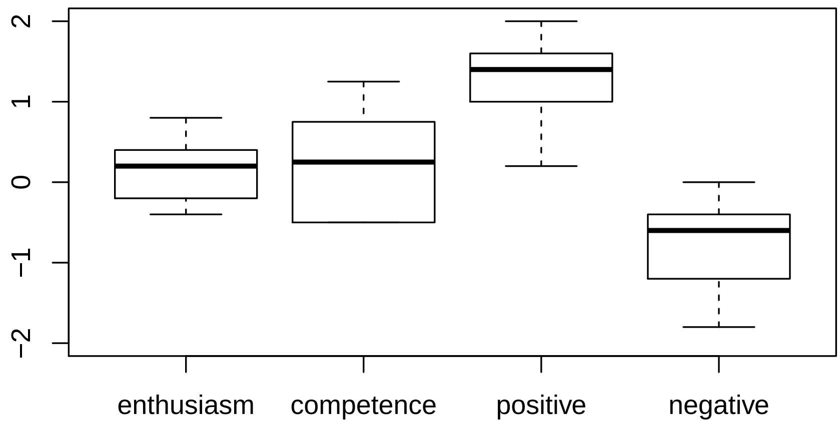


Figure 13.5.: Participants' attitudes towards electronic devices (TA-EG).

13.3.2. Participants, Apparatus, and Procedure

While the literature examined individual preferences about robot persona and politeness in the lab and WoZ studies, the in-situ study addressed the target population in their domestic environment. Nine participants (5 female, 4 male), aged 61 to 78 ($M = 68.33$, $SD = 5.59$), were recruited for an in-situ study in their domestic environment. All of them were native German speakers, and all texts presented by the robot were in German.

At the beginning of the study, the participants were asked to fill in the TA-EG questionnaire by Karrer et al. (2009) about their attitude towards electronic devices. It measures four aspects: *Enthusiasm*, *Competence*, *Positive Effects of Technology* and *Negative Effects of Technology*. Figure 13.5 illustrates the result of the questionnaire. The scale $[-2; 2]$ reflects the wording used in the questionnaire: -2 corresponds to "strongly disagree", 0 is "neutral" and 2 represents "strongly agree".

Participants scored rather high on the *positive* aspect ($M = 1.20$, $SD = 0.56$), indicating a general trust in electronic devices and belief that these can improve everyday life. These results align with the lower rating of the *negative* aspect ($M = -0.84$, $SD = 0.62$), indicating that participants attribute fewer negative consequences to electronic devices. In average, *enthusiasm* ($M = 0.16$, $SD = 0.40$) and *competence* ($M = 0.22$, $SD = 0.67$) are relatively neutral with a trend towards positive ratings. These results indicate a positive, optimistic opinion about electronic devices. Since participants did not attribute a lack of competence to themselves, they seemed confident in their ability to handle electronic devices, at least at a basic level. This phenomenon is not overly common among this demographic, possibly one reason why the test persons agreed to participate in the study.

First, the system was set up and configured according to the participant. This process included activating or deactivating applications (see section 8.5) and entering the participant's city name to provide the correct weather forecast. Apart from a two-sided manual for the hardware control panel and for starting, stopping, and interacting with the robotic companion, only a few further instructions were given.

Participants were told that they should pay particular attention to the robot's spoken language and that their feedback should be given depending on whether they liked or disliked how the robot expressed itself. They were also told to decide independently of the semantic content or factual information presented, e.g., in the context of information

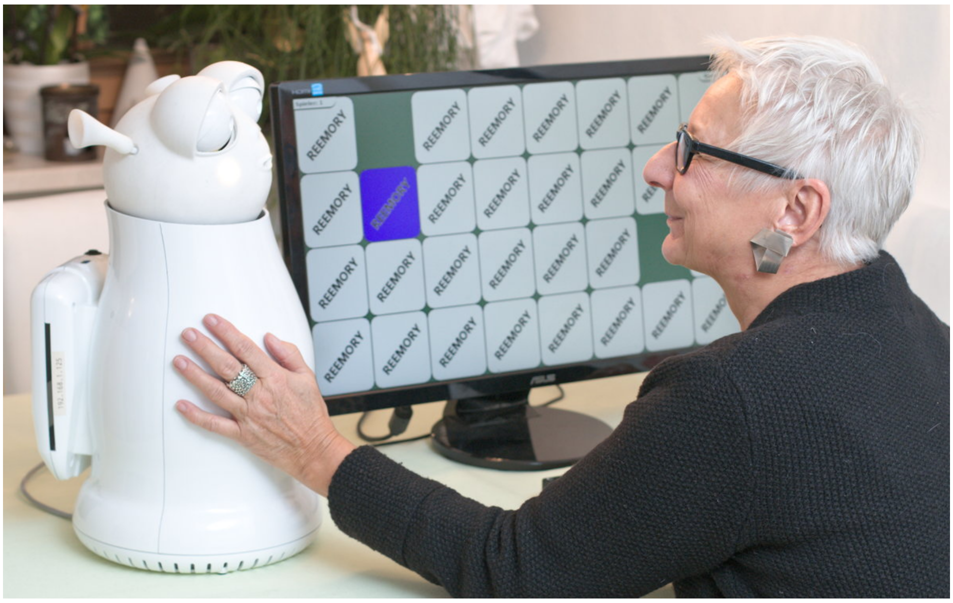


Figure 13.6.: A female test person interacts with the robotic companion.

retrieval tasks. Furthermore, they were told that giving positive and negative feedback is essential for the robot, and thus they should provide honest feedback whether they like or dislike the robot's behavior. These instructions were also part of the two written and illustrated manual pages. Participants were not told which aspects of the robot's behaviors were manipulated in detail. After providing all this information and reviewing the manual pages, the experimenter left the participant's domestic environment.

Each user interacted with the robotic companion for one week. They were free to interact with it whenever and as often as they wanted. The participants could explore the different applications at will: at any time, participants were able to pick whatever application they were interested in. During the study, the robot explained using the control panel and application when the application started for the first time. Additionally, the manual contained a labeled overview graphic with information about the control panel's keyboard assignment. At the end of the week, the experimenter revisited the participant. The participant filled out a final questionnaire addressing user experience.

13.3.3. Results

13.3.3.1. User Experience

After the study, when collecting the hardware, the participant had to fill in the final questionnaires to evaluate subjective measurements. First, the System Usability Scale (Brooke, 1996) evaluated the custom-made control panel to get an idea of potential operating errors or misunderstandings. It measures the usability of an electronic device on a scale from 0 to 100, with everything above 68 being better than average. The right side of Figure 13.7 illustrates the result. The control interface achieved good results (M = 80.00, SD = 12.93), which indicates that the participants perceived it as straightforward to use and did not experience many problems.

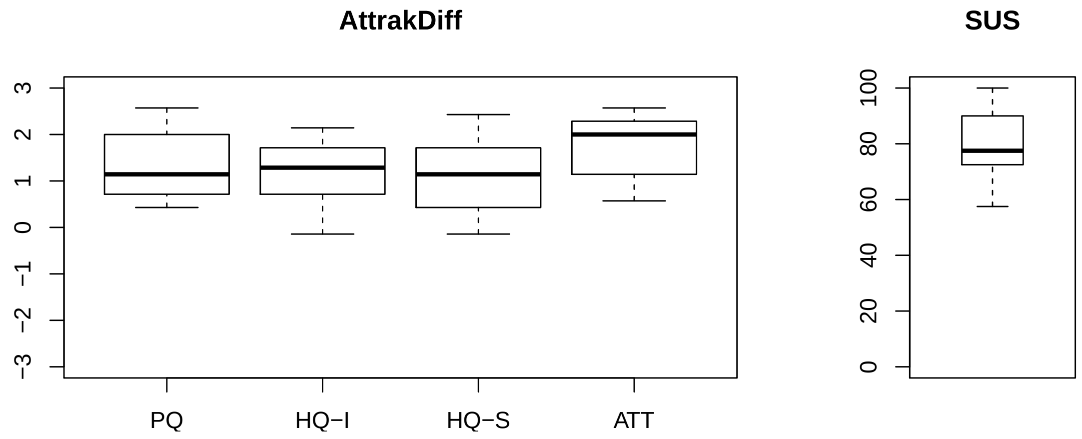


Figure 13.7.: Participants' ratings of the robotic companion (AttrakDiff) and hardware control panel (SUS).

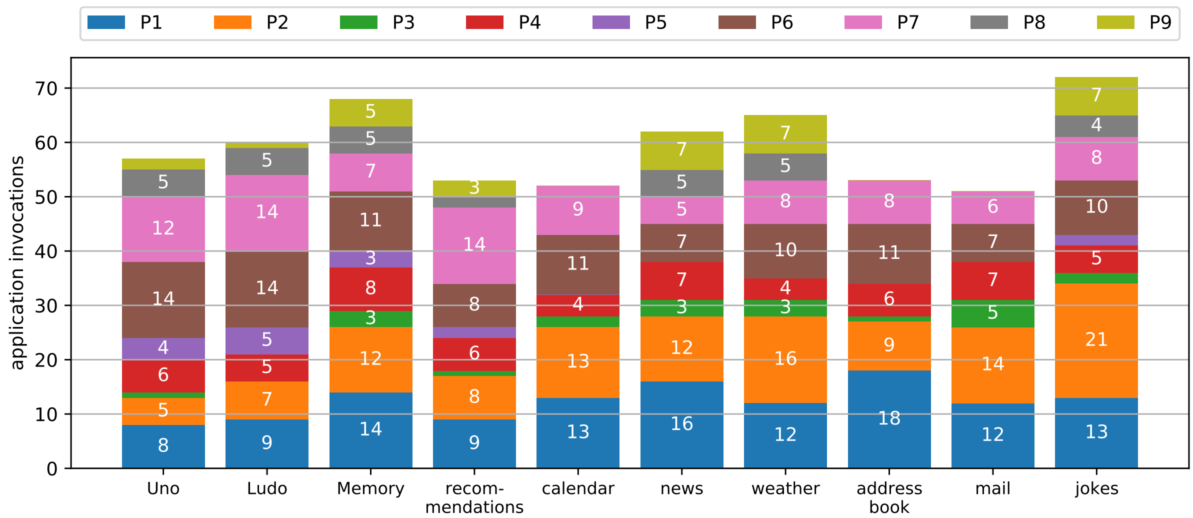


Figure 13.8.: Number of application invocations, i.e., how often each participant used each application.

Second, the AttrakDiff questionnaire by Hassenzahl, Burmester, and Koller (2003) measured the overall perception of the system. It consists of four subscales: *Pragmatic Quality* (PQ), *Hedonic Quality - Identity* (HQ-I), *Hedonic Quality - Stimulation* (HQ-S) and *Attractivity* (ATT). The results can be found on the left side of Figure 13.7 in range $[-3; 3]$. In average, all subscales were rated positively, especially with regard to ATT ($M = 1.81$, $SD = 0.72$) and PQ ($M = 1.32$, $SD = 0.83$). On the HQ-S ($M = 1.05$, $SD = 0.85$) and HQ-I ($M = 1.11$, $SD = 0.82$) subscale, the robot achieved lower scores, but still above the neutral value of zero. These results indicate that the robotic companion was perceived as an attractive product. In addition, the assigned pragmatic quality is also in line with the *positive* aspect in the TA-EG questionnaire, which revealed the users' attitude toward the positive effects of modern technology.

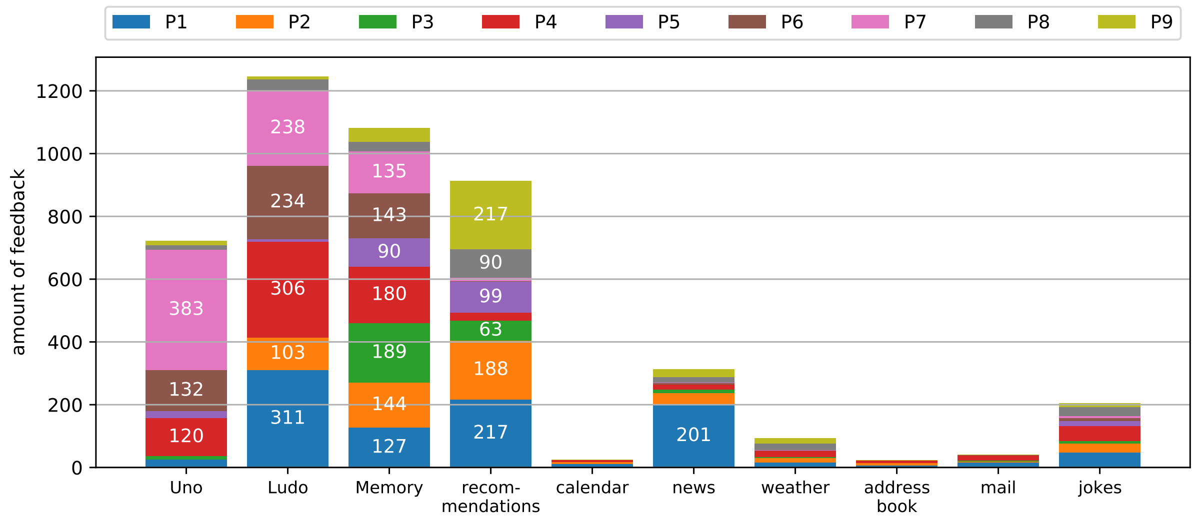


Figure 13.9.: Collected human feedback for each application.

13.3.3.2. Collected Human Feedback

Interaction measurements revealed insights into the overall adaptation process. Figure 13.8 gives an overview of all participants' use of the robot's functions. It visualizes how often each participant started each application. Five participants (no. 1, 2, 4, 6, 7) used all the robot's functions, one (no. 3) used everything except the Ludo game, and one (no. 5) used everything except for those functions that required internet access. The remaining two (no. 8 and 9) did not use the calendar, address book, and mail. The bar chart illustrates that interests in the different applications were diverse, but they tried each application at least once. Moreover, all participants used at least one application per learning agent, allowing them to provide feedback on the robot's adaptation mechanism.

While this information indicates what participants were most interested in and how often they started interacting with the robot, it does not necessarily correlate with the amount of human feedback for the adaptation process or the time spent per session.

Figure 13.9 plots the amount of feedback given by each user within each application, which is the most interesting information for the learning task. The bar graph includes both positive and negative ratings. It does not sum up the ratings' values but the number of times the buttons were pressed. Thus, the robot received the most feedback in the context of games, which provided the opportunity to give feedback after each of the robot's and user's moves. The recommender application also received a decent amount of feedback. In contrast, information retrieval tasks received fewer human ratings, which is also because these applications primarily present external content. Since the adaptation process did not control and manipulate these contents, the user often could not provide feedback to the robot. Only some of the robot's formulations (such as how it calls attention) provided the opportunity for rating, which accounts for a smaller fraction of assessable content than other applications. Thus, the amount of feedback in a certain amount of time spent on information retrieval tasks is smaller than in other contexts.

In order to compare the actual data used for adaptation, Figure 13.10 aggregates all applications which contributed to the same learning agent. It stresses that the agent learning about the mentor and opponent persona received the most feedback, followed

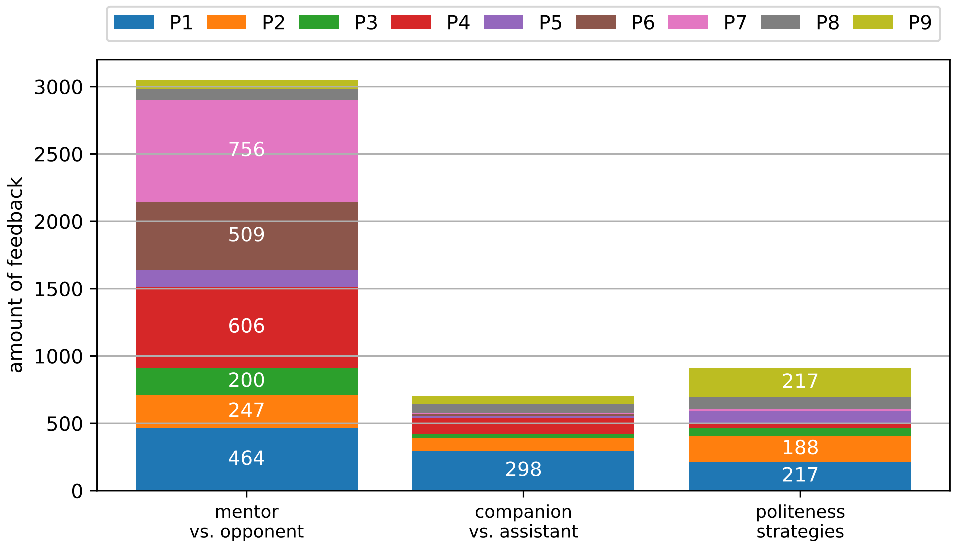


Figure 13.10.: Collected human feedback for each learning agent (corresponding applications combined).

by the agent for politeness strategies and the agent for companion vs. assistant persona. However, it does *not* follow from these graphs without more ado that the agent's learned preferences are most accurate for games and most improper for information retrieval tasks for all participants. Preferences are not aggregated but learned by each participant individually, ignoring knowledge from previous test persons. For example, participant 9 provided more feedback in the context of recommendations and less for games. Also, the amount of feedback provided is different between test persons, which becomes apparent in Figure 13.9 and Figure 13.10.

13.3.3.3. Exemplary Learning Progress

The learning agents identified individual preferences based on the collected participants' feedback, i.e., subjective measurements. As explained in section 13.1.2, the adaptation process calculates a value for each multi-armed bandit problem's actions. These values represent the estimated participant's preferences, which are approximated and refined over time, but also contribute to optimizing action selection during runtime.

Figure 13.11 illustrates this process based on real data of one participant in the context of recommendations. The top graph plots each of the eight politeness strategies' (i.e., the actions') values on the y-axis; the x-axis represents the time. Since users can stop interaction anytime, switch apps, or come back later, the system persists the samples after each step. The time axis strings together all learning steps from all recommender application invocations throughout the participant's study progress (one week). The bottom graph plots the user's feedback, which corresponds to button presses and represents positive or negative rewards. In each step, one selected action's value increases or decreases accordingly. It follows that the length of the x-axis (217 steps) corresponds to the amount of feedback given for recommendations by participant 9 as listed in Figure 13.10.

One can see clearly that value changes become smaller over time due to the stationary problem modeling and how the algorithm works. In the beginning, each reward greatly

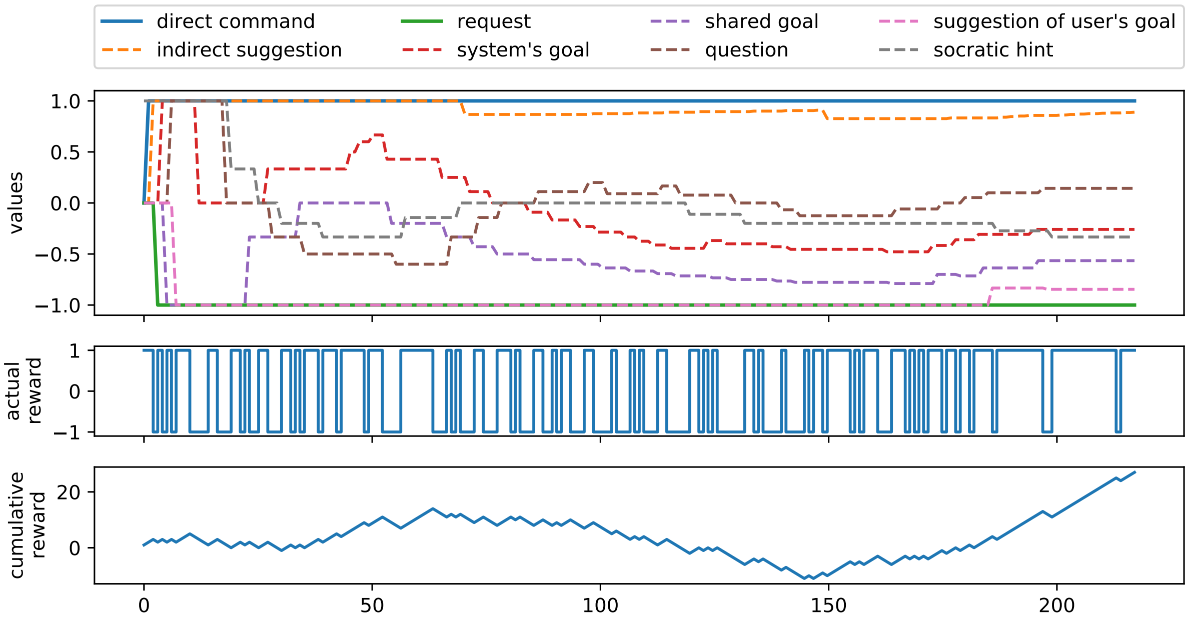


Figure 13.11.: Exemplary adaptation progress for politeness (P9).

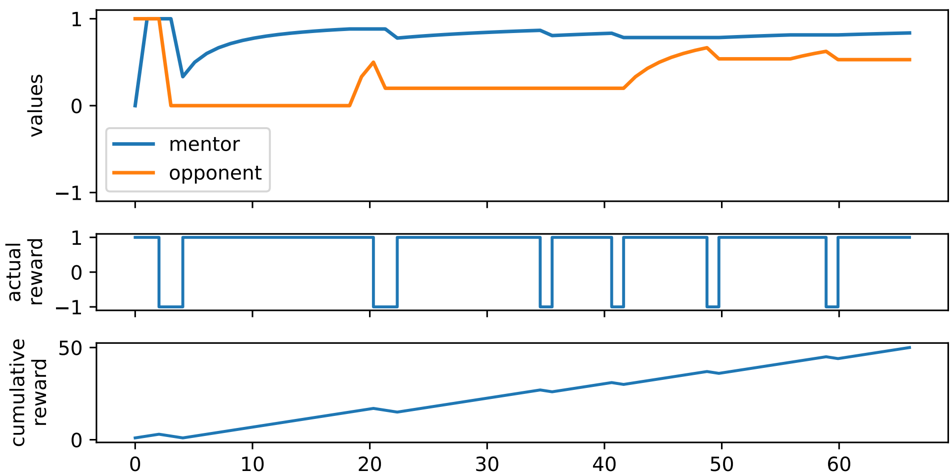


Figure 13.12.: Exemplary adaptation progress for games (P9).

influences the action's value. The more feedback provided, and the more often a specific action has been executed, the smaller the impact on the action's value. The resulting best and worst politeness strategies have a solid line; the remaining use a dashed line.

For each participant, this plot looks different. In this example, the agent identified the direct command as the best formulation, and requests performed worst for this participant. Interestingly, the participant consistently rated the best and worst actions exclusively positive or negative, which is why their values two never change. The values of the remaining actions received more varied human feedback, which is why lines cross every once in a while. Figure 13.12 illustrates similar learning progress for the agent focusing on the mentor vs. opponent persona in the context of games. It looks much simpler because the agent has two instead of eight actions, which also applies to the companion vs. assistant agent. Nonetheless, these plots illustrate the agent's adaptation progress for each user.

13.3.3.4. Learned User Preferences

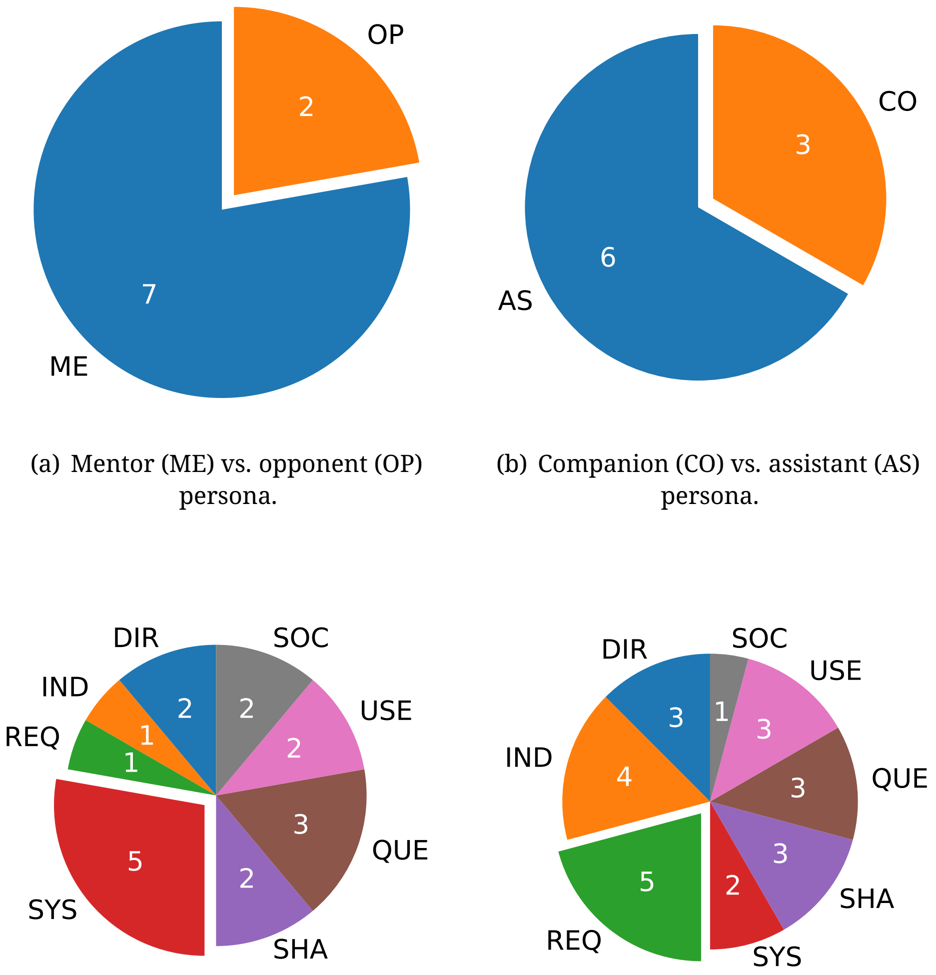
While the plots mentioned above give insight into each agent's learning progress in temporal terms and for each participant individually, the final task is to identify the resulting preferences. For this purpose, the corresponding best and worst actions (i.e., those with the greatest and smallest value) were identified for all participants. The results are as follows. In the context of games (see Figure 13.13(a)), the mentor behavior is identified as the preferred persona for seven participants, and, in turn, the estimated best persona for the remaining two users is the opponent persona. Similar applies to the information retrieval context (see Figure 13.13(b)): the assistant persona was superior six times, the companion persona three times. The corresponding worst actions result from the opposite distribution automatically since it is a binary problem with two actions.

For the eight politeness strategies, which were used in the context of recommendations, the results are listed in Figures 13.13(c) and 13.13(d). First of all, it must be noted that these numbers do not add up to the number of participants. It is not possible to identify *one* single best or worst strategy for some participants since they rated multiple strategies equally good or bad. For example, if each request or shared goal occurrence is rated consistently negative, the calculated value is −1 for both actions. Consequently, the final value of multiple actions may be −1 or +1, making the results even more interesting.

Based on the received human feedback, five participants rated the system's goal best, followed by the questions, which had maximum value for three users. The suggestion of the user's goal, shared goal, socratic hint, and direct command had maximum value in two cases. Indirect suggestions and requests are both represented once. On the other side, the worst performers include the request (five times), followed by indirect suggestions, which have the minimum value in four cases. Direct commands, questions, the suggestion of the user's goal, and formulations as shared goals occur three times each. The system's goal (two times) and socratic hint (once) appear as the worst options.

13.3.4. Discussion

Most interestingly, all politeness strategies appear as best and worst options, indicating that what is best for one user might be worst for another user and the other way round.



(a) Mentor (ME) vs. opponent (OP) persona.

(b) Companion (CO) vs. assistant (AS) persona.

(c) Best politeness strategies.

(d) Worst politeness strategies.

Figure 13.13.: Final distribution of all participants' preferences. DIR = direct command, IND = indirect suggestion, REQ = request, SYS = system's goal, SHA = shared goal, USE = suggestion of user's goal, QUE = question, SOC = socratic hint.

While the overall number of participants needs to be higher to derive generally valid findings, the results indicate that preferences can be diverse and of individual nature. For the participants at hand, one can observe the following aspects:

- There is a tendency towards the mentor and assistant persona in the context of games and information retrieval. The preference of assistant over companion persona contradicts the findings by Bartl et al. (2016) but is in line with Dautenhahn et al. (2005). The preference of mentor over opponent persona is in line with works in the context of games. For example, Altmeyer, Lessel, and Krüger (2018) observed that seniors aged 75+ primarily play to socialize, resulting in collaboration, caretaking, and avoiding competition.
- Regarding the robot's politeness, formulations as the system's goal had the best rating most of the time when compared with the other strategies.
- As the distribution of best and worst politeness strategies illustrates, there is not one single strategy that is best or worst for all users.

These results suggest that an adaptive approach for exploring a social robot's communication style and behavior is an opportunity to optimize the individual user's interaction experience. The goal is to estimate and identify the best behavior during runtime and to adapt the robot's verbal behaviors to the humans' individual preferences.

13.4. Conclusion

This chapter presented the first implementation of the conceptual framework for non-functional social robot behavior adaptation from chapter 12. The non-functional adaptation approach explores variations of politeness and persona as expressed in a domestic companion robot's verbal utterances. The contribution of this chapter is the model, implementation, simulation, and evaluation of a fully autonomous adaptation approach based on RL and explicit human feedback. Users provide feedback via button presses on a custom hardware control panel. The stateless learning agent uses associative search with three agents for different application contexts. A simulation identified a reasonable value for the parameter c of the UCB action selection mechanism. The subsequent in-situ study evaluated the fully autonomous system based on the hardware and software from chapter 8 in elderly participants' domestic environments for one week. Evaluation results indicate that there is not a single best or worst communication style for all users but that a learning approach can adapt the robot's behaviors to the users' liking.

14. Implicit Feedback

After investigating explicit feedback and associative search for stateless, but context-dependent RL in chapter 13, the chapter at hand focuses on the inclusion of human social signals in the adaptation process. As outlined in the structured conceptual framework from chapter 12, these signals and related interaction dynamics can be included in the user model, which allows the learning agent to react to changes in the user's state. Consequently, this chapter requires dynamic user models updated during the interaction.

The learning agent's goal in the following experiments is to optimize user experience by maximizing an interaction dynamic (engagement or affect), which is sensed with SSP techniques. The monitored data is used in the state space or for deriving the reward signal, thus guiding the agent toward its goal. There is no task-based reward in the presented experiments, i.e., human social signals are the only source of reward.

Motivated by the diversity of findings concerning user preferences on robot personality (see section 5.2.2), the first experiment focuses on the adaptation of a robot's expression of the introversion-extraversion trait in a storytelling application. The NLG approach from chapter 7 generates the robot's verbal behaviors for producing utterances with varying degrees of extraversion. Temporal changes in user engagement drive the adaptation process. User engagement is also used as an interaction dynamic in a range of user-adaptive HRI experiments in the literature (see section 6.2).

Afterward, human affect is used to adapt a robot's presented humor. The experiments are motivated by the reported positive effects of humor in HRI but rare attempts for adaptation of robot humor in the literature (see section 5.4) and much fewer experiments using RL (see section 6.2), let alone the dynamic generation of robot humor. The behavior generation techniques from chapter 9 produce multimodal the robot's behaviors. Temporal changes in human affect drive the adaptation process.

Again, chapter 12 serves as a blueprint for the model and implementation of socially-aware RL processes. They realize non-functional adaptation of the robot's behaviors, either verbal or combined verbal and non-verbal behaviors. The structure of each experiment is as follows: after a short overview of the general approach, the SSP technique is outlined first since it is a central element in the socially-aware learning processes. Afterward, details on the RL model and simulation/evaluation are provided.

14.1. Experiment: Adaptive Storytelling with Personality

This section introduces an approach to exploring human preferences regarding a social robot's extraversion-introversion trait. Motivated by the varying insights about human-robot compatibility, including similarity and complementarity attraction, as well as mixed findings (see section 5.2.2), the experiment at hand provides an approach for automatically adapting the robot's degree of extraversion to the user. Previous studies

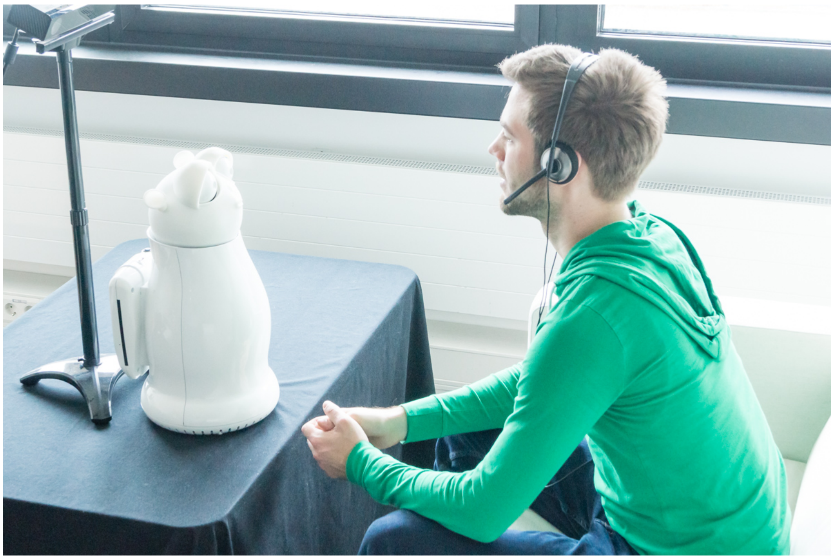


Figure 14.1.: The storytelling interaction scenario.

from the literature found out that the adaptation of agent personality can positively influence the user's liking of the robot, interaction experience, interaction dynamics (such as user engagement, see section 12.3.4), and the like (see section 5.2.2). Therefore, the socially-aware and nonstationary RL approach uses implicit feedback derived from sensed user engagement to optimize the degree of extraversion for the user's desired personality profile automatically based on the user's reactions. Section 14.2 presents the simulation of the adaptation approach.

The concept, simulation and implemented adaptation approaches were presented and reviewed in Ritschel and André (2017), Ritschel, Baur, and André (2017a), Ritschel, Baur, and André (2017b), and Ritschel (2018). The contents of this section expand these publications.

14.1.1. Overview

The following sections present the interaction scenario, SSP and RL model for adaptation of the robot's communication style. The learning agent manipulates parameters of the NLG approach from chapter 7, which, in turn, produces the robot's utterances. In contrast to the literature from section 5.2 and section 6.2 the adaptation approach controls the robot's utterances via the NLG module during runtime in order to keep interaction engaging while talking about the content of the book "Alice in Wonderland". A storytelling task was chosen as an appropriate setting for exploring the non-functional adaptation process since storytelling is one form of entertainment and fits well in domestic environments.

Figure 14.1 gives an overview of the storytelling scenario. The user and the robot sit opposite each other. The robot uses its language as the primary output medium. No GUI is used in this application, neither as input nor output modality. Instead, the interaction between the user and the root is realized via speech recognition. A quiet environment with a microphone for speech recognition, a Microsoft Kinect 2 sensor, and a powerful computer is required for SSP. The robot's language is presented with its internal TTS

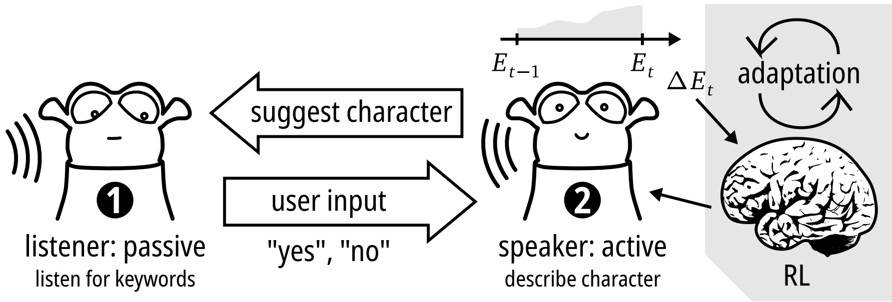


Figure 14.2.: Main interaction states from the perspective of the robot/application.

system. The resulting audio is played back with the robot's internal speaker.

14.1.2. Dialog Flow and Interaction

The general idea of the storytelling application is to provide the user with information about the book "Alice in Wonderland" by Lewis Carroll. The robot acts as the narrator, presenting the plot of different chapters and details about the main characters. Meanwhile, the user primarily acts as a listener.

The application starts as soon as the user greets the robot. It introduces itself to the listener, outlines its ability to talk about the book, and suggests the user select from a set of different characters. The robot asks whether it should talk about Alice, the white rabbit, or the queen of hearts. The user accepts or declines via speech commands and listens to the presented contents. As soon as no new information is left, the robot suggests another character. The selection of the book chapters works similarly by selecting chapter indices. Figure 14.2 illustrates the interaction loop:

1. the robot asks for input about which topic to talk about, and
2. the subsequent storytelling phase optimizes the robot's expressed degree of extraversion based on the user's engagement.

The robot suggests a character or book chapter (e.g., "Shall I tell you something about the white rabbit?") and waits for the human's response from the ASR. The user's answers are captured with the worn headset. As soon as the answer is identified, the SSP starts capturing and interpreting the user's non-verbal behaviors (see section 14.1.3) and sends the result to the RL (see section 14.1.4) process. During talking about the book, all the robot's presented utterances are generated dynamically (see section 7.2). The process restarts when the robot finishes its presentation. Then, the robot asks for the next character or chapter of interest, and the user instructs the robot on how to proceed.

14.1.3. Social Signal Processing

A central part of the adaptation process is processing the user's non-verbal social signals. The continuous estimation of human engagement is based on posture information.

Figure 14.3 illustrates the SSP setup. A Microsoft Kinect 2 sensor captures the user's posture and movements during the interaction. It uses a depth camera for skeletal body

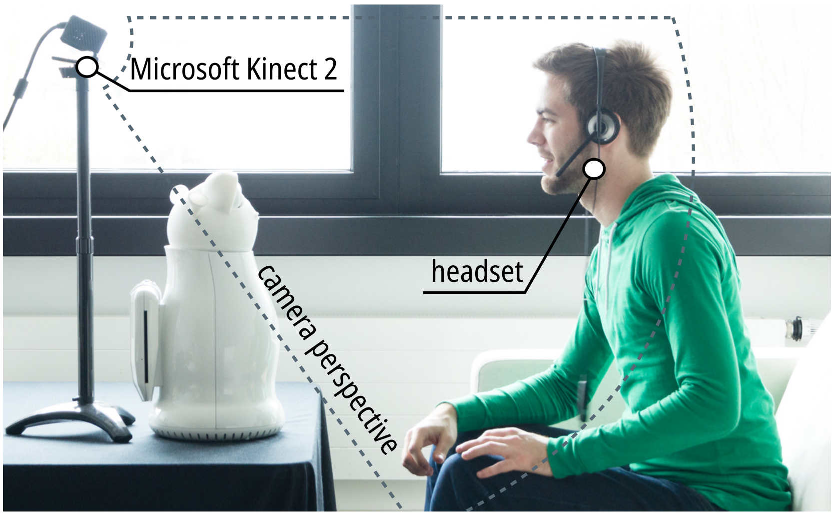


Figure 14.3.: Social signal processing is based on a Microsoft Kinect 2 sensor for posture estimation and a headset for speech recognition.

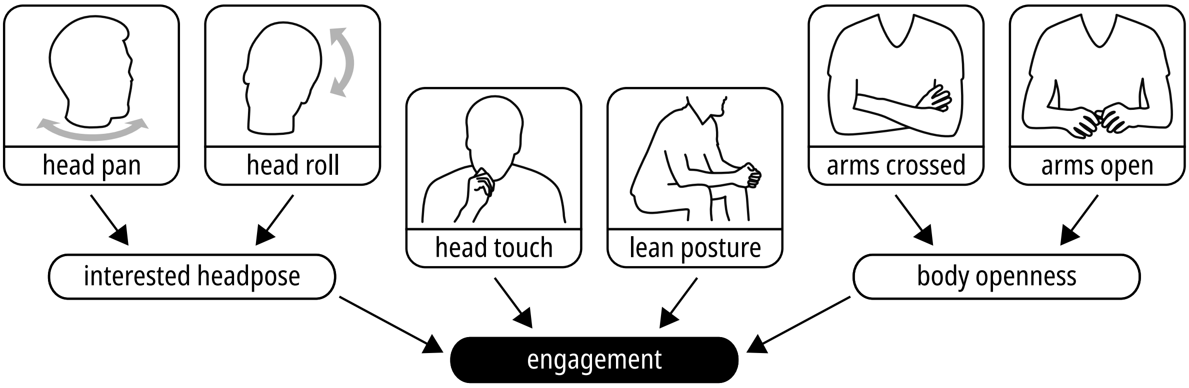


Figure 14.4.: The simplified Bayesian network for user engagement.

detection. The human's pose is passed on to the SSI framework (Wagner et al., 2013), which processes and interprets the skeleton data in real-time (see below). In addition, the user wears a headset with a microphone. Its sole purpose is to record the user's audio for ASR, which is realized with the SSI framework and the Microsoft Speech Platform. The headset has the main advantage of being near the audio source for better audio quality, making recognition more accurate. The recognized keywords are used for controlling the program flow (see section 14.1.2). They do not have an impact on engagement estimation.

User engagement at time t is defined as the floating point value $e_t \in [-1; 1]$. A positive value $e_t > 0$ indicates that the user is engaged, and a negative $e_t < 0$ indicates that they are disengaged. The absolute value determines the magnitude of (dis)engagement. $e_t = 0$ is the neutral value, i.e., neither engaged nor disengaged. As suggested in Baur, Schiller, and André (2017), the user's engagement e_t is estimated based on a Dynamic Bayesian network (BN), which is a directed, acyclic graph with nodes representing variables and edges describing conditional probabilities (Russell and Norvig, 2003). Moreover, in *Dynamic* BNs temporal dependencies between the current state of variables and their earlier states can be modeled.

Figure 14.5.: An engaged and disengaged user interacts with the storytelling robot.

Figure 14.4 illustrates a simplified abstraction of the Dynamic BN for user engagement. Each of the observed nodes contains two discrete states: *present* and *absent*. Based on the observations in these nodes, the probability of the values *present* and *absent* for the final node *engagement* can be inferred. The network further contains "hidden" values that may not be directly observed but have to be inferred from observable variables. As an example, the likelihood that the variable *interested head pose* has the value *present* is high if the value for the variable *head pan* tends towards *present* and the value for the variable *look away* would be close to *absent*. The evidence for these values is constantly updated in real time. For example, the probability that the variable *arms crossed* has the value *present* is high if the corresponding social cue has been recognized with high confidence.

Cues considered relevant for the scenario at hand include head tilt and orientation, indicating whether the user is interested in the current interaction. The openness of the body is determined by the arm posture (opened or closed/crossed). e_t increases or decreases depending on the user's gesture and posture over time. Figure 14.5 shows a user applying engaged and disengaged non-verbal behavior towards the robot. For example, users who lean forward are interpreted as more engaged than when they lean back. Further, the amount of conversational regulators (Ekman and Friesen, 1969), such as back-channels, indicates high engagement. The BNs at hand has been modeled with the GeNIe software (bayesfusion.com, 2021). The probabilities of the variables in the network were learned based on the NoXi corpus (aria agent.eu, 2021), which includes interactions of experts and novices about a certain topic, including audio, video, and Microsoft Kinect 2 depth streams.

Based on the floating point value e_t calculated by the BN, which is sent every 200 ms from SSI to the adaptation process, a moving average with a five seconds window is used to smooth the estimated value. Thus, the user's engagement e_t can be estimated at any time t.

With SSP being an inherent part of the adaptation process, different types of noise potentially occur during runtime (see section 12.3.1.3). Their impact on the learning approach is addressed in the simulation in section 14.2.

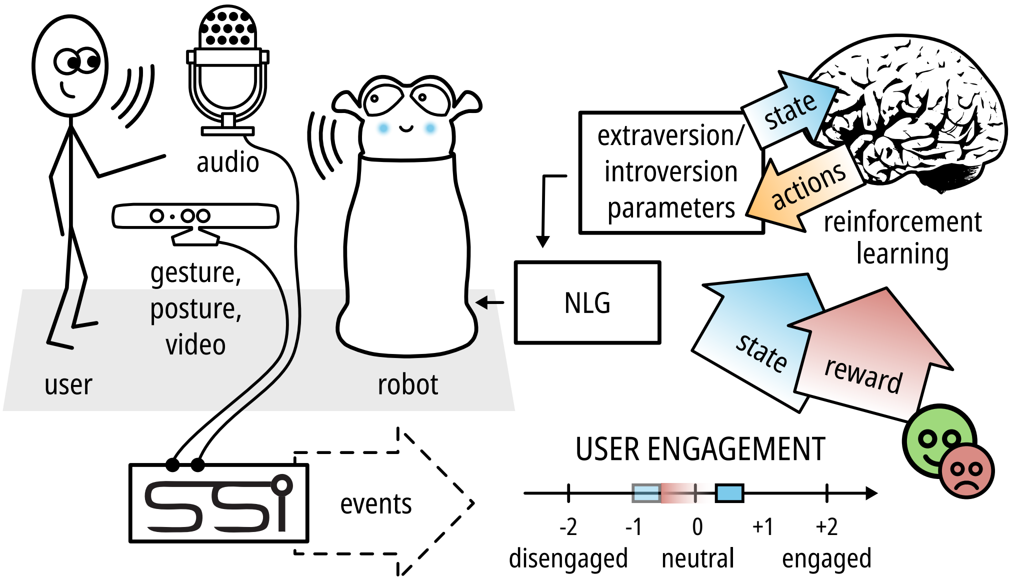


Figure 14.6.: Overview of the RL problem. It is modeled based on human engagement and social signals.

14.1.4. Adaptation Process

Figure 14.6 provides a general overview of the adaptation process. User engagement serves as short-term user information input (see section 6.1.3.2). The robot's goal is to maximize human engagement by adjusting the degree of extraversion in its spoken language. For this purpose, implicit human feedback is calculated automatically without additional interaction from the user.

The temporal development of user engagement plays a key role and is integrated into the socially-aware RL process both in the state space and reward signal. In turn, the agent's actions manipulate the robot's expressed extraversion over time *based on* the user's engagement over time; the robot thus learns how to react to these changes in user engagement. Section 12.3 described such *dependencies*: given a specific user state, the RL agent will choose a greedy action for the current degree of user engagement (see below).

The learning problem is modeled so that it aims to learn after each sentence and should receive feedback every few seconds. Similar to section 13.1.2, one RL time step corresponds to the robot's presentation of one generated description, which takes several seconds depending on the utterance length.

Due to user engagement being a short-term feature the task is expected to be nonstationary with potentially changing q_* values (see also section 12.4.1 and section 14.1.4.4). There is no final state by intention: even if the user reaches maximum engagement, the robot is not "finished" with its task because user engagement could drop afterward. The robot must continue and do its best to keep the user engaged. Therefore, the task at hand is a continuing problem (see section 2.4.3) and only stops when the interaction is finished.

Figure 14.7 illustrates the RL model as proposed in chapter 12. In contrast to section 13.1.2, the user is now involved in both reward and state space while the task itself does not contribute. Again, there is only one source of reward, and the actions of the

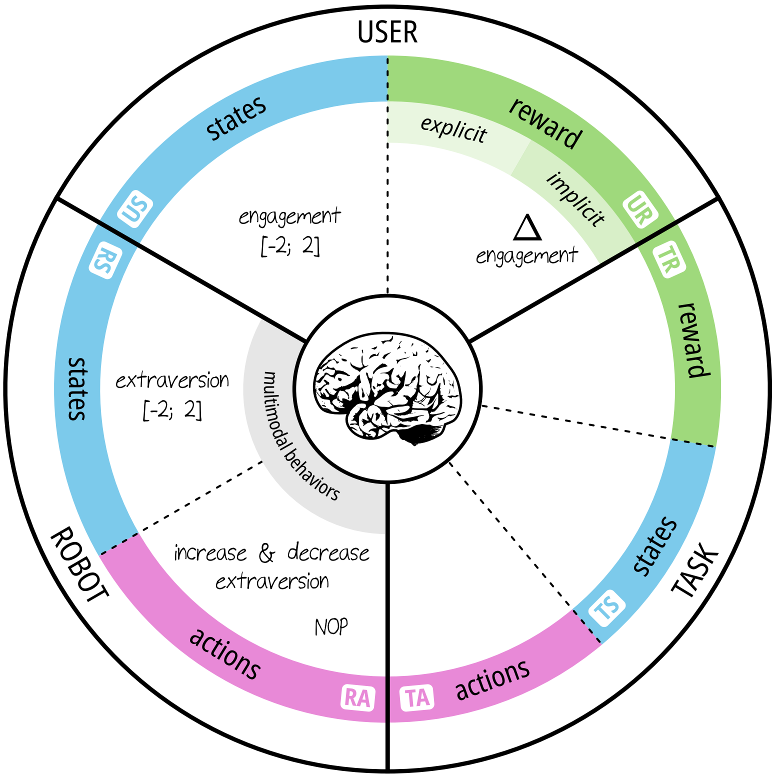


Figure 14.7.: General overview of the RL model.

learning agent manipulate the robot's behaviors. The robot's degree of extraversion and the estimated user engagement span the state space, allowing the learning agent to select greedy actions based on pairs of user engagement and robot extraversion as follows.

14.1.4.1. State Space

The state space has two dimensions:

1. *Estimated user engagement E*: The idea of the adaptation process is to be able to react to changes in user engagement. It means that the agent's learned policy will encode the knowledge of how to behave when user engagement increases, decreases or stays the same. Thus, the robot must be aware of the user's current engagement and differentiate between different degrees of engagement.

2. *Robot's extraversion X*: In combination with the action space (see below), the robot needs to know its current degree of extraversion. The combination of user engagement and the robot's degree of extraversion makes it possible to change its extraversion in smaller steps and learn when it should increase or decrease extraversion depending on the user's engagement.

The estimated user engagement as calculated by the SSP pipeline has been defined as a floating point value e_t in section 14.1.3. However, discretization is necessary for use in the tabular state space. The floating point value $e_t \in [-1; 1]$ is mapped to the integer interval $E \in [-2; 2]$ by splitting the original range $[-1; 1]$ into five equally big sections of size 0.4. For example, if $e_t \in [-1; -0.6]$ this corresponds to $E_t = -2$, for $e_t \in [-0.2; 0.2]$ corresponds to $E_t = 0$, etc.

The robot's degree of extraversion $X \in [-2; 2]$ is as previously defined in section 7.2 with $X = -2$ representing maximum introversion and $X = -2$ representing maximum extraversion. X influences the NLG parameters (see section 7.2), which cause the robot's next utterance to be generated more extravert or introvert accordingly.

With E and X both being integer values with five possible values each, a total of 25 distinct states represent the state space:

$$\mathcal{S} = X \times E$$

14.1.4.2. Action Space

There are three actions: increasing (INCR) and decreasing (DECR) the robot's degree of extraversion X, as well as doing nothing (NOP):

$$\mathcal{A} = \{\text{INCR}, \text{DECR}, \text{NOP}\}$$

In the case of INCR or DECR, the value X increments or decrements by one while limiting the result to the maximum range $[-2; +2]$. The manipulation of X then causes new utterances to be generated according to the new extraversion value by adjusting the parameter set as presented in section 7.2. X remains untouched when executing NOP. Without NOP, the robot would be forced to change its degree of extraversion in every time step.

$$X \longleftarrow \begin{cases} \min(X+1, 2), & \text{for } A_t = \text{INCR} \\ \max(X-1, -2), & \text{for } A_t = \text{DECR} \\ X, & \text{else} \end{cases}$$

Alternatively, one could choose one action for each of the robot's extraversion values in the range $[-2; 2]$, which would directly set the robot's extraversion to the specific value. In this case, X would not be required to be part of the state space since the actions would already represent the absolute degree of robot extraversion. However, the limitation to INCR, DECR, and NOP has an important advantage. It prevents the robot from changing X too fast, which would cause the generation of stylistically diverging utterances in terms of expressed extraversion within a very short time. With the presented solution, a change from maximum introvert to maximum extravert would require at least four steps as extraversion can increase or decrease only by one unit per step.

14.1.4.3. Reward

The learning agent aims to maximize user engagement. Thus, it is important to select actions that either increase or maintain human engagement, e.g., when engagement is already at its highest level. Thus, there is a direct mapping from reward to the human's progression of engagement over time.

In each time step, the pipeline measures the user's engagement e_t as a floating point value (see section 14.1.3) after each of the robot's actions, i.e., presented utterance. Based on the value e_{t-1} from the last learning step, the agent calculates the difference Δe_t, which indicates how much the user's engagement changed during the execution of the last action:

$$\Delta e_t = e_t - e_{t-1}$$

Δe_t is used as the reward signal for the agent since it reflects the temporal change in user engagement with higher accuracy than the discretized difference $E_t - E_{t-1}$. The robot gets a positive reward when the user's engagement increased and a negative reward when human engagement decreased. In contrast to section 13.1.2.2, where the reward was either +1 or −1, the magnitude of Δe_t also determines the magnitude of the reward signal here, which is more accurate information for the learning agent due to floating point operations:

$$R_t = \Delta e_t$$

14.1.4.4. Algorithm

In contrast to 13.1.2, the problem at hand is stateful and nonstationary, i.e., the optimal policy might change over time. Therefore, the Q-learning algorithm (see section 2.6.3) is used in combination with ϵ-greedy exploration (see section 2.3) for implementing adaptation. For real-time interaction, the learning agent uses the following parameter values:

- $\epsilon = 0.2$: An exploration rate of 20 % allows for greedy behavior in 80 % of action selection while maintaining a reasonable amount of exploration. For exploration, the agent uses uniform action selection with a one-third chance for each action.
- $\alpha = 0.1$: The learning rate is constant, which is essential for nonstationary problems to adapt to potential q_* changes. Moreover, it is small enough to prevent divergence in the long run, which could occur when the learning rate is too high.
- $\gamma = 0.9$: The discount factor is slightly smaller than 1 to let the agent focus on the most efficient long-sighted solution.

14.1.4.5. Reinforcement Learning Loop

In the beginning, X_{t-1} is initialized with 0, i.e., the robots starts with neutral introversion-extraversion. One iteration of the RL algorithm involves the following steps in this order:

1. User engagement e_{t-1} at the time just before presenting the next content is stored.
2. An action is selected based on X_{t-1}, E_{t-1} and ϵ-greedy.
3. The robot's extraversion updates according to the executed action. A new utterance is generated and presented with X_t serving as initialization for the NLG parameter set. In the meantime, social signals are continuously interpreted to estimate user engagement.

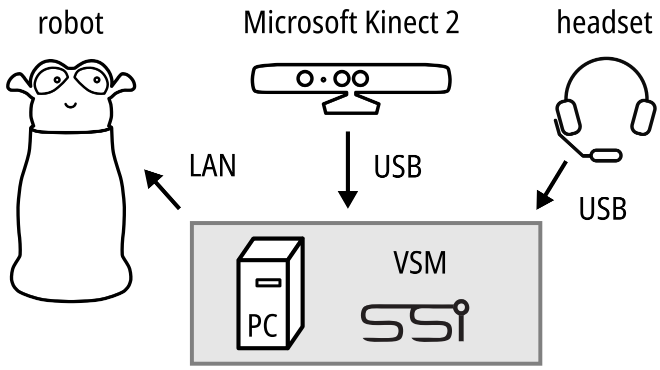


Figure 14.8.: Overview of the hardware and software setup.

4. When the robot stops speaking, the current value e_t is used to calculate the difference in user engagement $\Delta e_t = e_t - e_{t-1}$, which serves as reward.

5. The Q-learning algorithm is used to update the Q values. Afterward, the next time step $t + 1$ begins.

14.1.5. Hardware and Software

Figure 14.8 outlines the technical setup. It consists of the robot, the Microsoft Kinect 2 sensor, and a PC for handling the control flow of the application and SSP. The computer runs the Microsoft Windows operating system, which is required for interfacing with the Microsoft Kinect 2 sensor via its proprietary SDK, which is used by the SSI framework. A Visual SceneMaker (VSM) (Gebhard, Mehlmann, and Kipp, 2012) project implements the overall interaction flow and delegates specific tasks, such as RL, to external Java code. On the other side, VSM receives input from the SSI framework via network sockets, including estimated user engagement and keywords from the ASR, which are required for the interaction flow (see section 14.1.2).

14.2. Simulation

In addition to the running system, a simulation of the real-time adaptation process was implemented. The simulation is based on the same Q-learning and ϵ-greedy approach described above. Similar to section 13.2, the human is replaced with a simulated user to test the learning process. The main task of the simulation is to simulate the user's reactions to the robot's utterances, i.e., the simulation of a change in user engagement, given the robot's degree of extraversion.

14.2.1. Simulated User

The basic idea is that the simulated user's engagement increases when the robot's expressed personality matches the actual preferences and decreases otherwise. Completely deterministic user behavior is not realistic due to several types of noise, which might occur in a real interaction (see section 12.3.1.3, section 12.4.1 and section 12.4.2). The simulation introduces two types of noise. The first one, *user noise*, causes random changes in user engagement. They could occur, e.g., when the user is distracted from the interaction, and their reactions are not related to the robot's actions. The second one, *sensor noise*, causes deviations of the sensed value from the real E_t value in each learning step. For example, this could occur when the SSP fails due to many problems, such as wrong perspective, lighting, and more.

The RL reward calculation is the most important part of the simulation (see algorithm 3 for pseudocode). It introduces both user and sensor noise to the simulation. These probabilities $n_u \in [0;1)$ and $n_s \in [0;1)$ are constant throughout the simulation. In addition, the simulated user's real preferences $x_u \in [-2;2]$, the robot's current extraversion $x_r \in [-2;2]$, as well as the simulated last and current engagement $e_{last} \in [-2;2]$ and $e \in [-2;2]$ need to be provided as input.

Algorithm 3: Reward calculation for the storytelling simulation.

Input: user noise probability $n_u \in [0;1)$, sensor noise probability $n_s \in [0;1)$, user extraversion preference $x_u \in [-2;2]$, robot's current extraversion $x_r \in [-2;2]$, user engagement $e \in [-2;2]$, last user engagement $e_{\text{last}} \in [-2;2]$

Output: reward value

1 $r_u \longleftarrow$ a random float in $[0;1)$
2 **if** $r_u < n_u$ **then** *// random user reaction?*
3 $\quad r_{\text{tmp}} \longleftarrow$ a random integer in $[-2;2]$
4 $\quad e \longleftarrow \max(\min(e + r_{\text{tmp}}, 2), -2)$
5 **else**
6 $\quad \Delta x \longleftarrow \text{abs}(x_u - x_r)$
7 $\quad$ **if** $\Delta x = 0$ **then** *// robot extraversion matches user pref.?*
8 $\quad\quad c = 1$
9 $\quad$ **else**
10 $\quad\quad c = -1$
11 $\quad e \longleftarrow \max(\min(e + c, 2), -2)$
12 $r_s \longleftarrow$ a random float in $[0;1)$
13 **if** $r_s < n_s$ **then** *// sensor noise?*
14 $\quad e_{\text{sensed}} \longleftarrow$ a random integer in $[-2;2]$
15 **else**
16 $\quad e_{\text{sensed}} \longleftarrow e$
17 $\Delta e \longleftarrow e_{\text{sensed}} - e_{\text{last}}$
18 $R \longleftarrow \Delta e$
19 $e_{\text{last}} \longleftarrow e_{\text{sensed}}$
20 **return** R

First, a random number is drawn according to the user noise probability n_u. In the case of user noise, the simulated user's engagement increases or decreases by a random amount (see line 4). In the regular case, the rule-based approach checks how much the

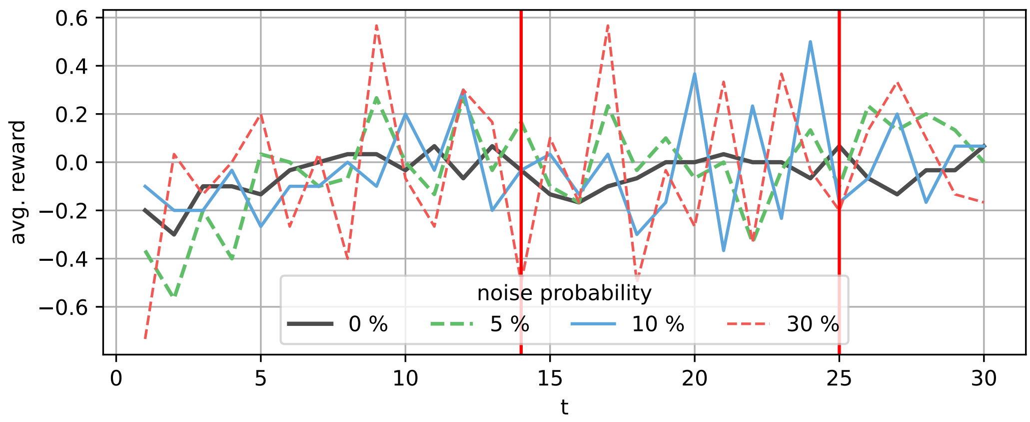


Figure 14.9.: Q-learning results (30 simulated users).

simulated user's preferences diverge from the robot's current extraversion (see line 6). In the case of equality, the simulated user's engagement increases. Otherwise, it decreases.

Second, another random number is drawn according to the sensor noise probability n_s. In the case of sensor noise, a random value is used as sensed user engagement e_{sensed} instead of the "correct" value e (see line 14).

Finally, the difference in current and previous user engagement (from the last step) Δe is calculated (see line 17). For $\Delta e \neq 0$, the reward signal is positive or negative according to Δe. While the real running system uses the more accurate floating-point difference from the SSP pipeline as the reward (see section 14.1.4.3), the simulation uses the integer difference $E_t - E_{t-1}$ since the SSP pipeline is not running.

14.2.2. Results

For the simulation at hand, 30 simulated users were used, each interacting for 30 steps. The simulation initializes the simulated user's preference randomly in each run. The robot starts with neutral extraversion $X = 0$, and the agent's Q table is initialized with zeros at the beginning. Each of the 30 runs stops after 30 steps, corresponding to 30 generated robot utterances and feedback from the human in terms of user engagement. The simulation uses the same learning parameter values as the interactive robot application (see section 14.1.4.4).

Introspective measurements are used to evaluate the self-motivated goal of the adaptation process as described in section 12.2.4. The average reward (see section 2.7.1) is calculated for all 30 users to illustrate the agent's performance. Figure 14.9 plots the result for different noise probabilities: 0 %, 5 %, 10 % and 30 %. They determine the amount of user and sensor noise that may occur simultaneously. The simulation runs for each probability, i.e., 30 simulated user interactions with 30 steps each. In order to make them more comparable, the same random seed is used for all noise probabilities.

In addition to the noise, the simulation includes another challenge to make the learning progress more visible. To evaluate how much time is needed to adapt to preference changes algorithmically, user preferences change at steps 14 and 25 (vertical lines) during the experiments to a new, random value. It represents a worst-case scenario as

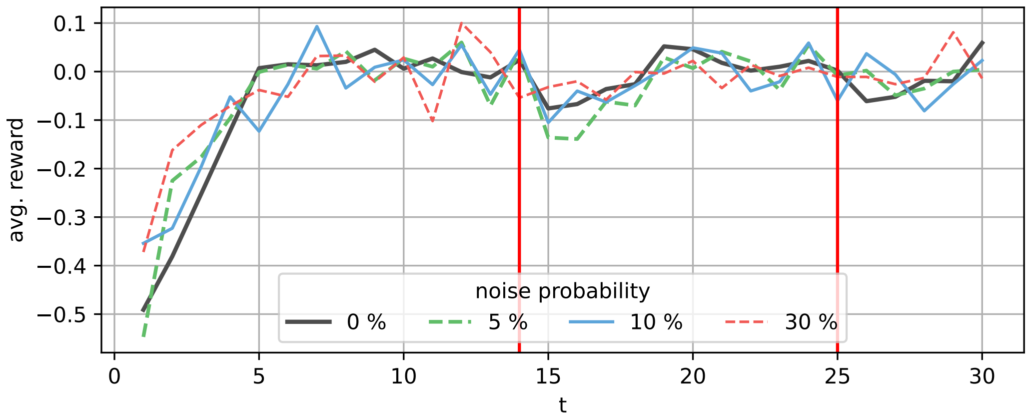


Figure 14.10.: Q-learning results (1000 simulated users).

preferences are expected to change gradually in real interaction and not within one step. However, it illustrates the agent's ability to adapt to the new preferences after a temporary performance loss.

Without noise, learning is quite robust. The average reward approaches zero: when the robot's extraversion level X equals the user's preference, it learns not to change X anymore. As a result, the user's engagement does not change (which results in zero rewards) apart from random noise. After the initial learning phase, positive and negative rewards can be attributed to exploration, noise, and resulting suboptimal behavior of the agent. Increasing noise leads to non-zero rewards more frequently. Figure 14.10 plots the result for 1000 users, which smooths the plot and gives an impression of performance on average.

14.3. Experiment: Adaptive Joke-Telling

This section introduces an experiment in the entertainment context, where a social robot's multimodal joke presentation gets adapted to the individual user. Similar to the scenario above, the socially-aware adaptation approach runs autonomously based on implicit feedback. It aims to maximize the spectator's amusement by personalizing the robot's verbal and non-verbal behaviors. The experiment explores two variations:

1. *Content selection*: The robot presents multimodal humor, including canned jokes in different joke categories, grimaces, and sounds (see section 9.1). The content is manually designed and prepared in advance. The adaptation process aims to select those contents and combinations of modalities that maximize the user's amusement.

2. *Multimodal augmentation*: The robot presents dynamically generated punning riddles, which are augmented with multimodal, paralinguistic cues (see section 9.2). The adaptation process aims to optimize the multimodal presentation, i.e., the specific use of paralinguistic cues, to maximize the user's amusement.

Figure 14.11.: The joke-telling interaction scenario.

While selected literature explored adaptation in terms of content selection in stand-up comedy shows (see section 5.4), the focus is here on adaptation to generate and personalize a multimodal performance by intelligently combining multiple modalities. Therefore – and similar to section 14.1 – a socially-aware and nonstationary RL approach with implicit feedback is implemented. Again, SSP is a central component of the process: human affect is included in the adaptation process as an interaction dynamic. The adaptation approach is simulated in section 14.4 and evaluated in a lab study in section 14.5.

The concept, simulation, and implemented adaptation approach were presented and reviewed in Ritschel and André (2018), Ritschel et al. (2020a), Ritschel et al. (2020b), Weber et al. (2018b) and Weber et al. (2018a). The contents of this section expand these publications.

14.3.1. Overview

This section describes the setup, SSP and RL model of the adaptation approach. The RL learning agent manipulates parameters of the joke-telling approaches from chapter 9, which results in a personalized multimodal performance. In contrast to the literature from section 5.4 and section 6.2 the focus is on the non-functional adaptation of the robot's multimodal humor in a joke-telling scenario, which fits well in domestic environments.

Making the robot acquire the human interaction partner's sense of humor quickly is one opportunity to improve the user experience in social interactions. As varied as human humor preferences are also the topics of jokes, such as gross-out, slapstick, or academic jokes. There is probably no topic without jokes about it. However, many jokes require background knowledge from a specific domain, which makes it hard to predict whether a person will understand and like the joke.

The social robot approaches entertainment by trying many humor stimuli and learning from human reactions. Depending on the interests of the interaction partner, jokes from different categories and visual and audible content work better or worse for different persons.

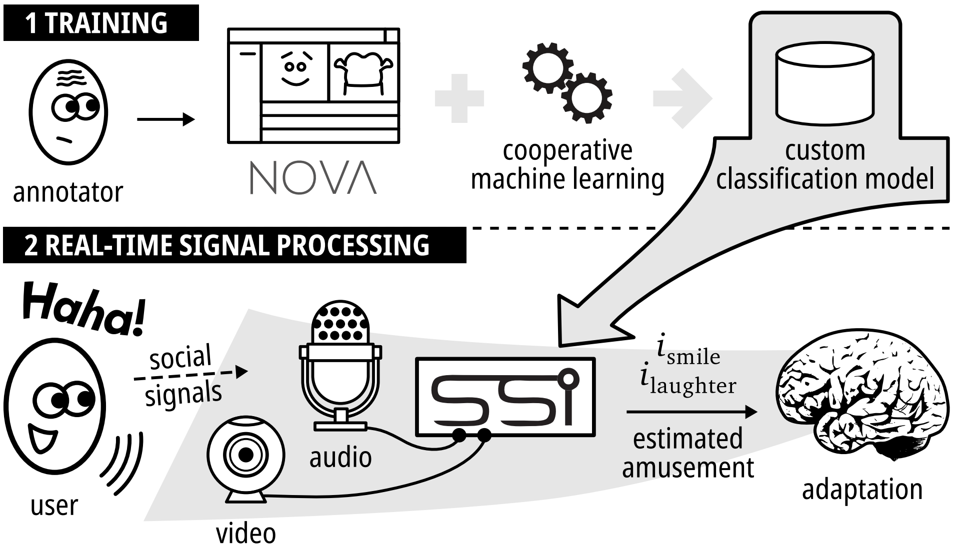


Figure 14.12.: Overview of the classification model training and real-time recognition process.

Figure 14.11 illustrates the interaction scenario. The robot presents jokes while the user is watching the performance. The humorous multimodal content includes jokes, funny sounds, and grimaces (see chapter 9). During the interaction, the user watches the robot's joke presentation. A webcam and microphone record the user. Similar to section 14.1 the processed human's social signals serve as input to the adaptation process.

All the robot's utterances are realized with the Cerevoice TTS software (w3.org, 2021) and played back with the robot's internal speaker. Grimaces are presented with the robot's face via internal motors. Besides the robot and the headset, a powerful computer handles SSP and the application's control flow.

14.3.2. Social Signal Processing

A crucial part of the adaptive joke-telling robot is its ability to understand the spectator's reaction to its performance. The SSP component constantly estimates the user's amusement. In particular, the robot can recognize human visual smiles and audiovisual laughter as a contemporary response to the robot's performance during the interaction. The estimated amusement is an essential input for the socially-aware RL process, which aims to optimize the show to the individual spectator's preferences (see section 14.3.3).

Laughter has for years been identified as a crucial part of social interaction by traditional conversation analysis (Glenn, 1989). It is the most evident reaction toward a successful punchline within a joke. Naturally, audible laughter is accompanied by a visual component, i.e., a smiling expression in the facial modality. The automated recognition of human laughter behavior has been applied to several conversational interfaces to generate an engaging and pleasant user experience (Urbain et al., 2010; Niewiadomski et al., 2013). The audience's smile and laughter serve also as user input for adaptation in some literature addressing robot stand-up comedy and humor (see section 5.4 and section 6.2).

The user's laughter and smile are key components to estimating user amusement. A two-step approach is presented in the following for the interpretation of these social signals (see Figure 14.12):

1. A custom machine learning model is trained offline to recognize human smiles and audible laughter.

2. The trained model is used during runtime for real-time classification of the visual and audible sensor input.

14.3.2.1. Annotation and Training

A custom machine learning model is trained offline to recognize laughter from the audio modality and smiles from video images. The models are trained using the NOVA annotation and cooperative machine learning tool (Wagner et al., 2018). The tool is designed to annotate social signals in continuous audiovisual recordings of social interactions between humans and between humans and social robots. Additionally, it supports cooperative machine learning during the annotation process to reduce the annotation labor and speed up the overall task.

To describe the paralinguistic content of voice, Mel-frequency cepstral coefficients (MFCCs) spectral, pitch, energy, duration, voicing, and voice quality features are employed, which were extracted using the EmoVoice toolbox (Vogt, André, and Bee, 2008). These features are used within an support vector machine (SVM) model. It is trained on excerpts of the Belfast Storytelling Database (McKeown et al., 2015), which contains spontaneous social interactions and dialogs with a laughter-focused annotation. Person-independent evaluation of the model on the training database showed an unweighted accuracy of 84 % for recognizing laughter frames. For detecting smiles in the video, transfer learning is applied to fine-tune the deep convolutional neural network VGGFace (Parkhi, Vedaldi, and Zisserman, 2015) by retraining it on the AffectNet (Mollahosseini, Hassani, and Mahoor, 2019) facial expression corpus.

14.3.2.2. Real-time Recognition

Audiovisual laughter recognition is carried out in real time during the interaction based on the trained models. The lower part of Figure 14.12 illustrates the approach: the robot continuously captures the spectator's social signals with a headset microphone and webcam. It analyzes the data with the SSI framework (Wagner et al., 2013) (see Figure 14.13). SSI monitors video and audio inputs from the sensors, voice activity, laughter, and smile probabilities.

The software detects bursts of laughter on a frame-by-frame basis: the audio signal is analyzed within a one-second sliding window that is shifted every 400 milliseconds, resulting in a decision rate of 2.5 Hz. The overall activity is monitored by applying a voice activity transformation to the signals via hamming windowing and intensity calculation. Coherent signal parts (i.e., frames) in which the mean of squared input values – multiplied by a Hamming window – that exceed predefined thresholds for intensity are identified as carriers of vocal activity. They serve as input for feature calculation and subsequent

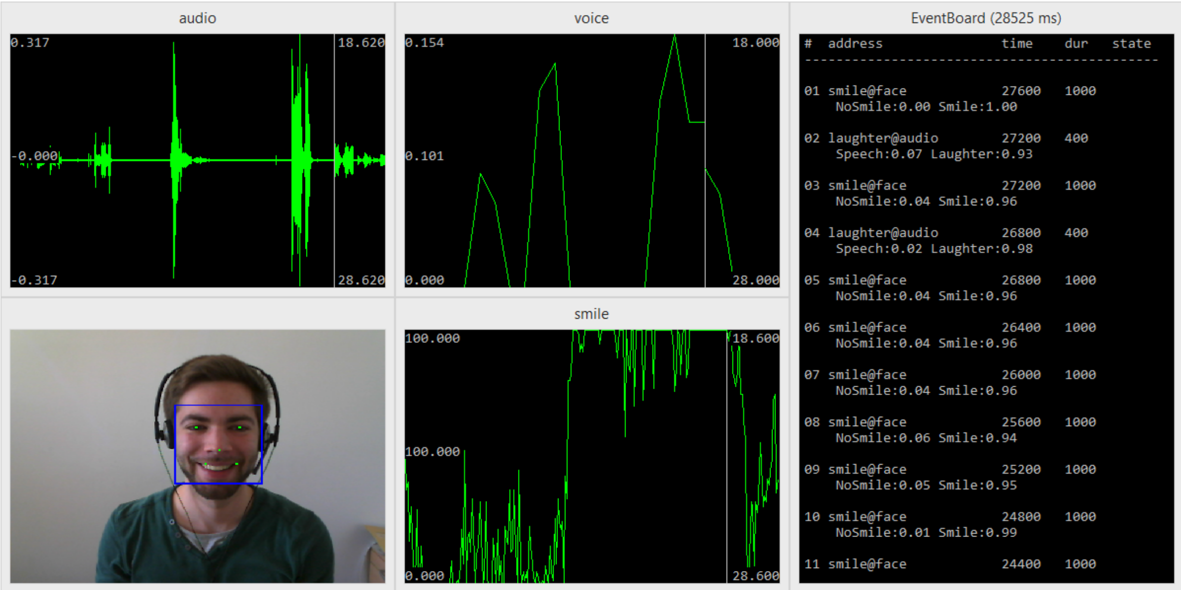


Figure 14.13.: Sensing the spectator's smile and laughter with the SSI framework.

classification. Video is captured at a rate of 15 frames per second. Each frame is classified with the neural network model described above. The probabilistic results are averaged with the same sliding window as the audio modality to gain equally clocked classification results from both input signals.

The final output of the recognition system is the intensity of recognized laughs and smiles in the form of continuous scores. McKeown and Curran (2015) state the strong connection of high-intensity laughs with perceived humor, a continuous relation that has already been proposed by Darwin mentioning a range "from violent to moderate laughter, to a broad smile, to a gentle smile" (Darwin, 1965). Following these guidelines, it makes sense to incorporate an intensity assessment. To this end, the confidence scores given by our laughter and smile classifiers are interpreted as intensity measurements. Though there are other sophisticated approaches to quantify the intensity of vocal laughs (Urbain et al., 2014) and facial smiles (Lynch, 2010), the restrictions of real-time capability must be taken into account. The probabilistic output of the implemented classification systems is a good intensity measure that can be computed efficiently.

The result of the SSP pipeline are two continuously updated and averaged floating point values, which serve as input to the adaptation process:

- $i_{\text{smile}} \in [0; 1]$: current intensity of visual smile
- $i_{\text{laughter}} \in [0; 1]$: current intensity of audible laughter.

14.3.3. Adaptation Process

The goal of the learning agent is to maximize user amusement over time. As mentioned previously, the approach is investigated in two variations. Due to the complexity of the task, each aspect is treated individually, i.e., different state and action spaces are presented for content selection and the paralinguistic presentation strategy. The rest of

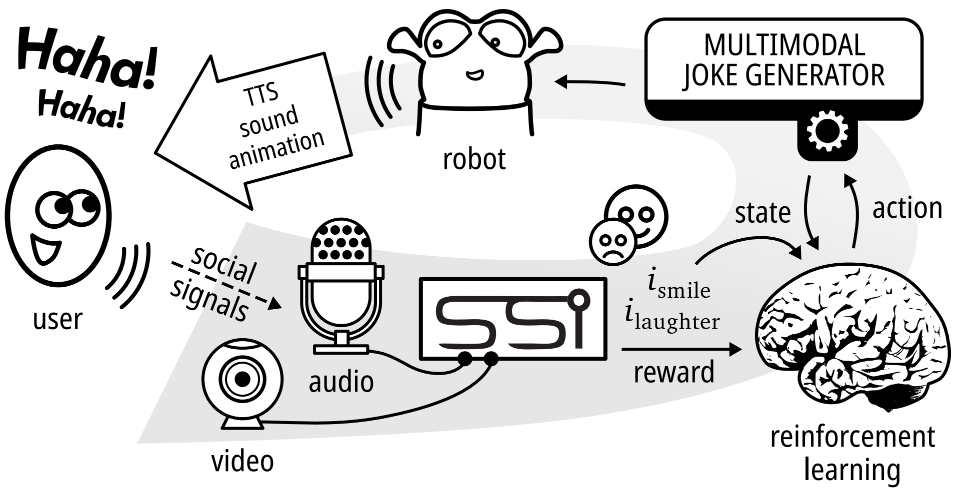


Figure 14.14.: Overview of the adaptation process.

the model is the same, as is the learning agent's task: find the optimal policy for the individual spectator, either concerning content selection or the paralinguistic presentation strategy. In both cases, the same SSP approach calculates the reward signal as described before.

Figure 14.14 gives an overview of the adaptation process. User affect is the user information (see section 6.1.3.2) that serves as input to the adaptation process. Similar to section 14.1.4, the change of the interaction dynamic over time determines the reward. Thus, it is used as implicit feedback in the socially-aware RL process.

The robot's actions determine the selection and use of jokes and multimodal behaviors. One RL time step corresponds to the presentation of one joke. Learning happens after each joke so that the robot receives feedback a few times per minute, depending on the duration of the presented content.

Similar to section 14.1.4 user affect is a short-term feature and thus the task is expected to be nonstationary with potentially changing q_* values (see also section 12.4.1 and section 14.1.4.4). Again, the task is non-episodic since the user's amusement should be preserved over a longer time. However, it could also drop at any point. Therefore, the task at hand is a continuing problem (see section 2.4.3) and only stops when the interaction finishes.

Figure 14.15 and Figure 14.16 illustrate the RL models as proposed in chapter 12. The reward signal is the same, but they differ in state and action space. In contrast to Figure 14.16, the user is involved both in terms of reward and state space in Figure 14.15, similar to section 14.1.4. Again, the user is the single source of reward. The learning agent's actions control the robot's generated behaviors.

14.3.3.1. State Space

Content Selection

The general idea of the content selection agent is to choose an amusing joke, grimace, sound, or one of their combinations *depending on the user's current amusement level* (see Figure 14.15). The user's current amusement level is part of the state space to make

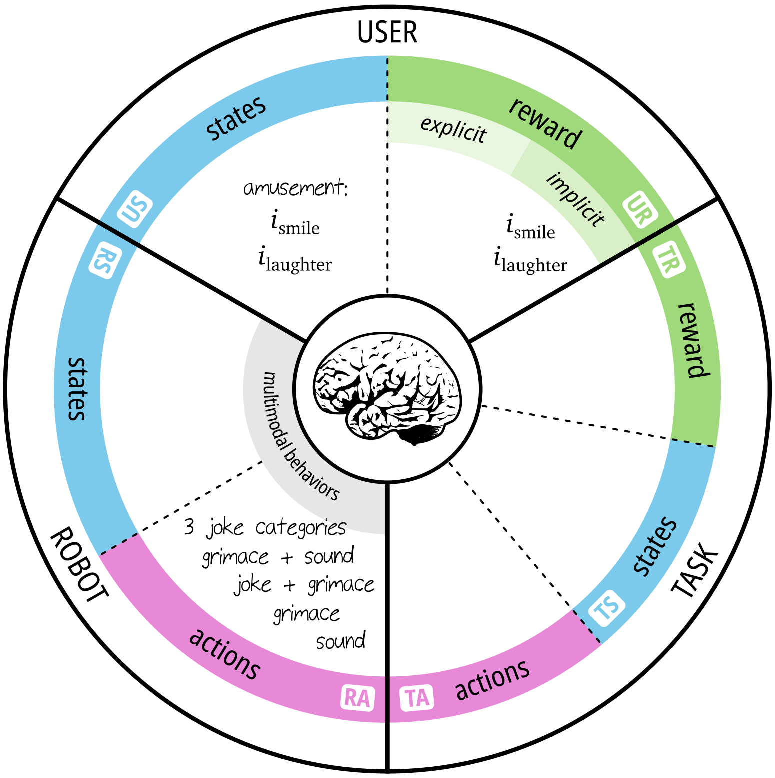


Figure 14.15.: General overview of the RL model (content selection).

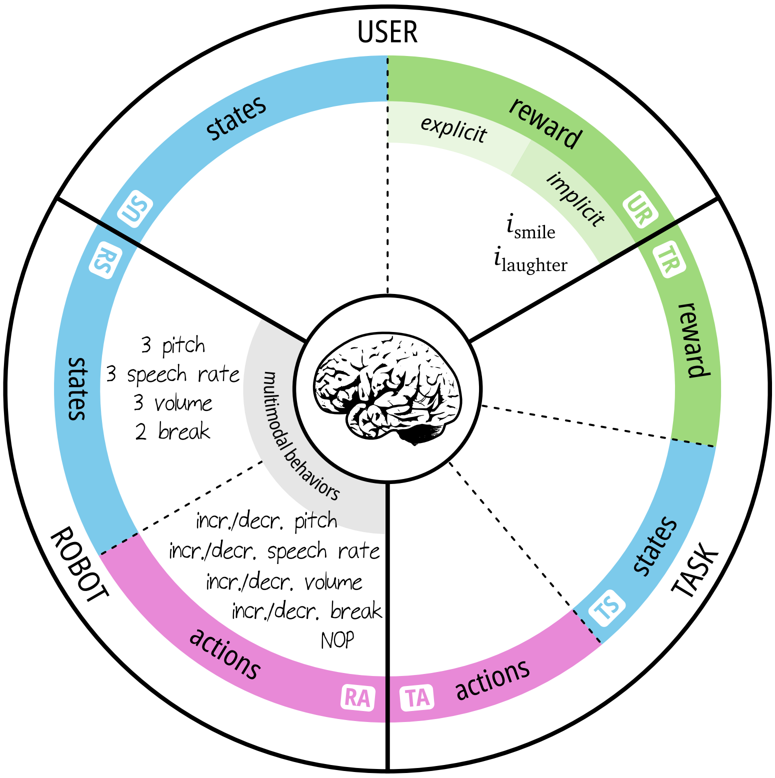


Figure 14.16.: General overview of the RL model (paralinguistic presentation strategy).

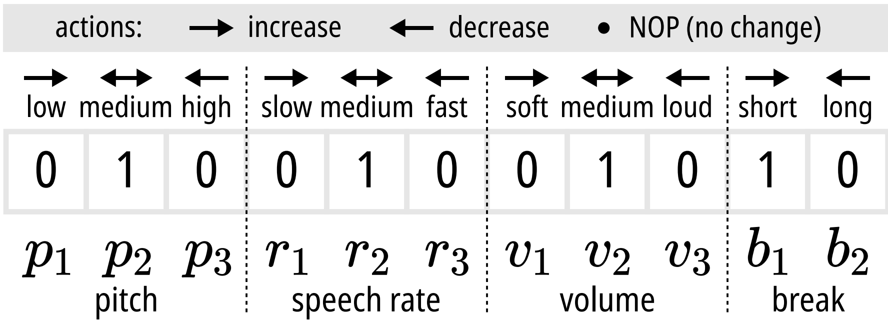


Figure 14.17.: State (initial configuration) and action space.

it possible to react to user amusement changes and learn the connection between the presented stimulus and the resulting degree of amusement. The state is defined as a two-dimensional non-discretized feature vector. It contains the current floating point intensities of smiles and laughter as calculated and reported by the SSP pipeline (see section 14.3.2):

$$\mathbf{x}(s) = \begin{pmatrix} i_{\text{smile}} \\ i_{\text{laughter}} \end{pmatrix}$$

Traversing the state space can be interpreted as the agent observing the user and deciding which action to take depending on the user's reactions to the last action. Having observed the result, the agent would then learn that choosing action A_t increases or decreases user amusement. This traversal would then be associated with a positive or negative reward (see below) and thus increase or decrease the probability of selecting the corresponding action A_t in the future in state $\mathbf{x}(S_t)$ and in similar states.

Paralinguistic Presentation Strategy

In contrast to the content selection problem, the paralinguistic presentation agent aims to adjust its multimodal cues *depending on the current configuration of its multimodal behaviors* (see Figure 14.16). The state encodes the robot's multimodal humor markers that are applied during the generation of the robot's next utterance, including the current configuration of *pitch*, *speech rate*, *volume* and *break* (see also section 9.2). A four-tuple (pitch, speech rate, volume, break) $\in \mathcal{S}$ is converted into an eleven-dimensional feature vector:

$$\mathbf{x}(s) = (p_1, p_2, p_3, r_1, r_2, r_3, v_1, v_2, v_3, b_1, b_2)^\top$$

Figure 14.17 illustrates the design of the vector. $\mathbf{x}(s)$ is divided into four sections. All components in the vector are associated with a specific manifestation of the respective marker: digits (1 or 0) are associated with the pitch attributes (p_1, p_2, p_3, i.e., *low*, *medium*, *high*), with the speech rate (r_1, r_2, r_3, i.e., *slow*, *medium*, *fast*), with the volume (v_1, v_2, v_3, i.e., *soft*, *medium*, *loud*) and with the length of the pause (b_1, b_2; i.e., *short*, *long*). Every section is one-hot-encoded, i.e., only one manifestation per marker can be 1, and the rest of the section is 0. The configuration in Figure 14.17 illustrates the initial state with medium pitch, speech rate, and volume, as well as a short break. Traversing the state space can be interpreted as gradually changing the multimodal presentation strategy step by step (see below).

14.3.3.2. Action Space

Content Selection

A set of seven actions is used for learning about the most effective humor stimuli in terms of modalities and their combination (see Figure 9.1 for an illustration). It includes presenting a grimace (GRC), presenting a sound (SND), telling jokes about three different topics (JOK1, JOK2, JOK3), presenting a grimace with a sound (GRC_SND), and presenting a joke with a grimace (JOK_GRC):

$$\mathcal{A}_{\text{cont}} = \{\text{GRC, SND, JOK1, JOK2, JOK3, GRC\_SND, JOK\_GRC}\}$$

Each action is associated with a set of different contents, i.e., grimaces, sounds, and multiple sets of jokes associated with different topics. For example, when executing the GRC action repeatedly, the robot shows different grimaces each time GRC is executed to provide variety and avoid repetitions. The selection within the sets of multimodal content is random; already presented contents are removed until all contents from the set have been presented.

19 grimaces and 23 sounds were created or selected by hand. The jokes were prepared in advance, too. However, one could combine them with the multimodal joke generation approach presented in section 9.2 by using the topic as input, e.g., for the STANDUP joke generator, as done for the paralinguistic presentation strategy. However, as the study participants were German native speakers, manually designed contents in the German language were used as a compromise (see also section 14.5).

Paralinguistic Presentation Strategy

The action space of the adaptation process for the optimization of the paralinguistic presentation strategy is tightly coupled with the state space. As illustrated in Figure 14.17 two actions exist for each of the multimodal markers to increase and decrease pitch (INCR_PITCH, DECR_PITCH), speech rate (INCR_RATE, DECR_RATE), volume (INCR_VOL, DECR_VOL) and break time (INCR_BREAK, DECR_BREAK). Moreover, the action NOP does not change anything and leaves the current configuration untouched. Otherwise, keeping the current marker configuration would be impossible, such as when the optimal presentation strategy has been found.

$$\begin{aligned}\mathcal{A}_{\text{para}} = \{&\text{INCR\_PITCH, DECR\_PITCH, INCR\_RATE, DECR\_RATE,}\\ &\text{INCR\_VOL, DECR\_VOL, INCR\_BREAK, DECR\_BREAK, NOP}\}\end{aligned}$$

Consequently, the robot can change one marker per learning step. Marker values cannot be set to a specific value directly to prevent switching between minima and maxima. First, it reduces the number of required actions and therefore speeds up learning due to less required exploration for the action space. Moreover, setting parameters from one extreme to another could make the robot's behavior appear strange.

The action space is state-dependent. In each state, the set of actions contains NOP and those actions that change the state. If a marker already has the minimum/maximum

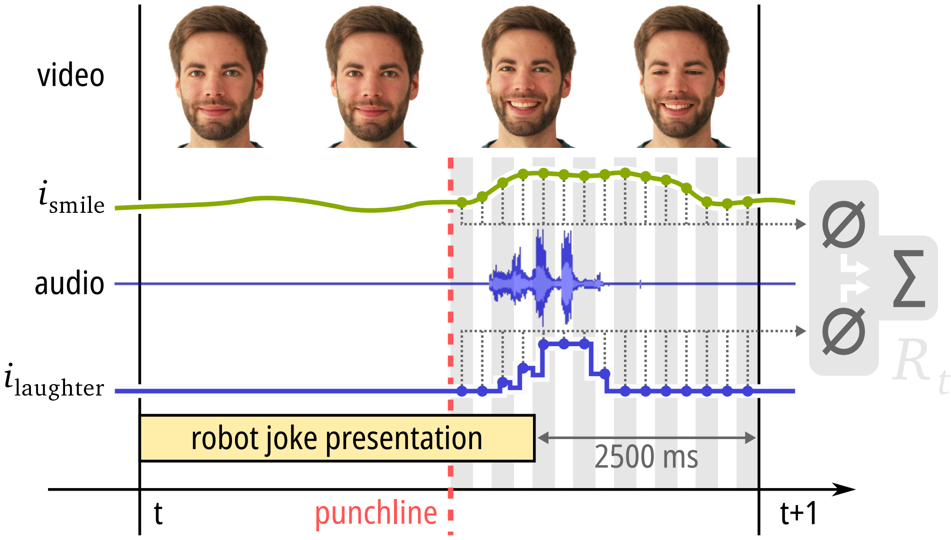


Figure 14.18.: The spectator's estimated amusement is monitored continuously. The relevant smile and laughter samples are averaged and summed up as the reward signal.

value, it cannot decrease/increase any further. Corresponding actions are removed from the action space for the particular state (see also Figure 14.17). For example, the action DECR_PITCH does not exist in states with a low pitch.

As illustrated in Figure 9.3, the robot's smile and laughter are randomized to a certain degree for variation. The adaptation process does not manipulate them.

14.3.3.3. Reward

Similar to the storytelling scenario in section 14.1, social signals determine the reward signal. The estimated human amusement (see section 14.3.2) is used to calculate the reward for the learning agent based on the intensity of smiles and laughter during and after the punchline. In contrast to section 14.1, it does not calculate the difference from the last time step but an absolute value.

Findings by Katevas, Healey, and Harris (2015) indicate that human reactions to a joke usually peak just after the punchline. As illustrated in Figure 14.18, user amusement is sensed continuously from the punchline until 2.5 seconds after the end. This timespan was hand-tuned to capture timely and delayed human social signals. Delayed reactions occur when the user needs time to think about the punchline until the joke is fully understood, and laughter occurs only a few moments later. It was observed in initial tests that some people slightly smile during the setup of a joke before the punchline because they anticipate an answer, expect a funny outcome, or are amused by the robot's visual embodiment. However, this is person- and joke-dependent. Thus, the sensed amusement during the joke setup is not considered for reward calculation, which is in line with the findings by Katevas, Healey, and Harris.

Grimaces and sounds do not have a punchline. Katevas, Healey, and Harris (2015) mention the best time for measurement is during and right after the action, but not before either. Thus, the reward calculation starts in parallel with sound playback or animation. Again, all signals accumulate until 2.5 seconds after the grimace/sound and contribute to the reward.

The relevant smile and laughter samples are averaged in order to eliminate outliers. The averaged smile ($\overline{i_{\text{smile}}}$) and laughter ($\overline{i_{\text{laughter}}}$) intensities are weighed and summed up to estimate the spectator's amusement, which is used as the reward signal:

$$R_t = \frac{1}{2} \cdot \overline{i_{\text{smile}}} + \frac{1}{2} \cdot \overline{i_{\text{laughter}}}$$

14.3.3.4. Algorithm

Similar to section 14.1, the experiment at hand is stateful and nonstationary. It uses the algorithm from section 2.8 for linear function approximation in combination with ϵ-greedy action selection (see section 2.3). Linear function approximation is used here to benefit from generalization between similar states. Generalization makes it possible to apply knowledge from known states in new states that have not been observed before. It is an important advantage over the table-based learning algorithms used in section 13.1 and section 14.1. The Fourier basis is used as described in Konidaris, Osentoski, and Thomas (2011) for learning potential *non*-linear dependencies between state values. Most likely, there will not be one optimal solution for all users, but preferences might differ individually and change over time.

For real-time interaction, the following parameter values are set:

- Initial $\epsilon = 0.5$: with an exploration rate of 50 % in the beginning, the agent explores half of the time to collect many samples. Over time, the exploration rate decreases by 0.05 after each step and is capped at a minimum of 0.1. Uniform action selection is used in case of exploration with the same chance for each action.
- $\alpha = 0.5/k$ with $k = 15$: the learning rate is set in relation to the chosen Fourier base and number of actions to guarantee convergence.
- $\gamma = 0.2$: the discount factor is chosen in relation to the learning rate. Smaller discount factors focus the agent primarily on the immediate reward, resulting in a more short-sighted behavior.

14.3.3.5. Reinforcement Learning Loop

In the beginning, the robot tries every action once to obtain at least one sample for each action. Afterward, it uses the ϵ-greedy strategy. The weights for the linear function approximator are initialized with a normal distribution based on Sutton et al. (2009).

One iteration of the RL algorithm involves the following steps in this order:

1. An action from $\mathcal{A}_{\text{cont}}$ or $\mathcal{A}_{\text{para}}$ is selected and executed.
2. When reaching the punchline, the application starts recording the calculated smile and laughter intensities over time. If there is no punchline (e.g., for sounds or grimaces without speech output), the application starts recording this data parallel with action execution.

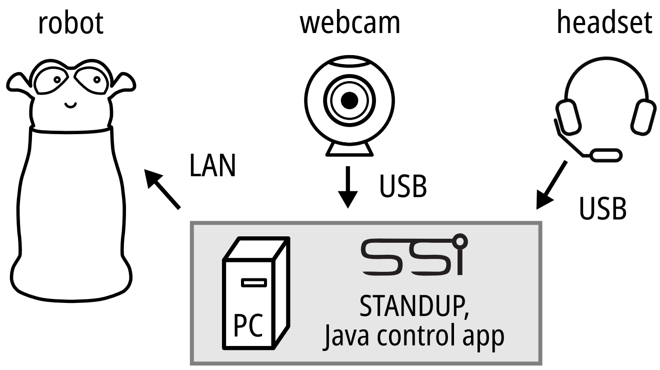


Figure 14.19.: Overview of the hardware and software setup.

3. After the humor presentation, the received data from the SSP pipeline are used to calculate $\overline{i_{\text{smile}}}$ and $\overline{i_{\text{laughter}}}$ and the resulting reward.

4. The RL algorithm updates the weights. Afterward, the next time step $t + 1$ begins.

14.3.4. Hardware and Software

Figure 14.19 gives an overview of the technical setup of the hardware and software. A computer with Microsoft Windows runs a Java application, which implements the overall program and interaction logic. The STANDUP joke generator by Manurung et al. (2008) generates the punning riddles. Robot grimaces, sounds, and texts are loaded from the hard drive. The Java control application interfaces with the robot via the custom *reeti-rest* software for multimodal animation generation (see chapter 11).

Since there is no GUI in this application, the robot's only input is from the user's headset microphone and the webcam. Both connect to the computer with a USB cable. The SSI software processes the streams of the sensors in real time. SSI and the Java control application with the RL logic communicate via a socket connection.

14.4. Simulation

Similar to section 13.2 and section 14.2 a simulation was implemented for the real-time adaptation process. Again, the purpose was to get a first impression of the learning process based on a simulated artificial user before evaluating it with real humans. The rule-based simulation uses the same learning algorithm from the running system described above. It focuses on the adaptation of the paralinguistic presentation strategy.

14.4.1. Simulated User

Similar to the simulation from section 14.2, the idea is to give each simulated user a set of user preferences unknown to the learning agent. These preferences are initialized randomly in each simulation run, corresponding to a set of different human users with different preferences. As in the interactive prototype, the task of the RL agent is to estimate the artificial user's preferences.

The underlying idea of the simulation is that the user's amusement increases if the robot's humor presentation style matches its preferences. For the simulation, this means that the more parameters of the paralinguistic presentation strategy match the user's preferences, the higher the reward (see below). Again, such a rule-based and deterministic user behavior is probably not what occurs in real-world interactions with the robot due to different types of noise. Similar to section 14.2, *user noise* and *sensor noise* are introduced to randomize the simulation to a certain degree (see section 12.3.1.3, section 12.4.1 and section 12.4.2). In contrast to section 14.2, both types of noise directly influence the reward in the implemented simulation at hand.

Algorithm 4: Reward calculation for the joke-telling simulation.

Input: user noise probability $n_u \in [0;1)$, sensor noise probability $n_s \in [0;1)$, user preferences c_u, robot's current configuration c_r
Output: reward value

$R \longleftarrow 0.0$
if $c_r[PITCH] = c_u[PITCH]$ **then** // *pitch matches user prefs?*
  $R \longleftarrow R + 0.25$
if $c_r[RATE] = c_u[RATE]$ **then** // *speech rate matches user prefs?*
  $R \longleftarrow R + 0.25$
if $c_r[VOLUME] = c_u[VOLUME]$ **then** // *volume matches user prefs?*
  $R \longleftarrow R + 0.25$
if $c_r[BREAK] = c_u[BREAK]$ **then** // *break matches user prefs?*
  $R \longleftarrow R + 0.25$
$r_u \longleftarrow$ a random float in $[0;1)$
if $r_u < n_u$ **then** // *random user reaction?*
  $r_{\text{tmp}} \longleftarrow$ a random float in $[-1;1]$
  $R \longleftarrow R + r_{\text{tmp}}$
$r_s \longleftarrow$ a random float in $[0;1)$
if $r_s < n_s$ **then** // *sensor noise?*
  $r_{\text{tmp}} \longleftarrow$ a random float in $[-1;1]$
  $R \longleftarrow R + r_{\text{tmp}}$
$R \longleftarrow \min(\max(R, 0.0), 1.0)$
return R

Algorithm 4 details the reward calculation. Both user and sensor noise are configured as probabilities $n_u \in [0;1)$ and $n_s \in [0;1)$. They are constant during each simulation run. In each run, the reward is initialized to 0.0. Then, the current configuration of the humor markers in terms of pitch, speech rate, volume, and break duration is compared to the simulated user's preferences. For each matching feature, the reward is increased by 0.25. With a total of four matching features, the maximum possible reward is 1.0, which is interpreted as making a real user very amused because of a successful joke presentation.

User noise, which occurs with probability n_u, adds a random value in the interval $[-1;1]$ to the reward. Semantically, this imitates a real user's amusement being biased

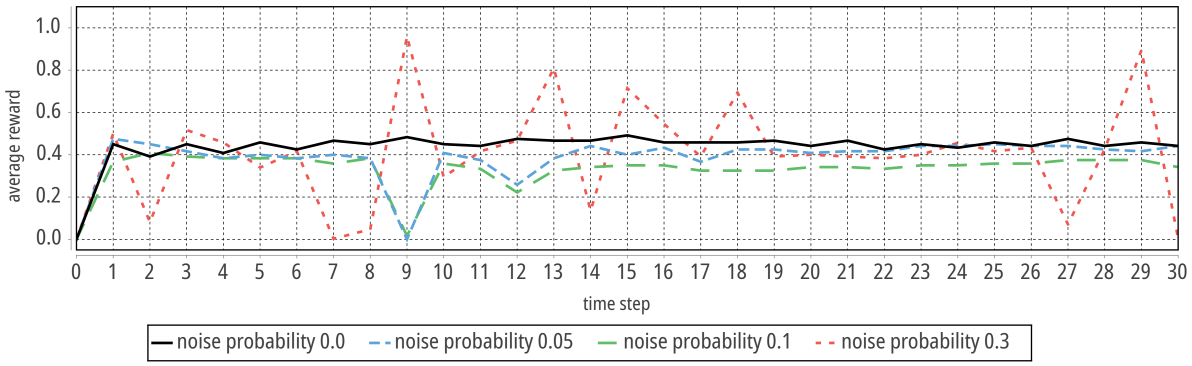


Figure 14.20.: Simulation results.

and this bias not being correlated with the learning agent's actions, which might occur in reality every once in a while. Sensor noise is implemented analogously, which translates to the SSP failing to estimate the real user's amusement correctly. Noise also manipulates the reward by adding a random number. The resulting reward value is trimmed to the interval [0; 1], with 0.0 being the smallest and 1.0 being the highest possible reward.

14.4.2. Results

The average reward (see section 2.7.1) is used as an introspective measurement to evaluate the self-motivated goal of the adaptation process (see section 12.2.4). It is calculated for all 30 users to illustrate the agent's performance. The plot in Figure 14.20 averages over 30 trials, each consisting of 30 steps. These numbers are analogous to a study with 30 participants, with the robot telling 30 jokes to each of them. The agent has no initial knowledge, and the learned policy resets between trials. Performance is evaluated for 0 % (baseline), 5 %, 10 %, and 30 % of both user and sensor noise, which randomizes the reward as described in section 14.4.1. The simulation uses the same parameter values as the interactive prototype (see section 14.3.3.4).

Without noise, learning results in a pretty stable reward of about 0.5. With increasing noise, the overall performance decreases as expected. On average, the reward is still very similar to the baseline most of the time, indicating that the learning approach can cope with noise. The initial random seed is the same for each noise probability experiment.

14.5. Evaluation: Lab Study

In order to explore the performance of the adaptation approach in real HRI, a within-subjects user study was conducted after the initial simulation in the lab. The evaluation focussed on the first learning agent design using the German language. Concerning the second agent design that learns the robot's use of humor markers, a pilot study revealed that German native speakers had problems understanding generated jokes. According to most participants, the English humor by the STANDUP joke generator was difficult to understand in terms of semantic content and vocabulary but also due to the pronunciation of the TTS system.

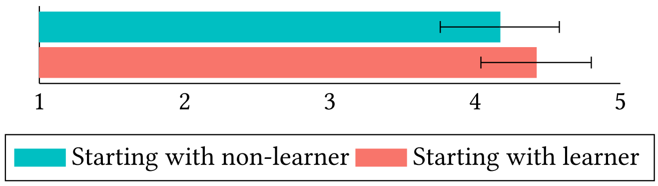


Figure 14.21.: Participants' self-rated sense of humor from non-humorous (1) to humorous (5). Error bars denote 95 % CI.

Two conditions were prepared:

1. a *learning* robot, which chooses jokes by learning from the user's reactions, i.e., social signals, and

2. a *non-learning* robot, which uses a random function to choose the next joke (baseline).

The performance of both robots was compared. Introspective and subjective measurements (see also section 6.1.4) were used to get insights into the agent's performance and user experience. In terms of subjective measurements, the study explored (1) if users laugh more when being entertained by a learning robot and (2) if users perceive a learning robot as more or less entertaining than a non-learning robot.

14.5.1. Participants, Apparatus, and Procedure

24 participants (12 female and 12 male), aged from 18 to 29 ($M = 22.04$, $SD = 3.25$) were recruited from a university campus. All participants were students. The study started by welcoming participants and providing an introduction to the user study. To this end, participants were handed out a short description, which informed them about the general study procedure, and a short questionnaire to report their self-perceived sense of humor. Moreover, participants were informed that there would be two sessions, one after the other, in which they would be asked to listen to a robot telling jokes. After each session, they would be asked to provide feedback about their general experience based on a questionnaire. The interaction was as described in section 14.3.1: the participant sat opposite the robot and was recorded with a webcam and a headset microphone for SSP (see Figure 14.11 and section 14.3.2). The setup was identical in both sessions.

Participants were told that the two sessions were different in terms of the program version uploaded to the robot. However, participants were not informed that they would experience the robot once telling random jokes and once telling jokes based on their reactions. At the end of all sessions, participants were asked which version of the robot they preferred overall.

In both sessions, 25 actions were executed, which took around 10–15 minutes. In both conditions, the robot measured the reward of every performed action as described in section 14.3.3.3. While the reward was only used in the *learner* condition for joke selection, the data was saved for both sessions for comparison. The order of the sessions was counter-balanced, with half of the participants starting with the *learner* version of the robot and the other half starting with the *non-learner* version of the robot. The above questionnaire revealed an almost equal rated sense of humor as depicted in Figure 14.21.

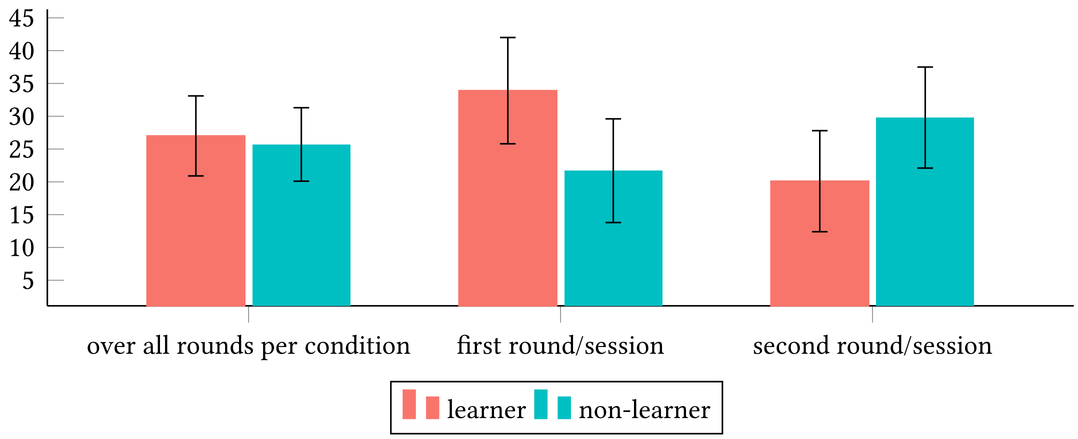


Figure 14.22.: Average reward of both sessions/amusement level of participants. Error bars denote 95 % CI.

14.5.2. Results

Figure 14.22 presents a frequency plot of the participants' amusement levels separated by condition, which shows no obvious difference over all collected output measures (i.e., amusement level/reward). However, when frequencies are separated by round, the following can be observed: in round one, participants were more amused by the learning robot, and in round two, participants were more amused by the non-learning robot.

One can also observe an interaction between the first and second round/session and between the *learner* and *non-learner* conditions. On the one hand, participants, who experienced the learning version of the robot, kept being amused even when, in the second session, the same robot told randomly selected jokes. On the other hand, participants, who started to experience the robot telling randomly chosen jokes, kept feeling less amused in the second session when the robot started to adapt its joke preferences to the user's reaction. Thus, a carry-over effect can be observed in the data addressed in the following statistical analysis.

14.5.2.1. Statistical Analysis

Since repeated measurements were collected from each participant, a paired *Student's t-test* was performed. As expected after exploring the data plot in Figure 14.22, it shows that the *learner* vs. *non-learner* condition does not affect participants' amusement levels over all collected data ($M = 0.0123$, $SE = 0.0157$), $t(23) = 1.714$, $p = 0.022$ (Field and Hole, 2003).

However, a significant effect occurs when focusing on evaluating round one, without including the results of round two, to avoid the ordering effect. The group is divided into a control group (users having watched the non-learning version first) and an experimental group (users having watched the learning version first) with 12 participants each. Using the independent *Student's t-test* shows a significant effect that is also substantial ($p = 0.02$, $d = 0.87$, see Figure 14.22) regarding the gained reward. Comparing the averaged amusement level over both rounds (see Figure 14.23) also shows an overall significant difference ($p = 0.03$, $d = 0.83$) between those two groups, including the second round.

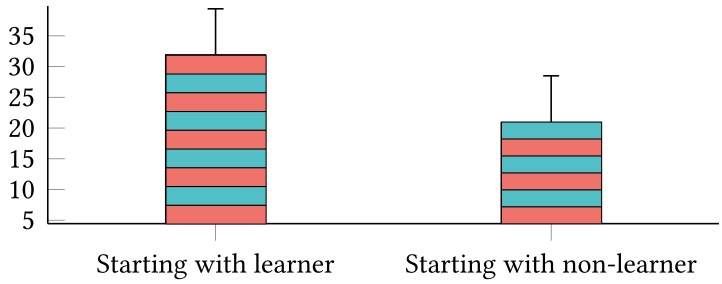


Figure 14.23.: Participants' amusement level (reward) averaged over both rounds. Error bars denote 95 % CI.

14.5.2.2. User Preferences

At the end of the study, participants were explicitly asked which version of the robot they preferred overall. Participants did not know how the two versions of the robot differed. Most participants preferred the version they experienced last: 54 % preferred the robot from the second round, 25 % preferred the robot from the first round, and the rest could not tell which version they preferred. Figure 14.24 provides an overview of self-reported data, which was utilized after each session to collect subjective data about how participants perceived the robot's performance.

14.5.3. Discussion

Many participants preferred the robot they listened to in the last session. This result was somewhat unexpected since the order of the *learner* and *non-learner* conditions was counter-balanced to prevent ordering and carry-over effects. The participants' collected amusement level data, which is a non-subjective measure, explains why participants started to prefer the second version of the robot. Considering the participants' amusement level in each round and condition, Figure 14.23 shows clearly that participants' amusement levels did not change across rounds.

This observation suggests, in retrospect, that choosing a repeated measurement setup might have been naive since the interactions (i.e., telling jokes) and effects (i.e., positive emotions and amusement) have been of slow nature and difficult to change in short periods. There seems to be only a very small decrease in amusement level in the second round, independent of condition, which could be attributed to a novelty aspect disappearing in the second round.

Consequently, the carry-over effect was addressed by dropping the data collected in the second round and analyzing the remaining data as between-subjects data. It revealed that the condition had a main effect on the amusement level and that the robot achieved significantly higher levels of amusement in the learning condition than one that does not learn from its user's implicit feedback. This procedure eliminates any carry-over effects and demonstrates that a robot can successfully learn the users' humor preferences based on the proposed socially-aware RL approach. In the joke-telling scenario, this resulted in a significant increase in amusement level compared to the non-learning baseline condition.

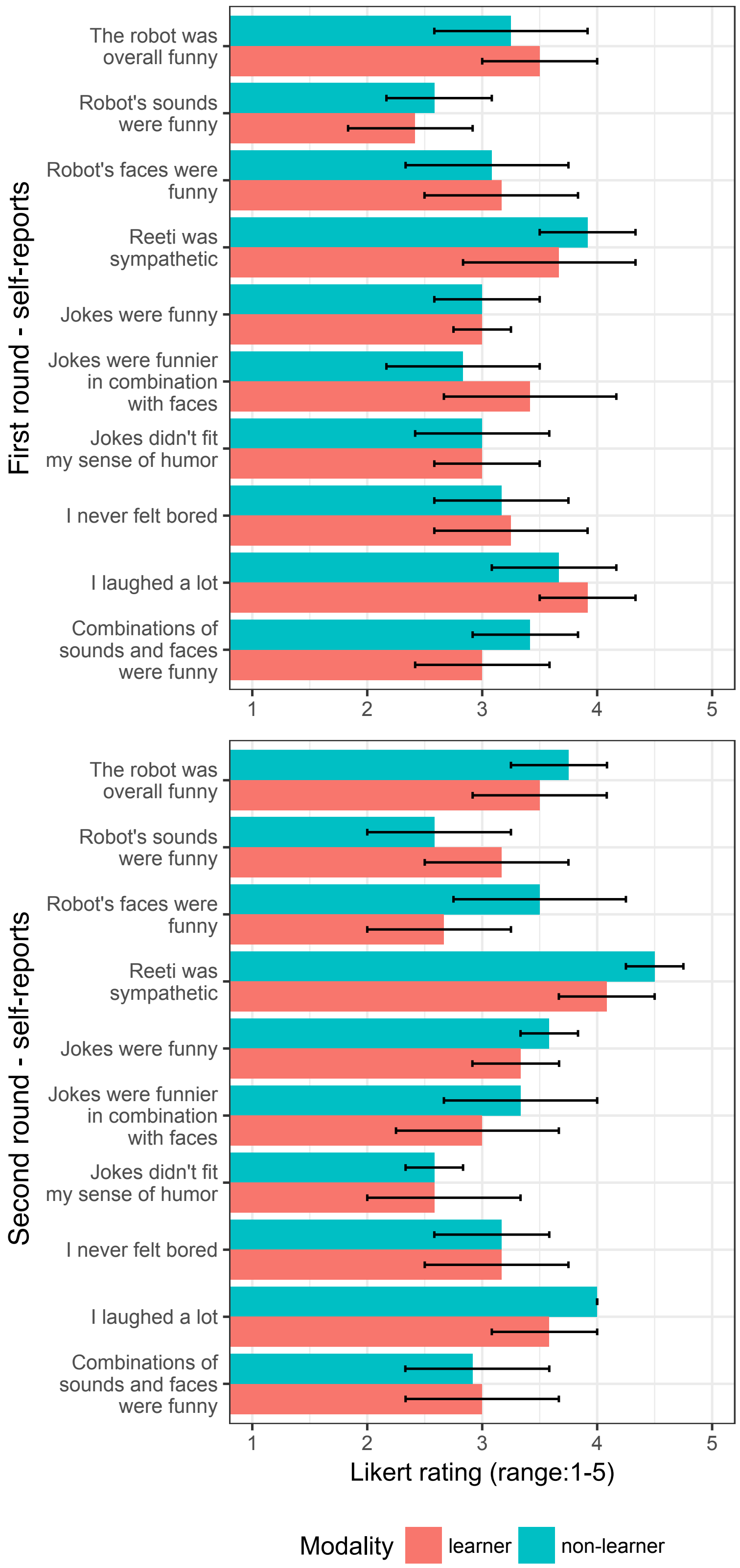


Figure 14.24.: Overview of participants' self-reports, ranging from strongly disagree (1) to strongly agree (5). Error bars denote 95 % CI.

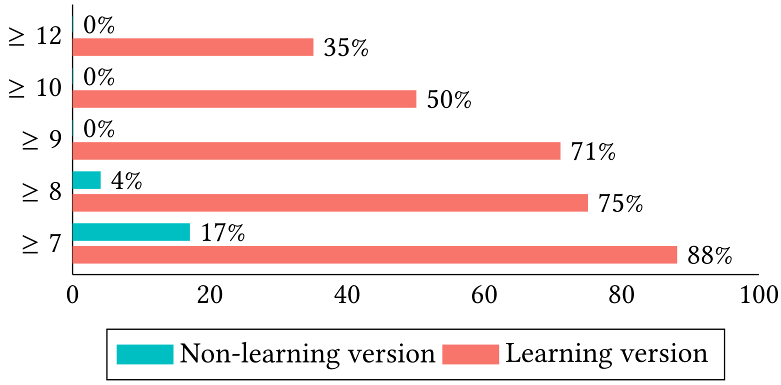


Figure 14.25.: The percentage of frequently chosen action categories.

14.5.3.1. The Importance of First Impressions

Figure 14.23 also shows that the amusement level was not only significantly influenced by the *learner* vs. *non-learner* condition (see Figure 14.22), but that the level stayed stable even when the robot adapted its humor presentation. This insight indicates that in a joke-telling scenario, the first impression of a robot is more important than having a robot adapt later on. It seems that the good impression in the first session left by the learning version of the robot made the users like the robot, and the following non-learning version was found almost equally good because of the first perceived impression. On the other hand, when participants experienced the non-learning version first, it seemed to have left a worse first impression. Although the robot changed its behavior and started to adapt to the users' preferences in the second session, the first impression seemed to have dominated the participants' opinions about the robot. This hypothesis is in line with our study data, which shows that in two out of three cases, the learning version made users laugh less in the second session (compared to only one out of six cases in the first session).

Consequently, applying the ability to adapt to human preferences can be in vain if users already have had a bad first impression of the robot, as mentioned above. It also shows that when trying to adapt to humans' preferences, the robot should not make too many mistakes and learn about their preferences as quickly as possible.

14.5.3.2. Learning and Reward Function

The SSP of both users' smiles and vocal laughter worked reliably during the experiment. Combining smile and vocal laughter performed well as a reward signal for the adaptation approach. As a result, the robot developed a tendency towards a specific action in contrast to the non-learning version based on the data collected during the study. For example, the percentage of action categories that were frequently chosen (more than seven times) was much higher for the learning robot (see Figure 14.25).

The crucial point is, however, to determine when to take measurements as an indicator of the user's level of amusement. As seen in Figure 14.18, the interval used for collecting the frames required for assessing the user's amusement was statically predefined by the punchline as suggested by Katevas, Healey, and Harris (2015). Sometimes, this interval did not optimally fit the user's reactions to a particular joke. Some users reacted before or after the interval, resulting in a lower reward than it could have been. In the worst case, they only reacted after the interval, when the new action was already in progress when it took them a while to understand the joke. Nevertheless, such shifts in expected reactions did not happen often, and this noise is likely overwritten in subsequent learning steps.

14.5.3.3. Benefits of Variation

The non-learning version of the robot was sometimes perceived as funnier, as also indicated by participants' preferences and self-reports (see Figure 14.24). For example, this can be seen for the items "The Reeti was sympathetic" and "Robot's faces were funny". This could be caused by the fact that the non-learning robot varies its actions more drastically due to its randomized behavior. Of course, this is not the goal of the learning robot, which continuously aims to determine which action the user prefers most. Consequently, this may result in less variation for the learning agent in shorter periods.

The collected data has shown that a robot should leave a good first impression by adapting to the user's sense of humor. It also indicated that later on, there is no harm in integrating variations. Apart from adaptation, additional moments of surprise and variety will also contribute to successful robot performance in a scenario that aims to entertain listeners.

14.6. Conclusion

This chapter presented another three models and experiments that implement the conceptual framework for non-functional social robot behavior adaptation from chapter 12. In contrast to chapter 13, the socially-aware and stateful RL processes use implicit human feedback so that the user does not need to provide an explicit assessment for each of the robot's actions. Instead, the reward signal is derived from human social signals automatically. User engagement and affect served as interaction dynamics and input to the adaptation process.

The presented socially-aware RL models differ in the design of their learning agents:

- The first model (section 14.1) adapts the robot's expressed extraversion in terms of generated linguistic content; the temporal progression of the user's engagement (which is estimated based on the user's pose) determines the reward. Here, the difference in the user's engagement from their engagement in the last step is interpreted as a reward signal, resulting in a positive/negative reward for increasing/decreasing user engagement. By including both the user state (engagement) and the configuration of the robot's extraversion in the state space, the model allows the agent to learn the dependencies between its communication style, the user's engagement, and the resulting effects.

- The second model (section 14.3) adapts the robot's content selection and combination of multimodal humor while the user's amusement determines the reward.

User amusement is estimated based on the probabilities of visual smiles and audible laughter observed during the robot's humor presentation. Similar to the first model, the inclusion of the user state (amusement) allows the agent to learn the dependencies between user amusement and the most effective communication style for the individual user.

- The third model (section 14.3) relies on the same reward signal as the second model. However, it uses dynamically generated behaviors for producing multimodal humor during the interaction. Here, the non-functional adaptation addresses the specific configuration of different modalities, including prosody and timing. Similar to the first model, the configuration of the robot's behaviors is part of the state space, which allows the agent to learn the dependencies between the user's amusement, the robot's use of multimodal cues, and the resulting effects.

The contributions of this chapter are the different models, implementations, simulations of the non-functional adaptation processes, and the insights of the evaluation, which report on the feasibility of the presented approaches. Furthermore, they illustrate the variety of application possibilities of the structured overview of socially-aware RL for non-functional robot adaptation from chapter 12. In particular, the contributions also include using real-time SSP techniques, which are fully integrated into the autonomous adaptation approaches. Apart from nondeterministic user behaviors (i.e., nonstationary problems), the simulations also address different types of noise in the context of SSP. Finally, the evaluation of the second model showed that a socially-aware RL process with implicit feedback achieves significantly higher levels of user amusement than one that does not learn from the user's implicit feedback.

Part V.

Conclusion

15. Contributions

This thesis presented and evaluated approaches for the real-time and non-functional adaptation of social robot behaviors. For reaching this goal, it contributed on the conceptual, technical, and empirical levels to two building blocks: behavior generation and adaptation. The following sections detail these contributions.

15.1. Conceptual Contributions

Based on an extensive literature review in Part II, the conceptual contributions cover models for behavior generation and adaptation in Part III and Part IV. Moreover, a comprehensive conceptual framework for implementing real-time socially-aware non-functional adaptation of social robot behaviors is one of the core contributions of this thesis. The following sections break down the conceptual contributions according to their focus on behavior generation or adaptation.

Generation of Multimodal Socially Intelligent Robot Behaviors

Social robot behavior generation is an essential part of the proposed adaptation approaches. Part III equips the robot with variable verbal and non-verbal behaviors, which an adaptation process can manipulate. They are provided as sets of single- or multimodal robot behaviors, relying on spoken language for verbal output and keyframe animations for non-verbal behaviors. The used robot hardware is relatively cheap, which makes it very affordable for the community replicating the proposed models. Since most social robots offer speech, gaze, and facial expressions, the proposed models for generating variable and socially-intelligent behaviors are also transferrable to many other social robots.

The models are grounded in the psychology literature from chapter 4. Several approaches for behavior generation were presented after thoroughly analyzing the literature in chapter 5, including rule-based dynamic generation approaches. Findings from human-human interaction were transferred to the robot. Their implementation enriches the research landscape with new techniques for robot behavior generation. Given the context of domestic companion robots, the thesis focused on the expression of robot personality, persona, politeness, and humor:

- In chapter 7, the dynamically generated expression of extraversion/introversion relied on a knowledge base and a traditional NLG pipeline approach with content planning, sentence planning, and surface realization. It allows the robot to generate descriptions of characters and the plot of a book with a configurable degree of extraversion/introversion.

- Chapter 8 generated different types of persona and politeness in the context of recommendations, information retrieval, and entertainment.
- Chapter 9 realized the generation of robot humor for the German and English languages. A particular contribution is the systematic multimodal augmentation of dynamically generated verbal humor with corresponding prosody and non-verbal robot behaviors.
- In addition, chapter 10 introduced a rule-based approach for dynamically generating multimodal robot irony based on NLP and NLG. It equips the robot with socially-intelligent behavior by reacting to the user's input and augmenting the spoken words with appropriate paralinguistic and non-verbal, irony-specific cues.

All models have been implemented successfully for the robot hardware at hand in Part III. These implementations provided a solid basis for the proposed non-functional adaptation approaches, allowing fine-grained manipulation during the interaction.

Real-Time Non-Functional Adaptation

The heart of the thesis is a structured overview and conceptual framework for real-time non-functional social robot behavior adaptation with RL in chapter 12. It describes and illustrates in detail a holistic view of the overall problem. It covers the general procedure of modeling, simulating, and evaluating user-adaptive robot behaviors based on the RL theory from chapter 2, the literature on user-adaptive interaction, and the analyzed related works from chapter 6. The conceptual overview provides researchers with guidance and an analysis tool for the design and implementation of user-adaptive and socially-aware social robot behaviors by pointing out key considerations:

- In the context of problem modeling, this thesis breaks down the connections between general and theoretic properties of RL problems, environments, algorithms, and their implications on models for real-world HRI scenarios. Moreover, this thesis presents and illustrates the mapping between the core aspects of RL models (state space, action space, reward, and learning algorithm) and non-functional robot adaptation, and dependencies and interrelations between these aspects. In that regard, a noteworthy contribution is the *Adaptation Triad*, which allows for getting an overview of the most important parts of the adaptation model at a glance by visualizing them in a circular diagram.
- Advantages and disadvantages, as well as specific challenges of simulations and human evaluations of adaptation models, are presented with their use and implications on the resulting experiment design. Different measures and their applications are listed in the different stages of simulating and evaluating a model.
- Apart from the general systematic breakdown of non-functional behavior adaptation with RL, the conceptual framework also provides details on the inclusion of the user in the adaptation process. In this context, another noteworthy contribution is the overview on *socially-aware RL*, which covers the inclusion of human social signals from chapter 3, related interaction dynamics, explicit and implicit feedback

in the RL loop. Potential design approaches are illustrated for use in state space and reward, as well as associated types of noise, including resulting algorithmic considerations.

Four adaptation models with explicit and implicit feedback implemented the conceptual framework in chapter 13 and chapter 14. These models also rely on the previously presented behavior generation approaches. The interaction scenarios are not goal-oriented, i.e., the robot cannot rely on measurable, task-related information for adaptation. Instead, it relies on explicit and implicit human feedback. Simulations, lab, and in-situ studies evaluated the following models with different complexity of behavior generation and adaptation:

- A stateless approach was chosen for estimating the user's preferences of the robot's expressed politeness and persona in chapter 13. The robot investigated different politeness strategies and sets of persona in different assistive and entertainment application contexts. The user provided explicit feedback via button presses on a hardware control panel.
- The adaptation of the robot's expressed extraversion/introversion in section 14.1 was realized based on implicit human feedback derived from sensed user engagement. The model combines user engagement and the robot's configuration of extraversion/introversion in the RL state space. The reward signal is based on the relative temporal progression of user engagement.
- In the first model for robot humor adaptation in section 14.3, the robot's multimodal configuration for behavior generation is part of the state space and manipulated by the learning agent's actions. It uses implicit feedback based on measured human amusement as the reward signal.
- The second model for robot humor in section 14.3 deviated from the first in the sense that only the user's amusement is part of the state space. The learning agent's actions trigger different randomized multimodal behaviors. Implicit feedback is based on measured human amusement and used as the reward signal.

The models illustrate the various implementations of the conceptual framework and its applicability for different use cases. The structured overview and models illustrate important considerations and implementation options. They serve as systematic guidance and inspiration for the research field.

15.2. Technical Contributions

Several technical contributions complement this thesis. All behavior generation and non-functional adaptation models were implemented for the robot at hand, resulting in conceptual and theoretic contributions and complete proofs of concept. Their implementations included verbal and non-verbal behaviors, technical simulations of the adaptation approaches, and real-time interactive research prototypes for conducting lab and in-situ studies.

Besides the technical implementation of the presented behavior generation and non-functional adaptation approaches, there were additional technical challenges whose resolutions are worth listing as standalone contributions. Listing all of them would go beyond the scope of this thesis, but the following overview briefly summarizes the most important ones:

- All proposed behavior generation approaches from Part III and non-functional adaptation approaches from Part IV were implemented for the robot hardware at hand. These implementations were the basis for the experiments, simulations, and user studies.
- Chapter 8 introduced an assistive robotic companion. It served as an autonomous evaluation platform for the in-situ study in chapter 13. It provides various assistance, entertainment, information retrieval, and communication applications. They were inspired by features of assistive robots in domestic environments as reported in the literature in chapter 5. The implementation targeted the in-situ evaluation of user-adaptive social robot behaviors in elderly users' domestic environments. Thus, it needed to cope with specific difficulties which arose from the participants' concerns about privacy, internet access, and more. The resulting platform is entirely autonomous and configurable to the individual participants' needs and wishes. It does not need any supervision or operator.
- The development of the *reeti-rest* software and API from chapter 11 was essential for realizing the proposed dynamically generated robot behavior approaches. Specifically, the software allows for dynamically generated keyframe animation with the parallel movement of all actuators and parallel playback of speech and sounds. Besides the robot's internal TTS software, the integration of additional third-party TTS software allows for more control over the robot's prosody. Moreover, the software also provides additional handy tools, such as a virtual robot for testing, pose and animation editors for authoring content, and remote access to essential robot settings, such as volume, audio playback, and more. The reeti-rest software was an essential contribution and a core technical component for all conducted experiments.

15.3. Empirical Contributions

The proposed behavior generation and non-functional adaptation approaches were evaluated in several experiments in Part III and Part IV. Results demonstrated that the implementations of the presented models work as expected and that a robot can adapt its behaviors to individual users. Moreover, the thesis reported insights into the user experience. Even though not all proposed adaptation approaches were evaluated in the lab or in-situ studies, working research prototypes and simulations constituted proof of concept for further investigation in future research.

A study in chapter 10 revealed that generated multimodal robot irony improved perceived social intelligence. Results have clearly shown that the irony generation approach worked well: participants were able to correctly identify their robotic conversation partner's use of irony and experienced associated humor, they associated a better user experience with the ironic robot, and overall, more participants preferred the ironic robot. Thus, the generation of robot humor has much potential for shaping the user's impressions of the robot since humor contributes to the expressed personality profile and perceived social intelligence.

This thesis conducted simulations for the proposed adaptation approaches from Part IV. Due to their nature, they do not provide insights into human user preferences or resulting impacts on user experience. However, they reveal the dependence of the learning agents' performance on different types of noise. A nondeterministic and nonstationary environment resulted from replacing the human with a simulated user of reduced complexity while introducing noise to the system. It allowed for parameter tuning and investigating the influence of different degrees of noise.

Afterward, user studies provided valuable insights into the proposed non-functional adaptation approaches and the underlying models. Chapter 13 presented an in-situ study with several elderly participants in their homes for one week. The evaluation results indicated that there is not one best or worst communication style for all users. Instead, a learning approach can be used to adapt the robot's politeness and persona to the users' liking. In addition, the lab study in chapter 14 evaluated the robot using adaptive humor. The results showed that the proposed socially-aware RL process with implicit feedback achieved significantly higher levels of user amusement than one that did not learn from the user's implicit feedback.

These insights indicate that socially-aware and non-functional adaptation of social robot behaviors is a promising direction for future research. The thesis illustrated that RL can be used for adapting generated verbal and non-verbal robot behaviors to individual users and that such adaptation can even result in improved user experience.

16. Future Work

The non-functional adaptation of social robot behaviors during interaction is still challenging in HRI. With robot hardware becoming more powerful and affordable, the time when social robots are part of our everyday life is getting closer and closer. Equipping robots with technology that makes them more expressive and appear more socially intelligent and understanding is a long-term goal for the future. This thesis suggested different approaches for multimodal behavior generation and corresponding adaptation based on RL. The following sections outline some limitations and perspectives for future research.

16.1. Non-Verbal Sounds

This thesis relied on verbal and non-verbal robot communication with speech, facial expression, and gaze since the Reeti robot has a limited set of actuators. The robot does not have extremities, and thus it cannot move or perform gestures. Of course, future work should also investigate the adaptation of the remaining modalities (see section 5.1.2). Besides using typical non-verbal communication channels, such as gestures, posture, movement, and proxemics (which often rely on mimicking human behaviors), many robots also have LED lights, displays, and sound playback as additional communication channels.

While lights and displays rely on visual contact, auditive communication with sounds is of specific interest for the future. It allows the machine to communicate complex information in a very short time without visual contact. In particular, robots can play back any sound: recorded samples (e.g., sounds from nature, human or animal sounds, and more), synthetic sounds (such as beeps), music, and much more. The state-of-the-art technology also allows for manipulating sounds on the fly with real-time effects for modifying pitch, adding reverb, distortion, and more. The same applies to polyphonic music, which can be created and played back with instruments and synthesizers. Unfortunately, the HRI literature has not extensively studied the use of non-verbal sounds (i.e., sounds that express messages in a short time independently of any particular language).

So far, the HRI literature primarily investigated the use of specific sound and music parameters (such as intonation, pitch, and timbre). For example, a typical use case is expressing the robot's internal state (e.g., different emotions, such as sadness, anger, fear, or joy) or intentions (e.g., affirmation or denial). One of the few works addressing the adaptation of robot sounds is Ritschel et al. (2019a), which presents an approach for shaping the robot's timbre based on an evolution strategy and explicit human comparative feedback. A set of musical themes with up to four voices is played back with a synthesizer. The evolution strategy manipulates the synthesizers' parameters (sawtooth and sine wave oscillator amplitude, sustain level, release time, high pass and low pass filter) of each voice and the tempo. The user listens to two variants of the same music theme with different parameters and selects the preferred one. This process repeats several times. The authors found that almost all participants preferred the personalized robot sounds, which indicates that adapting a robot's sonic interaction design is a promising field for future research.

16.2. Neural Networks and Deep Learning

This thesis focused on rule-based approaches for dynamically generating multimodal social robot behaviors. For future work, it is essential to look at data-driven generation paradigms, where neural networks are trained on large data corpora to synthesize stylistic variants of verbal and non-verbal behaviors automatically. These techniques are of specific interest for producing various dynamically generated verbal and non-verbal robot behaviors for expressing personality, persona, politeness, and humor. The literature uses neural network techniques for generating texts and animations that portray certain personality aspects.

Modern dialog systems use neural networks to generate an answer to arbitrary user input. For example, Nguyen, Morales, and Chin (2018) train a chatbot on TV show transcripts and movie dialogs to learn how to map user utterances on chatbot responses. They use an encoder to process user utterances and a decoder to produce the chatbot answer. Results of the evaluation show that the chatbot can generate answers that portray certain aspects of personality. Oraby et al. (2018) investigate how to train a neural model for task-based dialog that ensures not only stylistic variation but also semantic fidelity. The PERSONAGE NLG system by Mairesse and Walker (2010) generates a sufficient amount of training material, which consists of semantic representations of dialog acts and matching language output with different personality profiles. Experiments with various encoder-decoder setups illustrate the benefit of neural architectures for controlling stylistic variants in a task-based dialog.

For non-verbal behaviors, neural network approaches generate stylistic variants for character animations. A particular motion style modulates how a character is perceived and which emotion, mood, and personality are ascribed to it. Neural animation generation approaches have become popular for virtual agents, e.g., for animating a virtual violinist's hand (Kim, Cordier, and Magnenat-Thalmann, 2000), synthesizing emotional expressions of 3D faces (Hong, Wen, and Huang, 2002), or for producing appropriate gestures for a 3D skeleton for a given text (Kucherenko et al., 2020). Smith et al. (2019) train three separate networks for pose, timing, and foot contact to enable real-time style transfer. Style transfer helps reduce the amount of data, including combinations of heterogeneous actions and motion variants required to achieve character animations of sufficient quality that enhance the character's expressivity. Even though the work by Smith et al. focuses on a broader range of affective and non-affective components, the approach bears great promise for generating motion style variants that portray a character's personality. In the context of social robots, neural animation allows, e.g., for generating co-speech gestures for the NAO robot (Yoon et al., 2019) and the generation of affective robot movements (Suguitan, Bretan, and Hoffman, 2019).

Beyond the generation of robot behaviors, neural networks and deep learning are essential technologies for state-of-the-art SSP. Estimating different interaction dynamics (see section 12.3.4) allows for acquiring information about the user, which, in turn, serves as input to the adaptation process. Automated approaches for sensing and interpreting multimodal human social signals made massive improvements in recent years. For example, Lee et al. (2018) present a deep learning framework in the context of robot-assisted autism therapy with the NAO robot. It uses social signals with video (facial expressions, head movements, body movements, pose, and gestures), audio, and biosignals (heart rate, electrodermal activity, and body temperature) for estimating children's affective state (valence and arousal) and engagement. The trained framework allows the robot to adapt its perception of the multimodal input to different cultures (Asia and Europe) and individuals (based on the children's demographics, behavioral assessment scores, and individual characteristics), which allows for personalizing the interaction during the robot-assisted therapy process.

Apart from the use of RL for real-time adaptation during the interaction (as done in this thesis), the literature furthermore combines RL with deep learning approaches for automated training and optimization, e.g., in the context of SSP. For example, Zhang et al. (2018) present an approach for the automated design of machine learning architectures without manual feature-engineering and refinement in the context of infant vocalization analysis. Instead, an RL agent controls the design of different deep recurrent neural network structures, which are automatically instantiated, trained, and evaluated, thus automating the search for the best architecture without human interaction. In the context of speech recognition, Rajapakshe et al. (2022) introduce a RL policy for emotion recognition. The presented "Zeta policy" is based on deep RL for recognizing the emotions happy, sad, angry, and neutral in speech audio. Such technologies are essential for implementing SSP and adaptation, as speech is a central communication channel, and deep learning techniques open up more and more opportunities for acquiring information about the user.

This thesis focused on model-free and value-based RL with tabular learning and linear function approximation algorithms. Deep RL is a promising direction for the implementation of adaptation of social robots and virtual agents in the future. For example, in the context of dialog systems, Galland, Pelachaud, and Pecune (2022) adapt a socially interactive agent's conversational strategy with deep RL in a simulated environment. Each simulated user has preferences regarding different dialog acts and engagement, which impact non-verbal behaviors and turn-taking status. The estimated engagement contributes to the reward signal: the reward signal contains task-based information and the social goal of maintaining user engagement to ensure a high-quality user experience. Other RL techniques, such as batch RL (Lange, Gabel, and Riedmiller, 2012) can also be combined with deep learning (Fujimoto et al., 2019). They allow for efficient use of collected training samples from online learning.

16.3. Long-Term Adaptation and Lifelong Learning

This thesis used lab and in-situ studies to evaluate the proposed adaptation approaches. One week was the longest for evaluating the adaptive domestic companion with politeness and persona in section 13.3. The *long-term* evaluation of autonomous adaptive social robots is an open research question (Martins, Santos, and Dias, 2019). Continuous adaptation over an extended period is important for exploring the long-term viability of adaptation approaches for social robots and building a relationship with the user. So far, social robots have been used for exploring long-term HRI (Leite, Martinho, and Paiva, 2013) and adaptivity in HRI (Ahmad, Mubin, and Orlando, 2017) in the context of health care and therapy, education, work environments, public spaces and domestic environments for several weeks or months, but rarely for more than a year. Social robot adaptation will ideally be a process that accompanies the user for the rest of their life so that the robot can evolve with the user. In this context, the term *long-term* focuses on evaluating and adapting robot behaviors over an extended period.

There is also another, more algorithmic perspective. When using machine learning for adaptation, algorithms and approaches are needed that learn continuously and, at best, can transfer their knowledge to new and unknown tasks that they did not learn to solve yet. Instead of learning each task in isolation, a learning agent should utilize experience from past similar problems. The term *lifelong machine learning* describes the general idea of improving the agent's knowledge and effectiveness by accumulating and retaining knowledge from previous learning tasks and applying it to future tasks (Chen and Liu, 2018). While there are related approaches, such as hierarchical RL, which decomposes the overall MDP in sub-tasks (e.g., Dietterich (2000) and Sutton et al. (2011)), there is a conceptual difference to lifelong machine learning. Traditional RL is limited to solving *one* problem in *one* environment. An agent trained for one problem cannot share its knowledge with another agent for another problem. Different problems must be solved separately. For example, the RL agents from Part IV are distinct agents. The agent learning about persona cannot apply its knowledge in the context of politeness, extraversion, or joke-telling, nor can the other agents apply their knowledge in the other contexts. In contrast, lifelong RL (Chen and Liu, 2018) approaches investigate the use of previous knowledge on similar tasks, e.g., for an optimal exploration strategy (Garcia and Thomas, 2019), state abstractions (Abel et al., 2018a) and policy and value transfer (Abel et al., 2018b). Recent work has also investigated the integration of human expert feedback in lifelong RL (Jacquin, Perez, and Boulard Masson, 2022).

In general, lifelong learning and personalization in long-term HRI (Irfan et al., 2022) is a promising field of research for the future. Using lifelong learning techniques for social robot behavior adaptation seems obvious since humans also adapt and learn based on knowledge transfer. For example, a piano player does not learn every finger movement from scratch when practicing a new piece – the dexterity and musical appreciation from already played pieces are applied and transferred to the new piece. Nevertheless, of course, both can be refined and extended. Similar applies to human social behaviors and is supposed to be the case for adaptive robots: while solving the problem of telling a joke is an achievement, the intelligent use and transfer of this ability in more complex social contexts would be the next step. Lifelong learning is also of relevance for SSP. For example, Ren et al. (2020) demonstrate that lifelong learning saves more memory space than conventional multi-task learning for multi-task sequential learning in the context of deep speech emotion recognition.

16.4. Entrainment

This thesis aimed to adapt the robot's behaviors to the user. It did not consider the potential effects of such an adaptation process. For example, the user may notice the robot's adaptation and, thus, react by changing their behaviors, which could interfere with the robot's goal.

Entrainment (Benus, 2014) is "the tendency of interlocutors to become similar to each other in various aspects of their verbal and non-verbal behavi[o]r". According to Benus, several studies report this phenomenon across many languages and cultures in human-human spoken interaction. Speakers adapt their linguistic and paralinguistic behaviors *to the behaviors of their dialog partners*[1]. Human verbal and non-verbal behaviors are reported to be adapted, including overall structural linguistic features (e.g., syntactic structures or rules, lexical choice), acoustic-prosodic features (e.g., pitch, energy, speech rate), conversational patterns in turn-taking management (e.g., the duration for coordination of conversation partners) and other non-verbal behaviors (e.g., posture, mannerisms, facial expressions) (Benus, 2014).

Benus points out that entrainment is highly relevant in social robotics since – in human-human interaction – people who entrain their communication style to their interlocutors are perceived as more socially attractive, competent, likable, and intimate. Positive effects of entrainment have also been observed in HCI and HRI in both directions (user entrains to the machine and machine entrains to the user) (Benus, 2014). Thus, the adaptation of the robot's communication style has great potential.

The machine learning approaches in this thesis do not aim to implement robot entrainment, but robot entrainment might be a result of these adaptation processes. For example, the motivation for manipulating the robot's expressed extraversion was the variety of findings for preferred personality profiles in different contexts and scenarios in the literature with no consensus on similarity or complementarity attraction. However, adapting the robot's expressed personality could result in the robot entraining to the user if the robot learns to express a similar personality profile to the user.

An important question for future experiments is whether human entrainment may counteract or interfere with adaptation approaches similar to those presented in this thesis and how human entrainment should be taken into account from the machine learning and algorithmic perspective. As soon as the robot manipulates its behaviors, this could result in the human entraining to the robot. In turn, the robot may adapt to the already entrained user, and so on. The RL approaches in this thesis do not aim to entrain to the user (e.g., by mirroring), and the reward signal focuses on other interaction dynamics (e.g., the robot aims to maximize engagement or amusement). Thus, the approaches should be rather independent of human entrainment. However, a closer examination is necessary to investigate human entrainment and its impact on social robot behavior adaptation approaches based on RL.

[1] This is an essential distinction to the thesis at hand, where *adaptation* is not meant in terms of coordination, convergence, mirroring, synchrony and the like.

16.5. Ethical and Privacy Considerations

Using human data for adaptation raises ethical and privacy questions. Thus, this section outlines points of contact with ethical machine learning (Quadrianto, Schuller, and Lattimore, 2021) and the need for good practices regarding an automated adaptation of social robot behaviors.

First, an important question is whether the user is informed about the process of machine learning and behavior adaptation being applied by the robot. One might argue that many everyday technologies adapt to their users. For example, virtual keyboards on smartphone displays optimize the touch point to the individual user's fingers and typing. They also learn vocabulary and sentence fragments based on the user's input to reduce errors and increase efficiency. However, the device may also upload such data to cloud services – with the declaration of consent being deeply hidden in lengthy terms and conditions. For "social" robots, which already suggest by their embodiment and humanoid behaviors that they aim to represent social entities, it should be a matter of course to be transparent to their users on how they process the user's data.

From the perspective of adaptation, this thesis includes human data primarily in two ways in the adaptation process:

1. the acquisition of human feedback to train the machine, and
2. the use of collected feedback for decision making, which impacts subsequent robot behaviors and, consequently, the resulting user experience.

Especially in domestic environments, the former raises privacy concerns depending on the feedback modality. Section 6.3 introduced explicit and implicit feedback, differentiating whether the user is aware of giving feedback. Of course, relying on implicit feedback may be convenient but invasive when cameras or microphones need to record the interaction for SSP. Today's commercial voice assistants address this problem, e.g., by streaming data to the cloud only after being activated with a specific wake word and lighting up LEDs to communicate to the user that they are "listening". Such signals will also be needed in the context of social robot adaptation, even though the robot's embodiment might already suggest that the machine can "understand" the user. Nevertheless, transparency should be first priority, making clear *when* SSP is applied, *how*, *where*, and *why* it is processed. Hiding this information from the user will damage the user's privacy and be ethically questionable. Thus, this thesis processed all data locally on the machine connected to the robot, without sharing any data with third-party cloud services and giving participants control over which of the robot's applications they want to use and which not. Ethical guidelines and good practices for processing sensitive data for machine learning exist, e.g., in Batliner, Hantke, and Schuller (2022), who focus on speech data.

The latter is also worthy of discussion. Using human data for decision-making might sound straightforward, as long as the user is informed and agrees to this process. However, such an adaptation process might also result in manipulation. First, a RL agent always pursues an objective encoded in the system designer's reward function. Only by optimizing its behavior, the agent reaches the goal most efficiently. However, is the encoded goal also crucial for the specific user and their expectations about the interaction?

Moreover, RL is an online adaptation mechanism. It is the nature of RL that one never knows how the user would have reacted if the robot performed another action. Online learning during interaction includes the risk of missing a better opportunity or negatively impacting user experience due to suboptimal action selection, such as when performing exploration. Even small behavioral mistakes may permanently damage the relationship between the user and the robot.

In contrast to a RL agent, human behaviors often rely on cooperation and a degree of “give and take”: humans defer their own needs or interest to do some good to their friends or partner. However, the nature of RL is *greedy* behavior: the agent optimizes its behavior to achieve a single goal – maximize long-term rewards. Thus, a careful design of the reward signal is needed, as are additional mechanisms to make the learning process a bit “more human”.

Finally, Sharkey and Sharkey (2010) point out that social robots can relieve the feeling of loneliness for the elderly, stimulate interactions with other people and increase their quality of life. Adaptation processes will possibly be helpful for this by further optimizing the user experience and addressing people’s needs and preferences. Nevertheless, André (2015) points out that an emotional binding to the machine could also result in emotional manipulation, exploitation, or dependency. She mentions that robots will professionalize their ability to empathize and thus give a human counterpart the illusion of compassionate individuals in the future. However, it is also questionable whether this is desirable. This process could be amplified by robots expressing social intelligence, such as conversational humor, and their adaptation to the user, though communicating only an illusion of real “intelligence”.

Despite these mixed perceptions, socially-aware adaptation allows the user to actively shape and participate in the concrete characteristics of their robot’s behaviors. Such mechanisms allow future improvement of user experience without deeper technical knowledge and effort, but – maybe – with a twinkle in the user’s eye.

Bibliography

Abbeel, Pieter and Andrew Y. Ng (2004). “Apprenticeship learning via inverse reinforcement learning”. In: *Machine Learning, Proceedings of the Twenty-first International Conference (ICML 2004), Banff, Alberta, Canada, July 4-8, 2004*. Ed. by Carla E. Brodley. Vol. 69. ACM International Conference Proceeding Series. ACM.

Abel, David, Dilip Arumugam, Lucas Lehnert, and Michael L. Littman (2018a). “State Abstractions for Lifelong Reinforcement Learning”. In: *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*. Ed. by Jennifer G. Dy and Andreas Krause. Vol. 80. Proceedings of Machine Learning Research. PMLR, pp. 10–19.

Abel, David, Yuu Jinnai, Sophie Yue Guo, George Dimitri Konidaris, and Michael L. Littman (2018b). “Policy and Value Transfer in Lifelong Reinforcement Learning”. In: *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*. Ed. by Jennifer G. Dy and Andreas Krause. Vol. 80. Proceedings of Machine Learning Research. PMLR, pp. 20–29.

Adams Jr, Reginald B and Robert E Kleck (2005). “Effects of direct and averted gaze on the perception of facially communicated emotion.” In: *Emotion* 5.1, p. 3.

Addo, Ivor D. and Sheikh Iqbal Ahamed (2014). “Applying affective feedback to reinforcement learning in ZOEI, a comic humanoid robot”. In: *The 23rd IEEE International Symposium on Robot and Human Interactive Communication, IEEE RO-MAN 2014, Edinburgh, UK, August 25-29, 2014*. IEEE, pp. 423–428.

Ahmad, Muneeb Imtiaz, Omar Mubin, and Joanne Orlando (2017). “A Systematic Review of Adaptivity in Human-Robot Interaction”. In: *Multimodal Technol. Interact.* 1.3, p. 14.

Akalin, Neziha and Amy Loutfi (2021). “Reinforcement Learning Approaches in Social Robotics”. In: *Sensors* 21.4, p. 1292.

Al-Taee, Majid A., Waleed Al-Nuaimy, Zahra J. Muhsin, and Ali Al-Ataby (2017). “Robot Assistant in Management of Diabetes in Children Based on the Internet of Things”. In: *IEEE Internet of Things Journal* 4.2, pp. 437–445.

Altmeyer, Maximilian, Pascal Lessel, and Antonio Krüger (2018). “Investigating Gamification for Seniors Aged 75+”. In: *Proceedings of the 2018 on Designing Interactive Systems Conference 2018, DIS 2018, Hong Kong, China, June 09-13, 2018*. Ed. by Ilpo Koskinen, Youn-Kyung Lim, Teresa Cerratto Pargman, Kenny K. N. Chow, and William Odom. ACM, pp. 453–458.

Aly, Amir and Adriana Tapus (2013). “A model for synthesizing a combined verbal and nonverbal behavior based on personality traits in human-robot interaction”. In: *ACM/IEEE International Conference on Human-Robot Interaction, HRI 2013, Tokyo, Japan, March 3-6, 2013*. Ed. by Hideaki Kuzuoka, Vanessa Evers, Michita Imai, and Jodi Forlizzi. IEEE/ACM, pp. 325–332.

Aly, Amir and Adriana Tapus (2016). “Towards an intelligent system for generating an adapted verbal and nonverbal combined behavior in human-robot interaction”. In: *Auton. Robots* 40.2, pp. 193–209.

Amershi, Saleema, Maya Cakmak, W. Bradley Knox, and Todd Kulesza (2014). “Power to the People: The Role of Humans in Interactive Machine Learning”. In: *AI Magazine* 35.4, pp. 105–120.

Amin, Miriam and Manuel Burghardt (Dec. 2020). “A Survey on Approaches to Computational Humor Generation”. In: *Proceedings of the The 4th Joint SIGHUM Workshop on Computational Linguistics for Cultural Heritage, Social Sciences, Humanities and Literature*. Online: International Committee on Computational Linguistics, pp. 29–41.

André, Elisabeth (2015). “Empathische Reaktionen und ihre Modellierung im Computer”. In: *Das soziale Gehirn*. Ed. by Helmut Fink/Rainer Rosenzweig. Eisenbahnstraße 11, 48143 Münster, Germany: mentis Verlag GmbH, pp. 55–70.

Andrist, Sean, Bilge Mutlu, and Adriana Tapus (2015). “Look Like Me: Matching Robot Personality via Gaze to Increase Motivation”. In: *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems, CHI 2015, Seoul, Republic of Korea, April 18-23, 2015*. Ed. by Bo Begole, Jinwoo Kim, Kori Inkpen, and Woontack Woo. ACM, pp. 3603–3612.

Archakis, Argiris, Maria Giakoumelou, Dimitris Papazachariou, and Villy Tsakona (2010). “The prosodic framing of humour in conversational narratives: Evidence from Greek data”. In: *Journal of Greek Linguistics* 10.2, pp. 187–212.

Argyle, Michael and Brian R. Little (1972). “Do personality traits apply to social behaviour?” In: *Journal for the Theory of Social Behaviour* 2.1, pp. 1–35.

Aria agent.eu (2021). *Noxi DB*. http://noxi.aria-agent.eu. Accessed: 2021-04-14.

Arias, Enrique Alberto (2001). *Comedy in music: a Historical Bibliográphical resource guide*. Greenwood Publishing Group.

Attardo, Salvatore (2000a). “Irony as relevant inappropriateness”. In: *Journal of Pragmatics* 32.6, pp. 793–826.

Attardo, Salvatore (2000b). “Irony markers and functions: Towards a goal-oriented theory of irony and its processing”. In: *Rask* 12.1, pp. 3–20.

Attardo, Salvatore, Jodi Eisterhold, Jennifer Hay, and Isabella Poggi (2003). “Multimodal markers of irony and sarcasm”. In: *Humor* 16.2, pp. 243–260.

Attardo, Salvatore and Lucy Pickering (2011). “Timing in the performance of jokes”. In: *Humor-International Journal of Humor Research* 24.2, pp. 233–250.

Attardo, Salvatore, Lucy Pickering, and Amanda Baker (2011). “Prosodic and multimodal markers of humor in conversation”. In: *Pragmatics & Cognition* 19.2, pp. 224–247.

Attardo, Salvatore, Manuela Maria Wagner, and Eduardo Urios-Aparisi (2013). *Prosody and humor*. Vol. 55. John Benjamins Publishing.

Audrieth, Anthony L (1998). *The art of using humor in public speaking*. https://www.squaresail.com/auh.html. Accessed: 2022-01-25.

Bakker, Saskia and Karin Niemantsverdriet (2016). “The interaction-attention continuum: Considering various levels of human attention in interaction design”. In: *International Journal of Design* 10.2, pp. 1–14.

Barraquand, Rémi and James L. Crowley (2008). “Learning polite behavior with situation models”. In: *Proceedings of the 3rd ACM/IEEE international conference on Human robot interaction, HRI 2008, Amsterdam, The Netherlands, March 12-15, 2008*. Ed. by Terry Fong, Kerstin Dautenhahn, Matthias Scheutz, and Yiannis Demiris. ACM, pp. 209–216.

Bartl, Andrea, Stefanie Bosch, Michael Brandt, Monique Dittrich, and Birgit Lugrin (2016). “The Influence of a Social Robot’s Persona on How it is Perceived and Accepted by Elderly Users”. In: *Social Robotics - 8th International Conference, ICSR 2016, Kansas City, MO, USA, November 1-3, 2016, Proceedings*. Ed. by Arvin Agah, John-John Cabibihan, Ayanna M. Howard, Miguel A. Salichs, and Hongsheng He. Vol. 9979. Lecture Notes in Computer Science, pp. 681–691.

Batliner, Anton, Simone Hantke, and Björn W. Schuller (2022). “Ethics and Good Practice in Computational Paralinguistics”. In: *IEEE Trans. Affect. Comput.* 13.3, pp. 1236–1253.

Bauman, Richard (1986). *Story, performance, and event: Contextual studies of oral narrative*. Vol. 10. Cambridge University Press.

Baur, Tobias (2018). "Cooperative and transparent machine learning for the context-sensitive analysis of social interactions". PhD thesis. University of Augsburg, Germany.

Baur, Tobias, Dominik Schiller, and Elisabeth André (2017). "Modeling User's Social Attitude in a Conversational System". In: *Emotions and Personality in Personalized Services - Models, Evaluation and Applications*. Ed. by Marko Tkalcic, Berardina De Carolis, Marco de Gemmis, Ante Odic, and Andrej Kosir. Human-Computer Interaction Series. Springer, pp. 181–199.

bayesfusion.com (2021). *GeNIe Modeler – BayesFusion*. https://www.bayesfusion.com/genie/. Accessed: 2021-04-14.

Benus, Stefan (2014). "Social Aspects of Entrainment in Spoken Interaction". In: *Cogn. Comput.* 6.4, pp. 802–813.

Berry, Pauline M., Thierry Donneau-Golencer, Khang Duong, Melinda T. Gervasio, Bart Peintner, and Neil Yorke-Smith (2009). "Evaluating User-Adaptive Systems: Lessons from Experiences with a Personalized Meeting Scheduling Assistant". In: *Proceedings of the Twenty-First Conference on Innovative Applications of Artificial Intelligence, July 14-16, 2009, Pasadena, California, USA*. Ed. by Karen Zita Haigh and Nestor Rychtyckyj. AAAI.

Bethel, Cindy L. and Robin R. Murphy (2008). "Survey of Non-facial/Non-verbal Affective Expressions for Appearance-Constrained Robots". In: *IEEE Trans. Systems, Man, and Cybernetics, Part C* 38.1, pp. 83–92.

Binsted, Kim, Anton Nijholt, Oliviero Stock, Carlo Strapparava, G Ritchie, R Manurung, H Pain, Annalu Waller, and D O'Mara (2006). "Computational humor". In: *IEEE Intelligent Systems* 21.2, pp. 59–69.

Binsted, Kim and Graeme Ritchie (1997). "Computational rules for generating punning riddles". In: *HUMOR-International Journal of Humor Research* 10.1, pp. 25–76.

Bird, Christy (2011). "Formulaic jokes in interaction: The prosody of riddle openings". In: *Pragmatics & Cognition* 19.2, pp. 268–290.

blender.org (2021). *Home of the Blender project - Free and Open 3D Creation Software*. https://www.blender.org/. Accessed: 2021-04-14.

Bouzit, Sara, Gaëlle Calvary, Joëlle Coutaz, Denis Chêne, Éric Petit, and Jean Vanderdonckt (2017). "The PDA-LPA design space for user interface adaptation". In: *11th International Conference on Research Challenges in Information Science, RCIS 2017, Brighton, United Kingdom, May 10-12, 2017*. Ed. by Saïd Assar, Oscar Pastor, and Haralambos Mouratidis. IEEE, pp. 353–364.

Breazeal, Cynthia (2003). "Toward sociable robots". In: *Robotics and autonomous systems* 42.3-4, pp. 167–175.

Breazeal, Cynthia (2004). *Designing sociable robots*. MIT press.

Brooke, John (1996). "SUS - A quick and dirty usability scale". In: *Usability evaluation in industry* 189.194. Ed. by P.W. Jordan, B. Thomas, B. A. Weerdmeester, and A.L. McClelland, pp. 4–7.

Brown, Penelope and Stephen C. Levinson (Feb. 1987). *Politeness: Some Universals in Language Usage*. en. Cambridge University Press.

Buijzen, Moniek and Patti M. Valkenburg (2004). "Developing a Typology of Humor in Audiovisual Media". In: *Media Psychology* 6.2, pp. 147–167.

Busa-Fekete, Róbert and Eyke Hüllermeier (2014). "A Survey of Preference-Based Online Learning with Bandit Algorithms". In: *Algorithmic Learning Theory - 25th International Conference, ALT 2014, Bled, Slovenia, October 8-10, 2014. Proceedings*. Ed. by Peter Auer, Alexander Clark, Thomas Zeugmann, and Sandra Zilles. Vol. 8776. Lecture Notes in Computer Science. Springer, pp. 18–39.

Byrne, Donn (1997). “An overview (and underview) of research and theory within the attraction paradigm”. In: *Journal of Social and Personal Relationships* 14.3, pp. 417–431.

Byrne, Donn and William Griffitt (1969). “Similarity and awareness of similarity of personality characteristics as determinants of attraction.” In: *Journal of Experimental Research in Personality*.

Callejas, Zoraida, Birgit Lugrin, Jean-Claude Martin, Michael F. McTear, and Juliana Miehle (2021). “Adaptive Systems for Multicultural and Ageing Societies”. In: *Multimodal Agents for Ageing and Multicultural Societies : Communications of NII Shonan Meetings*. Ed. by Juliana Miehle, Wolfgang Minker, Elisabeth André, and Koichiro Yoshino. Singapore: Springer Singapore, pp. 1–20.

Canestrari, Carla (2010). “Meta-communicative signals and humorous verbal interchanges: A case study”. In: 23.3, pp. 327–349.

Cann, Arnie, Lawrence G Calhoun, and Janet S Banks (1997). “On the role of humor appreciation in interpersonal attraction: It’s no joking matter”. In: *Humor-International Journal of Humor Research* 10.1, pp. 77–90.

Carter, Judy (2010). *Stand-up comedy: The book*. Dell.

Carvalho, Paula, Luís Sarmento, Mário J. Silva, and Eugénio de Oliveira (2009). “Clues for Detecting Irony in User-generated Contents: Oh...!! It’s "So Easy" ;-)”. In: *Proceedings of the 1st International CIKM Workshop on Topic-sentiment Analysis for Mass Opinion*. TSA ’09. Hong Kong, China: ACM, pp. 53–56.

Cassell, Justine, Hannes Högni Vilhjálmsson, and Timothy W. Bickmore (2004). “BEAT: the Behavior Expression Animation Toolkit”. In: *Life-like characters - tools, affective functions, and applications*. Ed. by Helmut Prendinger and Mitsuru Ishizuka. Cognitive Technologies. Springer, pp. 163–186.

Çeliktutan, Oya and Hatice Gunes (2015). “Computational analysis of human-robot interactions through first-person vision: Personality and interaction experience”. In: *24th IEEE International Symposium on Robot and Human Interactive Communication, RO-MAN 2015, Kobe, Japan, August 31 - September 4, 2015*. IEEE, pp. 815–820.

cereproc.com (2021). *Academic Licensing | CereProc Text-to-Speech*. https://www.cereproc.com/en/products/academic. Accessed: 2021-04-14.

Chafe, Wallace (1994). *Discourse, consciousness, and time: The flow and displacement of conscious experience in speaking and writing*. University of Chicago Press.

Chartrand, Tanya L and John A Bargh (1999). “The chameleon effect: the perception–behavior link and social interaction.” In: *Journal of personality and social psychology* 76.6, p. 893.

Chen, Zhiyuan and Bing Liu (2018). *Lifelong Machine Learning, Second Edition*. Synthesis Lectures on Artificial Intelligence and Machine Learning. Morgan & Claypool Publishers.

Chiang, Yi-Shiu, Ting-Sheng Chu, Chung Dial Lim, Tung-Yen Wu, Shih-Huan Tseng, and Li-Chen Fu (2014). “Personalizing robot behavior for interruption in social human-robot interaction”. In: *2014 IEEE Workshop on Advanced Robotics and its Social Impacts, ARSO 2014, Evanston, IL, USA, September 11-13, 2014*. Ed. by Katsu Yamane and Jodi Forlizzi. IEEE, pp. 44–49.

Churamani, Nikhil, Pablo V. A. Barros, Erik Strahl, and Stefan Wermter (2018). “Learning Empathy-Driven Emotion Expressions using Affective Modulations”. In: *2018 International Joint Conference on Neural Networks, IJCNN 2018, Rio de Janeiro, Brazil, July 8-13, 2018*. IEEE, pp. 1–8.

Cignarella, Alessandra Teresa, Simona Frenda, Valerio Basile, Cristina Bosco, Viviana Patti, and Paolo Rosso (2018). “Overview of the EVALITA 2018 Task on Irony Detection in Italian Tweets (IronITA)”. In: *Proceedings of the Sixth Evaluation Campaign of Natural Language Processing and Speech Tools for Italian. Final Workshop (EVALITA 2018) co-located with the Fifth Italian*

Conference on Computational Linguistics (CLiC-it 2018), Turin, Italy, December 12-13, 2018. Ed. by Tommaso Caselli, Nicole Novielli, Viviana Patti, and Paolo Rosso. Vol. 2263. CEUR Workshop Proceedings. CEUR-WS.org.

Cope, Corrine S (1969). "Linguistic structure and personality development." In: *Journal of Counseling Psychology* 16.5p2, p. 1.

Cosar, Serhan, Manuel Fernández-Carmona, Roxana Agrigoroaie, Jordi Pagès, François Ferland, Feng Zhao, Shigang Yue, Nicola Bellotto, and Adriana Tapus (2020). "ENRICHME: Perception and Interaction of an Assistive Robot for the Elderly at Home". In: *Int. J. Soc. Robotics* 12.3, pp. 779–805.

Costa, Marco, Wies Dinsbach, Antony SR Manstead, and Pio Enrico Ricci Bitti (2001). "Social presence, embarrassment, and nonverbal behavior". In: *Journal of Nonverbal Behavior* 25.4, pp. 225–240.

Craenen, Bart G. W., Amol A. Deshmukh, Mary Ellen Foster, and Alessandro Vinciarelli (2018a). "Do We Really Like Robots that Match our Personality? The Case of Big-Five Traits, Godspeed Scores and Robotic Gestures". In: *27th IEEE International Symposium on Robot and Human Interactive Communication, RO-MAN 2018, Nanjing, China, August 27-31, 2018*. IEEE, pp. 626–631.

Craenen, Bart G. W., Amol A. Deshmukh, Mary Ellen Foster, and Alessandro Vinciarelli (2018b). "Shaping Gestures to Shape Personality: Big-Five Traits, Godspeed Scores and the Similarity-Attraction Effect". In: *Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems, AAMAS 2018, Stockholm, Sweden, July 10-15, 2018*. Ed. by Elisabeth André, Sven Koenig, Mehdi Dastani, and Gita Sukthankar. International Foundation for Autonomous Agents and Multiagent Systems Richland, SC, USA / ACM, pp. 2221–2223.

Cruz-Maya, Arturo and Adriana Tapus (2017). "Learning users' and personality-gender preferences in close human-robot interaction". In: *26th IEEE International Symposium on Robot and Human Interactive Communication, RO-MAN 2017, Lisbon, Portugal, August 28 - Sept. 1, 2017*. IEEE, pp. 791–798.

Culpeper, Jonathan (1996). "Towards an anatomy of impoliteness". In: *Journal of pragmatics* 25.3, pp. 349–367.

Damian, Ionut, Michael Dietz, and Elisabeth André (2018). "The SSJ Framework: Augmenting Social Interactions Using Mobile Signal Processing and Live Feedback". In: *Frontiers ICT* 5, p. 13.

Darwin, Charles (1965). *The expression of the emotions in man and animals*. University of Chicago press (Original work published 1872).

Dautenhahn, Kerstin (1998). "The Art of Designing Socially Intelligent Agents: Science, Fiction, and the Human in the Loop". In: *Applied Artificial Intelligence* 12.7-8, pp. 573–617.

Dautenhahn, Kerstin, Sarah Woods, Christina Kaouri, Michael L. Walters, Kheng Lee Koay, and Iain Werry (2005). "What is a robot companion - friend, assistant or butler?" In: *2005 IEEE/RSJ International Conference on Intelligent Robots and Systems, Edmonton, Alberta, Canada, August 2-6, 2005*. IEEE, pp. 1192–1197.

De Gorostiza Luengo, Francisco Javier Fernández, Fernando Alonso-Martín, Álvaro Castro-González, and Miguel Angel Salichs (2017). "Sound Synthesis for Communicating Nonverbal Expressive Cues". In: *IEEE Access* 5, pp. 1941–1957.

De Graaf, Maartje M. A. and Somaya Ben Allouch (2013). "Exploring influencing variables for the acceptance of social robots". In: *Robotics Auton. Syst.* 61.12, pp. 1476–1486.

De Valck, Marijke (2005). "THE SOUND GAG". In: *New Review of Film and Television Studies* 3.2, pp. 223–235.

Deaville, James and Agnes Malkinson (2014). “A laugh a second?: Music and sound in comedy trailers”. In: *Music, Sound, and the Moving Image* 8.2, pp. 121–140.

Den Hengst, Floris, Eoin Martino Grua, Ali el Hassouni, and Mark Hoogendoorn (2020). “Reinforcement learning for personalization: a systematic literature review”. In.

DeYoung, Colin G., Yanna J. Weisberg, Lena C. Quilty, and Jordan B. Peterson (2013). “Unifying the Aspects of the Big Five, the Interpersonal Circumplex, and Trait Affiliation”. In: *Journal of Personality* 81.5, pp. 465–475.

Dieterich, Hartmut, Uwe Malinowski, Thomas Kühme, and Matthias Schneider-Hufschmidt (1993). “State of the art in adaptive user interfaces”. In: *Human factors in information technology* 10, pp. 13–13.

Dietterich, Thomas G. (2000). “Hierarchical Reinforcement Learning with the MAXQ Value Function Decomposition”. In: *J. Artif. Intell. Res.* 13, pp. 227–303.

D’Mello, Sidney S, Patrick Chipman, and Art Graesser (2007). “Posture as a predictor of learner’s affective engagement”. In: *Proceedings of the Annual Meeting of the Cognitive Science Society*. Vol. 29. 29.

Dynel, Marta (2009). “Beyond a Joke: Types of Conversational Humour”. In: *Language and Linguistics Compass* 3.5, pp. 1284–1299.

Dynel, Marta (2014). “Isn’t it ironic? Defining the scope of humorous irony”. In: *Humor* 27.4, pp. 619–639.

Ekman, Paul (1993). “Facial expression and emotion.” In: *American psychologist* 48.4, p. 384.

Ekman, Paul (2009). “Lie catching and microexpressions”. In: *The philosophy of deception* 1.2, p. 5.

Ekman, Paul and Wallace V Friesen (1969). “The repertoire of nonverbal behavior: Categories, origins, usage, and coding”. In: *Nonverbal communication, interaction, and gesture*, pp. 57–106.

Ekman, Paul and Wallace V Friesen (1978). “Facial action coding system”. In: *Environmental Psychology & Nonverbal Behavior*.

Ekman, Paul and Wallace V Friesen (2010). *The repertoire of nonverbal behavior: Categories, origins, usage, and coding*. De Gruyter Mouton.

En, Looi Qin and See Swee Lan (2012). “Applying politeness maxims in social robotics polite dialogue”. In: *International Conference on Human-Robot Interaction, HRI’12, Boston, MA, USA - March 05 - 08, 2012*. Ed. by Holly A. Yanco, Aaron Steinfeld, Vanessa Evers, and Odest Chadwicke Jenkins. ACM, pp. 189–190.

Estroff, Sharon Duke and Stephen Nowicki Jr. (1992). “Interpersonal Complementarity, Gender of Interactants, and Performance on Puzzle and Word Tasks”. In: *Personality and Social Psychology Bulletin* 18.3, pp. 351–356.

Fast, Lisa A and David C Funder (2008). “Personality as manifest in word use: correlations with self-report, acquaintance report, and behavior.” In: *Journal of personality and social psychology* 94.2, p. 334.

Feil-Seifer, David and Maja J Mataric (2005). “Defining socially assistive robotics”. In: *9th International Conference on Rehabilitation Robotics, 2005. ICORR 2005.* IEEE, pp. 465–468.

Ferreira, Emmanuel and Fabrice Lefèvre (2015). “Reinforcement-learning based dialogue system for human-robot interactions with socially-inspired rewards”. In: *Comput. Speech Lang.* 34.1, pp. 256–274.

Field, A.P. and G. Hole (2003). *How to Design and Report Experiments*. Sage.

Flutura, Simon, Andreas Seiderer, Ilhan Aslan, Chi-Tai Dang, Raphael Schwarz, Dominik Schiller, and Elisabeth André (2018). “DrinkWatch: A Mobile Wellbeing Application Based on Interactive and Cooperative Machine Learning”. In: *Proceedings of the 2018 International Conference on Digital Health, DH 2018, Lyon, France, April 23-26, 2018*. Ed. by Patty Kostkova, Floriana

Grasso, Carlos Castillo, Yelena Mejova, Arnold Bosman, and Michael Edelstein. ACM, pp. 65–74.

Fong, Terrence, Illah R. Nourbakhsh, and Kerstin Dautenhahn (2003). “A survey of socially interactive robots”. In: *Robotics Auton. Syst.* 42.3-4, pp. 143–166.

Frenda, Simona (2016). “Computational rule-based model for Irony Detection in Italian Tweets”. In: *Proceedings of Third Italian Conference on Computational Linguistics (CLiC-it 2016) & Fifth Evaluation Campaign of Natural Language Processing and Speech Tools for Italian. Final Workshop (EVALITA 2016), Napoli, Italy, December 5-7, 2016*. Ed. by Pierpaolo Basile, Anna Corazza, Francesco Cutugno, Simonetta Montemagni, Malvina Nissim, Viviana Patti, Giovanni Semeraro, and Rachele Sprugnoli. Vol. 1749. CEUR Workshop Proceedings. CEUR-WS.org.

Frischen, Alexandra, Andrew P Bayliss, and Steven P Tipper (2007). “Gaze cueing of attention: visual attention, social cognition, and individual differences.” In: *Psychological bulletin* 133.4, p. 694.

Fujimoto, Scott, Edoardo Conti, Mohammad Ghavamzadeh, and Joelle Pineau (2019). “Benchmarking Batch Deep Reinforcement Learning Algorithms”. In: *CoRR* abs/1910.01708.

Furnham, Adrian (1990). “Language and personality.” In.

Gallaher, Peggy E (1992). “Individual differences in nonverbal behavior: Dimensions of style.” In: *Journal of personality and social psychology* 63.1, p. 133.

Galland, Lucie, Catherine Pelachaud, and Florian Pecune (2022). “Adapting conversational strategies to co-optimize agent’s task performance and user’s engagement”. In: *IVA ’22: ACM International Conference on Intelligent Virtual Agents, Faro, Portugal, September 6 - 9, 2022*. Ed. by Carlos Martinho, João Dias, Joana Campos, and Dirk Heylen. ACM, 23:1–23:3.

Gamborino, Edwinn and Li-Chen Fu (2018). “Interactive Reinforcement Learning based Assistive Robot for the Emotional Support of Children”. In: *2018 18th International Conference on Control, Automation and Systems (ICCAS)*, pp. 708–713.

Garcia, Francisco M. and Philip S. Thomas (2019). “A Meta-MDP Approach to Exploration for Lifelong Reinforcement Learning”. In: *Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent Systems, AAMAS ’19, Montreal, QC, Canada, May 13-17, 2019*. Ed. by Edith Elkind, Manuela Veloso, Noa Agmon, and Matthew E. Taylor. International Foundation for Autonomous Agents and Multiagent Systems, pp. 1976–1978.

Garivier, Aurélien and Eric Moulines (2008). “On upper-confidence bound policies for non-stationary bandit problems”. In: *arXiv preprint arXiv:0805.3415*.

Gatt, Albert and Ehud Reiter (2009). “SimpleNLG: A Realisation Engine for Practical Applications”. In: *ENLG 2009 - Proceedings of the 12th European Workshop on Natural Language Generation, March 30-31, 2009, Athens, Greece*. Ed. by Emiel Krahmer and Mariët Theune. The Association for Computer Linguistics, pp. 90–93.

Gatti, Lorenzo, Gözde Özbal, Oliviero Stock, and Carlo Strapparava (2017). “Automatic Generation of Lyrics Parodies”. In: *Proceedings of the 2017 ACM on Multimedia Conference, MM 2017, Mountain View, CA, USA, October 23-27, 2017*. Ed. by Qiong Liu, Rainer Lienhart, Haohong Wang, Sheng-Wei "Kuan-Ta" Chen, Susanne Boll, Yi-Ping Phoebe Chen, Gerald Friedland, Jia Li, and Shuicheng Yan. ACM, pp. 485–491.

Gebhard, Patrick, Gregor Mehlmann, and Michael Kipp (2012). “Visual SceneMaker - a tool for authoring interactive virtual characters”. In: *J. Multimodal User Interfaces* 6.1-2, pp. 3–11.

Gebhard, Patrick, Marc Schröder, Marcela Charfuelan, Christoph Endres, Michael Kipp, Sathish Pammi, Martin Rumpler, and Oytun Türk (2008). “IDEAS4Games: Building Expressive Virtual Characters for Computer Games”. In: *Intelligent Virtual Agents, 8th International Conference, IVA 2008, Tokyo, Japan, September 1-3, 2008. Proceedings*. Ed. by Helmut Prendinger, James

C. Lester, and Mitsuru Ishizuka. Vol. 5208. Lecture Notes in Computer Science. Springer, pp. 426–440.
Geist, Matthieu and Olivier Pietquin (2010). “Kalman Temporal Differences”. In: *J. Artif. Intell. Res.* 39, pp. 483–532.
Gifford, Robert (1991). “Mapping nonverbal behavior on the interpersonal circle.” In: *Journal of Personality and Social Psychology* 61.2, p. 279.
Gill, Alastair J and Jon Oberlander (2019). “Taking care of the linguistic features of extraversion”. In: *Proceedings of the Twenty-Fourth Annual Conference of the Cognitive Science Society*. Routledge, pp. 363–368.
Gill Woodall, W and Judee K Burgoon (1983). “Talking fast and changing attitudes: A critique and clarification”. In: *Journal of nonverbal behavior* 8.2, pp. 126–142.
Gironzetti, Elisa (2017). “Prosodic and multimodal markers of humor”. In: *The Routledge handbook of language and humor*, pp. 400–413.
Gironzetti, Elisa, Salvatore Attardo, and Lucy Pickering (2016). “Smiling, gaze, and humor in conversation: A pilot study”. In: *Metapragmatics of Humor: Current research trends* 14, p. 235.
Gironzetti, Elisa, Meichan Huang, Lucy Pickering, and Salvatore Attardo (Mar. 2015). “The Role of Eye Gaze and Smiling in Humorous Dyadic Conversations”. In.
github.com (2021). *GitHub - simplenlg/simplenlg: Java API for Natural Language Generation. Originally developed by Ehud Reiter at the University of Aberdeen’s Department of Computing Science and co-founder of Arria NLG. This git repo is the official SimpleNLG version.* https://github.com/simplenlg/simplenlg. Accessed: 2021-04-14.
Glenn, Phillip J (1989). “Initiating shared laughter in multi-party conversations”. In: *Western Journal of Communication (includes Communication Reports)* 53.2, pp. 127–149.
Gordon, Goren, Samuel Spaulding, Jacqueline Kory Westlund, Jin Joo Lee, Luke Plummer, Marayna Martinez, Madhurima Das, and Cynthia Breazeal (2016). “Affective Personalization of a Social Robot Tutor for Children’s Second Language Skills”. In: *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, February 12-17, 2016, Phoenix, Arizona, USA*. Ed. by Dale Schuurmans and Michael P. Wellman. AAAI Press, pp. 3951–3957.
Gratch, Jonathan, Jeff Rickel, Elisabeth André, Justine Cassell, Eric Petajan, and Norman I. Badler (2002). “Creating Interactive Virtual Humans: Some Assembly Required”. In: *IEEE Intell. Syst.* 17.4, pp. 54–63.
Griffith, Shane, Kaushik Subramanian, Jonathan Scholz, Charles L. Isbell Jr., and Andrea Lockerd Thomaz (2013). “Policy Shaping: Integrating Human Feedback with Reinforcement Learning”. In: *Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5-8, 2013, Lake Tahoe, Nevada, United States*. Ed. by Christopher J. C. Burges, Léon Bottou, Zoubin Ghahramani, and Kilian Q. Weinberger, pp. 2625–2633.
Grüneberg, Patrick and Kenji Suzuki (2014). “An Approach to Subjective Computing: A Robot That Learns From Interaction With Humans”. In: *IEEE Trans. Auton. Ment. Dev.* 6.1, pp. 5–18.
Guerini, Marco, Carlo Strapparava, and Oliviero Stock (2008). “Valentino: A Tool for Valence Shifting of Natural Language Texts”. In: *Proceedings of the International Conference on Language Resources and Evaluation, LREC 2008, 26 May - 1 June 2008, Marrakech, Morocco*. European Language Resources Association.
Hall, Edward Twitchell (1966). *The hidden dimension*. Vol. 609. Garden City, NY: Doubleday.
Hall, Judith A (1990). *Nonverbal sex differences: Accuracy of communication and expressive style.* Johns Hopkins University Press.
Hammer, Stephan, Birgit Lugrin, Sergey Bogomolov, Kathrin Janowski, and Elisabeth André (2016). “Investigating Politeness Strategies and Their Persuasiveness for a Robotic Elderly

Assistant”. In: *Persuasive Technology - 11th International Conference, PERSUASIVE 2016, Salzburg, Austria, April 5-7, 2016, Proceedings*. Ed. by Alexander Meschtscherjakov, Boris E. R. de Ruyter, Verena Fuchsberger, Martin Murer, and Manfred Tscheligi. Vol. 9638. Lecture Notes in Computer Science. Springer, pp. 315–326.

Haraguchi, Kazuki, Satoshi Aoki, Tomohiro Umetani, Tatsuya Kitamura, and Akiyo Nadamoto (2019). “Omotenashi Robots: Generating Funny Dialog Using Visitor Geographical Information”. In: *Advances in Networked-based Information Systems - The 22nd International Conference on Network-Based Information Systems, NBiS 2019, Oita, Japan, September 5-7, 2019*. Ed. by Leonard Barolli, Hiroaki Nishino, Tomoya Enokido, and Makoto Takizawa. Vol. 1036. Advances in Intelligent Systems and Computing. Springer, pp. 669–679.

Hassenzahl, Marc, Michael Burmester, and Franz Koller (2003). “AttrakDiff: Ein Fragebogen zur Messung wahrgenommener hedonischer und pragmatischer Qualität”. de. In: *Mensch & Computer 2003*. Ed. by Gerd Szwillus and Jürgen Ziegler. Vol. 57. Wiesbaden: Vieweg+Teubner Verlag, pp. 187–196.

Hassin, Ran and Yaacov Trope (2000). “Facing faces: studies on the cognitive aspects of physiognomy.” In: *Journal of personality and social psychology* 78.5, p. 837.

Hay, Jennifer (2001). “The pragmatics of humor support”. In.

Hayashi, Kotaro, Takayuki Kanda, Takahiro Miyashita, Hiroshi Ishiguro, and Norihiro Hagita (2008). “Robot *Manzai*: Robot Conversation as a Passive-Social Medium”. In: *I. J. Humanoid Robotics* 5.1, pp. 67–86.

Hee, Cynthia Van, Els Lefever, and Véronique Hoste (2018). “SemEval-2018 Task 3: Irony Detection in English Tweets”. In: *Proceedings of The 12th International Workshop on Semantic Evaluation, SemEval@NAACL-HLT 2018, New Orleans, Louisiana, USA, June 5-6, 2018*. Ed. by Marianna Apidianaki, Saif M. Mohammad, Jonathan May, Ekaterina Shutova, Steven Bethard, and Marine Carpuat. Association for Computational Linguistics, pp. 39–50.

Hong, Pengyu, Zhen Wen, and Thomas S. Huang (2002). “Real-time speech-driven face animation with expressions using neural networks”. In: *IEEE Trans. Neural Networks* 13.4, pp. 916–927.

Horowitz, Leonard M., Kelly R. Wilson, Bulent Turan, Pavel Zolotsev, Michael J. Constantino, and Lynne Henderson (2006). “How Interpersonal Motives Clarify the Meaning of Interpersonal Behavior: A Revised Circumplex Model”. In: *Personality and Social Psychology Review* 10.1, pp. 67–86.

Huijnen, Claire A. G. J., Atta Badii, Herjan van den Heuvel, Praminda Caleb-Solly, and Daniel Thiemert (2011). “"Maybe It Becomes a Buddy, But Do Not Call It a Robot" - Seamless Cooperation between Companion Robotics and Smart Homes”. In: *Ambient Intelligence - Second International Joint Conference on AmI 2011, Amsterdam, The Netherlands, November 16-18, 2011. Proceedings*. Ed. by David V. Keyson, Mary Lou Maher, Norbert A. Streitz, Adrian David Cheok, Juan Carlos Augusto, Reiner Wichert, Gwenn Englebienne, Hamid K. Aghajan, and Ben J. A. Kröse. Vol. 7040. Lecture Notes in Computer Science. Springer, pp. 324–329.

Irfan, Bahar, Aditi Ramachandran, Samuel Spaulding, German I. Parisi, and Hatice Gunes (2022). “Lifelong Learning and Personalization in Long-Term Human-Robot Interaction (LEAP-HRI)”. In: *Proceedings of the 2022 ACM/IEEE International Conference on Human-Robot Interaction*. HRI ’22. Sapporo, Hokkaido, Japan: IEEE Press, 1261–1264.

Isbister, Katherine and Clifford Nass (2000). “Consistency of personality in interactive characters: verbal cues, non-verbal cues, and user characteristics”. In: *Int. J. Hum. Comput. Stud.* 53.2, pp. 251–267.

Ishi, Carlos Toshinori, Ryusuke Mikata, and Hiroshi Ishiguro (2020). “Person-Directed Pointing Gestures and Inter-Personal Relationship: Expression of Politeness to Friendliness by Android Robots”. In: *IEEE Robotics Autom. Lett.* 5.4, pp. 6081–6088.

Izard, Carroll E (1991). *The psychology of emotions*. Springer Science & Business Media.

Jackson, Ryan Blake, Ruchen Wen, and Tom Williams (2019). “Tact in Noncompliance: The Need for Pragmatically Apt Responses to Unethical Commands”. In: *Proceedings of the 2019 AAAI/ACM Conference on AI, Ethics, and Society, AIES 2019, Honolulu, HI, USA, January 27-28, 2019*. Ed. by Vincent Conitzer, Gillian K. Hadfield, and Shannon Vallor. ACM, pp. 499–505.

Jackson, Ryan Blake, Tom Williams, and Nicole Smith (2020). “Exploring the Role of Gender in Perceptions of Robotic Noncompliance”. In: *HRI ’20: ACM/IEEE International Conference on Human-Robot Interaction, Cambridge, United Kingdom, March 23-26, 2020*. Ed. by Tony Belpaeme, James E. Young, Hatice Gunes, and Laurel D. Riek. ACM, pp. 559–567.

Jacquin, Thierry, Julien Perez, and Cécile Boulard Masson (Mar. 2022). “Human Influence in the Lifelong Reinforcement Learning Loop”. In: *Workshop on Lifelong Learning and Personalization in Long-Term Human-Robot Interaction (LEAP-HRI), HRI 2022*.

Jäncke, Lutz (1993). “Different facial EMG-reactions of extraverts and introverts to pictures with positive, negative and neutral valence”. In: *Personality and Individual Differences* 14.1, pp. 113–118.

Janowski, Kathrin, Hannes Ritschel, and Elisabeth André (2022). “Adaptive Artificial Personalities”. In: *The Handbook on Socially Interactive Agents: 20 Years of Research on Embodied Conversational Agents, Intelligent Virtual Agents, and Social Robotics Volume 2: Interactivity, Platforms, Application*. 1st ed. New York, NY, USA: Association for Computing Machinery, 155–194.

Johnson, W. Lewis, Richard E. Mayer, Elisabeth André, and Matthias Rehm (2005). “Cross-Cultural Evaluation of Politeness in Tactics for Pedagogical Agents”. In: *Artificial Intelligence in Education - Supporting Learning through Intelligent and Socially Informed Technology, Proceedings of the 12th International Conference on Artificial Intelligence in Education, AIED 2005, July 18-22, 2005, Amsterdam, The Netherlands*. Ed. by Chee-Kit Looi, Gordon I. McCalla, Bert Bredeweg, and Joost Breuker. Vol. 125. Frontiers in Artificial Intelligence and Applications. IOS Press, pp. 298–305.

Joosse, Michiel, Manja Lohse, Jorge Gallego Perez, and Vanessa Evers (2013). “What you do is who you are: The role of task context in perceived social robot personality”. In: *2013 IEEE International Conference on Robotics and Automation, Karlsruhe, Germany, May 6-10, 2013*. IEEE, pp. 2134–2139.

Jung, Soyoung, Hyoung taek Lim, Sanghun Kwak, and Frank A. Biocca (2012). “Personality and facial expressions in human-robot interaction”. In: *International Conference on Human-Robot Interaction, HRI’12, Boston, MA, USA - March 05 - 08, 2012*. Ed. by Holly A. Yanco, Aaron Steinfeld, Vanessa Evers, and Odest Chadwicke Jenkins. ACM, pp. 161–162.

Kahneman, Daniel (1973). *Attention and effort*. Vol. 1063. Citeseer.

Karrer, Katja, Charlotte Glaser, Caroline Clemens, and Carmen Bruder (Sept. 2009). “Technikaffinität erfassen – der Fragebogen TA-EG”. In: *ZMMS Spektrum* 29.

Katevas, Kleomenis, Patrick Healey, and Matthew Harris (2015). “Robot Comedy Lab: experimenting with the social dynamics of live performance”. In: *Frontiers in Psychology* 6, p. 1253.

Katevas, Kleomenis, Patrick GT Healey, and Matthew Tobias Harris (2014). “Robot stand-up: engineering a comic performance”. In: *Proceedings of the workshop on humanoid robots and creativity at the IEEE-RAS international conference on humanoid robots humanoids (Madrid)*. Citeseer.

Kausel, Edgar E and Jerel E Slaughter (2011). “Narrow personality traits and organizational attraction: Evidence for the complementary hypothesis”. In: *Organizational Behavior and Human Decision Processes* 114.1, pp. 3–14.

Kendon, Adam, Thomas A Sebeok, and Jean Umiker-Sebeok (2010). *Nonverbal communication, interaction, and gesture: selections from Semiotica*. Vol. 41. Walter de Gruyter.

Kidd, Cory D. and Cynthia Breazeal (2007). “A Robotic Weight Loss Coach”. In: *Proceedings of the Twenty-Second AAAI Conference on Artificial Intelligence, July 22-26, 2007, Vancouver, British Columbia, Canada*. AAAI Press, pp. 1985–1986.

Kidd, Cory D. and Cynthia Breazeal (2008). “Robots at home: Understanding long-term human-robot interaction”. In: *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems, September 22-26, 2008, Acropolis Convention Center, Nice, France*. IEEE, pp. 3230–3235.

Kiderle, Thomas, Hannes Ritschel, Kathrin Janowski, Silvan Mertes, Florian Lingenfelser, and Elisabeth Andre (2021). “Socially-Aware Personality Adaptation”. In: *2021 9th International Conference on Affective Computing and Intelligent Interaction Workshops and Demos (ACIIW)*, pp. 1–8.

Kim, E. S. and B. Scassellati (2007). “Learning to refine behavior using prosodic feedback”. In: *2007 IEEE 6th International Conference on Development and Learning*, pp. 205–210.

Kim, Junhwan, Frederic Cordier, and Nadia Magnenat-Thalmann (2000). “Neural Network-based Violinist’s Hand Animation”. In: *Computer Graphics International Conference, CGI 2000, Geneva, Switzerland, June 19-24, 2000*. IEEE Computer Society, pp. 37–41.

Knapp, Mark L, Judith A Hall, and Terrence G Horgan (2013). *Nonverbal communication in human interaction*. Cengage Learning.

Knight, Heather (2011). “Eight Lessons Learned about Non-verbal Interactions through Robot Theater”. In: *Social Robotics - Third International Conference, ICSR 2011, Amsterdam, The Netherlands, November 24-25, 2011. Proceedings*. Ed. by Bilge Mutlu, Christoph Bartneck, Jaap Ham, Vanessa Evers, and Takayuki Kanda. Vol. 7072. Lecture Notes in Computer Science. Springer, pp. 42–51.

Knight, Heather, Scott Satkin, Varun Ramakrishna, and Santosh Divvala (2011). “A savvy robot standup comic: Online learning through audience tracking”. In: *Workshop paper (TEI 2011, January 22-23, 2011, Funchal, Portugal)*.

Knox, W. Bradley and Peter Stone (2009). “Interactively shaping agents via human reinforcement: the TAMER framework”. In: *Proceedings of the 5th International Conference on Knowledge Capture (K-CAP 2009), September 1-4, 2009, Redondo Beach, California, USA*. Ed. by Yolanda Gil and Natasha Fridman Noy. ACM, pp. 9–16.

Knox, W. Bradley, Peter Stone, and Cynthia Breazeal (2013). “Training a Robot via Human Feedback: A Case Study”. In: *Social Robotics - 5th International Conference, ICSR 2013, Bristol, UK, October 27-29, 2013, Proceedings*. Ed. by Guido Herrmann, Martin J. Pearson, Alexander Lenz, Paul Bremner, Adam Spiers, and Ute Leonards. Vol. 8239. Lecture Notes in Computer Science. Springer, pp. 460–470.

Kohl, Nate and Peter Stone (2004). “Policy Gradient Reinforcement Learning for Fast Quadrupedal Locomotion”. In: *Proceedings of the 2004 IEEE International Conference on Robotics and Automation, ICRA 2004, April 26 - May 1, 2004, New Orleans, LA, USA*. IEEE, pp. 2619–2624.

Konidaris, George Dimitri, Sarah Osentoski, and Philip S. Thomas (2011). “Value Function Approximation in Reinforcement Learning Using the Fourier Basis”. In: *Proceedings of the Twenty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2011, San Francisco, California, USA, August 7-11, 2011*. Ed. by Wolfram Burgard and Dan Roth. AAAI Press.

Kristof-Brown, Amy, Murray R Barrick, and Cynthia Kay Stevens (2005). “When opposites attract: a multi-sample demonstration of complementary person-team fit on extraversion”. In: *Journal of personality* 73.4, pp. 935–958.

Kucherenko, Taras, Patrik Jonell, Sanne van Waveren, Gustav Eje Henter, Simon Alexandersson, Iolanda Leite, and Hedvig Kjellström (2020). “Gesticulator: A framework for semantically-aware speech-driven gesture generation". In: *ICMI '20: International Conference on Multimodal Interaction, Virtual Event, The Netherlands, October 25-29, 2020*. Ed. by Khiet P. Truong, Dirk Heylen, Mary Czerwinski, Nadia Berthouze, Mohamed Chetouani, and Mikio Nakano. ACM, pp. 242–250.

Kuzmanovic, Bojana, Alexandra L Georgescu, Simon B Eickhoff, Nadim J Shah, Gary Bente, Gereon R Fink, and Kai Vogeley (2009). "Duration matters: dissociating neural correlates of detection and evaluation of social gaze". In: *Neuroimage* 46.4, pp. 1154–1163.

La Torre, Fernando De and Jeffrey F. Cohn (2011). "Facial Expression Analysis". In: *Visual Analysis of Humans - Looking at People*. Ed. by Thomas B. Moeslund, Adrian Hilton, Volker Krüger, and Leonid Sigal. Springer, pp. 377–409.

Lacey, Cherie and Catherine Barbara Caudwell (2018). "The Robotic Archetype: Character Animation and Social Robotics". In: *Social Robotics - 10th International Conference, ICSR 2018, Qingdao, China, November 28-30, 2018, Proceedings*. Ed. by Shuzhi Sam Ge, John-John Cabibihan, Miguel Angel Salichs, Elizabeth Broadbent, Hongsheng He, Alan R. Wagner, and Álvaro Castro González. Vol. 11357. Lecture Notes in Computer Science. Springer, pp. 25–34.

Lange, Sascha, Thomas Gabel, and Martin A. Riedmiller (2012). "Batch Reinforcement Learning". In: *Reinforcement Learning*. Ed. by Marco A. Wiering and Martijn van Otterlo. Vol. 12. Adaptation, Learning, and Optimization. Springer, pp. 45–73.

Lasseter, John (1987). "Principles of traditional animation applied to 3D computer animation". In: *Proceedings of the 14th Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH 1987, Anaheim, California, USA, July 27-31, 1987*. Ed. by Maureen C. Stone. ACM, pp. 35–44.

Lee, Jaeryoung, Miles Dai, Björn W. Schuller, and Rosalind W. Picard (2018). "Personalized machine learning for robot perception of affect and engagement in autism therapy". In: *Sci. Robotics* 3.19.

Lee, Kwan Min, Wei Peng, Seung-A Jin, and Chang Yan (Nov. 2006). "Can Robots Manifest Personality?: An Empirical Test of Personality Recognition, Social Responses, and Social Presence in Human–Robot Interaction". In: *Journal of Communication* 56.4, pp. 754–772.

Lee, Nam Yeon, Jeonghun Kim, Eunji Kim, and Ohbyung Kwon (2017). "The Influence of Politeness Behavior on User Compliance with Social Robots in a Healthcare Service Setting". In: *Int. J. Soc. Robotics* 9.5, pp. 727–743.

Lee, Yeoreum, Jae-eul Bae, Sona S Kwak, and Myung-Suk Kim (2011). "The effect of politeness strategy on human-robot collaborative interaction on malfunction of robot vacuum cleaner". In: *RSS workshop on HRI*.

Leech, Geoffrey N (2016). *Principles of pragmatics*. Routledge.

Leite, Iolanda, Carlos Martinho, and Ana Paiva (2013). "Social Robots for Long-Term Interaction: A Survey". In: *Int. J. Soc. Robotics* 5.2, pp. 291–308.

Leite, Iolanda, André Pereira, Ginevra Castellano, Samuel Mascarenhas, Carlos Martinho, and Ana Paiva (2011). "Modelling Empathy in Social Robotic Companions". In: *Advances in User Modeling - UMAP 2011 Workshops, Girona, Spain, July 11-15, 2011, Revised Selected Papers*. Ed. by Liliana Ardissono and Tsvi Kuflik. Vol. 7138. Lecture Notes in Computer Science. Springer, pp. 135–147.

Leonardi, Chiara, Adriano Albertini, Fabio Pianesi, and Massimo Zancanaro (2010). "An exploratory study of a touch-based gestural interface for elderly". In: *Proceedings of the 6th Nordic Conference on Human-Computer Interaction 2010, Reykjavik, Iceland, October 16-20, 2010*. Ed. by Ebba Þóra Hvannberg, Marta Kristín Lárusdóttir, Ann Blandford, and Jan Gulliksen. ACM, pp. 845–850.

Lin, Jinying, Zhen Ma, Randy Gomez, Keisuke Nakamura, Bo He, and Guangliang Li (2020). “A Review on Interactive Reinforcement Learning From Human Social Feedback”. In: *IEEE Access* 8, pp. 120757–120765.

Lippa, Richard (1998). “The nonverbal display and judgment of extraversion, masculinity, femininity, and gender diagnosticity: A lens model analysis”. In: *Journal of Research in Personality* 32.1, pp. 80–107.

Löffler, Diana, Nina Schmidt, and Robert Tscharn (2018). “Multimodal Expression of Artificial Emotion in Social Robots Using Color, Motion and Sound”. In: *Proceedings of the 2018 ACM/IEEE International Conference on Human-Robot Interaction, HRI 2018, Chicago, IL, USA, March 05-08, 2018*. Ed. by Takayuki Kanda, Selma Sabanovic, Guy Hoffman, and Adriana Tapus. ACM, pp. 334–343.

Lynch, Robert (2010). “It’s funny because we think it’s true: laughter is augmented by implicit preferences”. In: *Evolution and Human Behavior* 31.2, pp. 141–148.

Mairesse, François (2008). “Learning to adapt in dialogue systems: data-driven models for personality recognition and generation.” PhD thesis. University of Sheffield.

Mairesse, François and Marilyn A. Walker (2010). “Towards personality-based user adaptation: psychologically informed stylistic language generation”. In: *User Model. User-Adapt. Interact.* 20.3, pp. 227–278.

Mairesse, François and Marilyn A. Walker (2011). “Controlling User Perceptions of Linguistic Style: Trainable Generation of Personality Traits”. In: *Computational Linguistics* 37.3, pp. 455–488.

Manning, Christopher D., Mihai Surdeanu, John Bauer, Jenny Rose Finkel, Steven Bethard, and David McClosky (2014). “The Stanford CoreNLP Natural Language Processing Toolkit”. In: *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, ACL 2014, June 22-27, 2014, Baltimore, MD, USA, System Demonstrations*. The Association for Computer Linguistics, pp. 55–60.

Manurung, Ruli, Graeme Ritchie, Helen Pain, Annalu Waller, Dave O’Mara, and Rolf Black (2008). “The Construction of a Pun Generator for Language Skills Development”. In: *Applied Artificial Intelligence* 22.9, pp. 841–869.

Markey, Patrick M., David C. Funder, and Daniel J. Ozer (2003). “Complementarity of Interpersonal Behaviors in Dyadic Interactions”. In: *Personality and Social Psychology Bulletin* 29.9, pp. 1082–1090.

Martins, Gonçalo S, Luıs Santos, and Jorge Dias (2015). “The GrowMeUp project and the applicability of action recognition techniques”. In: *Third workshop on recognition and action for scene understanding (REACTS). Ruiz de Aloza*.

Martins, Gonçalo S., Luís Santos, and Jorge Dias (2019). “User-Adaptive Interaction in Social Robots: A Survey Focusing on Non-physical Interaction”. In: *I. J. Social Robotics* 11.1, pp. 185–205.

Martins, Gonçalo S., Hend Al Tair, Luís Santos, and Jorge Dias (2019). “αPOMDP: POMDP-based user-adaptive decision-making for social robots”. In: *Pattern Recognit. Lett.* 118, pp. 94–103.

Martínez Estrada, Paloma (2020). “Stand-up comedy: using verbal communication and gestures to express humor in relationship jokes.” PhD thesis.

Mashimo, Ryo, Tomohiro Umetani, Tatsuya Kitamura, and Akiyo Nadamoto (2015). “Automatic generation of Japanese traditional funny scenario from web content based on web intelligence”. In: *Proceedings of the 17th International Conference on Information Integration and Web-based Applications & Services, iiWAS 2015, Brussels, Belgium, December 11-13, 2015*. Ed. by Gabriele Anderst-Kotsis and Maria Indrawan-Santiago. ACM, 21:1–21:9.

Matthews, Gerald, Ian J Deary, and Martha C Whiteman (2003). *Personality traits*. Cambridge University Press.

McArthur, Leslie Z and Reuben M Baron (1983). “Toward an ecological theory of social perception.” In: *Psychological review* 90.3, p. 215.

McCrae, Robert R and Paul T Costa (1989). “The structure of interpersonal traits: Wiggins’s circumplex and the five-factor model.” In: *Journal of personality and social psychology* 56.4, p. 586.

McCrae, Robert R and Paul T Costa Jr (1989). “Reinterpreting the Myers-Briggs type indicator from the perspective of the five-factor model of personality”. In: *Journal of personality* 57.1, pp. 17–40.

McCrae, Robert R. and Paul T. Costa Jr. (2008). “The five-factor theory of personality.” In: Handbook of personality: Theory and research, 3rd ed. New York, NY, US: The Guilford Press, pp. 159–181.

McGee, Elizabeth and Mark Shevlin (2009). “Effect of humor on interpersonal attraction and mate selection”. In: *The Journal of psychology* 143.1, pp. 67–77.

McKeown, Gary and Will Curran (2015). “The Relationship between laughter intensity and perceived humour”. In: *The Fourth Interdisciplinary Workshop on Laughter and other Non-Verbal Vocalisations in Speech*, pp. 27–29.

McKeown, Gary, William Curran, Johannes Wagner, Florian Lingenfelser, and Elisabeth André (2015). “The Belfast storytelling database: A spontaneous social interaction database with laughter focused annotation”. In: *2015 International Conference on Affective Computing and Intelligent Interaction, ACII 2015, Xi’an, China, September 21-24, 2015*. IEEE Computer Society, pp. 166–172.

McNeill, David (1985). “So you think gestures are nonverbal?” In: *Psychological review* 92.3, p. 350.

McNeill, David (2011). “Hand and Mind”. In: *Advances in Visual Semiotics: The Semiotic Web 1992-93*. Ed. by Thomas A. Sebeok and Jean Umiker-Sebeok. De Gruyter Mouton, pp. 351–374.

Mehl, Matthias R, Samuel D Gosling, and James W Pennebaker (2006). “Personality in its natural habitat: manifestations and implicit folk theories of personality in daily life.” In: *Journal of personality and social psychology* 90.5, p. 862.

Mehlmann, Gregor, Markus Häring, Kathrin Janowski, Tobias Baur, Patrick Gebhard, and Elisabeth André (2014). “Exploring a Model of Gaze for Grounding in Multimodal HRI”. In: *Proceedings of the 16th International Conference on Multimodal Interaction, ICMI 2014, Istanbul, Turkey, November 12-16, 2014*. Ed. by Albert Ali Salah, Jeffrey F. Cohn, Björn W. Schuller, Oya Aran, Louis-Philippe Morency, and Philip R. Cohen. ACM, pp. 247–254.

Mehrabian, Albert (1969). “Significance of posture and position in the communication of attitude and status relationships.” In: *Psychological bulletin* 71.5, p. 359.

Mehrabian, Albert et al. (1971). *Silent messages*. Vol. 8. 152. Wadsworth Belmont, CA.

Mileounis, Alexandros, Raymond H. Cuijpers, and Emilia I. Barakova (2015). “Creating Robots with Personality: The Effect of Personality on Social Intelligence”. In: *Artificial Computation in Biology and Medicine - International Work-Conference on the Interplay Between Natural and Artificial Computation, IWINAC 2015, Elche, Spain, June 1-5, 2015, Proceedings, Part I*. Ed. by José Manuel Ferrández de Vicente, José Ramón Álvarez Sánchez, Félix de la Paz López, F. Javier Toledo-Moreo, and Hojjat Adeli. Vol. 9107. Lecture Notes in Computer Science. Springer, pp. 119–132.

Miller, George A. (1995). “WordNet: A Lexical Database for English”. In: *Commun. ACM* 38.11, pp. 39–41.

Mirnig, Nicole, Susanne Stadler, Gerald Stollnberger, Manuel Giuliani, and Manfred Tscheligi (2016). “Robot humor: How self-irony and Schadenfreude influence people’s rating of robot likability”. In: *25th IEEE International Symposium on Robot and Human Interactive Communication, RO-MAN 2016, New York, NY, USA, August 26-31, 2016*. IEEE, pp. 166–171.

Mirnig, Nicole, Gerald Stollnberger, Manuel Giuliani, and Manfred Tscheligi (2017). "Elements of Humor: How Humans Perceive Verbal and Non-verbal Aspects of Humorous Robot Behavior". In: *Companion of the 2017 ACM/IEEE International Conference on Human-Robot Interaction, HRI 2017, Vienna, Austria, March 6-9, 2017*. Ed. by Bilge Mutlu, Manfred Tscheligi, Astrid Weiss, and James E. Young. ACM, pp. 211–212.

Mitsunaga, Noriaki, Christian Smith, Takayuki Kanda, Hiroshi Ishiguro, and Norihiro Hagita (2005). "Robot behavior adaptation for human-robot interaction based on policy gradient reinforcement learning". In: *2005 IEEE/RSJ International Conference on Intelligent Robots and Systems, Edmonton, Alberta, Canada, August 2-6, 2005*. IEEE, pp. 218–225.

Mitsunaga, Noriaki, Christian Smith, Takayuki Kanda, Hiroshi Ishiguro, and Norihiro Hagita (2008). "Adapting Robot Behavior for Human–Robot Interaction". In: *IEEE Trans. Robotics* 24.4, pp. 911–916.

Mollahosseini, Ali, Behzad Hassani, and Mohammad H. Mahoor (2019). "AffectNet: A Database for Facial Expression, Valence, and Arousal Computing in the Wild". In: *IEEE Trans. Affect. Comput.* 10.1, pp. 18–31.

Montoya, R. Matthew and Robert S. Horton (2013). "A meta-analytic investigation of the processes underlying the similarity-attraction effect". In: *Journal of Social and Personal Relationships* 30.1, pp. 64–94.

Mori, Masahiro (1970). "The uncanny valley". In: *Energy* 7.4, pp. 33–35.

Motti, Lilian Genaro, Nadine Vigouroux, and Philippe Gorce (2013). "Interaction techniques for older adults using touchscreen devices: a literature review". In: *Proceedings of the 25th IEME conference francophone on l'Interaction Homme-Machine, IHM '13, Talence, France, November 12-15, 2013*. ACM, pp. 125–134.

Murstein, Bernard I and Robert G Brust (1985). "Humor and interpersonal attraction". In: *Journal of personality assessment* 49.6, pp. 637–640.

Najar, Anis, Olivier Sigaud, and Mohamed Chetouani (2016). "Training a robot with evaluative feedback and unlabeled guidance signals". In: *25th IEEE International Symposium on Robot and Human Interactive Communication, RO-MAN 2016, New York, NY, USA, August 26-31, 2016*. IEEE, pp. 261–266.

Nejat, Goldie and Maurizio Ficocelli (2008). "Can I be of assistance? The intelligence behind an assistive robot". In: *2008 IEEE International Conference on Robotics and Automation, ICRA 2008, May 19-23, 2008, Pasadena, California, USA*. IEEE, pp. 3564–3569.

Ng, Andrew Y., Daishi Harada, and Stuart Russell (1999). "Policy Invariance Under Reward Transformations: Theory and Application to Reward Shaping". In: *Proceedings of the Sixteenth International Conference on Machine Learning (ICML 1999), Bled, Slovenia, June 27 - 30, 1999*. Ed. by Ivan Bratko and Saso Dzeroski. Morgan Kaufmann, pp. 278–287.

Nguyen, H., D. Morales, and T. Chin (2018). *A neural chatbot with personality*. Stanford NLP Course. Stanford University.

Niculescu, Andreea, Betsy van Dijk, Anton Nijholt, Haizhou Li, and See Swee Lan (2013). "Making Social Robots More Attractive: The Effects of Voice Pitch, Humor and Empathy". In: *I. J. Social Robotics* 5.2, pp. 171–191.

Niewiadomski, Radoslaw, Jennifer Hofmann, Jérôme Urbain, Tracey Platt, Johannes Wagner, Bilal Piot, Hüseyin Çakmak, Sathish Pammi, Tobias Baur, Stéphane Dupont, Matthieu Geist, Florian Lingenfelser, Gary McKeown, Olivier Pietquin, and Willibald Ruch (2013). "Laugh-aware virtual agent and its impact on user amusement". In: *International conference on Autonomous Agents and Multi-Agent Systems, AAMAS '13, Saint Paul, MN, USA, May 6-10, 2013*. Ed. by Maria L. Gini, Onn Shehory, Takayuki Ito, and Catholijn M. Jonker. IFAAMAS, pp. 619–626.

Nijholt, Anton (2007). "Conversational Agents and the Construction of Humorous Acts". In: *Conversational Informatics*. Wiley-Blackwell. Chap. 2, pp. 19–47.

Nijholt, Anton (2018). "Robotic Stand-Up Comedy: State-of-the-Art". In: *Distributed, Ambient and Pervasive Interactions: Understanding Humans - 6th International Conference, DAPI 2018, Held as Part of HCI International 2018, Las Vegas, NV, USA, July 15-20, 2018, Proceedings, Part I*. Ed. by Norbert A. Streitz and Shin'ichi Konomi. Vol. 10921. Lecture Notes in Computer Science. Springer, pp. 391–410.

Nomura, Tatsuya and Kazuma Saeki (2010). "Effects of Polite Behaviors Expressed by Robots: A Psychological Experiment in Japan". In: *Int. J. Synth. Emot.* 1.2, pp. 38–52.

Norrick, Neal R (2001). "On the conversational performance of narrative jokes: Toward an account of timing". In: *Humor* 14.3, pp. 255–274.

Norrick, Neal R. (2004). "Non-verbal humor and joke performance". In: 17.4, pp. 401–409.

Oakman, Jonathan, Shannon Gifford, and Natasha Chlebowsky (2003). "A Multilevel Analysis of the Interpersonal Behavior of Socially Anxious People". In: *Journal of Personality* 71.3, pp. 397–434.

Oberlander, Jon and Alastair J. Gill (2006). "Language With Character: A Stratified Corpus Comparison of Individual Differences in E-Mail Communication". In: *Discourse Processes* 42.3, pp. 239–270.

Oertel, Catharine, Ginevra Castellano, Mohamed Chetouani, Jauwairia Nasir, Mohammad Obaid, Catherine Pelachaud, and Christopher E. Peters (2020). "Engagement in Human-Agent Interaction: An Overview". In: *Frontiers Robotics AI* 7, p. 92.

Oliveira, Raquel, Patrícia Arriaga, Minja Axelsson, and Ana Paiva (2021). "Humor-Robot Interaction: A Scoping Review of the Literature and Future Directions". In: *Int. J. Soc. Robotics* 13.6, pp. 1369–1383.

openweathermap.org (2021). *Current weather and forecast - OpenWeatherMap*. https://openweathermap.org/. Accessed: 2021-04-14.

Oraby, Shereen, Lena Reed, Shubhangi Tandon, Sharath T. S., Stephanie M. Lukin, and Marilyn A. Walker (2018). "Controlling Personality-Based Stylistic Variation with Neural Natural Language Generators". In: *Proceedings of the 19th Annual SIGdial Meeting on Discourse and Dialogue, Melbourne, Australia, July 12-14, 2018*. Ed. by Kazunori Komatani, Diane J. Litman, Kai Yu, Lawrence Cavedon, Mikio Nakano, and Alex Papangelis. Association for Computational Linguistics, pp. 180–190.

Ortony, Andrew, Gerald L. Clore, and Allan Collins (1988). *The Cognitive Structure of Emotions*. Cambridge University Press.

Oudeyer, Pierre-Yves and Frederic Kaplan (2008). "How can we define intrinsic motivation?" In: *the 8th International Conference on Epigenetic Robotics: Modeling Cognitive Development in Robotic Systems*. Lund University Cognitive Studies, Lund: LUCS, Brighton.

Park, Hae Won, Ishaan Grover, Samuel Spaulding, Louis Gomez, and Cynthia Breazeal (2019). "A Model-Free Affective Reinforcement Learning Approach to Personalization of an Autonomous Social Robot Companion for Early Literacy Education". In: *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019*. AAAI Press, pp. 687–694.

Parkhi, Omkar M., Andrea Vedaldi, and Andrew Zisserman (2015). "Deep Face Recognition". In: *Proceedings of the British Machine Vision Conference 2015, BMVC 2015, Swansea, UK, September 7-10, 2015*. Ed. by Xianghua Xie, Mark W. Jones, and Gary K. L. Tam. BMVA Press, pp. 41.1–41.12.

Patompak, Pakpoom, Sungmoon Jeong, Itthisek Nilkhamhang, and Nak Young Chong (2020). "Learning Proxemics for Personalized Human-Robot Social Interaction". In: *Int. J. Soc. Robotics* 12.1, pp. 267–280.

Pedersen, Ted and Varada Kolhatkar (2009). "WordNet: : SenseRelate: : AllWords - A Broad Coverage Word Sense Tagger that Maximizes Semantic Relatedness". In: *Human Language Technologies: Conference of the North American Chapter of the Association of Computational Linguistics, Proceedings, May 31 - June 5, 2009, Boulder, Colorado, USA, Demos*. Ed. by Fred Popowich and Michael Johnston. The Association for Computational Linguistics, pp. 17–20.

Pelachaud, Catherine and Massimo Bilvi (2003). "Modelling Gaze Behavior for Conversational Agents". In: *Intelligent Virtual Agents*. Ed. by Thomas Rist, Ruth S. Aylett, Daniel Ballin, and Jeff Rickel. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 93–100.

Pennebaker, James W and Laura A King (1999). "Linguistic styles: language use as an individual difference." In: *Journal of personality and social psychology* 77.6, p. 1296.

Pennebaker, James W, Matthias R Mehl, and Kate G Niederhoffer (2003). "Psychological aspects of natural language use: Our words, our selves". In: *Annual review of psychology* 54.1, pp. 547–577.

Pentland, Alex (2005). "Socially Aware Computation and Communication". In: *IEEE Computer* 38.3, pp. 33–40.

Pentland, Alex (2007). "Social Signal Processing [Exploratory DSP]". In: *IEEE Signal Process. Mag.* 24.4, pp. 108–111.

Peters, Christopher E., Stylianos Asteriadis, and Genaro Rebolledo-Mendez (2009). "Modelling user attention for human-agent interaction". In: *10th Workshop on Image Analysis for Multimedia Interactive Services, WIAMIS 2009, London, United Kingdom, May 6-8, 2009*. IEEE Computer Society, pp. 266–269.

Pickering, Lucy, Marcella Corduas, Jodi Eisterhold, Brenna Seifried, Alyson Eggleston, and Salvatore Attardo (2009). "Prosodic markers of saliency in humorous narratives". In: *Discourse processes* 46.6, pp. 517–540.

Pittam, Jeff (1994). *Voice in social interaction*. Vol. 5. Sage.

Poggi, Isabella and Francesca D'Errico (2010). "Cognitive modelling of human social signals". In: *Proceedings of the 2nd international workshop on Social signal processing, SSPW@MM 2010, Firenze, Italy, October 29, 2010*. Ed. by Alessandro Vinciarelli, Maja Pantic, and Alex Pentland. ACM, pp. 21–26.

Posner, Jonathan, James A Russell, and Bradley S Peterson (2005). "The circumplex model of affect: An integrative approach to affective neuroscience, cognitive development, and psychopathology". In: *Development and psychopathology* 17.3, pp. 715–734.

Poyatos, Fernando (1984). "The multichannel reality of discourse: language-paralanguage-kinesics and the totality of communicative systems". In: *Language Sciences* 6.2, pp. 307–337.

Quadrianto, Novi, Björn W. Schuller, and Finnian Rachel Lattimore (2021). "Editorial: Ethical Machine Learning and Artificial Intelligence". In: *Frontiers Big Data* 4, p. 742589.

Rajapakshe, Thejan, Rajib Rana, Sara Khalifa, Jiajun Liu, and Björn W. Schuller (2022). "A Novel Policy for Pre-trained Deep Reinforcement Learning for Speech Emotion Recognition". In: *ACSW 2022: Australasian Computer Science Week 2022, Brisbane, Australia, February 14 - 18, 2022*. Ed. by David Abramson and Minh Ngoc Dinh. ACM, pp. 96–105.

Ramachandran, Aditi, Sarah Strohkorb Sebo, and Brian Scassellati (2019). "Personalized Robot Tutoring Using the Assistive Tutor POMDP (AT-POMDP)". In: *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in*

Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019. AAAI Press, pp. 8050–8057.

Raskin, Victor (2012). *Semantic mechanisms of humor*. Vol. 24. Springer Science & Business Media.

Raskin, Victor and Salvatore Attardo (1994). “Non-literalness and non-bona-fide in language: An approach to formal and computational treatments of humor”. In: *Pragmatics & Cognition* 2.1, pp. 31–69.

Rea, Daniel J., Sebastian Schneider, and Takayuki Kanda (2021). “"Is this all you can do? Harder!": The Effects of (Im)Polite Robot Encouragement on Exercise Effort”. In: *HRI ’21: ACM/IEEE International Conference on Human-Robot Interaction, Boulder, CO, USA, March 8-11, 2021*. Ed. by Cindy L. Bethel, Ana Paiva, Elizabeth Broadbent, David Feil-Seifer, and Daniel Szafir. ACM, pp. 225–233.

reeti.fr (2021). *reeti - Home Reeti*. http://reeti.fr/index.php/en/. Accessed: 2021-04-14.

Reis, Harry T and Susan Sprecher (2009). *Encyclopedia of human relationships*. Sage Publications.

Reiter, Ehud and Robert Dale (2000). *Building natural language generation systems*. Cambridge university press.

Ren, Zhao, Jing Han, Nicholas Cummins, and Björn W. Schuller (2020). “Enhancing Transferability of Black-Box Adversarial Attacks via Lifelong Learning for Speech Emotion Recognition Models”. In: *Interspeech 2020, 21st Annual Conference of the International Speech Communication Association, Virtual Event, Shanghai, China, 25-29 October 2020*. Ed. by Helen Meng, Bo Xu, and Thomas Fang Zheng. ISCA, pp. 496–500.

Ribeiro, Tiago and Ana Paiva (2012). “The illusion of robotic life: principles and practices of animation for robots”. In: *International Conference on Human-Robot Interaction, HRI’12, Boston, MA, USA - March 05 - 08, 2012*. Ed. by Holly A. Yanco, Aaron Steinfeld, Vanessa Evers, and Odest Chadwicke Jenkins. ACM, pp. 383–390.

Rich, Charles, C Sidner, Bahador Nooraei, and William Coon (2012). “Operating in a hierarchy of time scales for an always-on relational agent”. In: *Workshop on Real-Time Conversations with Virtual Agents, Santa Cruz, CA (September 2012)*.

Rieser, Verena and Oliver Lemon (2011). *Reinforcement Learning for Adaptive Dialogue Systems - A Data-driven Methodology for Dialogue Management and Natural Language Generation*. Theory and Applications of Natural Language Processing. Springer.

Riggio, Ronald E and Howard S Friedman (1986). “Impression formation: The role of expressive behavior.” In: *Journal of personality and social psychology* 50.2, p. 421.

Rist, Thomas, Andreas Seiderer, and Elisabeth André (2018). “Providing Life-Style-Intervention to Improve Well-Being of Elderly People”. In: *Entertainment Computing - ICEC 2018 - 17th IFIP TC 14 International Conference, Held at the 24th IFIP World Computer Congress, WCC 2018, Poznan, Poland, September 17-20, 2018, Proceedings*. Ed. by Esteban Clua, Licinio Roque, Artur Lugmayr, and Pauliina Tuomi. Vol. 11112. Lecture Notes in Computer Science. Springer, pp. 362–367.

Rist, Thomas, Andreas Seiderer, Stephan Hammer, Marcus Mayr, and Elisabeth André (2015). “CARE - extending a digital picture frame with a recommender mode to enhance well-being of elderly people”. In: *9th International Conference on Pervasive Computing Technologies for Healthcare, PervasiveHealth 2015, Istanbul, Turkey, 20-23 May 2015*. Ed. by Bert Arnrich, Cem Ersoy, Anind K. Dey, Kai Kunze, and Nadia Berthouze. ICST, pp. 112–120.

Ritschel, Hannes (2018). “Socially-Aware Reinforcement Learning for Personalized Human-Robot Interaction”. In: *Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems, AAMAS 2018, Stockholm, Sweden, July 10-15, 2018*. Ed. by Elisabeth André, Sven Koenig, Mehdi Dastani, and Gita Sukthankar. International Foundation for Autonomous Agents and Multiagent Systems Richland, SC, USA / ACM, pp. 1775–1777.

Ritschel, Hannes and Elisabeth André (2017). “Real-Time Robot Personality Adaptation based on Reinforcement Learning and Social Signals”. In: *Companion of the 2017 ACM/IEEE International Conference on Human-Robot Interaction, HRI 2017, Vienna, Austria, March 6-9, 2017*. Ed. by Bilge Mutlu, Manfred Tscheligi, Astrid Weiss, and James E. Young. ACM, pp. 265–266.

Ritschel, Hannes and Elisabeth André (Nov. 2018). “Shaping a social robot’s humor with Natural Language Generation and socially-aware reinforcement learning”. In: *Proceedings of the Workshop on NLG for Human–Robot Interaction*. Tilburg, The Netherlands: Association for Computational Linguistics, pp. 12–16.

Ritschel, Hannes, Ilhan Aslan, Silvan Mertes, Andreas Seiderer, and Elisabeth André (2019a). “Personalized Synthesis of Intentional and Emotional Non-Verbal Sounds for Social Robots”. In: *8th International Conference on Affective Computing and Intelligent Interaction, ACII 2019, Cambridge, United Kingdom, September 3-6, 2019*. IEEE, pp. 1–7.

Ritschel, Hannes, Ilhan Aslan, David Sedlbauer, and Elisabeth André (2019b). “Irony Man: Augmenting a Social Robot with the Ability to Use Irony in Multimodal Communication with Humans”. In: *Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent Systems, AAMAS ’19, Montreal, QC, Canada, May 13-17, 2019*. Ed. by Edith Elkind, Manuela Veloso, Noa Agmon, and Matthew E. Taylor. International Foundation for Autonomous Agents and Multiagent Systems, pp. 86–94.

Ritschel, Hannes, Tobias Baur, and Elisabeth André (2017a). “Adapting a Robot’s linguistic style based on socially-aware reinforcement learning”. In: *26th IEEE International Symposium on Robot and Human Interactive Communication, RO-MAN 2017, Lisbon, Portugal, August 28 - Sept. 1, 2017*. IEEE, pp. 378–384.

Ritschel, Hannes, Tobias Baur, and Elisabeth André (2017b). “Personalisierung der Mensch-Roboter-Interaktion durch sozial-sensitives Lernen”. In: *2. VDI-Fachkonferenz Humanoide Roboter 2017, Aschheim, Deutschland, 5.-6. Dezember 2017*.

Ritschel, Hannes, Kathrin Janowski, Andreas Seiderer, Stefan Wagner, and Elisabeth André (2019c). “Insights on usability and user feedback for an assistive robotic health companion with adaptive linguistic style”. In: *Proceedings of the 12th ACM International Conference on PErvasive Technologies Related to Assistive Environments, PETRA 2019, Island of Rhodes, Greece, June 5-7, 2019*. Ed. by Fillia Makedon. ACM, pp. 319–320.

Ritschel, Hannes, Thomas Kiderle, and Elisabeth André (2021). “Implementing Parallel and Independent Movements for a Social Robot’s Affective Expressions”. In: *2021 9th International Conference on Affective Computing and Intelligent Interaction Workshops and Demos (ACIIW)*, pp. 1–4.

Ritschel, Hannes, Thomas Kiderle, Klaus Weber, and Elisabeth André (Mar. 2020a). “Multimodal Joke Presentation for Social Robots based on Natural-Language Generation and Nonverbal Behaviors”. In: *Proceedings of the 2nd Workshop on NLG for Human–Robot Interaction*. Cambridge, UK.

Ritschel, Hannes, Thomas Kiderle, Klaus Weber, Florian Lingenfelser, Tobias Baur, and Elisabeth André (2020b). “Multimodal Joke Generation and Paralinguistic Personalization for a Socially-Aware Robot”. In: *Advances in Practical Applications of Agents, Multi-Agent Systems, and Trustworthiness. The PAAMS Collection - 18th International Conference, PAAMS 2020, L’Aquila, Italy, October 7-9, 2020, Proceedings*. Ed. by Yves Demazeau, Tom Holvoet, Juan M. Corchado, and Stefania Costantini. Vol. 12092. Lecture Notes in Computer Science. Springer, pp. 278–290.

Ritschel, Hannes, Andreas Seiderer, Kathrin Janowski, Stefan Wagner, and Elisabeth André (2019d). “Adaptive linguistic style for an assistive robotic health companion based on explicit human feedback”. In: *Proceedings of the 12th ACM International Conference on PErvasive*

Technologies Related to Assistive Environments, PETRA 2019, Island of Rhodes, Greece, June 5-7, 2019. Ed. by Fillia Makedon. ACM, pp. 247–255.

Robert, Lionel P., Rasha Alahmad, Connor Esterwood, Sangmi Kim, Sangseok You, and Qiaoning Zhang (2019). "A Review of Personality in Human-Robot Interactions". In: *Found. Trends Inf. Syst.* 4.2, pp. 107–212.

Ruckert, Jolina H. (2011). "Unity in multiplicity: searching for complexity of persona in HRI". In: *Proceedings of the 6th International Conference on Human Robot Interaction, HRI 2011, Lausanne, Switzerland, March 6-9, 2011*. Ed. by Aude Billard, Peter H. Kahn Jr., Julie A. Adams, and J. Gregory Trafton. ACM, pp. 237–238.

Ruhland, Kerstin, Sean Andrist, Jeremy B. Badler, Christopher E. Peters, Norman I. Badler, Michael Gleicher, Bilge Mutlu, and Rachel McDonnell (2014). "Look me in the Eyes: A Survey of Eye and Gaze Animation for Virtual Agents and Artificial Systems". In: *35th Annual Conference of the European Association for Computer Graphics, Eurographics 2014 - State of the Art Reports, Strasbourg, France, April 7-11, 2014*. Ed. by Sylvain Lefebvre and Michela Spagnuolo. Eurographics Association, pp. 69–91.

Russell, Stuart J. and Peter Norvig (2003). *Artificial intelligence - a modern approach, 2nd Edition*. Prentice Hall series in artificial intelligence. Prentice Hall.

Rutter, DR, Ian E Morley, and Jane C Graham (1972). "Visual interaction in a group of introverts and extraverts". In: *European Journal of Social Psychology* 2.4, pp. 371–384.

Salam, Hanan, Oya Çeliktutan, Isabelle Hupont Torres, Hatice Gunes, and Mohamed Chetouani (2017). "Fully Automatic Analysis of Engagement and Its Relationship to Personality in Human-Robot Interactions". In: *IEEE Access* 5, pp. 705–721.

Salem, Maha, Micheline Ziadee, and Majd F. Sakr (2013). "Effects of Politeness and Interaction Context on Perception and Experience of HRI". In: *Social Robotics - 5th International Conference, ICSR 2013, Bristol, UK, October 27-29, 2013, Proceedings*. Ed. by Guido Herrmann, Martin J. Pearson, Alexander Lenz, Paul Bremner, Adam Spiers, and Ute Leonards. Vol. 8239. Lecture Notes in Computer Science. Springer, pp. 531–541.

Salem, Maha, Micheline Ziadee, and Majd F. Sakr (2014). "Marhaba, how may i help you?: effects of politeness and culture on robot acceptance and anthropomorphization". In: *ACM/IEEE International Conference on Human-Robot Interaction, HRI'14, Bielefeld, Germany, March 3-6, 2014*. Ed. by Gerhard Sagerer, Michita Imai, Tony Belpaeme, and Andrea Lockerd Thomaz. ACM, pp. 74–81.

Schatzmann, Jost, Karl Weilhammer, Matthew N. Stuttle, and Steve J. Young (2006). "A survey of statistical user simulation techniques for reinforcement-learning of dialogue management strategies". In: *Knowl. Eng. Rev.* 21.2, pp. 97–126.

Scherer, Klaus Rainer (1979). *Personality markers in speech*. Cambridge University Press.

Schiller, Dominik, Katharina Weitz, Kathrin Janowski, and Elisabeth André (2019). "Human-Inspired Socially-Aware Interfaces". In: *Theory and Practice of Natural Computing - 8th International Conference, TPNC 2019, Kingston, ON, Canada, December 9-11, 2019, Proceedings*. Ed. by Carlos Martín-Vide, Geoffrey T. Pond, and Miguel A. Vega-Rodríguez. Vol. 11934. Lecture Notes in Computer Science. Springer, pp. 41–53.

Schmidt, Albrecht (2000). "Implicit Human Computer Interaction Through Context". In: *Pers. Ubiquitous Comput.* 4.2/3, pp. 191–199.

Schneider, Sebastian and Franz Kummert (2017). "Exploring embodiment and dueling bandit learning for preference adaptation in human-robot interaction". In: *26th IEEE International Symposium on Robot and Human Interactive Communication, RO-MAN 2017, Lisbon, Portugal, August 28 - Sept. 1, 2017*. IEEE, pp. 1325–1331.

Schroeter, Ch., Steffen Müller, Michael Volkhardt, Erik Einhorn, Claire A. G. J. Huijnen, Herjan van den Heuvel, Andreas van Berlo, Andreas Bley, and Horst-Michael Gross (2013). "Realization and user evaluation of a companion robot for people with mild cognitive impairments". In: *2013 IEEE International Conference on Robotics and Automation, Karlsruhe, Germany, May 6-10, 2013*. IEEE, pp. 1153–1159.

Seiderer, Andreas, Stephan Hammer, Elisabeth André, Marcus Mayr, and Thomas Rist (2015). "Exploring Digital Image Frames for Lifestyle Intervention to Improve Well-being of Older Adults". In: *Proceedings of the 5th International Conference on Digital Health 2015, Florence, Italy, May 18-20, 2015*. Ed. by Patty Kostkova and Floriana Grasso. ACM, pp. 71–78.

Senju, Atsushi and Gergely Csibra (2008). "Gaze following in human infants depends on communicative signals". In: *Current biology* 18.9, pp. 668–671.

Sharkey, Noel and Amanda Sharkey (2010). "Living with robots: Ethical tradeoffs in eldercare". In: *Close Engagements with Artificial Companions*. John Benjamins, pp. 245–256.

Shrout, Patrick E. and Donald W. Fiske (1981). "Nonverbal behaviors and social evaluation". In: *Journal of Personality* 49.2, pp. 115–128.

Sidner, Candace L., Timothy W. Bickmore, Bahador Nooraie, Charles Rich, Lazlo Ring, Mahni Shayganfar, and Laura Vardoulakis (2018). "Creating New Technologies for Companionable Agents to Support Isolated Older Adults". In: *TiiS* 8.3, 17:1–17:27.

Siegman, Aron W and Benjamin Pope (1965). "Personality variables associated with productivity and verbal fluency in the initial interview." In: *Proceedings of the Annual Convention of the American Psychological Association*. American Psychological Association.

Sjöbergh, Jonas and Kenji Araki (2008). "Robots Make Things Funnier". In: *New Frontiers in Artificial Intelligence, JSAI 2008 Conference and Workshops, Asahikawa, Japan, June 11-13, 2008, Revised Selected Papers*. Ed. by Hiromitsu Hattori, Takahiro Kawamura, Tsuyoshi Idé, Makoto Yokoo, and Yohei Murakami. Vol. 5447. Lecture Notes in Computer Science. Springer, pp. 306–313.

Smith, Bruce L., Bruce L. Brown, William J. Strong, and Alvin C. Rencher (1975). "Effects of Speech Rate on Personality Perception". In: *Language and Speech* 18.2, pp. 145–152.

Smith, Harrison Jesse, Chen Cao, Michael Neff, and Yingying Wang (2019). "Efficient Neural Networks for Real-time Motion Style Transfer". In: *Proc. ACM Comput. Graph. Interact. Tech.* 2.2, 13:1–13:17.

Snyder, Mark and William Ickes (1985). "Personality and social behavior". In: *Handbook of social psychology* 2.3, pp. 883–947.

So, HeeSeon, Myungsuk Kim, and KwangMyung Oh (2008). "People's perceptions of a personal service robot's personality and a personal service robot's personality design guide suggestions". In: *The 17th IEEE International Symposium on Robot and Human Interactive Communication, RO-MAN 2008, Munich, Germany, August 1-3, 2008*. Ed. by Martin Buss and Kolja Kühnlenz. IEEE, pp. 500–505.

Socher, Richard, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Y. Ng, and Christopher Potts (2013). "Recursive Deep Models for Semantic Compositionality Over a Sentiment Treebank". In: *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, EMNLP 2013, 18-21 October 2013, Grand Hyatt Seattle, Seattle, Washington, USA, A meeting of SIGDAT, a Special Interest Group of the ACL*. ACL, pp. 1631–1642.

sourceforge.net (2021). *A. L. I. C. E. and AIML download | SourceForge.net*. https://sourceforge.net/projects/alicebot. Accessed: 2021-04-14.

Srinivasan, Vasant and Leila Takayama (2016). "Help Me Please: Robot Politeness Strategies for Soliciting Help From Humans". In: *Proceedings of the 2016 CHI Conference on Human Factors* *in Computing Systems, San Jose, CA, USA, May 7-12, 2016*. Ed. by Jofish Kaye, Allison Druin, Cliff Lampe, Dan Morris, and Juan Pablo Hourcade. ACM, pp. 4945–4955.

Srivastava, Ajitesh and Naomi T. Fitter (2021). “A Robot Walks into a Bar: Automatic Robot Joke Success Assessment”. In: *IEEE International Conference on Robotics and Automation, ICRA 2021, Xi’an, China, May 30 - June 5, 2021*. IEEE, pp. 2710–2716.

Stahl, John S (1999). “Amplitude of human head movements associated with horizontal saccades”. In: *Experimental brain research* 126.1, pp. 41–54.

Stern, John A, Larry C Walrath, and Robert Goldstein (1984). “The endogenous eyeblink”. In: *Psychophysiology* 21.1, pp. 22–33.

Stock, Oliviero and Carlo Strapparava (2002). “HAHAcronym: Humorous agents for humorous acronyms”. In: *Stock, Oliviero, Carlo Strapparava, and Anton Nijholt. Eds*, pp. 125–135.

Strait, Megan, Priscilla Briggs, and Matthias Scheutz (2015). “Gender, more so than age, modulates positive perceptions of language-based human-robot interactions”. In: *4th international symposium on new frontiers in human robot interaction*, pp. 21–22.

Strait, Megan, Cody Canning, and Matthias Scheutz (2014). “Let me tell you! investigating the effects of robot communication strategies in advice-giving situations based on robot appearance, interaction modality and distance”. In: *ACM/IEEE International Conference on Human-Robot Interaction, HRI’14, Bielefeld, Germany, March 3-6, 2014*. Ed. by Gerhard Sagerer, Michita Imai, Tony Belpaeme, and Andrea Lockerd Thomaz. ACM, pp. 479–486.

Suay, Halit Bener and Sonia Chernova (2011). “Effect of human guidance and state space size on Interactive Reinforcement Learning”. In: *20th IEEE International Symposium on Robot and Human Interactive Communication, RO-MAN 2011, Atlanta, Georgia, USA, July 31 - August 3, 2011*. Ed. by Henrik I. Christensen. IEEE, pp. 1–6.

Suguitan, Michael, Mason Bretan, and Guy Hoffman (2019). “Affective Robot Movement Generation Using CycleGANs”. In: *14th ACM/IEEE International Conference on Human-Robot Interaction, HRI 2019, Daegu, South Korea, March 11-14, 2019*. IEEE, pp. 534–535.

Sutton, Richard S. and Andrew G. Barto (2018). *Reinforcement Learning - An Introduction (Second Edition)*. Adaptive Computation and Machine Learning. MIT Press.

Sutton, Richard S., Hamid Reza Maei, Doina Precup, Shalabh Bhatnagar, David Silver, Csaba Szepesvári, and Eric Wiewiora (2009). “Fast gradient-descent methods for temporal-difference learning with linear function approximation”. In: *Proceedings of the 26th Annual International Conference on Machine Learning, ICML 2009, Montreal, Quebec, Canada, June 14-18, 2009*. Ed. by Andrea Pohoreckyj Danyluk, Léon Bottou, and Michael L. Littman. Vol. 382. ACM International Conference Proceeding Series. ACM, pp. 993–1000.

Sutton, Richard S., Joseph Modayil, Michael Delp, Thomas Degris, Patrick M. Pilarski, Adam White, and Doina Precup (2011). “Horde: a scalable real-time architecture for learning knowledge from unsupervised sensorimotor interaction”. In: *10th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2011), Taipei, Taiwan, May 2-6, 2011, Volume 1-3*. Ed. by Liz Sonenberg, Peter Stone, Kagan Tumer, and Pinar Yolum. IFAAMAS, pp. 761–768.

Swaminathan, Janani, Jane Akintoye, Marlena R. Fraune, and Heather Knight (2021). “Robots That Run their Own Human Experiments: Exploring Relational Humor with Multi-Robot Comedy”. In: *30th IEEE International Conference on Robot & Human Interactive Communication, RO-MAN 2021, Vancouver, BC, Canada, August 8-12, 2021*. IEEE, pp. 1262–1268.

Szafir, Daniel and Bilge Mutlu (2012). “Pay attention!: designing adaptive agents that monitor and improve user engagement”. In: *CHI Conference on Human Factors in Computing Systems, CHI ’12, Austin, TX, USA - May 05 - 10, 2012*. Ed. by Joseph A. Konstan, Ed H. Chi, and Kristina Höök. ACM, pp. 11–20.

Tao, Vincent, Kristi Moy, and Vibhuti Arya Amirfar (2016). “A little robot with big promise may be future of personalized health care”. In: *Pharmacy Today* 22.9, p. 38.

Tapus, Adriana and Maja J. Mataric (2008). “Socially Assistive Robots: The Link between Personality, Empathy, Physiological Signals, and Task Performance”. In: *Emotion, Personality, and Social Behavior, Papers from the 2008 AAAI Spring Symposium, Technical Report SS-08-04, Stanford, California, USA, March 26-28, 2008*. AAAI, pp. 133–140.

Tapus, Adriana, Cristian Tapus, and Maja J. Mataric (2008). “User - robot personality matching and assistive robot behavior adaptation for post-stroke rehabilitation therapy”. In: *Intelligent Service Robotics* 1.2, pp. 169–183.

Tenorio-González, Ana Cecilia, Eduardo F. Morales, and Luis Villaseñor Pineda (2010). “Dynamic Reward Shaping: Training a Robot by Voice”. In: *Advances in Artificial Intelligence - IBERAMIA 2010, 12th Ibero-American Conference on AI, Bahía Blanca, Argentina, November 1-5, 2010. Proceedings*. Ed. by Ángel Fernando Kuri Morales and Guillermo Ricardo Simari. Vol. 6433. Lecture Notes in Computer Science. Springer, pp. 483–492.

Teufel, Christoph, Dean M Alexis, Nicola S Clayton, and Greg Davis (2010). “Mental-state attribution drives rapid, reflexive gaze following”. In: *Attention, Perception, & Psychophysics* 72.3, pp. 695–705.

Thomaz, Andrea Lockerd and Cynthia Breazeal (2007). “Asymmetric Interpretations of Positive and Negative Human Feedback for a Social Learning Agent”. In: *IEEE RO-MAN 2007, 16th IEEE International Symposium on Robot & Human Interactive Communication, August 26-29, 2007, Jeju Island, South Korea, Proceedings*. IEEE, pp. 720–725.

Thorson, James A. and F. C. Powell (1993). “Sense of humor and dimensions of personality”. In: *Journal of Clinical Psychology* 49.6, pp. 799–809.

Tolba, Rahma M, Taha Al-Arif, and El-Sayed M El Horbaty (2018). “Realistic Facial Animation Review: Based on Facial Action Coding System”. In: *Egyptian Computer Science Journal* 42.1, pp. 1–9.

Torrey, Cristen, Susan R. Fussell, and Sara B. Kiesler (2013). “How a robot should give advice”. In: *ACM/IEEE International Conference on Human-Robot Interaction, HRI 2013, Tokyo, Japan, March 3-6, 2013*. Ed. by Hideaki Kuzuoka, Vanessa Evers, Michita Imai, and Jodi Forlizzi. IEEE/ACM, pp. 275–282.

Tseng, Shih-Huan, Feng-Chih Liu, and Li-Chen Fu (2018). “Active Learning on Service Providing Model: Adjustment of Robot Behaviors Through Human Feedback”. In: *IEEE Trans. Cogn. Dev. Syst.* 10.3, pp. 701–711.

ubuntu.com (2021). *Enterprise Open Source and Linux*. https://ubuntu.com/. Accessed: 2021-04-14.

Umetani, Tomohiro, Satoshi Aoki, Kazuhiro Akiyama, Ryo Mashimo, Tatsuya Kitamura, and Akiyo Nadamoto (2015). “Manzai robot system with scalability based on distributed software components”. In: *2015 International Symposium on Micro-NanoMechatronics and Human Science, MHS 2015, Nagoya, Japan, November 23-25, 2015*. IEEE, pp. 1–5.

Umetani, Tomohiro, Akiyo Nadamoto, and Tatsuya Kitamura (2017). “Manzai robots: entertainment robots as passive media based on autocreated Manzai scripts from web news articles”. In: *Handbook of Digital Games and Entertainment Technologies*, pp. 1041–1068.

Urbain, Jérôme, Hüseyin Çakmak, Aurelie Charlier, Maxime Denti, Thierry Dutoit, and Stéphane Dupont (2014). “Arousal-Driven Synthesis of Laughter”. In: *IEEE J. Sel. Top. Signal Process.* 8.2, pp. 273–284.

Urbain, Jérôme, Radoslaw Niewiadomski, Elisabetta Bevacqua, Thierry Dutoit, Alexis Moinet, Catherine Pelachaud, Benjamin Picart, Joëlle Tilmanne, and Johannes Wagner (2010). “AVLaughterCycle”. In: *J. Multimodal User Interfaces* 4.1, pp. 47–58.

Valitutti, Alessandro and Tony Veale (2015). "Inducing an ironic effect in automated tweets". In: *2015 International Conference on Affective Computing and Intelligent Interaction, ACII 2015, Xi'an, China, September 21-24, 2015*. IEEE Computer Society, pp. 153–159.

Vilk, John and Naomi T. Fitter (2020). "Comedians in Cafes Getting Data: Evaluating Timing and Adaptivity in Real-World Robot Comedy Performance". In: *HRI '20: ACM/IEEE International Conference on Human-Robot Interaction, Cambridge, United Kingdom, March 23-26, 2020*. Ed. by Tony Belpaeme, James E. Young, Hatice Gunes, and Laurel D. Riek. ACM, pp. 223–231.

Vinciarelli, Alessandro, Maja Pantic, and Hervé Bourlard (2009). "Social signal processing: Survey of an emerging domain". In: *Image Vision Comput.* 27.12, pp. 1743–1759.

Vogt, Thurid, Elisabeth André, and Nikolaus Bee (2008). "EmoVoice - A Framework for Online Recognition of Emotions from Voice". In: *Perception in Multimodal Dialogue Systems, 4th IEEE Tutorial and Research Workshop on Perception and Interactive Technologies for Speech-Based Systems, PIT 2008, Kloster Irsee, Germany, June 16-18, 2008, Proceedings*. Ed. by Elisabeth André, Laila Dybkjær, Wolfgang Minker, Heiko Neumann, Roberto Pieraccini, and Michael Weber. Vol. 5078. Lecture Notes in Computer Science. Springer, pp. 188–199.

w3.org (2021). *Speech Synthesis Markup Language (SSML) Version 1.1*. https://www.w3.org/TR/speech-synthesis11/. Accessed: 2021-04-14.

Wada, Kazuyoshi and Takanori Shibata (2006). "Robot Therapy in a Care House - its Sociopsychological and Physiological Effects on the Residents". In: *Proceedings of the 2006 IEEE International Conference on Robotics and Automation, ICRA 2006, May 15-19, 2006, Orlando, Florida, USA*. IEEE, pp. 3966–3971.

Wagner, Johannes, Tobias Baur, Dominik Schiller, Yue Zhang, Björn Schuller, Michel Valstar, and Elisabeth Andre (2018). "Show Me What You've Learned: Applying Cooperative Machine Learning for the Semi-Automated Annotation of Social Signals". In: *IJCAI/ECAI, Workshop on XAI 2018*.

Wagner, Johannes, Florian Lingenfelser, Tobias Baur, Ionut Damian, Felix Kistler, and Elisabeth André (2013). "The social signal interpretation (SSI) framework: multimodal signal processing and recognition in real-time". In: *ACM Multimedia Conference, MM '13, Barcelona, Spain, October 21-25, 2013*. Ed. by Alejandro Jaimes, Nicu Sebe, Nozha Boujemaa, Daniel Gatica-Perez, David A. Shamma, Marcel Worring, and Roger Zimmermann. ACM, pp. 831–834.

Walker, Marilyn A., Diane J. Litman, Candace A. Kamm, and Alicia Abella (1997). "PARADISE: A Framework for Evaluating Spoken Dialogue Agents". In: *35th Annual Meeting of the Association for Computational Linguistics and 8th Conference of the European Chapter of the Association for Computational Linguistics, Proceedings of the Conference, 7-12 July 1997, Universidad Nacional de Educación a Distancia (UNED), Madrid, Spain*. Ed. by Philip R. Cohen and Wolfgang Wahlster. Morgan Kaufmann Publishers / ACL, pp. 271–280.

Waller, Annalu, Rolf Black, David A. O'Mara, Helen Pain, Graeme Ritchie, and Ruli Manurung (2009). "Evaluating the STANDUP Pun Generating Software with Children with Cerebral Palsy". In: *TACCESS* 1.3, 16:1–16:27.

Weber, Klaus, Hannes Ritschel, Ilhan Aslan, Florian Lingenfelser, and Elisabeth André (2018a). "How to Shape the Humor of a Robot - Social Behavior Adaptation Based on Reinforcement Learning". In: *Proceedings of the 2018 on International Conference on Multimodal Interaction, ICMI 2018, Boulder, CO, USA, October 16-20, 2018*. Ed. by Sidney K. D'Mello, Panayiotis G. Georgiou, Stefan Scherer, Emily Mower Provost, Mohammad Soleymani, and Marcelo Worsley. ACM, pp. 154–162.

Weber, Klaus, Hannes Ritschel, Florian Lingenfelser, and Elisabeth André (2018b). "Real-Time Adaptation of a Robotic Joke Teller Based on Human Social Signals". In: *Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems, AAMAS 2018,* *Stockholm, Sweden, July 10-15, 2018*. Ed. by Elisabeth André, Sven Koenig, Mehdi Dastani, and Gita Sukthankar. International Foundation for Autonomous Agents and Multiagent Systems Richland, SC, USA / ACM, pp. 2259–2261.

Wennerstrom, Ann (2001). *The music of everyday speech: Prosody and discourse analysis*. Oxford University Press.

Wetzel, Christopher G and Chester A Insko (1982). "The similarity-attraction relationship: Is there an ideal one?" In: *Journal of Experimental Social Psychology* 18.3, pp. 253–276.

Whittaker, Steve, Yvonne Rogers, Elena Petrovskaya, and Hongbin Zhuang (2021). "Designing Personas for Expressive Robots: Personality in the New Breed of Moving, Speaking, and Colorful Social Home Robots". In: *J. Hum.-Robot Interact.* 10.1.

Whitworth, Brian (2005). "Polite computing". In: *Behaviour & Information Technology* 24.5, pp. 353–363.

Whitworth, Brian and Tong Liu (2009). "Politeness as a social computing requirement". In: *Human computer interaction: Concepts, methodologies, tools, and applications*. IGI Global, pp. 2675–2692.

Williams, Jason A, Erin L Burns, and Elizabeth A Harmon (2009). "Insincere utterances and gaze: eye contact during sarcastic statements". In: *Perceptual and motor skills* 108.2, pp. 565–572.

Williams, John L (1971). "Personal space and its relation to extraversion-introversion." In: *Canadian Journal of Behavioural Science/Revue canadienne des sciences du comportement* 3.2, p. 156.

Williams, Tom, Daniel H. Grollman, Mingyuan Han, Ryan Blake Jackson, Jane Lockshin, Ruchen Wen, Zachary Nahman, and Qin Zhu (2020). ""Excuse Me, Robot": Impact of Polite Robot Wakewords on Human-Robot Politeness". In: *Social Robotics - 12th International Conference, ICSR 2020, Golden, CO, USA, November 14-18, 2020, Proceedings*. Ed. by Alan R. Wagner, David Feil-Seifer, Kerstin Sophie Haring, Silvia Rossi, Thomas Williams, Hongsheng He, and Shuzhi Sam Ge. Vol. 12483. Lecture Notes in Computer Science. Springer, pp. 404–415.

Wilson, Deirdre and Dan Sperber (1992). "On verbal irony". In: *Lingua* 87.1, pp. 53 –76.

Woods, Sarah, Kerstin Dautenhahn, Christina Kaouri, Rene te Boekhorst, and Kheng Lee Koay (2005). "Is this robot like me? Links between human and robot personality traits". In: *5th IEEE-RAS International Conference on Humanoid Robots, Humanoids 2005, Tsukuba, Japan, December 5-7, 2005*. IEEE, pp. 375–380.

Yang, Chiao-Yu, Ming-Jen Lu, Shih-Huan Tseng, and Li-Chen Fu (2017). "A companion robot for daily care of elders based on homeostasis". In: *2017 56th Annual Conference of the Society of Instrument and Control Engineers of Japan (SICE)*, pp. 1401–1406.

Yeong Tan, Diana Tze and Ramadhar Singh (1995). "Attitudes and attraction: A developmental study of the similarity-attraction and dissimilarity-repulsion hypotheses". In: *Personality and Social Psychology Bulletin* 21.9, pp. 975–986.

Yoon, Youngwoo, Woo-Ri Ko, Minsu Jang, Jaeyeon Lee, Jaehong Kim, and Geehyuk Lee (2019). "Robots Learn Social Skills: End-to-End Learning of Co-Speech Gesture Generation for Humanoid Robots". In: *International Conference on Robotics and Automation, ICRA 2019, Montreal, QC, Canada, May 20-24, 2019*. IEEE, pp. 4303–4309.

Zarinbal, Marzieh, Azadeh Mohebi, Hesamoddin Mosalli, Razieh Haratinik, Zahra Jabalameli, and Farnoush Bayatmakou (2019). "A New Social Robot for Interactive Query-Based Summarization: Scientific Document Summarization". In: *Interactive Collaborative Robotics - 4th International Conference, ICR 2019, Istanbul, Turkey, August 20-25, 2019, Proceedings*. Ed. by Andrey Ronzhin, Gerhard Rigoll, and Roman V. Meshcheryakov. Vol. 11659. Lecture Notes in Computer Science. Springer, pp. 330–340.

Zhang, Zixing, Jing Han, Kun Qian, and Björn W. Schuller (2018). “Evolving Learning for Analysing Mood-Related Infant Vocalisation”. In: *Interspeech 2018, 19th Annual Conference of the International Speech Communication Association, Hyderabad, India, 2-6 September 2018*. Ed. by B. Yegnanarayana. ISCA, pp. 142–146.

A. Publications and Reviews

Some contents of this thesis have appeared partially in the following peer-reviewed publications and book chapters.

A.1. Conference and Workshop Papers

Kiderle, Thomas, Hannes Ritschel, Kathrin Janowski, Silvan Mertes, Florian Lingenfelser, and Elisabeth André (2021). "Socially-Aware Personality Adaptation". In: *2021 9th International Conference on Affective Computing and Intelligent Interaction Workshops and Demos (ACIIW)*, pp. 1–8.

Ritschel, Hannes (2018). "Socially-Aware Reinforcement Learning for Personalized Human-Robot Interaction". In: *Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems, AAMAS 2018, Stockholm, Sweden, July 10-15, 2018*. Ed. by Elisabeth André, Sven Koenig, Mehdi Dastani, and Gita Sukthankar. International Foundation for Autonomous Agents and Multiagent Systems Richland, SC, USA / ACM, pp. 1775–1777.

Ritschel, Hannes and Elisabeth André (2017). "Real-Time Robot Personality Adaptation based on Reinforcement Learning and Social Signals". In: *Companion of the 2017 ACM/IEEE International Conference on Human-Robot Interaction, HRI 2017, Vienna, Austria, March 6-9, 2017*. Ed. by Bilge Mutlu, Manfred Tscheligi, Astrid Weiss, and James E. Young. ACM, pp. 265–266.

Ritschel, Hannes and Elisabeth André (Nov. 2018). "Shaping a social robot's humor with Natural Language Generation and socially-aware reinforcement learning". In: *Proceedings of the Workshop on NLG for Human–Robot Interaction*. Tilburg, The Netherlands: Association for Computational Linguistics, pp. 12–16.

Ritschel, Hannes, Ilhan Aslan, Silvan Mertes, Andreas Seiderer, and Elisabeth André (2019a). "Personalized Synthesis of Intentional and Emotional Non-Verbal Sounds for Social Robots". In: *8th International Conference on Affective Computing and Intelligent Interaction, ACII 2019, Cambridge, United Kingdom, September 3-6, 2019*. IEEE, pp. 1–7.

Ritschel, Hannes, Ilhan Aslan, David Sedlbauer, and Elisabeth André (2019b). "Irony Man: Augmenting a Social Robot with the Ability to Use Irony in Multimodal Communication with Humans". In: *Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent Systems, AAMAS '19, Montreal, QC, Canada, May 13-17, 2019*. Ed. by Edith Elkind, Manuela Veloso, Noa Agmon, and Matthew E. Taylor. International Foundation for Autonomous Agents and Multiagent Systems, pp. 86–94.

Ritschel, Hannes, Tobias Baur, and Elisabeth André (2017). "Adapting a Robot's linguistic style based on socially-aware reinforcement learning". In: *26th IEEE International Symposium on Robot and Human Interactive Communication, RO-MAN 2017, Lisbon, Portugal, August 28 - Sept. 1, 2017*. IEEE, pp. 378–384.

Ritschel, Hannes, Tobias Baur, and Elisabeth André (2017). “Personalisierung der Mensch-Roboter-Interaktion durch sozial-sensitives Lernen”. In: *2. VDI-Fachkonferenz Humanoide Roboter 2017, Aschheim, Deutschland, 5.-6. Dezember 2017*.

Ritschel, Hannes, Kathrin Janowski, Andreas Seiderer, Stefan Wagner, and Elisabeth André (2019c). “Insights on usability and user feedback for an assistive robotic health companion with adaptive linguistic style”. In: *Proceedings of the 12th ACM International Conference on PErvasive Technologies Related to Assistive Environments, PETRA 2019, Island of Rhodes, Greece, June 5-7, 2019*. Ed. by Fillia Makedon. ACM, pp. 319–320.

Ritschel, Hannes, Thomas Kiderle, and Elisabeth André (2021). “Implementing Parallel and Independent Movements for a Social Robot’s Affective Expressions”. In: *2021 9th International Conference on Affective Computing and Intelligent Interaction Workshops and Demos (ACIIW)*, pp. 1–4.

Ritschel, Hannes, Thomas Kiderle, Klaus Weber, and Elisabeth André (Mar. 2020a). “Multimodal Joke Presentation for Social Robots based on Natural-Language Generation and Nonverbal Behaviors”. In: *Proceedings of the 2nd Workshop on NLG for Human–Robot Interaction*. Cambridge, UK.

Ritschel, Hannes, Thomas Kiderle, Klaus Weber, Florian Lingenfelser, Tobias Baur, and Elisabeth André (2020b). “Multimodal Joke Generation and Paralinguistic Personalization for a Socially-Aware Robot”. In: *Advances in Practical Applications of Agents, Multi-Agent Systems, and Trustworthiness. The PAAMS Collection - 18th International Conference, PAAMS 2020, L’Aquila, Italy, October 7-9, 2020, Proceedings*. Ed. by Yves Demazeau, Tom Holvoet, Juan M. Corchado, and Stefania Costantini. Vol. 12092. Lecture Notes in Computer Science. Springer, pp. 278–290.

Ritschel, Hannes, Andreas Seiderer, and Elisabeth André (Mar. 2020). “Pianobot: An Adaptive Robotic Piano Tutor”. In: *Proceedings of the Workshop on Exploring Creative Content in Social Robotics*. Cambridge, UK.

Ritschel, Hannes, Andreas Seiderer, Kathrin Janowski, Ilhan Aslan, and Elisabeth André (2018). “Drink-O-Mender: An Adaptive Robotic Drink Adviser”. In: *Proceedings of the 3rd International Workshop on Multisensory Approaches to Human-Food Interaction, MHFI@ICMI 2018, Boulder, CO, USA, October 16, 2018*. Ed. by Anton Nijholt, Carlos Velasco, Marianna Obrist, Katsunori Okajima, and Charles Spence. ACM, 3:1–3:8.

Ritschel, Hannes, Andreas Seiderer, Kathrin Janowski, Stefan Wagner, and Elisabeth André (2019d). “Adaptive linguistic style for an assistive robotic health companion based on explicit human feedback”. In: *Proceedings of the 12th ACM International Conference on PErvasive Technologies Related to Assistive Environments, PETRA 2019, Island of Rhodes, Greece, June 5-7, 2019*. Ed. by Fillia Makedon. ACM, pp. 247–255.

Seiderer, Andreas, Hannes Ritschel, and Elisabeth André (2020). “Development of a Privacy-By-Design Speech Assistant Providing Nutrient Information for German Seniors”. In: *GoodTechs ’20: 6th EAI International Conference on Smart Objects and Technologies for Social Good, Antwerp, Belgium, September 14-16, 2020*. Ed. by Catia Prandi and Johann Márquez-Barja. ACM, pp. 114–119.

Weber, Klaus, Hannes Ritschel, Ilhan Aslan, Florian Lingenfelser, and Elisabeth André (2018a). “How to Shape the Humor of a Robot - Social Behavior Adaptation Based on Reinforcement Learning”. In: *Proceedings of the 2018 on International Conference on Multimodal Interaction, ICMI 2018, Boulder, CO, USA, October 16-20, 2018*. Ed. by

Sidney K. D'Mello, Panayiotis G. Georgiou, Stefan Scherer, Emily Mower Provost, Mohammad Soleymani, and Marcelo Worsley. ACM, pp. 154–162.

Weber, Klaus, Hannes Ritschel, Florian Lingenfelser, and Elisabeth André (2018b). "Real-Time Adaptation of a Robotic Joke Teller Based on Human Social Signals". In: *Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems, AAMAS 2018, Stockholm, Sweden, July 10-15, 2018*. Ed. by Elisabeth André, Sven Koenig, Mehdi Dastani, and Gita Sukthankar. International Foundation for Autonomous Agents and Multiagent Systems Richland, SC, USA / ACM, pp. 2259–2261.

A.2. Book Chapters

Janowski, Kathrin, Hannes Ritschel, and Elisabeth André (2022). "Adaptive Artificial Personalities". In: *The Handbook on Socially Interactive Agents: 20 Years of Research on Embodied Conversational Agents, Intelligent Virtual Agents, and Social Robotics Volume 2: Interactivity, Platforms, Application*. 1st ed. New York, NY, USA: Association for Computing Machinery, 155–194.

Janowski, Kathrin, Hannes Ritschel, Birgit Lugrin, and Elisabeth André (2018). "Sozial interagierende Roboter in der Pflege". In: *Pflegeroboter*. Ed. by Oliver Bendel. Wiesbaden: Springer Fachmedien Wiesbaden, pp. 63–87.

A.3. Awards

Best Technical Paper

Ritschel, Hannes, Andreas Seiderer, Kathrin Janowski, Stefan Wagner, and Elisabeth André (2019d). "Adaptive linguistic style for an assistive robotic health companion based on explicit human feedback". In: *Proceedings of the 12th ACM International Conference on PErvasive Technologies Related to Assistive Environments, PETRA 2019, Island of Rhodes, Greece, June 5-7, 2019*. Ed. by Fillia Makedon. ACM, pp. 247–255.

Best Poster

Ritschel, Hannes, Kathrin Janowski, Andreas Seiderer, Stefan Wagner, and Elisabeth André (2019c). "Insights on usability and user feedback for an assistive robotic health companion with adaptive linguistic style". In: *Proceedings of the 12th ACM International Conference on PErvasive Technologies Related to Assistive Environments, PETRA 2019, Island of Rhodes, Greece, June 5-7, 2019*. Ed. by Fillia Makedon. ACM, pp. 319–320.

A.4. Honorable Mentions

Best Paper Runner-Up

Weber, Klaus, Hannes Ritschel, Ilhan Aslan, Florian Lingenfelser, and Elisabeth André (2018a). “How to Shape the Humor of a Robot - Social Behavior Adaptation Based on Reinforcement Learning”. In: *Proceedings of the 2018 on International Conference on Multimodal Interaction, ICMI 2018, Boulder, CO, USA, October 16-20, 2018*. Ed. by Sidney K. D’Mello, Panayiotis G. Georgiou, Stefan Scherer, Emily Mower Provost, Mohammad Soleymani, and Marcelo Worsley. ACM, pp. 154–162.

A.5. Reviews

- “Autonomous Agents and Multi-Agent Systems” 2018, 2019, 2020 (Journal, Springer)
- “Frontiers in Robotics and AI” 2020 (Journal, Frontiers)
- “IEEE Transactions on Affective Computing” 2021 (Journal, IEEE)
- “Human-Robot Interaction” 2017 (Conference)
- “International Conference on Social Robotics” 2015 (Conference)

B. Teaching

B.1. Lectures, Practical Courses, and Seminars

Reinforcement Learning (Lecture & Tutorial) The Master lecture was developed by the thesis author from ground up based on the book by Sutton and Barto. It covers foundations, algorithms and techniques, which are applied, practiced and implemented in the tutorial. The course was held by the thesis author in all winter terms from 2015 to 2022.

Reinforcement Learning (Practical Course) The Master practical course was developed by the thesis author from ground up and covers selected contents of the lecture. Topics were changed annually, including applications in the context of HCI and HRI. The course was held by the thesis author in all summer terms from 2016 to 2022.

Multimedia Project (Practical Course) The Bachelor course covers varying topics on multimedia. The focus is on implementing a working prototype in the context of HCI or HRI. The thesis author supervised student topics in all winter terms from 2017 to 2020.

Advanced Topics in Multimodal Dialogue and Interaction (Seminar) The thesis author supervised student topics in summer term 2015.

B.2. Supervised Student Theses

B.2.1. Bachelor Theses

Makowski, David (2018). “Visualisierung des internen Zustands eines autonomen Agenten mit Bestärkendem Lernen im Rahmen einer Handelssimulation”. University of Augsburg.

Mertes, Silvan (2018). “Melodiegenerierung auf Basis von Naive-Bayes-Klassifikation”. University of Augsburg.

Wagner, Stefan (2016). “Natürliches Blickverhalten für den Reeti-Roboter”. University of Augsburg.

B.2.2. Master Theses

Kiderle, Thomas (2019). “Multimodaler Ausdruck von dynamisch generiertem Humor in der Mensch-Roboter-Interaktion”. University of Augsburg.

Krüger, Simon (2022). “Advanced Persistent Threat Emulation with Reinforcement Learning”. University of Augsburg.
Mastaller, Dominik (2019). “Multimodaler Ausdruck von Extraversion durch einen sozialen Roboter in einem Storytelling Szenario”. University of Augsburg.
Mertes, Silvan (2019). “Personalisierte Soundgenerierung durch Kombination von erzeugenden gegnerischen Netzwerken und evolutionären Algorithmen”. University of Augsburg.
Rothmeier, Thomas (2018). “Generating melodies using deep reinforcement learning based on rewards from music theory and human preferences”. University of Augsburg.
Schmid, Michael (2017). “Optimierung von Bestärkendem Lernen in Dialogsystemen durch Wizard-of-Oz-Experimente am Beispiel einer Telefonbuchsuche”. University of Augsburg.
Sedlbauer, David (2018). “Natürlichsprachliche Generierung und multimodaler Ausdruck von Ironie für einen Chat-basierten sozialen Roboter”. University of Augsburg.
Wagner, Stefan (2019). “Sprachliche Anpassung von Höflichkeit und Rolle für einen den Alltag unterstützenden, sozialen Roboter”. University of Augsburg.
Weber, Klaus (2017). “Adaption eines sozialen Roboters auf Basis von Bestärkendem Lernen mit linearer Funktionsapproximation und sozialen Signalen”. University of Augsburg.