Real-Time Topic and Sentiment Analysis in Human-Robot Conversation

Elise Russell
Marquette University

REAL-TIME TOPIC AND SENTIMENT ANALYSIS IN HUMAN-ROBOT CONVERSATION

by

Elise W. B. Russell, B.A.

A Thesis submitted to the Faculty of the Graduate School,
Marquette University,
in Partial Fulfillment of the Requirements for
the Degree of Master of Science

Milwaukee, Wisconsin

December 2015

ABSTRACT
REAL-TIME TOPIC AND SENTIMENT ANALYSIS IN HUMAN-ROBOT CONVERSATION

Elise W. B. Russell, B.A.

Marquette University, 2015

Socially interactive robots, especially those designed for entertainment and companionship, must be able to hold conversations with users that feel natural and engaging for humans. Two important components of such conversations include adherence to the topic of conversation and inclusion of affective expressions. Most previous approaches have concentrated on topic detection or sentiment analysis alone, and approaches that attempt to address both are limited by domain and by type of reply. This thesis presents a new approach, implemented on a humanoid robot interface, that detects the topic and sentiment of a user's utterances from text-transcribed speech. It also generates domain-independent, topically relevant verbal replies and appropriate positive and negative emotional expressions in real time.

The front end of the system is a smartphone app that functions as the robot's face. It displays emotionally expressive eyes, transcribes verbal input as text, and synthesizes spoken replies. The back end of the system is implemented on the robot's onboard computer. It connects with the app via Bluetooth, receives and processes the transcribed input, and returns verbal replies and sentiment scores. The back end consists of a topic-detection subsystem and a sentiment-analysis subsystem. The topic-detection subsystem uses a Latent Semantic Indexing model of a conversation corpus, followed by a search in the online database ConceptNet 5, in order to generate a topically relevant reply. The sentiment-analysis subsystem disambiguates the input words, obtains their sentiment scores from SentiWordNet, and returns the averaged sum of the scores as the overall sentiment score.

The system was hypothesized to engage users more with both subsystems working together than either subsystem alone, and each subsystem alone was hypothesized to engage users more than a random control. In computational evaluations, each subsystem performed weakly but positively. In user evaluations, users reported a higher level of topical relevance and emotional appropriateness in conversations in which the subsystems were working together, and they reported higher engagement especially in conversations in which the topic-detection system was working. It is concluded that the system partially fulfills its goals, and suggestions for future work are presented.

ACKNOWLEDGMENTS

Elise W. B. Russell, B.A.

I would like to thank Elizabeth Russell, Meg Russell, Allen Russell, John Russell, and Daniel Homernik for a great deal of support and patience, and also for inspiring conversations and editorial assistance. I would like to thank the students of the H.E.I.R Lab at Marquette University, particularly Matthew Morris, for the use of their robots and for their extensive contributions toward the success of this project. My thanks also to Paul D. Gessler, the creator of the LaTeXclass `mugsthesis`, which was very helpful in the writing of this paper. I owe a great debt to my committee members, Dr. Richard J. Povinelli, Dr. Henry Medeiros, and Dr. Andrew B. Williams, for their guidance and support. I am also extremely grateful to my advisor, Dr. Williams, for giving me the opportunity to pursue my first degree in engineering.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 Problem Statement

This study attempts to augment a conversational robot interface such that it produces topically relevant replies and emotionally appropriate expressions during real-time interactions with users. To do this, the interface is integrated with a back-end system composed of basic topic-detection and sentiment-analysis subsystems, which work in tandem to interpret user utterances and to generate relevant verbal and emotional reactions.

A successful topic-detection subsystem is defined as one which generates replies that users judge to be more topically relevant than the replies of a random control. Similarly, a successful sentiment-analysis subsystem is defined as one which generates emotional expressions that users judge to be more appropriate to the user's words than that of a random control.

The integration of both successful subsystems with the robot's interface is hypothesized to be more engaging to users than the integration of either subsystem alone, and either subsystem alone is hypothesized to be more engaging to users than the random control.

1.2 Rationale

Research reported in Fong et al.'s review suggests that emotions, dialogue, and personality are among the necessary characteristics of successful socially interactive robots [1]. According to at least one study that Fong reported, displays of emotion make the robot "more compelling to interact with," and emotions are also described as helping to "facilitate believable human-robot interaction."

Dialogue, aside from helping to coordinate interactions with humans, is also useful for conveying the robot's personality. As Fong writes, "There is reason to believe that if a robot had a compelling personality, people would be more willing to interact with it and to establish a relationship with it." Thus, user engagement can plausibly be tied to a robot's ability to convey emotion and personality in dialogue.

In her book *Designing Sociable Robots*, Breazeal remarks that the commercial success of domestic robots "hinges on their ability to be part of a person's daily life. As a result, the robots must be responsive to and interact with people in a natural and intuitive manner" [2]. Additionally, in his review of verbal and non-verbal human-robot communication, Mavridis suggests ten desiderata of a conversational robot, which include "breaking the 'simple commands only' barrier," "mixed initiative dialogue," and "affective interaction" [3]. He suggests that the development of robots implementing these desiderata would allow them to cooperate with humans without requiring these users to adapt their behavior, and he comments on "the desirability of natural fluid interaction with humans." Thus it is sensible to pursue conversational ability in the effort to engage normal human users with a robotic agent.

1.3 Project Goals and Scope

The scope of this project is the improvement of a target robot's conversational interactions by the implementation of a basic, real-time input analysis and reaction system. Since the goal is user engagement, personality is emphasized in the way in which these reactions are generated. Additionally, this research forms a foundation for the development of later, more elaborate conversational systems on the target robot's interface, which requires certain workarounds due to its construction.

The target robot is a 3-foot-tall humanoid robot known as the MU-L8 robot of the H.E.I.R Lab [4]. This robot, pictured in Figure 1.1, was designed and built with convenience and human interaction in mind, especially interaction with children. Although it can interact with users emotionally and verbally via the smartphone in its face, these interactions were previously restricted to templated and predictable formats [5]. This project aims to expand the MU-L8's form of conversation such that it can reply to user speech in new, less predictable, and more engaging ways.

The MU-L8 robot's smartphone interface is called the SMILE app, and it is described in more detail in Section 4.1 [6, 7]. The app has one main input modality: it collects user speech by transcribing it into text, a process which can often be uncertain or incomplete. The interface's reactions are limited to a set of six categorical emotional expressions, as well as the synthesis of spoken verbal replies.

Therefore, in order to map the given input to the allowed outputs, this research is required to find one mapping from uncertain text-transcribed user utterances to appropriate emotional expressions, and another mapping from uncertain text-transcribed user utterances to relevant verbal replies. A simple outline of the mapping is as follows:

- Transcribed user utterance → Analyze for sentiment → Generate matching emotional expression
- Transcribed user utterance → Analyze for topic → Generate relevant verbal reply

A basic sentiment-analysis system is required for the first mapping and a basic topic-detection system for the second mapping.

Since the goal of this project is to engage users in conversation with the robot without constraining their conversations to specific topics, the goal of the

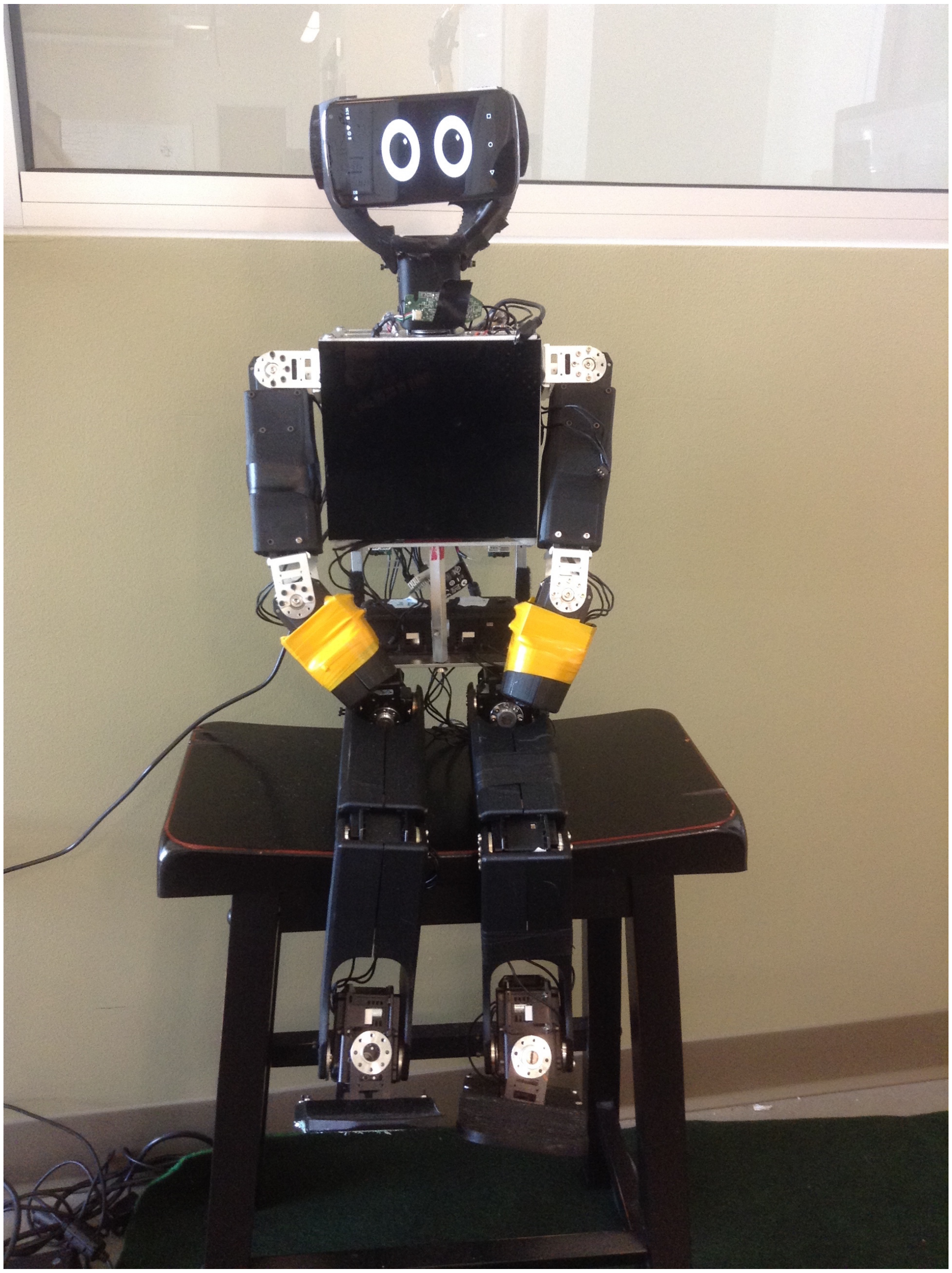


Figure 1.1: The MU-L8 humanoid robot.

topic-detection task is not simply to classify speech inputs into predetermined topical categories, but rather to use the content of these inputs to generate *topically related* replies. From previous research by this author (currently in review for publishing), it was determined that in order to convey personality, such replies should avoid rigid templates and repetition, go beyond rephrasing or generalizing the content of the user's utterance, and preferably contain some element of entertainment or humor [8].

For these reasons, a balance must be struck in reply generation. The replies must be relevant enough to the user's speech that the user perceives that the robot is reacting to what was said, but the replies must also contain enough new or unexpected information that the user has something to talk about in their next turn, thus encouraging continued conversation.

Therefore, in selecting the approaches to implement for these two subsystems, approaches that meet the following criteria are preferred:

- Widely available and relatively simple to integrate.
- Capable of processing an utterance and generating a reaction in a matter of four seconds or less.
- Unrestricted by any particular topical domain or set of categories.
- Capable of generating verbal reactions in a non-deterministic way.
- Capable of generating emotional reactions that correspond to some subset of the robot's expressions.

Due to these restrictions, rule-based reply methods such as those used in chatbots are avoided, as are domain-specific sentiment analysis classifiers and any approach that cannot be operated in close to real time.

Once integrated with the robot's interface, the resulting system is evaluated both computationally and by human users. Each subsystem is evaluated separately, and the integrated system is also evaluated as a whole.

1.4 Organization of the Thesis

This thesis is organized in the following manner. In Chapter 2, a literature review discusses the three fields of study that contribute to this work, and notable advances in each field are described. In Chapter 3, an overview is given of the tools, databases, and software packages that are used specifically in this work, along with the details of their contents and interfaces.

In Chapter 4, the robot's smartphone interface is described, followed by the back-end system created and integrated with the interface. This section is split into a discussion of the query-handling processes, both for the topic-detection subsystem and for the sentiment-analysis subsystem, as well as a description of the system build that supports these processes. A sample dialogue with the completed system is presented at the end of the chapter.

In Chapter 5, the evaluation of the resulting system is described. Methods, results, and analyses are presented and discussed for both the computational "Off-Line" evaluations of each subsystem and for the user experience study that constituted the "On-Line" evaluation. In Chapter 6, the results of the research overall are discussed and summed up, and plans for future work are presented.

CHAPTER 2

LITERATURE REVIEW

Three main areas of study inform this research. These are Topic Detection, Sentiment Analysis, and Human-Robot Interaction.

2.1 Topic Detection

The field of Topic Detection is interested in automatically categorizing or clustering documents, sections of documents, news stories, or conversational utterances using natural language processing and often machine learning techniques [9, 10]. Some applications are interested in segmenting documents by topic or extracting arguments from text [11, 12], while others are concerned with classifying documents according to pre-labeled sets of topics [13, 14], and others aim to extract categories and clusters of documents from the data given [15, 16].

Sebastiani conducted a review of different text classification techniques and their performance on different versions of the Reuters corpus; depending on version, the best results achieved F_1 scores of .753 to .920, using classifiers ranging from decision rules to support vector machines to AdaBoost [9]. A more recent work using domain kernels achieves an F_1 score of .928 on one of the same Reuters corpus versions [17]. While this level of accuracy is impressive, these works use static sets of predefined topic categories in which to classify their documents, and are therefore not useful to the current project.

Topic Detection and Tracking, a more specific application within the field, attempts to handle a stream of new documents by either labeling them with previously discovered topics or by determining that they belong to new topics and creating these new labels as needed [18]. Often these techniques use news stories as domain data, and the attempt is to classify which news stories talk about previously detected events and which ones talk about new, breaking news events [19]. Although these works resemble the current project in that they deal with a temporal domain and a dynamic set of topics, they are restricted by their focus on event-based documents.

The current project therefore turns to unsupervised learning techniques to model and explore conversational topical content without the use of predetermined topic sets. One notable set of unsupervised, corpus-based learning techniques is Latent Semantic Analysis and its more statistically grounded derivatives, Probabilistic Latent Semantic Analysis and Latent Dirichlet Allocation [20, 21, 22]. These techniques use statistical and matrix-based methods to calculate topic models and topic distributions based on word co-occurrences. Latent Semantic Analysis, the technique used in the current project, is described in detail in Section 3.1.

2.2 Sentiment Analysis

Sentiment Analysis is a natural language processing sub-field that is concerned with extracting opinions or sentiments from text data, for example from consumer reviews of movies or products, from articles about politics, or even from microblogging posts on Twitter [23, 24]. Companies and political parties especially are interested in aggregating sentiment data about their products and images in order to understand the effects of their branding and develop new strategies. In addition, consumers benefit from a high-level view of general opinion about products and services that they may be interested in purchasing [25].

Major techniques include simple lexicon-based approaches, machine-learning approaches, and conceptual approaches [26]. Lexicon-based approaches attempt to determine the positivity or negativity of a document from its vocabulary content, by comparing its content to a labeled dictionary; such a dictionary could be created manually or automatically [27, 28]. Machine learning approaches use corpora of labeled documents to build classifiers, and they are generally concerned with finding the best text features to use for their particular domain [29, 30]. Conceptual approaches build full ontologies of domain-specific knowledge with which to compare document content; this is often accomplished using seed-based algorithms and databases such as ConceptNet [31, 32]. Although most approaches are concerned with discrete, unrelated documents of various lengths found in large bodies on the Internet, a few applications also attempt to analyze sentiment in real-time online conversations, as well as in multi-modal data including audio and video [33, 34, 35, 36].

Since sentiment analysis is interested in binary (positive vs. negative) or ternary (positive vs. negative vs. objective) classifications, its state-of-the-art accuracy is quite high. In 2013, Mukherjee and Joshi achieved accuracies of 71.38% to 76.06% using their model based on a ConceptNet ontology [37], and Socher, et al., achieved accuracy as high as 86.4% on the MPQA opinion dataset using recursive auto-encoders [38].

Unfortunately, most applications are extremely domain-dependent; it can be difficult to apply a sentiment analysis algorithm developed on movie reviews, for example, to house-cleaning product reviews, which express positive and negative opinions in a very different way. However, a recent approach involving deep learning successfully crossed domain lines within product reviews for a set of 22 different types of products [39]. It remains to be seen whether this approach can be generalized to domains such as political discussions or microblogging.

The field of emotion detection overlaps with that of sentiment analysis, but it is neither a subset nor a superset of it. Emotion detection is concerned with recognizing and labeling human emotions given auditory, visual, or even textual data [40, 41]. There are debates in this field that parallel debates in the field of emotional psychology; specifically whether to model and label emotions using a categorical, dimensional, or cognitive, context-based model [42, 43]. Many approaches in this field deal with robots or embodied agents, and there are several multimodal approaches that incorporate analyses of features from different input domains [44, 45, 46].

Since the classifications in emotion detection tend to involve more categories and dimensions than sentiment analysis, the average accuracy of these approaches is somewhat lower. In 2008, Strapparava reviewed the accuracy of several recent text-based systems including variations on a system that the authors had created; each system attempted to score a set of news headlines for emotional content in six emotional categories (anger, disgust, fear, joy, sadness, and surprise) [47]. On average across all emotions, the UPAR7 system achieved the highest fine-grained Pearson correlation with gold standard scores, at $r = 28.38$. The highest coarse-grained precision and recall scores were achieved by two different versions of the authors' system, measured at 38.28% and 90.22%, respectively.

The current project deals with small documents consisting of text-transcribed conversational utterances; these documents are likely to average around 20 words each. An approach with such a dataset benefits from keeping both the task and the implementation simple in order to maximize speed and accuracy; for this reason, the problem was defined here as a ternary classification problem in which utterances are labeled as positive, negative, or neutral. A lexicon-based approach was adopted, using a state-of-the-art, publicly-available lexicon called SentiWordNet, which is described in detail in Section 3.5.

2.3 Human-Robot Interaction

The field of Human-Robot Interaction is concerned with improving the ability of robots to interact with humans in ways that humans find natural. In Fong, et al.'s review, this problem is broken up into design approaches for embodiment, emotion, dialogue, personality, human-oriented perception, user modeling, socially situated learning, and intentionality [1]. Breazeal describes the ideal sociable robot as

> ...able to communicate and interact with us, understand and even relate to us, in a personal way. It should be able to understand us and itself in social terms. We, in turn, should be able to understand it in the same social terms...In short, a sociable robot is socially intelligent in a human-like way, and interacting with it is like interacting with another person [2].

Recent approaches in conversational Human-Robot Interaction have differentiated between "conversational service robots" and "conversational entertainment robots" [48]. Among the former, advances have been made in dialogue and action planning [48] and in the parsing of a user's speech flow in order to determine the parts of the environment that the user is talking about [49].

The current project falls into the category of "conversational entertainment robot." For such robots, the purpose of dialogue is to entertain and engage users. One such robot was used to find a positive relationship between perceived enjoyment and intention-to-use among elderly populations [50], and another was designed to model turn-taking in multiparty conversations and to offer relevant information about baseball scores at the proper juncture [51]. Another robot, rather than participating in dialogue, uses turn-taking rules and perceived user speech content to offer appropriate back-channel feedback while listening [52].

However, the above robotic approaches are restricted by domain and by type of reply. A different approach created a simulated robot that is not restricted in this way, but interprets user expressions of interest to determine which subject to talk about next [53]. Although this robot's speech is not restricted by domain, it is generated directly from Wikipedia articles, and conversations with it amount to verbally browsing Wikipedia.

One other recent approach uses video information to analyze the emotional expressions on a user's face, and the robot then categorizes the user's speech with a Latent Semantic Analysis model built on a set of topical articles [46]. The robot's replies, however, are generated from these classifications using a rule base, which again restricts its reactions to a predetermined set.

No recent approaches have attempted to analyze both the emotional and the topical content of the user's text-transcribed speech and to form domain-independent reactions based on this information alone, as is attempted in the current project.

CHAPTER 3

OVERVIEW OF PROJECT RESOURCES

The following is an overview of methods, corpora, databases, and programming packages used directly in the implementation of this project.

3.1 Latent Semantic Indexing

Latent Semantic Analysis, also known as Latent Semantic Indexing or LSI, is a long-established technique in document retrieval and linguistic modeling. Introduced by Dumais et al., in 1988, it has proven to be a robust and versatile method for modeling semantic spaces in large corpora [21, 54], and its many uses range from indexing documents to simulating language acquisition [55].

LSI is implemented in the following manner, illustrated in Figure 3.1. A large set of documents is converted to a matrix representation in $n \times m$ matrix A, where each of the m documents in the corpus is represented by a column, and each of the n words in the corpus is represented by a row. Each cell a_{ij} contains a count of the number of times that word i appears in document j. A matrix formed by this method is usually very large and very sparse.

A weighting transformation is then applied to the matrix in order to balance the counts more fairly by the importance of the word in the corpus. Either the “term frequency – inverse document frequency” transformation or the “log entropy” transformation is usually preferred for this purpose.

The weighted matrix A then undergoes Singular Value Decomposition, or SVD, in which it is broken down into three matrices: the $n \times s$ matrix U, the $s \times s$ diagonal matrix S, and the $m \times s$ matrix V. These three matrices can be multiplied back together to produce the original matrix:

$$A = U \cdot S \cdot V^T \tag{3.1}$$

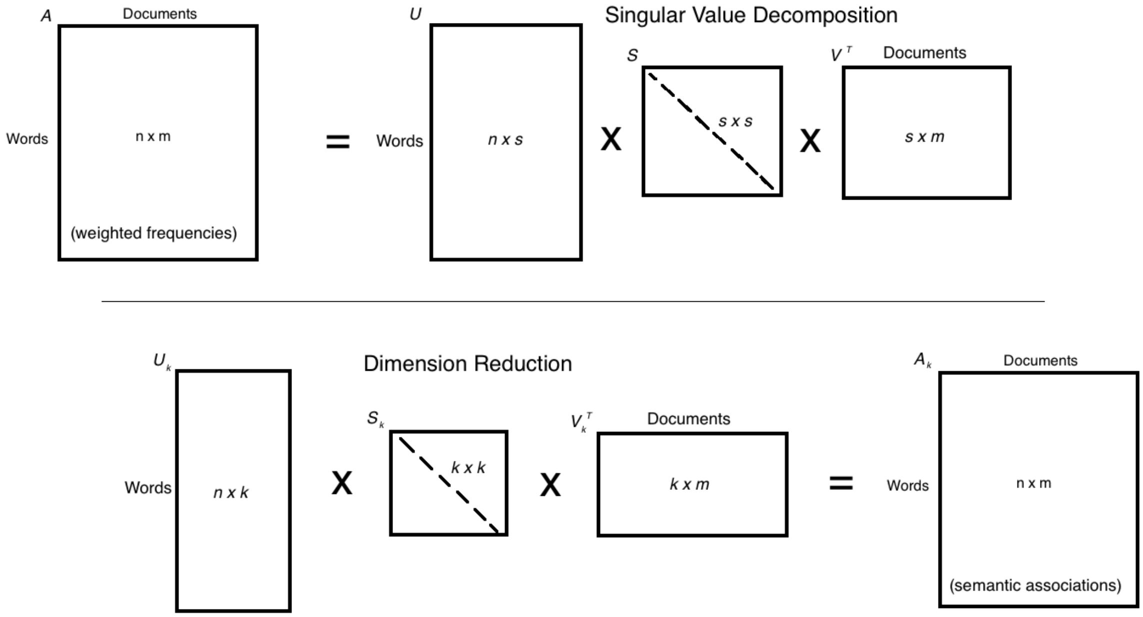


Figure 3.1: Illustration of the creation and dimension-reduction of an LSI model from a corpus of documents.

The rows of the U matrix correspond to the rows of A, that is, to the words in the corpus. The rows of the V matrix correspond to the columns of A, that is, to the documents in the corpus. The matrix S is a diagonal matrix with entries decreasing in magnitude. These entries, the singular values of the matrix A, each represent a topic in the model.

The next step is to reduce the influence of noise in the model by reducing its dimensionality to a predetermined size, k. This number is determined empirically and represents the predicted number of distinct, significant topics in the corpus; all others are assumed to be noise. Thus, all but the k largest entries in S are discarded by deleting the last $s - k$ rows and columns of S, resulting in the smaller diagonal matrix S_k. This effectively also discards the last $s - k$ columns in both U and V, producing shortened matrices U_k and V_k.

Words in the corpus may be compared to other words, and documents to other documents, by calculating the vector cosines of their respective vectors in

U_k or in V_k, respectively. These cosines are essentially semantic similarity scores. Additionally, the reduced matrices U_k, S_k, and V_k can be multiplied together to produce an approximation of the original matrix, A_k, with the same dimensionality as A but with smoothed-over data:

$$A_k = U_k \cdot S_k \cdot V_k^T \tag{3.2}$$

Each entry $a_{k:ij}$ now represents, essentially, the predicted number of times that word i should appear in document j, given the topical content of document j. In this way, documents can also be compared to words for semantic similarity.

A new document can be used to query the model by first being converted to a bag-of-words vector, a, equivalent to a column in the original matrix A. The same weighting function is applied to a, and it is then converted to vector v_k, equivalent to a row vector in V_k, by multiplication as follows:

$$v_k = a^T \cdot U_k \cdot S_k^{-1} \tag{3.3}$$

It can then be compared to other documents in the corpus using vector cosines with the other vectors in V_k.

Another use of the LSI model is to retrieve the top t words that are the most semantically related to a document. If the document is already in the model, then its vector in the reconstituted A_k matrix can be searched for the t cells with the highest values. The words corresponding to these rows in the matrix are the t words in the corpus that are the most semantically related to that document, and they may or may not actually appear in the document.

To get this list of words for a new document, the document is first converted to a bag-of-words vector a and transformed by the weighting function. Then, the following set of multiplications is used to directly on the weighted vector to find the document's reconstituted a_k vector, equivalent to a vector in the A_k matrix:

$$a_k = U_k \cdot S_k \cdot (a^T \cdot U_k \cdot S_k^{-1})^T \tag{3.4}$$

This vector can then be searched for its t highest-scored words, which are the t words with the strongest semantic association to the document.

Since the technique of Latent Semantic Indexing was first introduced, several further techniques have been developed based on it, with the goal of improving its probabilistic model and therefore its performance at document indexing and retrieval. These techniques most notably include Probabilistic Latent Semantic Indexing (pLSI) [22] and Latent Dirichlet Allocation (LDA) [20], and they do improve significantly upon LSI's performance.

However, LSI was chosen for the current project in part because of the relative speed of model creation and use, and primarily because the matrix calculations are accessible enough to produce direct document-to-word comparisons. The goal of this project was not to use the model for document retrieval, as is the goal of most implementations of LSI, pLSI, and LDA, but rather for the retrieval of words closely relating to a new document or query.

The `gensim` implementation of LSI was used in this project because it provides a simple and effective Python API, as well as memory-efficient, fast, and clean model creation and querying [56].

3.2 Fisher English Training Transcripts Corpus

The Fisher English Training corpus is a natural language speech corpus published in 2004 (Part 1) and 2005 (Part 2) by the Linguistic Data Consortium (LDC) to serve as a machine learning resource for automatic speech recognition [57]. The complete text transcripts of these speech recordings were published concurrently with the recordings, and they constitute the Fisher English Training Transcripts corpus, Parts 1 and 2.

With Parts 1 and 2 combined, the Transcripts corpus consists of 11,699 text-transcribed phone conversations between strangers who volunteered for the study in 2003. The participants consisted of over 12,000 English-speakers recruited from all over the United States, and each spoke in one to three conversations. Each conversation was at maximum 10 minutes long and began with a system-generated topic prompt drawn from a set of 40 (see Appendix A), although speakers could and did go off topic. Conversations were mediated by a "robot operator" at the LDC, which made the calls to participants, paired conversational partners, delivered topic prompts, and recorded the conversations.

The recorded calls were manually transcribed, about 12% by the LDC and the rest by BBN/Wordwave. Each transcription document consists of one conversation, in which the individual utterances are separated by line. Each line is marked with a set of timestamps denoting the start and end of the utterance, and a letter indicating which of the two speakers, A or B, is speaking the utterance. Sequential lines may be spoken by the same speaker.

The text of the conversations is all lowercase and contains no numerals. The only punctuation that appears consists of the following marks: apostrophes for contractions and possessives, hyphens for hyphenated words, hyphens to end tokens that denote unfinished words (such as wha- or th-), special markings for abbreviations (such as t._v. or u._s._a.), brackets around tokens that denote noises (such as [laughter] or [mn]), and double parentheses around uncertain transcriptions.

The following is a short selection from a conversation in the corpus, selected to demonstrate the types of tokens that often appear in the corpus:

190.16 191.74 B: i like um [lipsmack]
192.06 195.45 B: some of the shows they have on h._b._o. now i think they're better
195.74 200.17 B: but we don't get it so i have to go over to a friends apartment to watch it
197.67 200.73 A: yeah no i don't either [laughter] i don't either
201.48 204.99 A: i don't and you know some of these um
201.69 202.44 B: ((yeah))
205.31 207.77 A: survival shows you know
207.43 209.17 B: oh i can't watch them

The metadata for each conversation includes its ID number, the location where it was transcribed, and the topic prompt given for that conversation, among other information about the speakers which is not relevant to this project. The average number of utterances per conversation is 381.7, and the average number of tokens (words) per conversation is 1,920.9. Of the 11,699 conversations, the topic prompt is not recorded for 301 of them, or 2.6%.

As a set of text-transcribed natural language conversations about a wide variety of topics, this corpus forms an ideal machine learning database for the current project, which aims to deal with text-transcribed natural language data on domain-independent topics.

3.3 ConceptNet 5

ConceptNet 5 is the latest version of a large, open-source database of common-sense knowledge, currently developed and maintained by Luminoso Technologies, Inc., in collaboration with the MIT Media Lab [58]. The original ConceptNet was founded in the Media Lab as a crowd-sourced data gathering project called the Open Mind Common Sense project. Over the course of several versions, it has become a vast, multilingual project containing knowledge contributed by various databases, dictionaries, online games, and human users from around the world.

The knowledge in ConceptNet 5 is stored as a graph database: the nodes of the graph are "concepts" in the form of normalized words and phrases, and the relationships between the concepts are the edges. These relationships, or "relations," can be of several dozen different types, including "PartOf," "AtLocation," "DefinedAs," and "RelatedTo," or a negation of any of these.

As an example, two concepts in English might be "cat" and "chase_mouse," and a relation between them might be "cat Desires chase_mouse". Each such assertion is also associated with a weight denoting its certainty. For example, the weight of the above assertion is +3.322, a relatively high weight for the database. Negative weights indicate negated assertions, or assertions that are known to be *not* true. In general, a negative assertion known to be true is represented using a negated relation type, as in: "cat NotIsA dog."

Speer reported that by April 2012, ConceptNet 5:

> ...contains 12.5 million edges, representing about 8.7 million assertions connecting 3.9 million concepts. 2.78 million of the concepts appear in more than one edge...11.5 million of the edges contain at least one English concept [58].

Versions of the ConceptNet database have been used for a variety of machine learning and human-computer interaction purposes. It has been used in ontology-matching and creation [59, 32], emotion and sentiment detection [31, 37], and the generation of new associations and analogies via dimension reduction [60], among many other applications. It can be downloaded, built, and run in a variety of formats including SQL and JSON, or it can be queried over the web using the existing web API.

Web queries consist of links containing search terms, types, and filters, and they return JSON structures of the information found. One can search for a particular concept and a list of the *n* highest-weighted edges using that concept, or one can search for edges containing any start node, end node, edge type, or data source. For example, the search:

```
http://conceptnet5.media.mit.edu/data/5.4/search?
    start=/c/en/cat&end=/c/en/mouse&limit=10
```

will return a JSON structure containing the 10 highest-weighted edges starting with the concept "cat" and ending with the concept "mouse." Each edge contains a variety of meta-information, including the data source, the edge weight, the start and end nodes, the relation type, and a best-guess English sentence expressing the meaning of the edge, called the "surface text." As an example, the surface text for the edge "cat Desires chase_mouse" is "Cat wants to chase mice."

The current project uses the aforementioned web API for queries to ConceptNet 5, largely in order to save the space that would be required to download and run a version of the database on the robot. As long as the robot maintains an internet connection, it can use the ConceptNet 5 resource to search for common-sense relationships between words.

3.4 NLTK Text Processing Tools

When working with large amounts of text data, whether from a corpus or from user queries, it is often necessary to process the text to improve the richness and accuracy of the information gathered. Such processing can include tokenization, removal of punctuation and stopwords, part-of-speech tagging, stemming or lemmatization, and word-sense disambiguation. The Python package `NLTK` provides a wide variety of text-processing tools to accomplish these tasks [61]; the tools used in this project are described here.

`NLTK`'s Snowball Stemmer is used on lists of words to remove stopwords and to obtain stemmed versions of non-stopwords. Stemming involves using an extensive, language-specific algorithm to remove the suffixes from words that differentiate their use in different parts of speech. For example, “thinking”’ becomes “think,” “cried” becomes “cri,” and “ladies” becomes “ladi.” The stems returned from the algorithm are often not equivalent to any real word, but can be used to aggregate counts of when a particular word is used in different contexts. Stemming is not perfect; sometimes completely different words have the same stem, or the same word in different forms will return two different stems. For example, “mice” stems to “mice,” and “mouse” stems to “mous.”

The Snowball Stemmer provided by NLTK can be set to ignore stop words, or words that add no real meaning to the language processing task at hand. These words are usually articles, prepositions, pronouns, and helping verbs, for example words such as: “and, the, but, or, for, he, she, it, by, to, from, have, has, would, could,” etc. The default stop word list was modified for the current project to include a set of stop words specific to the application. For example, the Fisher English Training Transcripts corpus, described above, includes many conversational filler words such as: “hi, uh-huh, mm, yeah, right, yep, oh, ah, er, ha, okay,” etc. Since these words do not contribute to a topical or sentiment analysis of the conversation, they were also removed during processing.

Another useful tool in the NLTK kit is the Maximum Entropy Treebank Part-of-Speech Tagger, which is the default part-of-speech tagger provided by the package. This pre-trained tool takes an array of tokens and returns an array of tuples, each containing the original token and a tag indicating its probable part of speech. For example, the tag “NN” indicates a noun, “JJ” indicates an adjective, “RB” indicates an adverb, and so on. This information can then be used in further processing, such as lemmatization and word-sense disambiguation, described below.

One especially powerful tool provided by the NLTK package is access to Princeton University’s WordNet 3.0, a graph-formed dictionary/thesaurus with a unique organization based on word sense and synonyms [62]. In WordNet, there are two main forms of word representation: lemma and synset. A lemma is a basic form of a word, such as “car” or “jump.” A synset, on the other hand, is a set of lemmas that are synonyms, along with the meaning expressed by them in this sense. For example, the synset with the name “car.n.01” is defined as “a motor vehicle with four wheels; usually propelled by an internal combustion engine,” and it contains the lemmas “car,” “auto,” “automobile,” “machine,” and “motorcar.” A synset’s name is usually composed of one of its more prominent lemma members, a letter indicating its part of speech, and a number.

A synset has one definition and many possible lemmas that can express that definition; a lemma may have many definitions and therefore can be part of many synsets. For example, the lemma “machine,” from the synset “car.n.01” above, is also a lemma in the synset “machine.v.01,” which has the definition “turn, shape, mold, or otherwise finish by machinery.”

WordNet defines many relations between lemmas and between synsets, such as hypernyms, holonyms, and antonyms, and it can be used to find similarity measures between word senses based on the lengths of relationship paths between them. Notably, it can also be used to lemmatize words.

Lemmatizing is similar to stemming in that it reduces a word used in any of several different parts of speech to the same form; however, unlike stemming, it always returns a real word in the target language. Lemmatizing attempts to find the base form of a word given its current form and context; for example, “thinking” becomes “think,” “cried” becomes “cry,” and “ladies” becomes “lady.” Lemmatizing is often much slower than stemming, but it does offer cleaner and sometimes more useful output. NLTK provides a WordNet-based lemmatizer that accomplishes this task with the help of part-of-speech information, which can be obtained using the part-of-speech tagger described above.

Using the above tools, it is possible to interpret a text string into an array of part-of-speech tagged stems or lemmas. But given the power of WordNet's synset interface, it is desirable to ascertain not just the lemmas, but also the synsets that each word belongs to. That is, it is important not only to have the base form of the word, but also the sense that it is being used in. However, since each lemma can be a part of many different synsets, in order to find the correct synset for each lemma, a word-sense disambiguation algorithm is needed.

Although word-sense disambiguation is an entire field of its own, with many complex solutions, a quick-and-dirty method for lemma-to-synset disambiguation is available in NLTK. This is a simple implementation of the Lesk algorithm, which compares the context of a word against the definitions of all of the word's synsets that share its part-of-speech. The synset definition that has the most overlap with the context is the one chosen as the correct word sense.

NLTK's algorithm is very simplistic; for this reason, it was modified for this project to have slightly more power by processing the text of the context, and also the text of each definition. In this processing, punctuation and stopwords were removed, and the remaining words were then stemmed in order to lessen the impact of irrelevant information on the overlap comparison. References to the word being disambiguated were also removed from the context, in order to avoid giving preference to definitions written in terms of the word itself, which are usually not useful or correct for the context.

3.5 SentiWordNet 3.0

SentiWordNet 3.0 is the latest version of an opinion-mining resource based on WordNet 3.0, developed by a team at the Instituto di Scienza e Tecnologie dell'Informazione [63]. This resource maps a large number of WordNet synsets to sentiment scores indicating their positivity, negativity, and objectivity. That is, if a

WordNet synset is included in SentiWordNet, it is assigned a tuple (pos, neg, obj) such that pos indicates the synset's level of positivity, neg indicates its negativity, and obj indicates its sentiment-neutrality, and such that $pos + neg + obj = 1$. For example, the score tuple for the synset "love.n.01" is (0.625, 0.0, 0.375), and the score tuple for the synset "pain.n.01" is (0.0, 0.75, 0.25).

This resource was built in a two-step process, the first of which involved semi-supervised learning based on a seed set of positive and negative synsets, and the second of which involved a random walk that propagated the positive, negative, and objective scores via synset glosses and definitions. After a final normalization step, the full SentiWordNet resource was created. This resource is kindly distributed for free for research purposes, and in fact `NLTK` provides an interface for the English portion of the resource that meshes well with its interface for WordNet.

CHAPTER 4

APPROACH

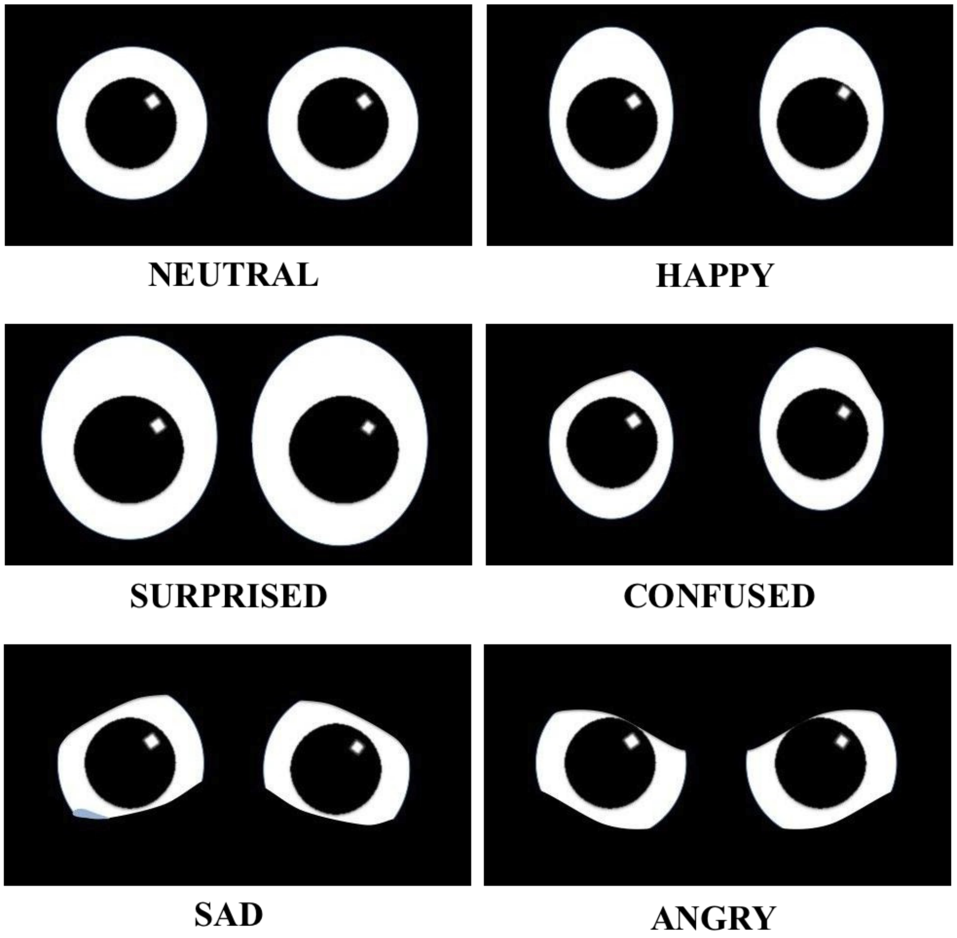
The current project implements a conversational system for a humanoid robot in two parts: the front end, in a smartphone app, and the back end, on the robot's onboard computer. The front end of the system was largely developed before the start of this project, whereas the back end is entirely in the domain of this project. Since each part is relevant, they are described separately below.

4.1 The SMILE App

The front end of the system is an Android app called the SMartphone Intuitive Likeness and Expression App, or the SMILE App. This app was developed in Java by students of the H.E.I.R. Lab as the emotional and conversational interface for their humanoid robot design, the MU-L8 robot [6, 7]. A smartphone running the app is positioned on the front of the robot's head, and the app then acts as the robot's face.

The app displays a pair of cartoon-like eyes that blink every few seconds and can be animated to assume any of six expressions: Neutral, Happy, Surprised, Confused, Sad, or Angry. These expressions are displayed in Figure 4.1. The app can also verbally interact with the user in a variety of different modes: Normal, Learn, Command, or Conversation.

The app speaks to the user via the Android TextToSpeech module. This module converts strings to synthesized speech and delivers this speech aloud, and the tone and rate of delivery are modified depending on the robot's current emotional expression. For example, when Surprised, the app speaks quickly with a higher tone, and when Sad, the app speaks slowly with a lower tone.



NEUTRAL

HAPPY

SURPRISED

CONFUSED

SAD

ANGRY

Figure 4.1: Emotional expressions that the SMILE app can assume.

Users can begin a conversation in Normal mode by tapping on the screen, upon which the app begins listening for user speech input to transcribe. For this purpose, the Android SpeechRecognizer module is used. This module records speech until it detects a “breakpoint” or pause, after which it sends the recording to Google for immediate processing, and finally returns a string that is its best guess as to the correct transcription of the speech. If no speech was heard, or if the speech was unintelligible, it returns an error. In Normal mode, the app deals with an error by simply re-starting the listener for new speech; in other modes, it may ask for clarification before doing so.

Recognized speech is dealt with differently in different modes. In Normal mode, the app listens for keywords or keyphrases that have been hard-coded into it or taught to it using the Learn mode. If it recognizes any of these, then it animates its eyes to the prescribed expression and speaks the verbal reply that it was taught. If it does not recognize any keywords or keyphrases, it simply repeats the speech that it did recognize back to the user. Some keyphrases include the commands to enter the other modes, such as “learn,” “command mode,” and “conversation mode.”

In Learn mode, the app runs through a protocol to help the user teach it a new keyword or keyphrase to recognize, along with an associated emotional expression and verbal reply. These are then saved so that even if the app is turned off and restarted later, it will remember the new keyword or keyphrase and react to it appropriately when required.

In Command mode, the app connects via Bluetooth to the MU-L8 robot’s onboard computer, a mini-PC that also controls its actuators. The user can then verbally command the robot to perform preprogrammed actions such as “sit,” “stand,” “kick” a ball, “relax” its motors, “track” a ball with its head, or “walk.”

Conversation mode was developed specifically to be the front end interface for the current project. In Conversation mode, the app again connects via Bluetooth to the robot’s onboard computer, in order to communicate with the topic-detection and sentiment-analysis subsystems installed on it. The onboard computer, with its higher computational power and storage capabilities, is much more suited to the task of running these systems than is the mobile phone running the app.

When Conversation mode is entered, the robot asks what the user wants to talk about, then prompts the user to talk about their opinions on that topic. The app begins listening for user speech, and it restarts the listener as many times as needed until it has transcribed at least 15 words from the user. It then sends this transcription to the onboard computer via Bluetooth and receives in return a string that contains:

1. A number indicating an emotion to express, as determined by the sentiment-analysis system (positive for Happy, negative for Sad, and zero for Neutral).
2. A text sentence to speak in reply, as generated by the topic-detection system.

The app animates to the new expression and speaks the indicated reply, and then it begins listening again for user speech. This process continues until the user chooses to exit Conversation mode by either tapping the screen or saying “exit” or “stop.”

4.2 Back End System

The system installed on the robot’s onboard computer is implemented in Python, and it consists of a text processing layer, the topic-detection subsystem, and the sentiment-analysis subsystem. These subsystems work together to analyze text-transcribed user speech input and to generate appropriate and relevant robot reactions.

The processing of an utterance, described below, is also illustrated by a flowchart in Figure 4.2. In this chart, the information from the SMILE app first enters text processing, then splits left and right to be processed by the topic-detection and sentiment-analysis subsystems, respectively, before returning to the app at the top with a reply and a sentiment score. The two resources at the bottom of the chart, the Synsets LSI model and the Sentiment Dictionary, were created during system build to support the subsystems. The operation of the subsystems on a query will be described first, followed by the system build.

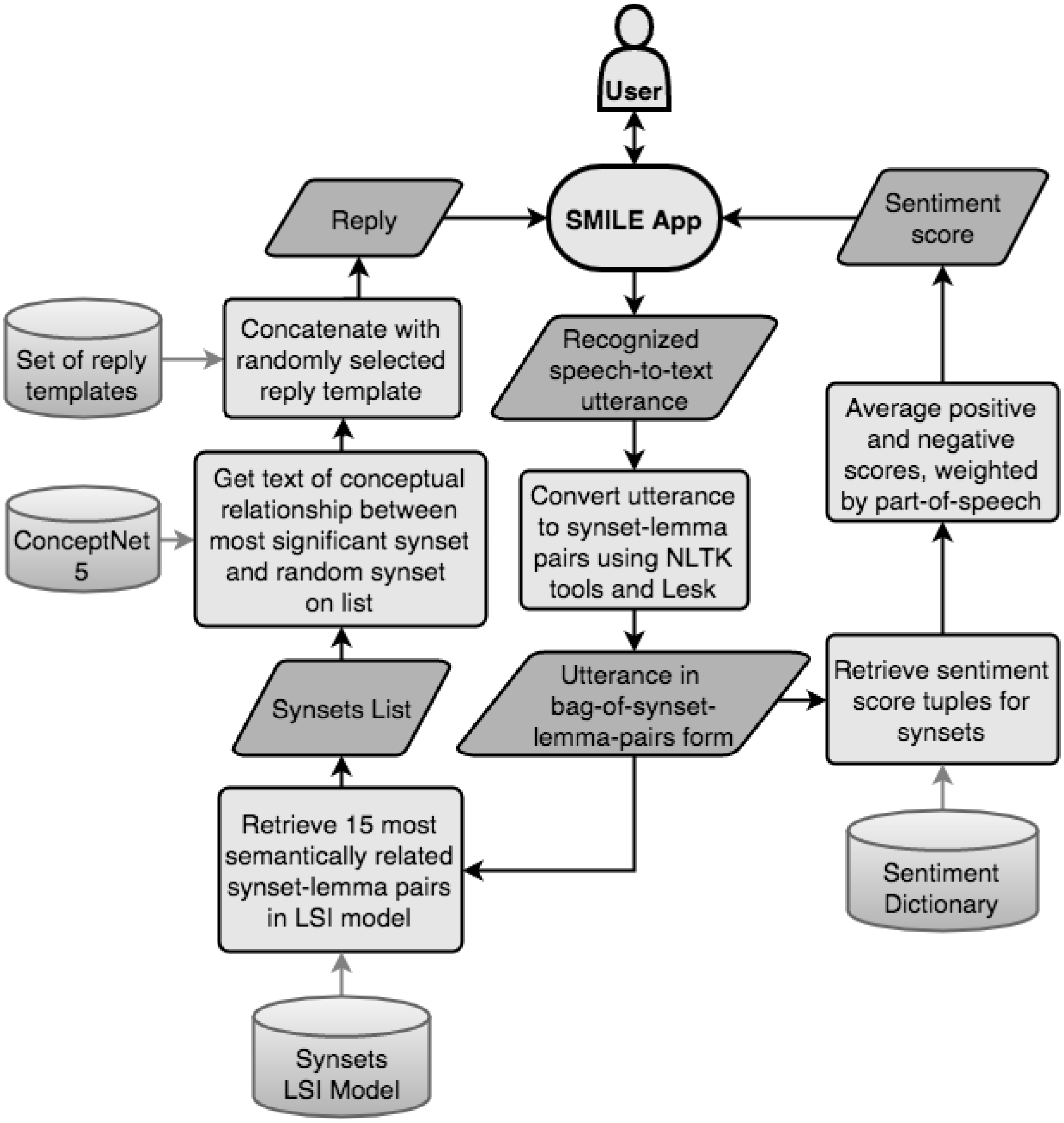

User
SMILE App
Reply
Sentiment score
Recognized speech-to-text utterance
Concatenate with randomly selected reply template
Set of reply templates
Average positive and negative scores, weighted by part-of-speech
Get text of conceptual relationship between most significant synset and random synset on list
ConceptNet 5
Convert utterance to synset-lemma pairs using NLTK tools and Lesk
Synsets List
Utterance in bag-of-synset-lemma-pairs form
Retrieve sentiment score tuples for synsets
Retrieve 15 most semantically related synset-lemma pairs in LSI model
Sentiment Dictionary
Synsets LSI Model

Figure 4.2: Flowchart illustrating how the the system processes a user utterance and generates a sentiment score and reply.

4.2.1 Text Processing

When a string containing a transcribed user utterance is submitted to the back end of the system, it must be converted to a form that the subsystems can work with before it can be analyzed.

First, the string is tokenized (split into a list of words), and all stop words are removed. Next, each remaining word is tagged for part-of-speech using `NLTK`'s default part-of-speech tagger, and this information is then used to lemmatize it with `NLTK`'s WordNet lemmatizer.

The lemmatized word with its part-of-speech tag is submitted to the modified Lesk algorithm for sense disambiguation, along with the rest of the utterance as context. The algorithm returns its best guess as to the WordNet synset that encapsulates the sense of the word, as used in the utterance.

Finally, this synset's name is concatenated with the lemma to form a "synset-lemma pair," a token that denotes both the word that was used (in its base form) and the way in which it was used. For example, if the word "machine" was used in reference to an automobile, and therefore belongs to the synset "car.n.01," then its synset-lemma pair would be "car.n.01.machine."

When this process has been completed, the user's utterance is now in bag-of-words form, where each word is a synset-lemma pair. This "bag-of-synset-lemma-pairs" form can be used by the subsystems to generate sentiment scores and replies.

4.2.2 Sentiment Analysis Subsystem

The sentiment analysis subsystem takes a user utterance in bag-of-synset-lemma-pairs form, as described above, and returns a decimal number that describes the overall sentiment in the utterance, positive or negative, and its relative strength. Relative strength is currently not used by the front end of the system, which is concerned only with the sign of the number, but it is included for future enhancements to the system.

First, the system iterates through each synset-lemma pair and looks up each synset in the Sentiment Dictionary (a resource created during system build, described below in Section 4.2.4). If a sentiment tuple for synset i is returned, it will be of the form (pos_i, neg_i, obj_i), and will be saved for aggregation with the rest of the scores.

The scores for all of the synsets are aggregated by calculating the average weighted difference between their positive and negative components. The weight, w_i, is determined by the part of speech of the synset i: nouns and verbs are weighted one-third as heavily as adjectives, and adverbs are weighted two-thirds as heavily as adjectives, the assumption being that these parts of speech carry different levels of importance in conveying spoken sentiment. Averaging is used in order to ensure scores between -1 and 1; having these bounds on the score's range simplifies the evaluation of the system discussed in Section 5.1.1. The equation for this sum is as follows:

$$SentiScore = \frac{1}{n} \sum_{i=1}^{n} w_i \cdot (pos_i - neg_i) \tag{4.1}$$

where n is the number of tuples returned from the Sentiment Dictionary.

The final sentiment score, indicating a Happy, Sad, or Neutral emotion when the score is positive, negative, or zero respectively, is then returned to the front end of the system.

4.2.3 Topic Detection Subsystem

The topic detection subsystem takes a user utterance in bag-of-synset-lemma-pairs form, as described above, and returns a string containing a topically relevant reply designed to engage the user in further conversation.

First, the system obtains 15 synset-lemma pairs with high semantic similarity to the utterance, whether or not they actually appear in it. To do this, it uses the utterance to query the Synsets LSI Model (a resource created during system build, described below in Section 4.2.4) using the matrix-multiplication method described by equation 3.4. This Synsets List of highly semantically related words is used to generate a reply that is both topically relevant and original, possibly introducing new ideas that are related to the topic of the user's utterance.

Two synset-lemma pairs are selected from the Synsets List, and their lemmas are used to query ConceptNet 5 for a conceptual relationship between the words. The first word is selected based on its semantic similarity to the utterance: the higher, the better, and even better if it is also present in the utterance itself. This is to ensure that the conceptual edge found is very relevant to the user's speech, even if the second word introduces a new idea. The second word is selected randomly from the rest of the list, and any words that have been used in prior replies in the conversation are avoided. This is to encourage variability and originality in replies, in order to improve user engagement.

ConceptNet 5 is queried for common-sense relationships connecting the concepts denoted by these two words, and it returns a list of the highest-weighted (most likely to be true) edges between them. These are searched for one highly-weighted edge with a useful relationship type; types such as "TranslationOf" and "DerivedFrom," for example, are avoided because they are concerned with verbal relationships rather than conceptual relationships. Once selected, the edge's surface text sentence, which describes the relationship in plain English, is extracted and used as the main text of the reply.

In order to further improve user engagement, a phrase is randomly selected from a set of reply templates and concatenated with the sentence. These reply templates are designed to be entertaining and even humorous, especially when spoken by a robot, and they include phrases such as the following:

> "A little bird tells me that..."
> "In my humble opinion..."
> "As the philosophers have noted..."
> "Don't quote me on this, but..."
> "What are you even thinking? ..."

A subset of these templates are only available to select when the sentiment returned by the sentiment-analysis subsystem is negative, and these include:

> "It pains me greatly to say that..."
> "Alas..."

Another subset of these templates are only available to select when the sentiment returned is positive, and these include:

> "I rejoice to inform you that..."
> "The sun is shining, and..."

In this way, the topic-detection and sentiment-analysis subsystems work together to return a coordinated reaction.

The final reply, consisting of the reply template concatenated with the surface text sentence, is then returned to the front end of the system.

4.2.4 System Build

The goal of this system is to maintain a topically relevant, emotionally appropriate, and engaging conversation with the user in real time. Thus, outputs from the system must be generated in minimal processing time, and any text processing of the user's utterances should be usable by both subsystems.

In order to analyze the user's utterances for positive and negative sentiments, they are converted to WordNet synsets and the relevant SentiWordNet scores for these synsets are retrieved. In order to produce replies that are topically relevant, yet original, an LSI model of the Fisher English Training Transcripts corpus is used to get a list of semantically similar words to the user's utterances. Since the sentiment-analysis subsystem requires that the utterance be converted into WordNet synsets, which contain useful discriminative information about word sense, it is convenient and beneficial to have the topic-detection subsystem's LSI model use synsets as well. Therefore, the final LSI model should represent words as synset-lemma pairs.

To meet these requirements, two resources for use in system querying, the "Synsets LSI Model" and the "Sentiment Dictionary," were built in the following manner. The process is also illustrated by a flowchart in Figure 4.3.

First, the Fisher English Training Transcripts corpus was processed to create two different LSI models, in which the first model was used to help create the second and final LSI model with greater accuracy.

For the first model, the corpus was tokenized and all stop words were removed, including tokens denoting noises, unfinished words, uncertain transcriptions, and back-channel words such as "mm" and "yeah." The remaining words were then stemmed using `NLTK`'s Snowball Stemmer, and the stemmed corpus was converted into an LSI model in which each conversation was represented by a column, and each word stem was represented by a row. The log-entropy transformation was used for the weighting step, and the resulting LSI model was called the "Stemmed LSI Model."

The corpus was then processed from scratch a second time. In this case, all of the transcribed speech in each conversation was concatenated, then broken up into 20-word substrings. For each substring, the stop words were again removed,

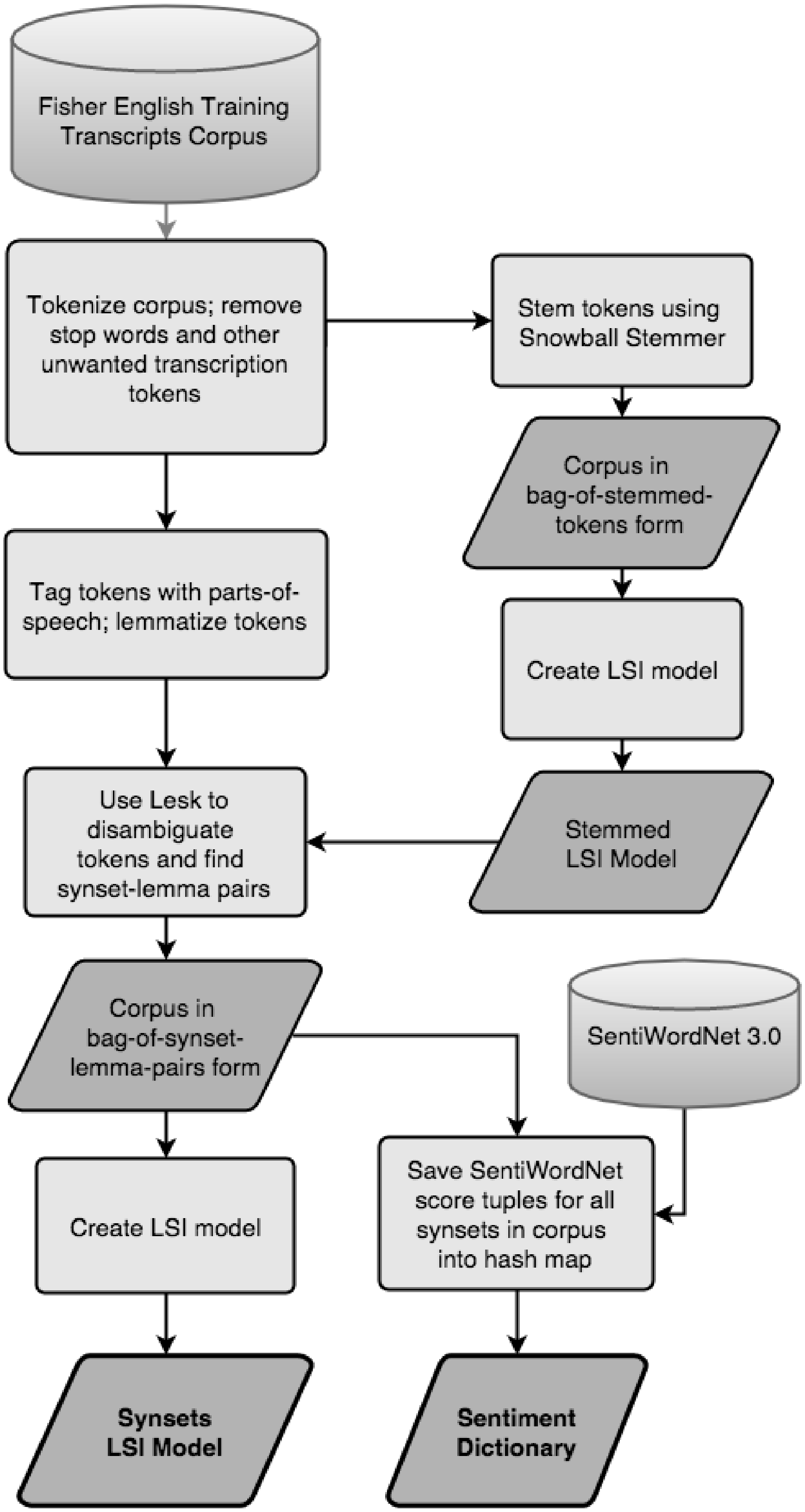

Fisher English Training Transcripts Corpus
Tokenize corpus; remove stop words and other unwanted transcription tokens
Stem tokens using Snowball Stemmer
Corpus in bag-of-stemmed-tokens form
Tag tokens with parts-of-speech; lemmatize tokens
Create LSI model
Use Lesk to disambiguate tokens and find synset-lemma pairs
Stemmed LSI Model
Corpus in bag-of-synset-lemma-pairs form
SentiWordNet 3.0
Create LSI model
Save SentiWordNet score tuples for all synsets in corpus into hash map
Synsets LSI Model
Sentiment Dictionary

Figure 4.3: Flowchart illustrating the process for creating the Synsets LSI Model and Sentiment Dictionary used in system operation.

and each non-stop-word was tagged for its part of speech and then lemmatized. Each lemma with its tag was then submitted to the Lesk algorithm to disambiguate its sense and find the correct WordNet synset for that lemma.

However, the Lesk algorithm requires the context of the word in order to operate. For this purpose, the rest of the non-stop-words in the 20-word substring were stemmed and submitted to the Stemmed LSI Model in order to obtain a list of the 15 most semantically-related stems for that substring (again, using the matrix-multiplication method described by equation 3.4). These 15 related words, along with the words in the substring, were used as the context for the Lesk algorithm, which increased the likelihood of overlap with the correct synset definition and thereby increased its accuracy.

Once a lemma's disambiguated synset was returned from the Lesk algorithm, it was concatenated with the lemma to form the word's synset-lemma pair, as described in Section 4.2.1. When the whole corpus was converted into non-stop-word synset-lemma pairs, it was then used to build a new LSI model in which each conversation was represented by a column, and each synset-lemma pair was represented by a row. Again, the matrix was weighted with a log-entropy transformation before undergoing Singular Value Decomposition. The resulting model was called the "Synsets LSI Model," and it was the model used to process utterances in the final system.

Concurrently to the build of this LSI model, the synset from each synset-lemma pair was submitted to SentiWordNet, and if it returned a sentiment score tuple, then this tuple was saved in a hash map with the synset name as the key. This hash map was called the "Sentiment Dictionary," and since its lookup time was far more efficient than that of the NLTK SentiWordNet interface, the use of this dictionary vastly improved the efficiency of utterance processing.

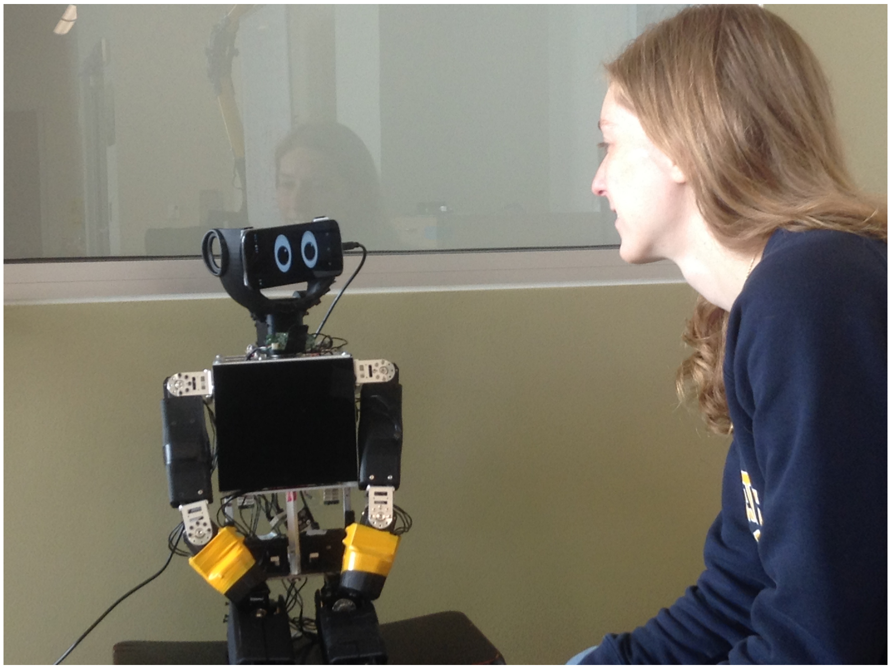


Figure 4.4: A user interacts with the system on the MU-L8 robot.

4.3 System Integration

Since the MU-L8 robot's onboard computer operates on an Ubuntu installation, the completed system is designed to run on Linux. With the processing power of the robot's onboard computer (1.8-2.8 GHz Intel NUC 3rd generation i5, 8GB RAM), the system generates replies to queries in around 3 seconds. For evaluation and demonstration purposes, a Macbook Pro laptop with an Ubuntu partition was substituted for the onboard computer, in order to monitor the ongoing text transcription of the conversation. This computer's processing power (2.26 GHz Intel Core Duo, 4GB RAM) allowed for reply generation in approximately 2 seconds. Reply generation time also varied depending on the speed of the internet connection and the number of times that ConceptNet 5 was queried before a satisfactory edge was found.

The following is an example transcription of a user conversation with the robot, with capitalization and punctuation added for clarity. The robot's emotional expressions are shown in square brackets and its verbal replies are italicized:

> “I like eating at home because I can cook for myself. I made some tuna pasta the other day that was really good.”
> [HAPPY] *Well, as they say, food can be eaten.*
> “I sure ate all that pasta! I also cook waffles in the morning, and hard boiled eggs.”
> [NEUTRAL] *A wise man once said, the effect of eating is gaining weight.*
> “What, are you calling me fat? I don’t gain too much weight from my cooking because I cook healthy.”
> [HAPPY] *A wise man once said, weight lifting is good exercise.*
> “Yeah, I do try to exercise, and I weightlift most mornings. I know you’re not supposed to exercise too much, so I try to keep it small.”
> [NEUTRAL] *Don’t quote me on this, but I think exercise is good for you.*
> “Thanks! I guess it is good for me. I like doing it because I feel better about myself.”
> [HAPPY] *I don’t know if it matters, but feeling comfortable speaking in public takes practice for many people.*
> “Thats true, it’s another thing that needs a lot of practice. I have practiced it a lot because I’ve worked as a teacher once.”
> [NEUTRAL] *A little bird tells me that a teacher can forget to grade papers.*

CHAPTER 5

EVALUATION

Both computational and perceptual evaluations were performed on the completed system. Computational evaluations tested each subsystem specifically and computationally for performance at its intended task, while the perceptual evaluation tested the performance of the system in interactions with actual human users.

5.1 Computational Evaluations

In order to perform computational evaluations on both subsystems, a set of conversational utterances was required for use as a test set. For this purpose, the Fisher English Training Transcripts corpus was split into a Training Set and a Test Set of conversation documents. Approximately 5% of the documents were randomly selected as the Test Set; any of these documents that lacked a topic prompt label were returned to the Training Set in order to ease the evaluation process for the topic-detection subsystem. This resulted in a split of 11,129 documents (95.13%) in the Training Set and 570 documents (4.87%) in the Test Set. In order to produce fair test results, the system was then rebuilt as described in Section 4.2.4, but using only the Training Set of documents from the corpus to create both LSI models and the Sentiment Dictionary.

5.1.1 Evaluation of the Sentiment Analysis Subsystem

In order to evaluate the sentiment analysis subsystem computationally, its task was considered both in terms of regression and in terms of classification. Regression was used to determine the fine-grained accuracy of the system in judging the relative strength of emotion in an utterance, while classification was used to determine its performance in returning an overall emotion category, as this is the granularity of emotion that is currently displayed on the front-end of the system.

A set of 120 test utterances was drawn from the Test Set of the corpus, and the system-generated sentiment scores for these utterances were compared against scores assigned by a human annotator.

The utterances were selected from the Test Set as follows: for each of the 40 topic prompts, seven documents were randomly drawn from the documents with that prompt label in the Test Set. Each conversational document was then scanned for its ten longest single-speaker utterances, and of these ten, one was randomly chosen and written to an output file. The final output file thus consisted of 280 utterances. This file was manually pruned for utterances that were grammatically clean, relatively understandable without conversational context, and generally on topic. At the end of the pruning process, three utterances from each topic remained, making 120 utterances in total, which was considered a reasonable number for a human to annotate in the amount of time available.

The annotation process was as follows: the set of utterances was first converted to a more human-readable form by capitalizing proper nouns and abbreviations, deleting noise tokens such as [mn] and [laughter], deleting uncertain-transcription tokens such as ((and)), and deleting unfinished-word tokens such as “y-” and “wha-.” No punctuation or other clarifications were added.

A human annotator was given the instruction to score each utterance for emotional content with a decimal number in the interval [-1, +1]. A score of -1 signified a completely negative utterance, +1 a completely positive utterance, and 0 an utterance without emotional content. Any number of decimal places could be used to represent intermediate scores. For ambiguous or mixed-emotion

Table 5.1: Correlation Results for Fine-Grained Computational Evaluation of Sentiment Analysis Subsystem

Pearson's R	N	Sig.
0.1940	120	0.0337*

utterances, the annotator was instructed to assign a score based on the amount to which positive or negative emotion dominated in the utterance.

System scores were obtained for the same set of utterances, using the system build from the Training Set of the corpus as described above. The resulting sets of scores were considered in a fine-grained manner, as a regression problem, and in a coarse-grained manner, as a classification problem.

In the fine-grained test, the Pearson's R correlation between the human-generated scores and the system-generated scores was calculated. Since Pearson's R requires that the variables be approximately normally distributed, both the human-generated data and the system-generated data were first tested for normality using the Shapiro-Wilk test. Both sets of data failed the test for normality.

In order to retain the power of the test and to minimize Types I and II errors, both sets of data were transformed using Blom's rank-based inverse-normality transformation before proceeding with the Pearson's R calculation [64]. The results of the correlation, shown in Table 5.1, display a weakly positive but statistically significant relationship ($p < .05$). A scatter plot of these scores can be seen in Figure 5.1. Due to the combining and averaging process that the system uses to produce its scores, the system scores fall in a much smaller range than the human annotator's scores; however, the correlation is the same even if this range is normalized.

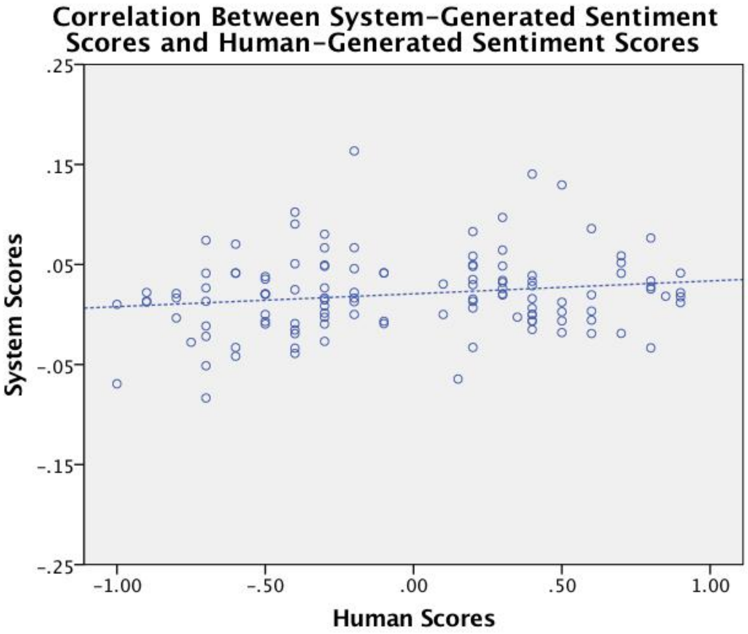


Figure 5.1: System-generated sentiment scores plotted against human-generated sentiment scores for 120 test utterances.

In the coarse-grained test, only the sign of each score was considered: positive (+) or negative (-), with neutral (0) as an intermediate category. The system was considered to have made a correct judgment when it assigned an utterance a score with the same sign as the human annotator's score. The confusion matrix for this test is shown in Table 5.2, and the system's precision, recall, and F_1 scores for each of the three categories are shown in Table 5.3.

Table 5.2: Confusion Matrix for Coarse-Grained Computational Evaluation of Sentiment Analysis Subsystem

		Human Scores			
		Positive	Neutral	Negative	Total
System Scores	Positive	42	0	40	82
	Neutral	3	0	2	5
	Negative	12	0	21	33
	Total	57	0	63	120

Table 5.3: Precision, Recall, and F_1 Scores for Coarse-Grained Computational Evaluation of Sentiment Analysis Subsystem

	Precision	Recall	F_1
Positive	0.512	0.737	0.604
Neutral	0	0	0
Negative	0.636	0.333	0.437

The system correctly classified 63 out of the 120 utterances, or 52.5%. It correctly classified positive utterances much more often than negative utterances: of the 57 utterances that the human annotator labeled as positive, it correctly classified 42, or 73.7% of them, while of the 63 utterances that the human annotator labeled as negative, it correctly classified only 21, or 33.3% of them. However, in general the system appears to have correctly avoided scoring utterances as neutral when they actually did contain emotion: the human annotator scored none of the utterances as neutral, and the system only scored 5 of them as neutral, or 4.2% of the whole set.

Cohen's Kappa (weighted) was used to measure the amount of agreement between the human and system scores in this confusion matrix. The results indicated a slight agreement above random chance: $\kappa = .1098$ (95% CI, -.0611 to .2807).

Upon examining the exact test utterances that the system and the human annotator disagreed about, it seems that the system often failed to notice emotionally relevant subtext, for example in this utterance:

> well I remember when 9-11 happened he was they were supposed to settled their contract two weeks after it was like September thirtieth they were supposed to sign everything and it was supposed to go into effect and then as soon as September eleventh happened boom

This utterance was rated as -0.9 by the human annotator, and the system rated it as 0.022. From a purely lexical point of view, there are no negative-indicating words in the sentence; however, any English-speaker reading the utterance would understand that it refers to some very negative subject matter.

Other confusion appeared to arise from the system lacking the ability to interpret negations and emphasis words. These complexities seem to affect the interpretation of positive utterances less often than negative utterances, which may have given rise to the relatively high accuracy in categorizing positive utterances versus negative utterances. For example, the following two utterances were both given very small positive ratings by the system and moderate to large negative ratings by the human annotator:

> they I mean they they don’t have good spelling they’re not equipped with good spelling tools they um they don’t write as well I don’t think um and they you know they’re just taking all these shortcuts they they don’t have good you know...

> well and the other thing is to you know a girlfriend of mine also I mean after years of fighting with her insurance company reached a settlement when she was in a really really bad car accident

Phrases such as “don’t have good spelling” and “don’t write as well” would be interpreted as positive by the system due to the presence of the words “good” and “well,” while it would ignore the stop word “don’t,” leading to an erroneous positive overall score. Phrases such as “years of fighting” and “really really bad” in the second utterance are clearly indicative of a very negative sentiment, but the system ignores their enhanced effect, and the average overall calculation returns a very small positive rating.

Although only one human annotator was available during the time permitted for this evaluation, a clearer picture of the system’s accuracy could also be obtained by comparing it to scores from multiple annotators.

5.1.2 Evaluation of the Topic Detection Subsystem

The topic-dection subsystem is not a classifier, nor does it produce continuous numerical data for regression; therefore evaluating it computationally presents some challenges. Since this subsystem is concerned with generating topically relevant, engaging replies to user utterances, the question to be answered in the computational evaluation is: "How topically relevant are the system's replies?" In other words, given an utterance, how close to that utterance is the reply generated in terms of objective semantic similarity?

The LSI model can be used to judge semantic similarity between an utterance and its reply, by converting each to LSI vectors and finding their vector cosine. However, the resulting similarity score is likely to be biased, since the LSI model is used in the generation of the reply itself. In order to remove some of this bias, the model can instead be used to compare utterances and replies against an external set of documents, thus pinning down their relative locations in a hypothetical topic space as defined by those documents.

The underlying assumption is that if Sentence A is topically similar to Sentence B, then Sentence A's semantic relationship to a set of external documents should resemble Sentence B's semantic relationship to that same set of documents. That is, if Sentence A is more similar to some external documents and less similar to others, then Sentence B will be more similar to the same external documents and less similar to the same other documents.

The LSI model can be used to find semantic similarity scores between any query sentence and any external document, by converting them both to LSI vectors and finding their vector cosine. The whole set of external documents can then be ranked according to each document's similarity score against the query sentence.

Thus, Sentence A’s relationship to the set of external documents will be defined as the ranked list of that set of documents, ordered according to vector cosine with Sentence A. Sentence B’s relationship to the set of external documents is similarly defined. In order to compare Sentence A and Sentence B, the correlation between their ranked lists is found. If the two sentences are topically similar to each other, relative to the set of documents, then their ranked lists will be positively correlated.

The 40 topic titles and prompts given in the Fisher English Training Transcripts corpus (see Appendix A) are used in this evaluation as the set of external documents, since they differ widely in topic and none of them appear directly in the corpus itself. By processing the topic prompts’ text and indexing them as vectors using the system’s LSI model, their vector cosines can be obtained relative to any query. These vector cosines are essentially semantic similarity scores, and they can be used to rank the set of topic prompts according to their semantic similarity to the query: from 1 (most similar) to 40 (least similar).

Thus, the above-described process can be used to compare utterances against their system-generated replies by correlating their ranked lists. In order to evaluate the intermediate steps of the reply generation process, the utterances can also be compared against their system-generated Synset Lists, from which the two words are picked to generate a reply. To do this, each Synset List is concatenated and treated as an utterance itself.

Thus, for each utterance, two topical similarity correlations will be found:

1. The correlation between the utterance’s ranked list and its intermediate synset-list’s ranked list.
2. The correlation between the utterance’s ranked list and its final reply’s ranked list.

A test set of utterances was obtained as follows: for every document in the Test Set of the corpus, all contiguous utterances by the same speaker were concatenated, and the five longest such utterances in each conversation were found. This process resulted in 2,850 testable utterances.

Since each utterance comes from a document with a label indicating the topic prompt assigned to that conversation, it can also be inferred that the utterance is likely to be more similar to that topic prompt than to the other topic prompts. As a result, that topic prompt is likely to appear higher in the utterance's ranked list than the mid-point. Since the synset-lists and replies should also be more similar to that topic prompt than to the others, that topic prompt should be higher in their ranked lists as well. Therefore, each ranked list is additionally searched for the rank of the "correct" topic prompt in that list, and this rank number is saved. Average rank numbers less than 20 would indicate that the system produces synset-lists and replies that are more similar to their correct topic prompts than to the other topic prompts.

The evaluation process, illustrated in a flowchart in Figure 5.2, is described as follows:

1. For each test utterance utt with correct topic prompt label $topic$, find synset-list $list$ and generate reply $reply$.
2. Create a ranked list, $Ranks_{utt}$, of the 40 topic prompts, ordered according their vector cosines against utt.
3. Create a ranked list, $Ranks_{list}$, of the 40 topic prompts, ordered according their vector cosines against $list$.
4. Create a ranked list, $Ranks_{reply}$, of the 40 topic prompts, ordered according their vector cosines against $reply$.
5. Find the Spearman's correlation and its p-value between $Ranks_{utt}$ and $Ranks_{list}$; save them as $Correlation_{utt-list}$

6. Find the Spearman's correlation and its p-value between $Ranks_{utt}$ and $Ranks_{reply}$; save them as $Correlation_{utt-reply}$
7. Find the rank of the correct topic $topic$ in $Ranks_{list}$; save it as $TopicRank_{list}$
8. Find the rank of the correct topic $topic$ in $Ranks_{reply}$; save it as $TopicRank_{reply}$
9. Repeat for every utterance utt_i in the test set.

Once these metrics are found for every utterance in the test set, the averages of all the $Correlation_{utt-list}$ and $Correlation_{utt-reply}$ values and their significances can be calculated. The averages of the rankings $TopicRank_{list}$ and $TopicRank_{reply}$ can also be calculated, and they can be tested with a two-tailed, single-sample t-test for any significant difference from the mid-point ranking of 20 out of 40. An

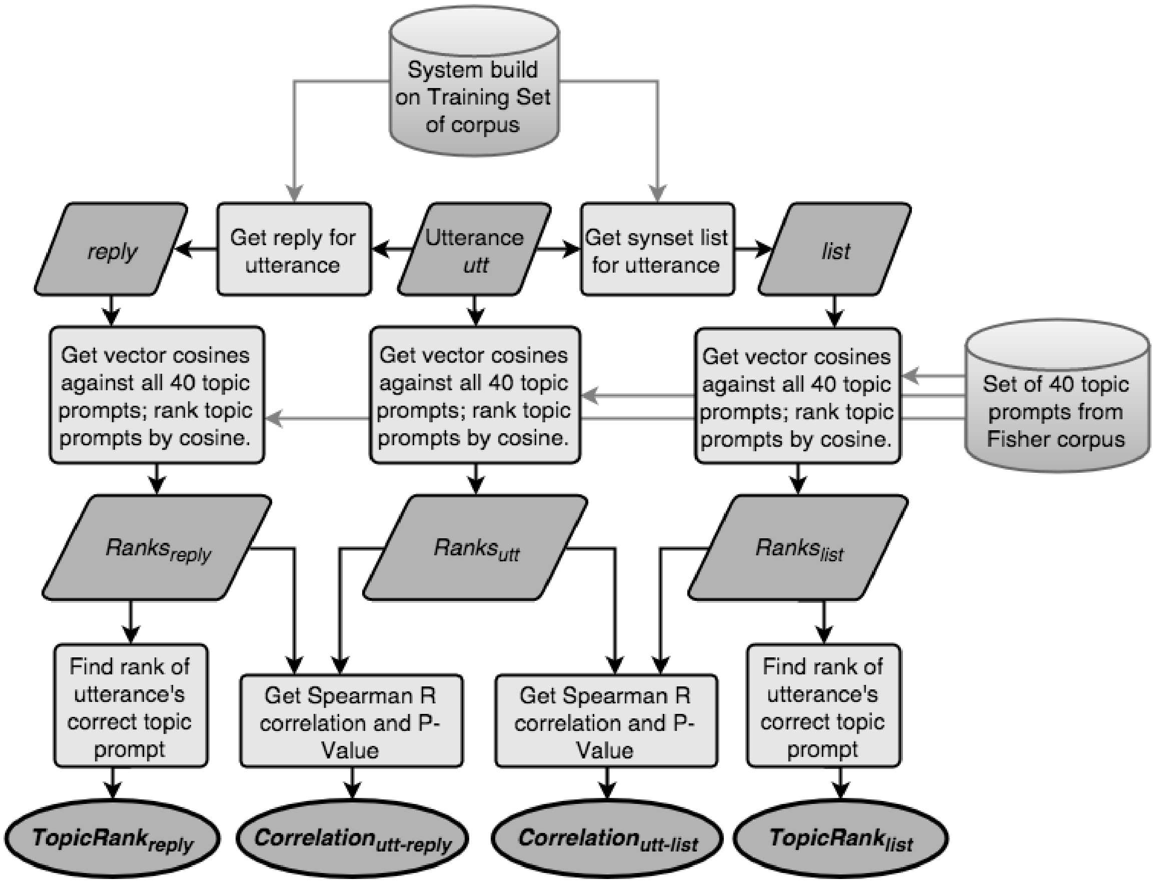

System build on Training Set of corpus
reply
Get reply for utterance
Utterance utt
Get synset list for utterance
list
Get vector cosines against all 40 topic prompts; rank topic prompts by cosine.
Get vector cosines against all 40 topic prompts; rank topic prompts by cosine.
Get vector cosines against all 40 topic prompts; rank topic prompts by cosine.
Set of 40 topic prompts from Fisher corpus
Ranks_reply
Ranks_utt
Ranks_list
Find rank of utterance's correct topic prompt
Get Spearman R correlation and P-Value
Get Spearman R correlation and P-Value
Find rank of utterance's correct topic prompt
TopicRank_reply
Correlation_utt-reply
Correlation_utt-list
TopicRank_list

Figure 5.2: Flowchart illustrating the calculation of evaluation metrics for each utterance in the computational evaluation of the topic-detection subsystem.

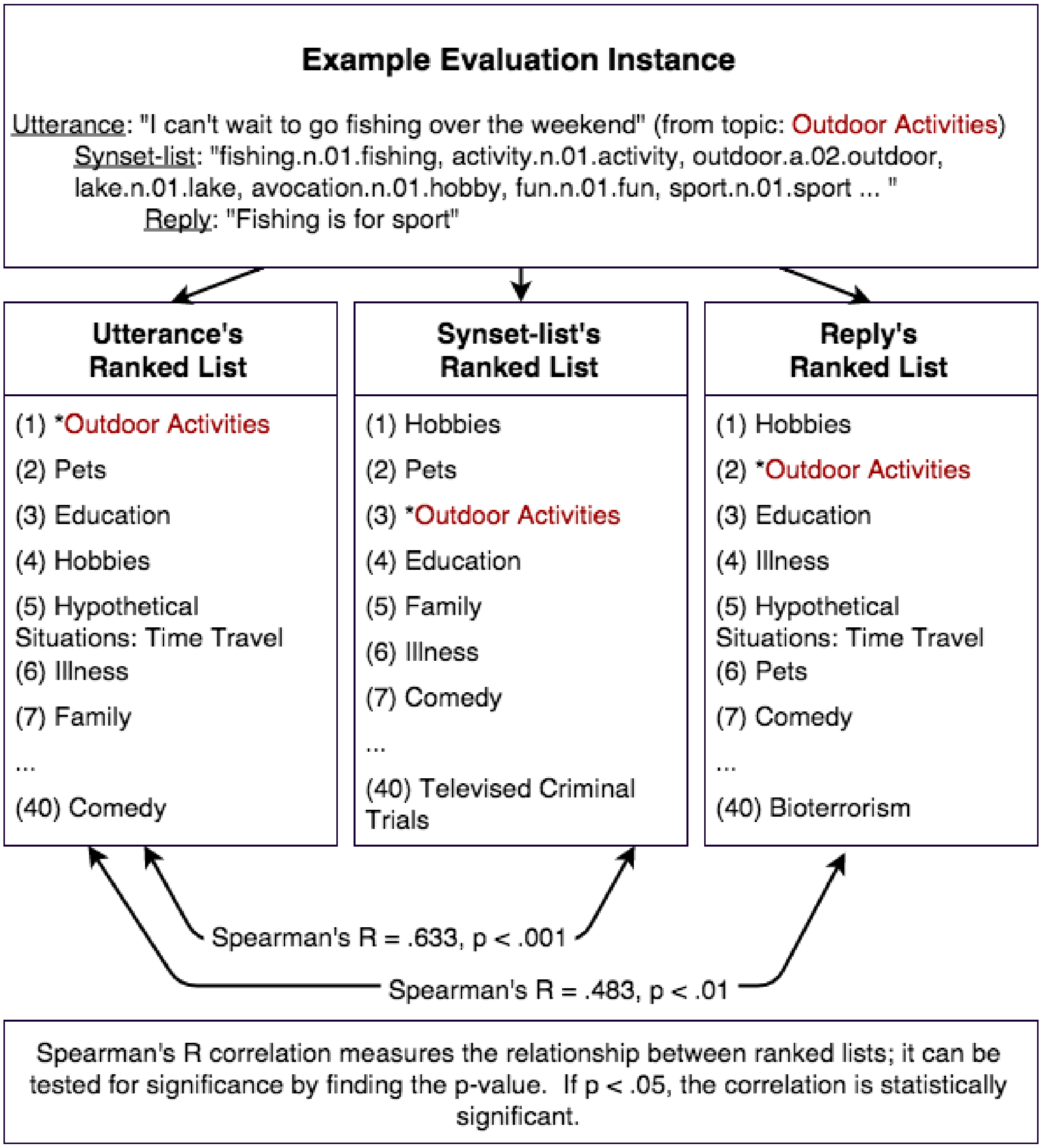

Example Evaluation Instance
Utterance: "I can't wait to go fishing over the weekend" (from topic: Outdoor Activities)
Synset-list: "fishing.n.01.fishing, activity.n.01.activity, outdoor.a.02.outdoor, lake.n.01.lake, avocation.n.01.hobby, fun.n.01.fun, sport.n.01.sport ... "
Reply: "Fishing is for sport"
Utterance's Ranked List
(1) *Outdoor Activities
(2) Pets
(3) Education
(4) Hobbies
(5) Hypothetical Situations: Time Travel
(6) Illness
(7) Family
...
(40) Comedy
Synset-list's Ranked List
(1) Hobbies
(2) Pets
(3) *Outdoor Activities
(4) Education
(5) Family
(6) Illness
(7) Comedy
...
(40) Televised Criminal Trials
Reply's Ranked List
(1) Hobbies
(2) *Outdoor Activities
(3) Education
(4) Illness
(5) Hypothetical Situations: Time Travel
(6) Pets
(7) Comedy
...
(40) Bioterrorism
Spearman's R = .633, p < .001
Spearman's R = .483, p < .01
Spearman's R correlation measures the relationship between ranked lists; it can be tested for significance by finding the p-value. If p < .05, the correlation is statistically significant.

Figure 5.3: Illustration of a single instance in the computational evaluation of the topic-detection subsystem.

illustration of an instance in this evaluation loop, with an example utterance, synset-list, and reply, is shown in Figure 5.3.

The results of this process can be seen in Table 5.4. The average Spearman's correlation between utterance ranked lists and synset-list ranked lists was a moderate positive correlation, but it was only marginally significant ($p = 0.068$). The average Spearman's correlation between utterance ranked lists

Table 5.4: Correlation and Average Ranking Results for Computational Evaluation of Topic-Detection Subsystem

	Average Rank of Correct Topic	t	Sig.	Average Spearman's R	Average Sig.
Synset-lists	19.3039	-3.2150	0.0013**	0.4712	0.0676
Replies	19.5446	-2.0896	0.0367*	0.3609	0.1767

and reply ranked lists was moderately low, but it was not significant ($p = 0.177$). The average ranking of the correct topic in the synset-list ranked list was 19.30 out of 40, and while this is a small improvement over a random average ranking of 20, the difference is significant according to the t-test ($p < .005$). The average ranking of the correct topic in the reply ranked list was 19.54 out of 40, and while this is also a small improvement over a random average ranking of 20, the difference is again significant ($p < .05$).

This suggests that while the synset-list bears a small but present topical connection with the utterance, the reply in general bears a very small, if any, topical connection with the utterance.

One caveat to note is that the system-produced replies tend to be very short, making comparisons against them difficult. Additionally, the topic prompts themselves are short, only one to three sentences long, and may not have been rich enough to create a well-defined topic space in which to make comparisons. Finally, the replies contain by design at least one concept with a more tenuous connection to the topical content of the utterance than the other. The resulting variability in replies, intended to heighten user engagement, may have weakened the results in this computational evaluation.

5.2 Perceptual Evaluation

The system was also tested in its target scenario, by having actual conversations with human users. There were three variables of interest in the perceptual evaluation:

1. User perception of the *topical relevance* of the robot's replies.
2. User perception of the *emotional appropriateness* of the robot's expressions.
3. *User engagement* during conversations.

The sentiment-analysis subsystem and the topic-detection subsystem were evaluated both separately and together in each of these areas, and each subsystem's performance was measured against random control settings.

The control setting for the sentiment-analysis subsystem displayed a uniformly randomly selected expression from the set {Sad, Neutral, Happy}. The control setting for the topic-detection subsystem generated a reply based on uniformly randomly selected words from the corpus dictionary, as opposed to semantically similar words obtained from the LSI model.

Thus there were four conversational conditions to compare, each with a different combination of settings:

1. *Control Condition*: the robot displays a random expression and generates a random reply.
2. *Topics Condition*: the robot displays a random expression and generates a topic-detection-based reply.
3. *Emotions Condition*: the robot displays a sentiment-analysis-based expression and generates a random reply.
4. *Integrated Condition*: the robot displays a sentiment-analysis-based expression and generates a topic-detection-based reply.

The research hypotheses for the user experience ratings and their differences in these conditions are as follows:

- Hypothesis 1: The Topics and Integrated conditions will each have higher mean topical relevance ratings than the Emotions and Control conditions.
- Hypothesis 2: The Emotions and Integrated conditions will each have higher mean emotional appropriateness ratings than the Topics and Control conditions.
- Hypothesis 3: The Integrated condition will have a higher mean user engagement rating than the Emotions and Topics conditions, which will in turn each have higher mean user engagement ratings than the Control condition.

These hypotheses are displayed graphically in Figure 5.4. In these diagrams, an arrow from one condition to another signifies that the first condition's mean rating is predicted to be statistically significantly greater than the second condition's mean rating.

5.2.1 Study Design

The evaluation study recruited 24 participants in all: 12 male and 12 female, ages 19 to 64, with a median age of 24.5 years. Participants self-reported their level of technical experience on a scale of 1 (minimal) to 7 (extensive), and the mean level reported was 4.88 with a standard deviation of 1.48.

During the appointment, the participant drew four topic prompts from a set of eighteen, which were modified from a subset of the topic prompts used in the Fisher English Training Transcripts corpus (see Appendix A). These prompts served as starting-points for each of four conversations that the participant would have with the robot, although there was no requirement to stay on-topic.

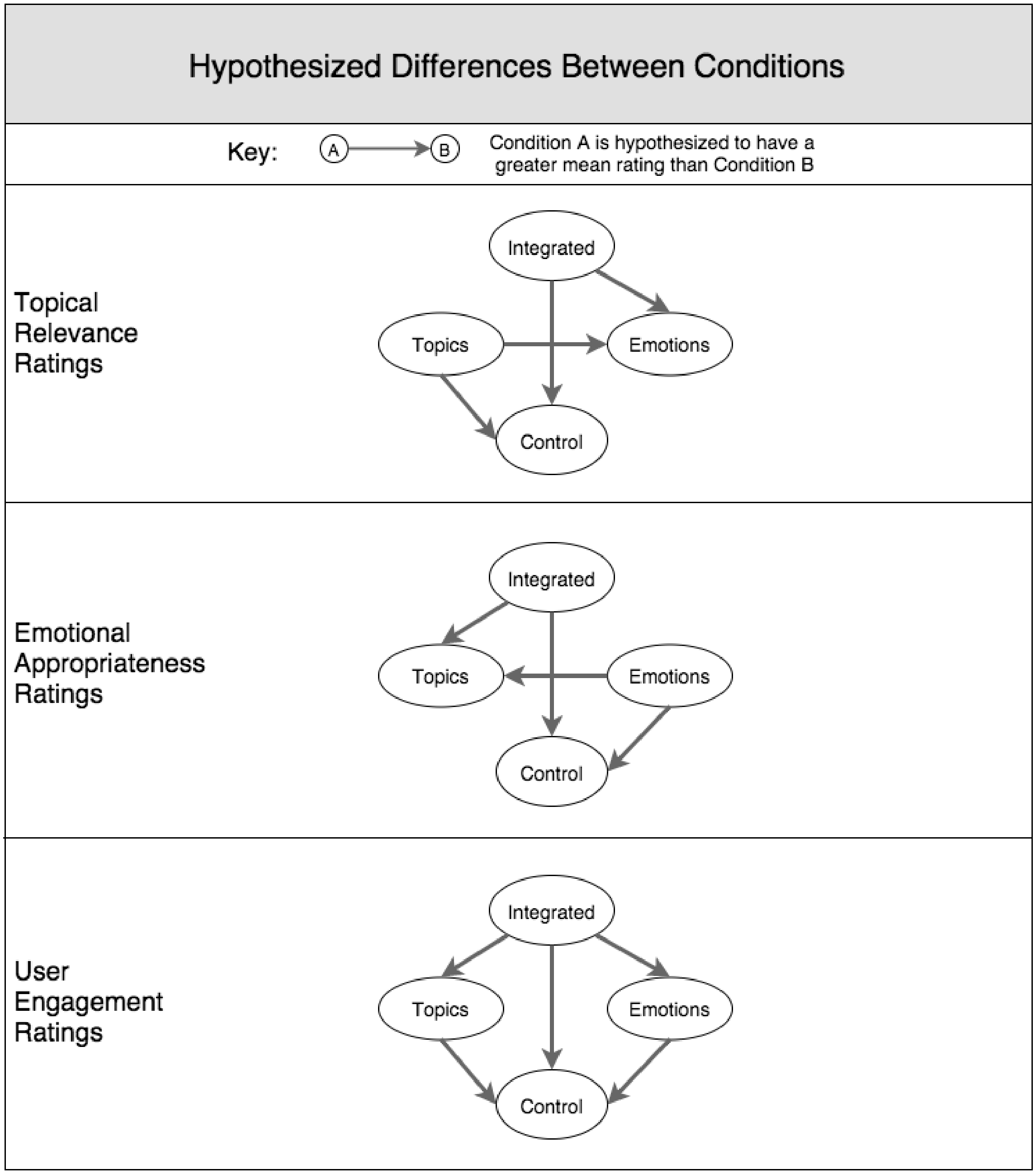

Hypothesized Differences Between Conditions
Key:
A
B
Condition A is hypothesized to have a greater mean rating than Condition B
Topical Relevance Ratings
Integrated
Topics
Emotions
Control
Emotional Appropriateness Ratings
Integrated
Topics
Emotions
Control
User Engagement Ratings
Integrated
Topics
Emotions
Control

Figure 5.4: Illustration of the hypothesized differences between conditions within each dependent variable for the perceptual evaluation.

Participants were also allowed to discard prompts and continue drawing until four topics were acceptable to them.

The four conversations were used to present each of the four conditions listed above. In order to avoid carryover effects from certain conditions always preceding others, the condition order was block randomized. All possible orderings of four conditions were obtained by enumerating all 24 permutations of the integers 1 through 4. At the beginning of each appointment, the experimenter randomly selected, without replacement, the permutation indicating the order in which this participant would encounter the conditions. In this way, all 24 possible orderings of the conversational conditions were used over the course of 24 participant appointments.

Each of the four conversations consisted of one prompt, spoken by the robot, followed by seven utterance-reply pairs. In each pair, the user spoke until the robot had gathered fifteen or more words, at which point it processed this speech and generated an emotional expression and verbal reply depending on the current condition. At the end of the conversation, after its seventh reply, the robot thanked the user for the conversation and stopped listening.

After each conversation, the participant filled out a section of a questionnaire. The same six questions were asked for each conversation, and each consisted of a statement for the user to indicate agreement or disagreement with. The user answered each question by circling a number from 1 (strongly disagree) to 7 (strongly agree), yielding six Likert-style user ratings for each conversation.

These ratings were evenly divided between the three dependent variables; there were two ratings, one reverse-scored, for each of them. The two ratings for each dependent variable were then averaged, yielding one overall rating for each dependent variable in each condition. The questions and the scoring process are illustrated in Figure 5.5.

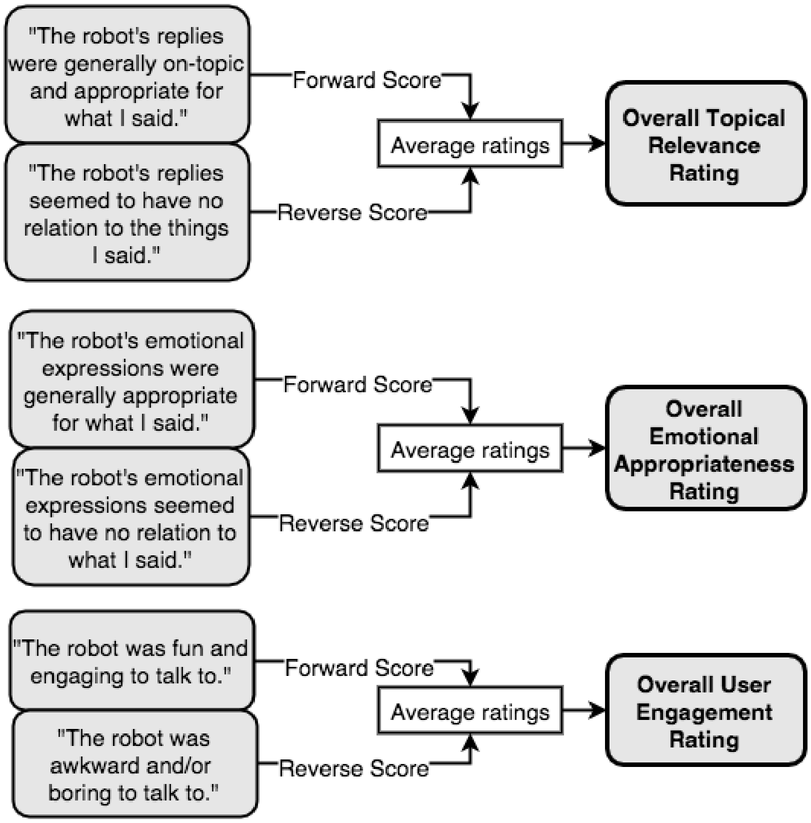

"The robot's replies were generally on-topic and appropriate for what I said."
"The robot's replies seemed to have no relation to the things I said."
Forward Score
Reverse Score
Average ratings
Overall Topical Relevance Rating
"The robot's emotional expressions were generally appropriate for what I said."
"The robot's emotional expressions seemed to have no relation to what I said."
Forward Score
Reverse Score
Average ratings
Overall Emotional Appropriateness Rating
"The robot was fun and engaging to talk to."
"The robot was awkward and/or boring to talk to."
Forward Score
Reverse Score
Average ratings
Overall User Engagement Rating

Figure 5.5: Scoring of Likert-style ratings collected from users during the perceptual evaluation.

5.2.2 Analysis and Results

Data analysis was performed on each dependent variable separately. Since the ratings are technically rankings rather than continuous data, they were not expected to conform to the assumption of normality, and this would ordinarily have led to the use of a non-parametric test. However, only three out of the twelve groups of overall ratings (one for each dependent variable in each condition) did not pass the Shapiro-Wilk test for normality. Since the repeated-measures ANOVA is quite robust to violations of normality, this parametric test was chosen over a more complicated non-parametric analysis.

In the case of significant ANOVA results, post-hoc tests were conducted with Tukey's HSD. This is appropriate given that this research is interested in all pairwise comparisons, and given the tighter Confidence Intervals produced by Tukey's HSD as compared to the more common Bonferroni correction.

Topical Relevance

The topical relevance ratings were normally distributed in two out of the four condition levels according to the Shapiro-Wilk test. By Mauchly's Test of Sphericity, the assumption of sphericity was not violated ($\chi^2(2) = 8.493, p = .131$). Descriptive statistics for these ratings can be seen in Table 5.5, including mean, median, range, and standard deviation.

A one-way repeated measures ANOVA determined that the mean topical relevance ratings differed statistically significantly between conditions ($F(3, 69) = 45.691, p < 0.001$). The results of this test are shown in in Table 5.6.

Post hoc testing using Tukey's HSD showed that the Topics condition had a significantly higher mean rating than both the Emotions condition ($p < 0.001$) and the Control condition ($p < 0.001$), and the Integrated condition had a significantly higher mean rating than the Topics condition ($p = 0.018$), the Emotions condition ($p < 0.001$), and the Control condition ($p < 0.001$). The results of these pairwise comparisons are shown in Table 5.7; each cell contains the difference in means and its significance value. A graph of the topical relevance rating means is displayed in Figure 5.6.

Table 5.5: Descriptive Statistics for Topical Relevance by Condition

Condition	N	Mean	Median	St. Deviation	Minimum	Maximum
Control	24	2.1458	2.00	1.11783	1.00	4.50
Topics	24	4.6458	4.75	1.35518	2.00	7.00
Emotions	24	2.4375	2.00	1.48406	1.00	5.50
Integrated	24	5.3333	5.50	1.18566	2.50	7.00

Table 5.6: Repeated-Measures ANOVA for Topical Relevance with Sphericity Assumed

Source	Type III Sum of Squares	df	Mean Square	F	Sig.	Partial Eta Squared
Condition	181.383	3	60.461	45.691	.000***	.665
Error(Condition)	91.305	69	1.323			

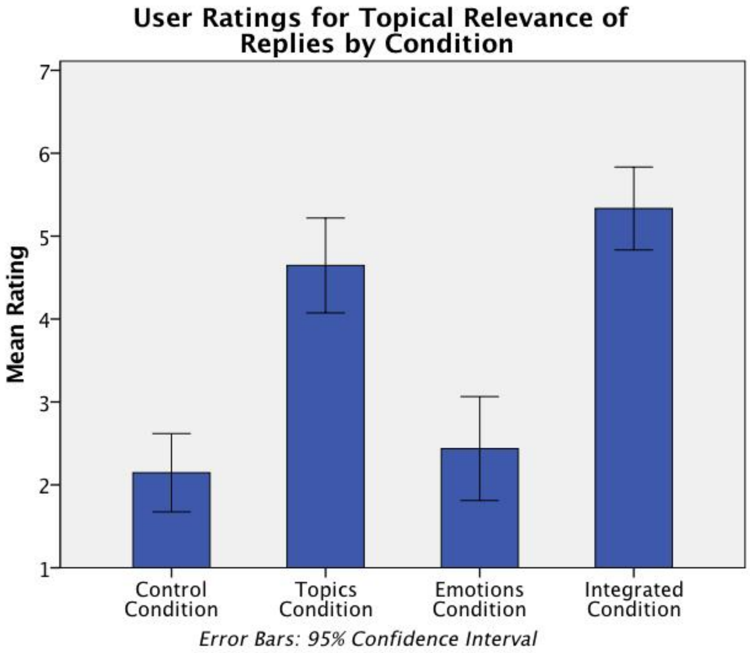


Figure 5.6: Mean topical relevance ratings in each condition.

Table 5.7: Pairwise Comparisons of Topical Relevance Means Using Tukey's HSD

	Control	Topics	Emotions	Integrated
Control	0	$2.500, p < .001$***	$.292, p = .354$	$3.118, p < .001$***
Topics		0	$-2.208, p < .001$***	$.688, p = .018$*
Emotions			0	$2.896, p < .001$***
Integrated				0

Emotional Appropriateness

The emotional appropriateness ratings were normally distributed in three out of the four condition levels according to the Shapiro-Wilk test. Mauchly's Test of Sphericity showed that the assumption of sphericity was not violated ($\chi^2(2) = 6.745, p = .241$). Descriptive statistics for these ratings can be seen in Table 5.8, including mean, median, range, and standard deviation.

A one-way repeated measures ANOVA determined that the mean emotional appropriateness ratings differed statistically significantly between conditions ($F(3, 69) = 8.379, p < 0.001$). The results of this test are shown in Table 5.9.

Post hoc testing using Tukey's HSD showed that the Topics condition had a significantly higher mean rating than the Control condition ($p = 0.017$), while the Emotions condition had a marginally significantly higher mean rating than the Control condition ($p = 0.051$), and the Integrated condition had a significantly higher mean rating than the Topics condition ($p = 0.004$), the Emotions condition ($p = 0.010$), and the Control condition ($p < 0.001$). The results of these pairwise comparisons are shown in Table 5.10; each cell contains the difference in means and its significance value. A graph of the emotional appropriateness rating means is displayed in Figure 5.7.

Table 5.8: Descriptive Statistics for Emotional Appropriateness by Condition

Condition	N	Mean	Median	St. Deviation	Minimum	Maximum
Control	24	3.0625	3.25	1.27102	1.00	5.00
Topics	24	3.8958	4.00	1.23340	1.00	6.00
Emotions	24	3.6667	4.00	1.60615	1.00	6.00
Integrated	24	4.6667	4.50	1.35668	2.00	7.00

Table 5.9: Repeated-Measures ANOVA for Emotional Appropriateness with Sphericity Assumed

Source	Type III Sum of Squares	df	Mean Square	F	Sig.	Partial Eta Squared
Condition	31.677	3	10.559	8.379	.000***	.267
Error(Condition)	86.948	69	1.260			

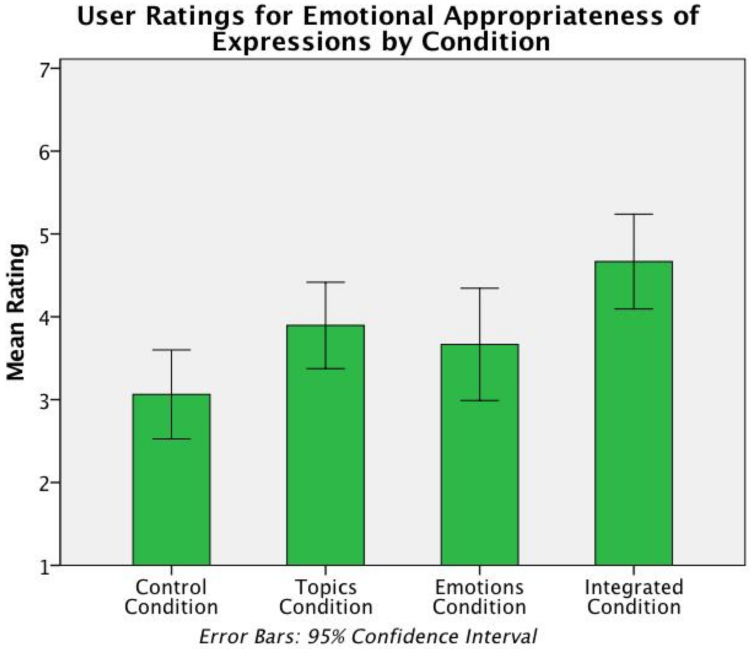


Figure 5.7: Mean emotional appropriateness ratings in each condition.

Table 5.10: Pairwise Comparisons of Emotional Appropriateness Means Using Tukey's HSD

	Control	Topics	Emotions	Integrated
Control	0	.833, $p = .017$*	.604, $p = .051$	1.604, $p < .001$***
Topics		0	$-.229$, $p = .532$	.771, $p = .004$**
Emotions			0	1.000, $p = .010$*
Integrated				0

User Engagement

The user engagement ratings were normally distributed in all condition levels, according to the Shapiro-Wilk test. However, according to Mauchly's Test of Sphericity, the assumption of sphericity was violated ($\chi^2(2) = 12.363, p < .05$). Therefore the Greenhouse-Geisser correction was used in interpreting the ANOVA results. Descriptive statistics for these ratings can be seen in Table 5.11, including mean, median, range, and standard deviation.

A one-way repeated measures ANOVA with Greenhouse-Geisser correction determined that the mean user engagement ratings differed statistically significantly between conditions ($F(2.135, 49.114) = 4.209, p < 0.02$). The results of this test are shown in Table 5.12.

Post hoc testing using Tukey's HSD showed that the Topics condition had a significantly higher mean rating than the Control condition ($p = 0.031$), and the Integrated condition had a significantly higher mean rating than both the Emotions condition ($p = 0.037$) and the Control condition ($p = 0.014$). The results of these pairwise comparisons are shown in Table 5.13; each cell contains the difference in means and its significance value. A graph of the user engagement rating means is displayed in Figure 5.8.

Table 5.11: Descriptive Statistics for User Engagement by Condition

Condition	N	Mean	Median	St. Deviation	Minimum	Maximum
Control	24	4.0000	4.00	1.31049	1.00	6.50
Topics	24	4.5625	4.75	1.18241	2.00	6.00
Emotions	24	4.0417	4.00	1.75026	1.00	7.00
Integrated	24	4.9792	5.00	1.13711	2.50	7.00

Table 5.12: Repeated-Measures ANOVA for User Engagement with Greenhouse-Geisser Correction

Source	Type III Sum of Squares	df	Mean Square	F	Sig.	Partial Eta Squared
Condition	15.604	2.135	7.307	4.209	.019*	.155
Error(Condition)	85.271	49.114	1.736			

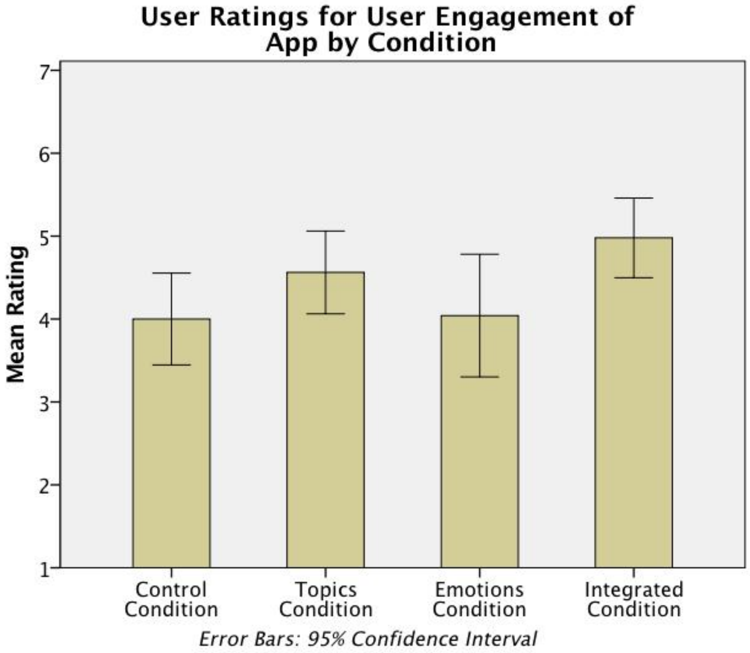


Figure 5.8: Mean user engagement ratings in each condition.

Table 5.13: Pairwise Comparisons of User Engagement Means Using Tukey's HSD

	Control	Topics	Emotions	Integrated
Control	0	$.563, p = .031$*	$.042, p = .871$	$.979, p = .014$*
Topics		0	$-.521, p = .099$	$.417, p = .170$
Emotions			0	$.938, p = .037$*
Integrated				0

5.2.3 Discussion of Results

The results of the evaluation study confirmed some of the hypotheses and failed to confirm others:

- Hypothesis 1 is confirmed: the Topics and Integrated conditions had higher mean topical relevance ratings than the Emotions and Control conditions.
- Hypothesis 2 is partially confirmed: the Integrated condition had a higher mean emotional appropriateness rating than the Topics and Control conditions. The Emotions condition mean, while higher than the Control condition mean, was only marginally significantly different, and it was not at all significantly different from the Topics condition mean.
- Hypothesis 3 is partially confirmed: the Integrated condition had a higher mean user engagement rating than the Emotions and Control conditions, but not the Topics condition. And while the Topics condition mean was higher than the Control condition mean, the Emotions condition mean was not significantly different from the Control condition mean.

These relationships are displayed graphically in Figure 5.9. In these diagrams, an arrow from one condition to another signifies that the first condition was either hypothesized or confirmed to have a significantly higher mean than the second. The arrows are color-coded to represent confirmed hypotheses, unconfirmed hypotheses, and relationships that were not hypothesized.

Several statistical relationships that were not hypothesized additionally became apparent in these tests:

- The Integrated condition had a higher mean topical relevance rating than the Topics condition.
- The Integrated condition had a higher mean emotional appropriateness rating than the Emotions condition.

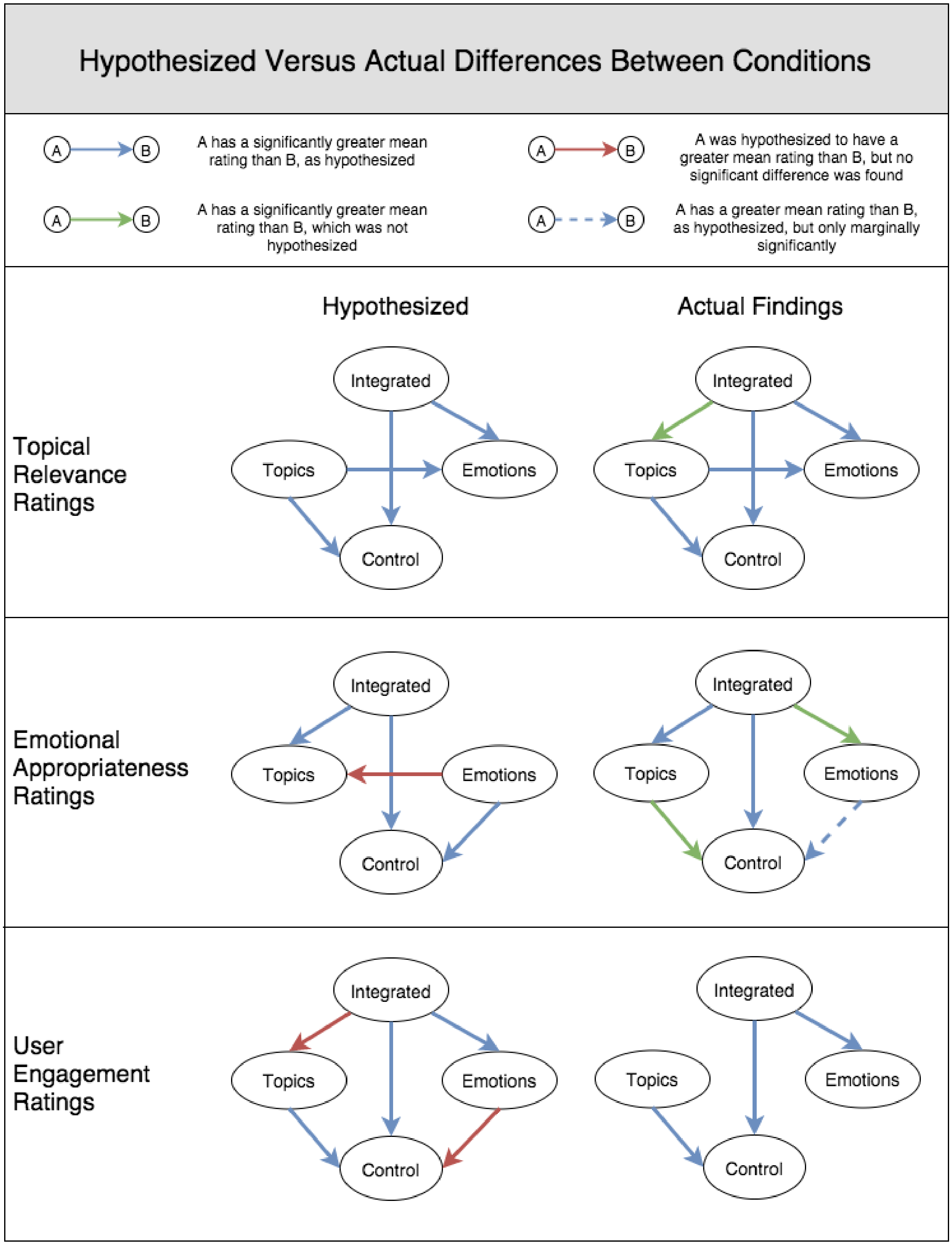

Hypothesized Versus Actual Differences Between Conditions
A
B
A has a significantly greater mean rating than B, as hypothesized
A has a significantly greater mean rating than B, which was not hypothesized
A was hypothesized to have a greater mean rating than B, but no significant difference was found
A has a greater mean rating than B, as hypothesized, but only marginally significantly
Hypothesized
Actual Findings
Topical Relevance Ratings
Emotional Appropriateness Ratings
User Engagement Ratings
Integrated
Topics
Emotions
Control

Figure 5.9: Comparisons between hypothesized and actual differences between conditions within each dependent variable.

- The Topics condition had a higher mean emotional appropriateness rating than the Control condition.

In other words, users perceived the Integrated condition as being both more topically relevant and more emotionally appropriate than any other condition, including both the Topics and Emotions conditions. They also perceived random emotions as being more appropriate when the replies were topically relevant, as in the Topics condition.

This suggests that the combination of the topic detection system and the emotional appropriateness system is more powerful at creating the impression of topical relevance and emotional appropriateness than either system is alone. However, another explanation is that users may have found the random control for one condition disconcerting enough to detract from their perception of the efficacy of the other system, which was operating in parallel with the control.

For example, according to comments made both during and after appointments, users frequently interpreted the robot's emotional expressions as relating to the content of the robot's verbal reply, as opposed to the content of the user's utterance just before it. This is not unreasonable, since the verbal reply and the expression were delivered simultaneously. Therefore, in the Emotions condition, when the robot was speaking random verbal replies, it may have given the impression that it was not "paying attention" to the user's utterances in any way, including emotionally.

Another interesting result relates to the last hypothesis. It was hypothesized that users would find a robot with topically relevant and emotionally appropriate replies more engaging than one with random emotions or random replies, and that the presence of at least one working system would make the robot more engaging than the completely random control condition. However, the results suggest that in general, users found that the topic-detection system contributed more to their engagement than the sentiment-analysis system, although as Figure 5.8 shows, the differences between means in the user engagement ratings were not large.

These results could be influenced by the fact that some users, according to their comments and reactions, found the randomly-generated replies and expressions surprising and entertaining enough to make up for their lack of relevance and appropriateness. These perceptions may have affected the user engagement ratings in some cases.

5.2.4 Technical Limitations

A certain amount of technical interference resulted from the speech-to-text module in the system not being fully adapted for conversational use. Often, users would pause in the middle of sentences, and the Android SpeechRecognizer module would interpret this to mean that their speech was finished. It would then stop listening and send the speech for processing, ignoring the fact that the user had begun speaking again. Alternatively, users would sometimes talk without pause for several sentences, and the module would fail to find a match when it attempted to transcribe these long utterances. This would force users to repeat themselves or keep talking even longer before the robot could reply.

Conditions such as these led to some awkwardness and stiltedness in the conversations. Solutions for these problems would involve using a different speech-to-text transcription module, or building a new one in-house, possibly including a top-down grammatical or semantic interpreter to find likely breakpoints in speech that do not rely wholly on pause length.

Another type of interference resulted from the visual design of the robot's emotional expressions. The robot's "Happy" expression is very similar to its "Neutral" expression, especially as compared to its "Sad" expression. As shown

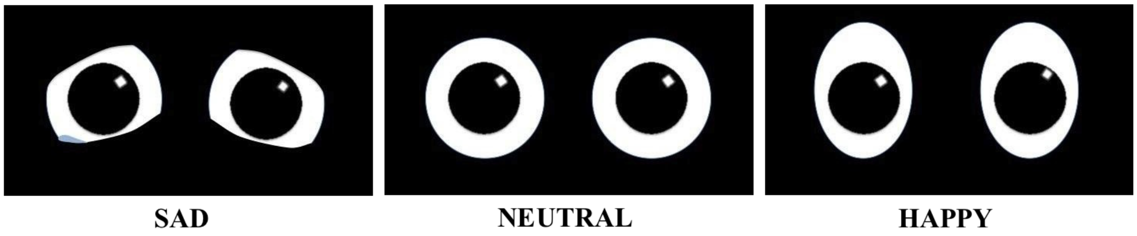


Figure 5.10: The robot’s negative, neutral and positive expressions.

in Figure 5.10, the “Happy” expression differs from the “Neutral” expression only in that its round eyes are stretched vertically, making them taller and more oval, whereas in the “Sad” expression, the eyes assume a different shape entirely and in fact include a small tear in the corner of one eye. Thus the “Sad” expression denotes sadness much more dramatically than the “Happy” expression denotes happiness.

This seemed to cause some uncertainty among users as to when the robot was interpreting their words positively versus neutrally, and when it interpreted their words negatively, they often perceived it as reacting with relative extremeness. This may have affected the emotional appropriateness ratings especially in the random emotion conditions, when robot tended to use the Sad expression more often.

Comments such as these, collected from participants both during and after the appointments, revealed several unanticipated bugs and areas which needed improvement; these will be addressed in future iterations of the system.

CHAPTER 6

CONCLUSION AND FUTURE WORK

6.1 Discussion of Research Hypotheses

The original research hypothesis was that the topic-detection and sentiment-analysis subsystems, if found to be successful and integrated with the robot's interface, would engage users more than either subsystem alone, and that either subsystem alone would engage users more than a random control. This hypothesis was partially confirmed.

Each subsystem was successful in that it generally produced reactions that were judged to be better than random controls, and they both performed weakly but positively in computational evaluations. However, the sentiment-analysis subsystem was judged by users to be less successful than the topic-detection subsystem, and it mainly affected user ratings positively when it was combined with the topic-detection subsystem.

Overall, user engagement was increased by the simultaneous integration of these two subsystems with the interface, albeit unevenly in that the topic-detection subsystem appeared to contribute more to user engagement than the sentiment-analysis subsystem. The research hypothesis was thus partially confirmed, although with some caveats due to technical limitations.

6.2 Future Work

After conducting evaluations and receiving user feedback, it became clear that several improvements to the system are needed. First, the speech-to-text transcription module should be adapted to facilitate a smoother conversational flow. Second, the robot's "Happy" and "Sad" expressions should be redesigned to denote happiness more actively and sadness less dramatically, in order to even out the perceived emotional distance between these two expressions and the “Neutral” expression.

Judging by the computational evaluations, the sentiment-analysis subsystem needs to incorporate negation-handling and emphasis-handling in order to improve its classification accuracy. Furthermore, this subsystem would greatly benefit from being able to change the robot’s emotional expressions both while the user is talking and while the robot itself is talking, and in each case base the expressions on the words being said at the time.

The implementation of this system was additionally intended to be a foundation for the development of more complex conversational systems on the MU-L8 robot. In particular, it is desirable to expand beyond ternary sentiment classifications, in order to take full advantage of the robot’s emotional range. Such an application could be attempted by the use of WordNetAffect [65], for example by replicating Liu et al.’s emotion-detection walk algorithm using ConceptNet [31].

In terms of the topic-detection subsystem, greater accuracy could be achieved by incorporating more conversational corpora and thus expanding the robot’s knowledge base. Additionally, if a Latent Dirichlet Allocation model could be adapted to provide lists of similar words to a query, as opposed to similar documents, then the use of this more statistically-grounded model would enhance the topic-detection subsystem’s performance.

One notable improvement for long-term research would involve seeking conceptual connections between whole user utterances within a conversation, and then using this ability to create a goal-directed conversational style. Such a conversation could end with the robot saving a particular piece of information about the user (such as “User *A* likes cooking”) after having first allowed the user to broach the topic, then having steered the conversation in order to establish this piece of information.

This would constitute a useful user model in the case of future conversations with the same user; for example, the robot could introduce topics related to those known to be interesting to the user, and possibly infer the user's opinions on analagous topics. The robot could even build a model of its own opinions by aggregating the opinions it has heard before and deciding whether to agree or disagree with the user based on the sentiments previously expressed. A robot with its own opinions to discover and relate to could make a much more engaging conversational partner than one without any self-knowledge.

6.3 Conclusion

This project attempted to improve user engagement in a conversational entertainment robot's interface by implementing a back-end system consisting of basic topic-detection and sentiment-analysis subsystems. These subsystems used machine learning and lexical techniques to analyze the topical and sentiment content of user utterances, then to generate relevant verbal replies and appropriate emotional expressions as the robot's reactions. These reactions were generated with the intent to express the robot's personality and continue the conversation with the user in real time.

In evaluations, the system overall performed positively, although in some areas not as strongly as expected. The sentiment-analysis subsystem in particular needs improvement in order to be more effective at engaging users. General user feedback was collected and reviewed, and these comments were used to determine the direction that this research should take next. Further and more complex conversational systems are planned for the MU-L8 robot using this system as a foundation.

REFERENCES

[1] T. Fong, I. Nourbakhsh, and K. Dautenhahn, “A survey of socially interactive robots,” *Robotics and autonomous systems*, vol. 42, no. 3, pp. 143–166, 2003.

[2] C. L. Breazeal, *Designing Sociable Robots*. MIT Press, 2004.

[3] N. Mavridis, “A review of verbal and non-verbal human–robot interactive communication,” *Robotics and Autonomous Systems*, vol. 63, pp. 22–35, 2015.

[4] A. B. Stroud, M. Morris, K. Carey, J. C. Williams, C. Randolph, and A. B. Williams, “MU-L8: The design architecture and 3-D printing of a teen-sized humanoid soccer robot.,” in *Proceedings of the 8th Workshop on Humanoid Soccer Robots, 13th IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, (Atlanta, GA), 2013.

[5] E. Russell, A. B. Stroud, J. Christian, D. Ramgoolam, and A. B. Williams, “SMILE: A portable humanoid robot emotion interface,” in *Proceedings of the 9th ACM/IEEE International Conference on Human-Robot Interaction, Workshop on Applications for Emotional Robots (HRI '14)*, (Bielefeld University, Germany), March 2014.

[6] E. Russell and A. B. Williams, “Effects of SMILE emotional model on humanoid robot user interaction,” in *Proceedings of the 10th ACM/IEEE International Conference on Human-Robot Interaction*, (Portland, Oregon), March 2015.

[7] A. B. Stroud, E. Russell, R. Chinang, K. Carey, J. C. Williams, D. Ramgoolam, and A. B. Williams, “SMILE: A verbal and graphical user interface tool for speech-control of soccer robots,” in *Proceedings of the 14th IEEE-RAS International Conference on Humanoid Robots, 9th Workshop on Humanoid Soccer Robots (Humanoids '14)*, (Madrid, Spain), 2014.

[8] E. Russell, R. J. Povinelli, and A. B. Williams, “Conversational topic detection in human-robot dialogue,” in *Proceedings of the Eleventh Annual ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, ACM, In Review.

[9] F. Sebastiani, “Machine learning in automated text categorization,” *ACM computing surveys (CSUR)*, vol. 34, no. 1, pp. 1–47, 2002.

[10] Y. Yang and J. O. Pedersen, “A comparative study on feature selection in text categorization,” in *ICML*, vol. 97, pp. 412–420, July 1997.

[11] T. Brants, F. Chen, and I. Tsochantaridis, “Topic-based document segmentation with probabilistic latent semantic analysis,” in *Proceedings of the eleventh international conference on Information and knowledge management*, pp. 211–218, ACM, 2002.

[12] H. V. Nguyen and D. J. Litman, “Extracting argument and domain words for identifying argument components in texts,” in *Proceedings of the 2nd Workshop on Argumentation Mining*, pp. 22–28, Association for Computational Linguistics, June 2015.

[13] C.-Y. Lin and E. Hovy, “The automated acquisition of topic signatures for text summarization,” in *Proceedings of the 18th conference on Computational linguistics*, vol. 1, pp. 495–501, Association for Computational Linguistics, July 2000.

[14] O. Medelyan, I. H. Witten, and D. Milne, “Topic indexing with Wikipedia,” in *Proceedings of the AAAI WikiAI workshop*, vol. 1, pp. 19–24, July 2008.

[15] K.-Y. Chen, L. Luesukprasert, and S. cho T. Chou, “Hot topic extraction based on timeline analysis and multidimensional sentence modeling,” *IEEE Transactions on Knowledge and Data Engineering*, vol. 19, no. 8, pp. 1016–1025, 2007.

[16] D. Ramage, C. D. Manning, and S. Dumais, “Partially labeled topic models for interpretable text mining,” in *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 457–465, ACM, August 2011.

[17] A. Gliozzo and C. Strapparava, “Domain kernels for text categorization,” in *Proceedings of the Ninth Conference on Computational Natural Language Learning (CoNLL-2005)*, (Ann Arbor), pp. 56–63, Association for Computational Linguistics, June 2005.

[18] J. Allan, ed., *Topic Detection and Tracking: Event-based Information Organization*, vol. 12 of *The Kluwer International Series on Information Retrieval*. Boston, MA: Kluwer Academic Publishers, 2002.

[19] J. Makkonen, H. Ahonen-Myka, and M. Salmenkivi, “Simple semantics in topic detection and tracking,” *Information Retrieval*, vol. 7, no. 3-4, pp. 347–368, 2004.

[20] D. M. Blei, A. Y. Ng, and M. I. Jordan, “Latent dirichlet allocation,” *The Journal of Machine Learning Research*, vol. 3, pp. 993–1022, 2003.

[21] S. Deerwester, S. Dumais, G. W. Furnas, and T. K. Landauer, "Indexing by latent semantic analysis," *Journal of the American Society for Information Science*, pp. 391–407, 1990.

[22] T. Hofmann, "Unsupervised learning by probabilistic latent semantic analysis," *Machine Learning*, vol. 42, no. 1-2, pp. 177–196, 2001.

[23] A. Agarwal, B. Xie, I. Vovsha, O. Rambow, and R. Passonneau, "Sentiment analysis of twitter data," in *Proceedings of the Workshop on Languages in Social Media*, pp. 30–38, Association for Computational Linguistics, June 2011.

[24] E. Kouloumpis, T. Wilson, and J. Moore, "Twitter sentiment analysis: The good the bad and the OMG!," *ICWSM*, vol. 11, pp. 538–541, 2011.

[25] B. Pang and L. Lee, "Opinion mining and sentiment analysis," *Foundations and trends in information retrieval*, vol. 2, no. 1-2, pp. 1–135, 2008.

[26] E. Cambria, B. Schuller, Y. Xia, and C. Havasi, "New avenues in opinion mining and sentiment analysis," *IEEE Intelligent Systems*, vol. 28, no. 2, pp. 15–21, 2013.

[27] T. Nasukawa and J. Yi, "Sentiment analysis: Capturing favorability using natural language processing," in *Proceedings of the 2nd international conference on Knowledge capture*, pp. 70–77, ACM, October 2003.

[28] M. Taboada, J. Brooke, M. Tofiloski, K. Voll, and M. Stede, "Lexicon-based methods for sentiment analysis," *Computational Linguistics*, vol. 37, no. 2, pp. 267–307, 2011.

[29] R. McDonald, K. Hannan, T. Neylon, M. Wells, and J. Reynar, "Structured models for fine-to-coarse sentiment analysis," in *Annual Meeting-Association For Computational Linguistics*, vol. 45, pp. 432–439, Association for Computational Linguistics, June 2007.

[30] T. Wilson, J. Wiebe, and P. Hoffmann, "Recognizing contextual polarity in phrase-level sentiment analysis," in *Proceedings of the conference on human language technology and empirical methods in natural language processing*, pp. 347–354, Association for Computational Linguistics, October 2005.

[31] H. Liu, H. Lieberman, and T. Selker, "A model of textual affect sensing using real-world knowledge," in *Proceedings of the 8th International Conference on Intelligent User Interfaces*, pp. 125–132, ACM, 2003.

[32] A. Sureka, V. Goyal, D. Correa, and A. Mondal, "Generating domain-specific ontology from common-sense semantic network for target-specific sentiment analysis," in *Fifth International Conference of the Global WordNet Association (GWC)*, 2010.

[33] E. Boiy, P. Hens, K. Deschacht, and M.-F. Moens, "Automatic sentiment analysis in on-line text," in *ELPUB*, pp. 349–360, June 2007.

[34] L.-P. Morency, R. Mihalcea, and P. Doshi, "Towards multimodal sentiment analysis: Harvesting opinions from the web," in *Proceedings of the 13th international conference on multimodal interfaces*, pp. 169–176, ACM, November 2011.

[35] G. Paltoglou, S. Gobron, M. Skowron, M. Thelwall, and D. Thalmann, "Sentiment analysis of informal textual communication in cyberspace," in *In Proceedings of Engage 2010, Springer LNCS State-of-the-Art Survey*, pp. 13–25, 2010.

[36] S. Raaijmakers, K. Truong, and T. Wilson, "Multimodal subjectivity analysis of multiparty conversation," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pp. 446–474, Association for Computational Linguistics, October 2008.

[37] S. Mukherjee and S. Joshi, "Sentiment aggregation using ConceptNet ontology," in *Proceedings of the 6th International Joint Conference on Natural Language Processing, IJCNLP*, (Nagoya, Japan), pp. 570–578, October 2013.

[38] R. Socher, J. Pennington, E. H. Huang, A. Y. Ng, and C. D. Manning, "Semi-supervised recursive autoencoders for predicting sentiment distributions," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pp. 151–161, Association for Computational Linguistics, July 2011.

[39] X. Glorot, A. Bordes, and Y. Bengio, "Domain adaptation for large-scale sentiment classification: A deep learning approach," in *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*, pp. 513–520, 2011.

[40] F. Berthelon and P. Sander, "Emotion ontology for context awareness," in *4th IEEE International Conference on Cognitive Infocommunications (CogInfoCom)*, pp. 59–64, December 2013.

[41] C.-H. Wu, Z.-J. Chuang, and Y.-C. Lin, "Emotion recognition from text using semantic labels and separable mixture models," *ACM transactions on Asian language information processing (TALIP)*, vol. 5, pp. 165–182, June 2006.

[42] S. Marsella and J. Gratch, "Computationally modeling human emotion," *Communications of the ACM*, vol. 57, no. 12, pp. 56–67, 2014.

[43] M. Munezero, C. S. Montero, E. Sutinen, and J. Pajunen, "Are they different? affect, feeling, emotion, sentiment, and opinion detection in text," *IEEE transactions on affective computing*, vol. 5, pp. 101–111, April-June 2014.

[44] I. Chazanovitz and M. Greenwald, "Text based emotion estimation," tech. rep., Ben-Gurion University of the Negev Department of Computer Science, September 2008.

[45] Z.-J. Chuang and C.-H. Wu, "Multi-modal emotion recognition from speech and text," *Computational Linguistics and Chinese Language Processing*, vol. 9, pp. 45–62, August 2004.

[46] L. Zhang, M. Jiang, D. Farid, and M. A. Hossain, "Intelligent facial emotion recognition and semantic-based topic detection for a humanoid robot," *Expert Systems with Applications*, vol. 40, no. 13, pp. 5160–5168, 2013.

[47] C. Strapparava and R. Mihalcea, "Learning to identify emotions in text," in *Proceedings of the 2008 ACM symposium on Applied computing*, pp. 1556–1560, ACM, March 2008.

[48] M. Nakano, Y. Hasegawa, K. Nakadai, T. Nakamura, J. Takeuchi, T. Torii, H. Tsujino, N. Kanda, and H. G. Okuno, "A two-layer model for behavior and dialogue planning in conversational service robots," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2005)*, pp. 3329–3335, IEEE, 2005.

[49] J. F. Maas, T. Spexard, J. Fritsch, B. Wrede, and G. Sagerer, "BIRON, what's the topic? A multi-modal topic tracker for improved human-robot interaction," in *The 15th IEEE International Symposium on Robot and Human Interactive Communication (ROMAN)*, pp. 26–32, IEEE, September 2006.

[50] M. Heerink, B. Kröse, B. Wielinga, and V. Evers, "Enjoyment intention to use and actual use of a conversational robot by elderly people," in *Proceedings of the 3rd ACM/IEEE international conference on Human Robot Interaction*, pp. 113–120, ACM, 2008.

[51] Y. Matsusaka, S. Fujie, and T. Kobayashi, "Modeling of conversational strategy for the robot participating in the group conversation," in *INTERSPEECH*, vol. 1, pp. 2173–2176, September 2001.

[52] S. Fujie, K. Fukushima, and T. Kobayashi, “A conversation robot with back-channel feedback function based on linguistic and nonlinguistic information,” in *Proceedings of the ICARA International Conference on Autonomous Robots and Agents*, pp. 379–384, 2004.

[53] K. Jokinen and G. Wilcock, “Constructive interaction for talking about interesting topics,” in *LREC*, pp. 404–410, 2012.

[54] S. T. Dumais, G. W. Furnas, T. K. Landauer, S. Deerwester, and R. Harshman, “Using latent semantic analysis to improve access to textual information,” in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pp. 281–285, ACM, 1988.

[55] T. K. Landauer, P. W. Foltz, and D. Laham, “An introduction to latent semantic analysis,” *Discourse Processes*, vol. 25, no. 2-3, pp. 259–284, 1998.

[56] R. Řehůřek and P. Sojka, “Software framework for topic modelling with large corpora,” in *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*, (Valletta, Malta), pp. 45–50, ELRA, May 2010.

[57] C. Cieri, D. Graff, O. Kimball, D. Miller, and K. Walker, “Fisher English Training parts 1 and 2, speech and transcripts,” *Linguistic Data Consortium, Philadelphia*, 2005.

[58] R. Speer and C. Havasi, “Representing general relational knowledge in ConceptNet 5,” in *LREC*, pp. 3679–3686, May 2012.

[59] M. Keshavarz and Y.-H. Lee, “Ontology matching by using ConceptNet,” in *Proceedings of the Asia Pacific Industrial Engineering & Management Systems Conference 2012*, pp. 1917–1925, 2012.

[60] R. Speer, C. Havasi, and H. Lieberman, “AnalogySpace: Reducing the dimensionality of common sense knowledge,” in *AAAI*, vol. 8, pp. 548–553, July 2008.

[61] S. Bird, E. Klein, and E. Loper, *Natural Language Processing with Python*. O’Reilly Media, Inc., 2009.

[62] C. Fellbaum, ed., *WordNet: An Electronic Lexical Database*. Cambridge, MA: MIT Press, 1998.

[63] S. Baccianella, A. Esuli, and F. Sebastiani, “SentiWordNet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining,” in *LREC*, vol. 10, pp. 2200–2204, May 2010.

[64] A. J. Bishara and J. B. Hittner, “Testing the significance of a correlation with nonnormal data: comparison of Pearson, Spearman, transformation, and resampling approaches,” *Psychological Methods*, vol. 17, no. 3, p. 399, 2012.

[65] C. Strapparava and A. Valitutti, “WordNet-Affect: an affective extension of WordNet,” in *Proceedings of the 4th International Conference on Language Resources and Evaluation (LREC)*, vol. 4, (Lisbon), pp. 1083–1086, May 2004.

APPENDIX A

LIST OF TOPIC PROMPTS USED IN THE FISHER ENGLISH TRAINING TRANSCRIPTS CORPUS

The following is a table of the 40 conversational topic prompts of the Fisher English Training Transcripts Corpus [57]. These prompts are used in both the computational and perceputal evaluations of this project. An asterisk (*) next to the topic ID indicates one of the subset of topic prompts that was adapted for the perceptual evaluation, described in Section 5.2.

ID	Topic Name	Prompt
ENG01	Professional Sports on TV	Do either of you have a favorite TV sport? How many hours per week do you spend watching it and other sporting events on TV?
ENG02*	Pets	Do either of you have a pet? If so, how much time each day do you spend with your pet? How important is your pet to you?
ENG03*	Life Partners	What do each of you think is the most important thing to look for in a life partner?
ENG04	Minimum Wage	Do each of you feel the minimum wage increase – to $5.15 an hour – is sufficient?
ENG05*	Comedy	How do you each draw the line between acceptable humor and humor that is in bad taste?
ENG06	Hypothetical Situations. Perjury	Do either of you think that you would commit perjury for a close friend or family member?

ID	Topic Name	Prompt
ENG07	Hypothetical Situations. One Million Dollars to Leave the US	Would either of you accept one million dollars to leave the US and never return? If you were willing to leave, where would you go, what would you do? What would you miss the most about the US? What would you not miss?
ENG08	Hypothetical Situations. Opening Your Own Business	If each of you could open your own business, and money were not an issue, what type of business would you open? How would you go about doing this? Do you feel you would be a successful business owner?
ENG09*	Hypothetical Situations. Time Travel	If each of you had the opportunity to go back in time and change something that you had done, what would it be and why?
ENG10	Hypothetical Situations. An Anonymous Benefactor	If an unknown benefactor offered each of you a million dollars - with the only stipulation being that you could never speak to your best friend again - would you take the million dollars?
ENG11	US Public Schools	In your opinions, is there currently something seriously wrong with the public school system in the US, and if so, what can be done to correct it?
ENG12	Affirmative Action	Do either of you think affirmative action in hiring and promotion within the business community is a good policy?

ID	Topic Name	Prompt
ENG13	Movies	Do each of you enjoy going to the movies in a theater, or would you rather rent a movie and stay home? What was the last movie that you saw? Was it good or bad and why?
ENG14	Computer Games	Do either of you play computer games? Do you play these games on the internet or on CD- ROM? What is your favorite game?
ENG15*	Current Events	How do both of you keep up with current events? Do you get most of your news from TV, radio, newspapers, or people you know?
ENG16*	Hobbies	What are your favorite hobbies? How much time do each of you spend pursuing your hobbies? Do you feel that every person needs at least one hobby?
ENG17*	Smoking	How do you both feel about the movement to ban smoking in all public places? Do either of you think Smoking Prevention Programs, Counter-smoking ads, Help Quit hotlines and so on, are a good idea?
ENG18	Terrorism	Do you think most people would remain calm, or panic during a terrorist attack? How do you think each of you would react?

ID	Topic Name	Prompt
ENG19*	Televised Criminal Trials	Do either of you feel that criminal trials, especially those involving high-profile individuals, should be televised? Have you ever watched any high-profile trials on TV?
ENG20	Drug Testing	How do each of you feel about the practice of companies testing employees for drugs? Do you feel unannounced spot-checking for drugs to be an invasion of a person's privacy?
ENG21	Family Values	Do either of you feel that the increase in the divorce rate in the US has altered your behavior? Has it changed your views on the institution of marriage?
ENG22*	Censorship	Do either of you think public or private schools have the right to forbid students to read certain books?
ENG23*	Health and Fitness	Do each of you exercise regularly to maintain your health or fitness level? If so, what do you do? If not, would you like to start?
ENG24	September 11	What changes, if any, have either of you made in your life since the terrorist attacks of Sept 11, 2001?
ENG25	Strikes by Professional Athletes	How do each of you feel about the recent strikes by professional athletes? Do you think that professional athletes deserve the high salaries they currently receive?

ID	Topic Name	Prompt
ENG26	Airport Security	Do either of you think that heightened airport security lessens the chance of terrorist incidents in the air?
ENG27	Issues in the Middle East	What does each of you think about the current unrest in the Middle East? Do you feel that peace will ever be attained in the area? Should the US remain involved in the peace process?
ENG28	Foreign Relations	Do either of you consider any other countries to be a threat to US safety? If so, which countries and why?
ENG29	Education	What do each of you think about computers in education? Do they improve or harm education?
ENG30*	Family	What does the word family mean to each of you?
ENG31	Corporate Conduct in the US	What do each of you think the government can do to curb illegal business activity? Has the cascade of corporate scandals caused the mild recession and decline in the US stock market and economy? How have the scandals affected you?
ENG32*	Outdoor Activities	Do you like cold weather or warm weather activities the best? Do you like outside or inside activities better? Each of you should talk about your favorite activities.

ID	Topic Name	Prompt
ENG33*	Friends	Are either of you the type of person who has lots of friends and acquaintances or do you just have a few close friends? Each of you should talk about your best friend or friends.
ENG34*	Food	Which do each of you like better – eating at a restaurant or at home? Describe your perfect meal.
ENG35*	Illness	When the seasons change, many people get ill. Do either of you? What do you do to keep yourself well? There is a saying, ‘A cold lasts seven days if you don’t go to the doctor and a week if you do.’ Do you both agree?
ENG36*	Personal Habits	According to each of you, which is worse: gossiping, smoking, drinking alcohol or caffeine excessively, overeating, or not exercising?
ENG37*	Reality TV	Do either of you watch reality shows on TV. If so, which one or ones? Why do you think that reality based television programming, shows like ’Survivor’ or ’Who Wants to Marry a Millionaire’ are so popular?
ENG38	Arms Inspections in Iraq	What, if anything, do you both think the US should do about Iraq? Do you think that disarming Iraq should be a major priority for the US?

ID	Topic Name	Prompt
ENG39*	Holidays	Do either of you have a favorite holiday? Why? If either of you you could create a holiday, what would it be and how would you have people celebrate it?
ENG40	Bioterrorism	What do you both think the US can do to prevent a bioterrorist attack?