A Usage-Based Model for the Acquisition of Syntactic Constructions and its Application in Spoken Language Understanding

Dissertation

zur Erlangung des Grades

Doctor rerum naturalium (Dr. rer. nat.)

der technischen Fakultät der Universität Bielefeld

vorgelegt von

Judith Gaspers

Bielefeld, Juli 2014

Gutachter:
Prof. Dr. Philipp Cimiano
Prof. Dr.-Ing. Britta Wrede
Prof. Dr. Suzanne Stevenson

Prüfungsausschuss:
Prof. Dr.-Ing. Stefan Kopp
Prof. Dr. Philipp Cimiano
Prof. Dr.-Ing. Britta Wrede
Prof. Dr. Suzanne Stevenson
Dr. Robert Haschke

Gedruckt auf alterungsbeständigem Papier nach ISO 9706

Abstract

According to usage-based approaches to language acquisition, linguistic knowledge is represented in the form of constructions as pairings of meaning and form at multiple levels of abstraction and complexity. The emergence of syntactic knowledge is assumed to be a result of the gradual abstraction of lexically specific and item-based knowledge. In this thesis, we first explore how the gradual emergence of a network consisting of constructions at varying degrees of complexity and abstraction can be modeled computationally by formalizing ideas proposed within usage-based theories to language acquisition and construction grammar. Similar to a child, the model learns language by observing natural language utterances, represented as sequences of words, in an ambiguous context. Starting from ambiguous contexts, the model establishes form-meaning mappings based on cross-situational statistics, a mechanism also referred to as cross-situational learning. In contrast to previous models investigating cross-situational learning, which typically focused on word-referent mappings, we explore how the same cross-situational learning mechanism can be applied consistently to establish form-meaning pairings beyond such simple mappings. We present empirical results, showing that the model can learn a compact and precise representation of the input data which generalizes well to unseen data. In line with findings from psycholinguistic studies with children, language learning in the model proceeds gradually. The model's generalization abilities are initially limited, increase over time, and finally converge, suggesting that the employed mechanisms allow accurate learning without (severe) deterioration of knowledge already captured by the network during further processing of examples. In addition, we present empirical results that are in line with recent findings from psycholinguistic studies with children and that show i) how our model is able to perform cross-situational verb learning by storing information about possible referents with verb entries and ii) how it can establish verb entries based on syntactic information alone. The model thus suggests learning mechanisms that may be at play during the emergence of verb-general constructions and the representation and refinement of verb entries.

In a further step, we explore learning from spoken utterances, instead of sequences of words. We assume no predefined linguistic resources other than a task-independent phoneme recognizer, and thus also address lexical acquisition. Applying a phoneme-based speech recognizer has several advantages over applying a word-based one: It yields low costs for training, makes it easy to adapt the system to novel tasks and supports the acquisition of a potentially unrestricted vocabulary. While previous research has addressed learning novel words from speech without word transcription, we are not aware of other algorithms learning syntactic constructions using ambiguous non-linguistic contexts. We present empirical results, showing that i) when applied to a written language understanding task, our algorithm achieves state-of-the-art performance, ii) when applied to a spoken language understanding task still several novel utterances can be understood and, in fact, iii) performance similar to applying a word-based in-domain speech recognizer can be expected. Further, we show how knowledge about syntactic patterns can be utilized to improve segmentation and language learning performance and how semantic speech recognition grammars, which have typically been created manually or learned in a supervised setting, can be induced using ambiguous contextual representations.

Acknowledgements

First of all, I would like to thank my supervisor Philipp Cimiano for providing me the opportunity to work in such an interesting and challenging research field. I am deeply grateful for his constant support, valuable feedback and inspiring ideas which have been implemented in this thesis. I am also grateful to Britta Wrede for fruitful discussions, constructive feedback, and for accepting to review this thesis. Furthermore, I would like to thank Suzanne Stevenson for serving as my external examiner, and in particular for coming to Bielefeld to attend my defense, and Stefan Kopp and Robert Haschke for joining my PhD committee.

I would like to thank my colleagues at the Semantic Computing group, Maximilian Panzner, Timo Reuter, Christina Unger, Sebastian Walter, John McCrae, Roman Klinger, Peter Menke, Cord Wiljes, Matthias Hartung, and Sherzod Hakimov, for the supportive working atmosphere and enjoyable time. Special thanks go to Anouschka Foltz for useful discussions and feedback concerning psycholinguistic aspects of child language acquisition and Sascha Griffiths and Oliver Beyer for inspiring discussions in the context of our former interest group on symbol grounding and language learning.

My thanks also go to my former student assistants Valerie van Hövell and Sascha Hinte for annotating the RoboCup corpus and Frederike Strunz for annotating the Eve corpus. Furthermore, I want to thank David Horn for sharing the JADIOS implementation of the ADIOS algorithm and Rolf Schatten for sharing background information on associative networks.

I wish to thank my family and friends for their great support, and my parents also for supporting my studies over so many years. I want to especially thank my mother for always believing in my abilities and for her interest in my work and Martin for all his support and for feedback on this thesis.

I am also grateful for financial support which has been provided by the German Research Foundation DFG in the context of the Collaborative Research Center 673

Alignment in Communication and the Center of Excellence *Cognitive Interaction Technology*.

Contents

1. Introduction 15
   - 1.1. Computational modeling of child language acquisition 17
   - 1.2. Learning from speech without word transcriptions 19
   - 1.3. Contributions 20
   - 1.4. Outline 22

2. Background 25
   - 2.1. Automatic Speech Recognition 26
   - 2.2. Spoken Language Understanding 30
   - 2.3. Segmentation of a continuous sequence of discrete units into word-like units 34
   - 2.4. Learning linguistic structures from speech without word transcriptions 36
   - 2.5. Word-to-meaning mapping 38
   - 2.6. Syntactic acquisition from raw text 39
   - 2.7. Models for the acquisition of syntactic constructions 41
   - 2.8. Context information in computational language learning 44
   - 2.9. Summary 46

3. A computational model for the acquisition of verb-specific constructions 49
   - 3.1. Introduction 49
   - 3.2. Learning problem 55
     - 3.2.1. Input 55
     - 3.2.2. Goal 56
   - 3.3. The computational model 58
     - 3.3.1. Basic components 59
     - 3.3.2. Network structure 67

3.3.3. Generalization 69
3.3.4. Confidence 71
3.3.5. Language learning algorithm 73
3.3.6. Retrieval of constructions 78
3.4. Experimental evaluation and discussion 80
3.4.1. Dataset 80
3.4.2. Semantic parsing 81
3.4.3. Qualitative analysis of acquired constructions 84
3.4.4. Learning over time 87
3.4.5. Parameters and confidence 89
3.4.6. Fast mapping 93
3.5. General discussion 97
3.5.1. Semantic parsing with ambiguous context information 97
3.5.2. Usage-based language acquisition 99
3.5.3. Cross-situational learning 101
3.5.4. Limitations and possible extensions 102
3.6. Summary 106

4. From verb-specific to verb-general constructions 109
4.1. Introduction 109
4.2. Learning problem 111
4.3. The extended computational model 112
4.3.1. Network structure 112
4.3.2. Generalization and confidence 113
4.3.3. Language learning 114
4.3.4. Retrieval of constructions and syntactic frames 117
4.4. Experimental evaluation and discussion 118
4.4.1. Input data 118
4.4.2. Cross-situational verb learning 120
4.4.3. Syntax as a zooming lens into semantics 122
4.5. General discussion 125
4.6. Summary 128

5. Learning a semantic parser from speech without word transcriptions 131
5.1. Introduction 132
5.2. Learning problem 134
5.3. Algorithm 137
5.3.1. Acquisition of an initial lexicon 142

5.3.2. Bottom-up creation of alignments 142
5.3.3. Creating a parser 144
5.3.4. Top-down creation of alignments 144
5.3.5. Parsing 145
5.4. Experimental evaluation and discussion 145
5.4.1. Datasets 146
5.4.2. Application to written text 147
5.4.3. Application to simulated ASR word transcriptions 148
5.4.4. Application to ASR phoneme transcriptions 151
5.4.5. On the role of knowledge about syntactic patterns in segmentation 153
5.4.6. Phonemic variation and its effects on parsing and learning . . 155
5.4.7. Application for the induction of semantic speech recognition grammars 159
5.5. General discussion 165
5.6. Summary 172

6. Summary 175

Bibliography 181

A. Similarity computation between phoneme strings 199

List of Figures

1. An example for an SRGS grammar including semantic information in SISR. 29

1. Example of a network capturing associations between form and meaning at the lexical level, and the execution of an update step changing the contained weights. 60
2. Example for the incorporation of new nodes at the word level. 62
3. Example for the combination of weights in an associative network. . . 64
4. Example for combining the information of two associative networks. . 65
5. Example for the induction of sets of elements by merging paths. . . . 67
6. Network modeling three levels of association 68
7. Example illustrating the incorporation of nodes and connections when starting with an empty network. 75
8. Example of a network representing two concrete natural language utterances along with semantic correspondences. 76
9. Example of a network representing a syntactic pattern along with semantic correspondences. 77
10. Example of a small network. 79
11. Change in F_1, precision and recall over the number of observed examples. 88
12. Number of observed and stored *NL* patterns over time. 89
13. Number of observed and stored *NL* patterns over time. 90
14. Change in F_1 when varying a rating threshold while keeping the remaining parameters constant 91
15. Change in F_1 when varying an entropy threshold while keeping the remaining parameters constant 92

1. Example of a verb-specific construction referring to the predicate *see* stored in the network. 117
2. Example of a verb-specific construction referring to the predicate *take* stored in the network. 118
3. Example of a verb-general construction stored in the network. 119
4. Proportion of the model's choice of the correct semantic representation for the novel verbs in the transitive and intransitive sentences. . 121
5. Proportion of the model's choice of the causative scene in the transitive and conjoined-subject intransitive conditions. 124
6. Proportion of the model's choice of the causative scene in the conjoined-subject intransitive conditions. 125

1. Overview of the learning process. 135
2. The algorithm's work flow. 138
3. Example of an alignment between form and meaning 143
4. F_1 over the number of top-down re-estimations. 153
5. Percentage of correctly segmented players over the number of top-down re-estimations. 154
6. A subset of weighted speech recognition grammar rules. 160
7. A subset of a purely syntactic speech recognition grammar rules obtained via ADIOS 161

List of Tables

3.1. Some statistics for the RoboCup training dataset 81
3.2. F_1 scores for different construction grammars using different datasets and with varying times training data was seen. 83
3.3. Predicates in the RoboCup corpus along with their counts in the training dataset and the number of comments annotated with them in the gold standard, and the five most frequent verb forms present in the training data. 85
3.4. Number of patterns associated with the individual meanings averaged over all folds. 86

5.1. Some statistics for the RoboCup training dataset, the grapheme-to-phoneme converted data, and the speech data transcribed by a phoneme recognizer. 147
5.2. Semantic parsing results for written text 148
5.3. Semantic parsing results for learning from and testing on data containing different amounts of simulated recognition errors. 151
5.4. Results for the application to ASR transcribed phoneme sequences. . 152
5.5. Sequences appearing most frequently in the data for a number of semantic referents, along with the top entry in the initial lexicon and the sequence extracted for the referent most frequently by the system. 156
5.6. Top five most frequent sequences in the data and aligned by the system exemplary for *purple10*. 157
5.7. F_1, precision and recall when taking a different number of top sequences into account for each semantic referent during parsing. 158
5.8. Speech recognition results using different language models. 162
5.9. Semantic parsing results on written text and on speech transcribed using different language models 163

A.1. Overview of distinctive features utilized in this thesis based on Yu & Ballard (2002). 201

Chapter 1

Introduction

During language acquisition, children accomplish several learning tasks, such as learning a comprehensive vocabulary and mastering syntax. However, while extensive knowledge concerning child language acquisition exists, the principles underlying the learning process remain relatively unclear, and a theory explaining language acquisition from birth until the age of five, which is generally accepted and verified, does not yet exist (Räsänen, 2012). This thesis aims to shed light on the potential learning mechanisms at play during child language acquisition by exploring how a lexicon and an inventory of syntactic constructions can be acquired and proposing a computational model for this. For this purpose we formalize ideas proposed within one of the major theoretical approaches to language acquisition and explore further representations and learning mechanisms.

The major competing theoretical approaches to language acquisition are generativist or nativist approaches (Eisenbeiß, 2009) on the one hand and usage-based or emergentist approaches (Behrens, 2009) on the other hand. While the traditional generative approach originally proposed by Chomsky (1965) assumes that language ability is hard-wired/innate in form of a "universal grammar" (UG), usage-based approaches assume that linguistic knowledge is learned through interaction with and exposure to language in some context or environment. Specifically, usage-based approaches typically assume linguistic knowledge to be acquired and represented in terms of *constructions* as proposed within the framework of construction grammar. According to Goldberg (Goldberg & Suttle, 2010; Goldberg, 2003), construction grammar assumes that linguistic knowledge is represented in the form of form-meaning-pairings – so-called constructions – at varying degrees of complexity and abstraction. These constructions are assumed to be captured by an interrelated network which comprises both item-specific information and generalized patterns, and natural language utterances are assumed to be created based on the network by combining comprised constructions. For instance, an utterance like “what did Mia eat?” is, among other things, composed of single word constructions, i.e. “what”, “Mia”, “did”, “eat”, but also of more complex ones such as a noun phrase construction, a verbal phrase construction and a question construction. It is further assumed that the network of constructions is learned on the basis of positive input coupled with domain-independent learning mechanisms (Goldberg & Suttle, 2010; Goldberg, 2003).

Both construction grammar and usage-based approaches to language acquisition assume that language learning proceeds gradually from item-based and formulaic linguistic knowledge to abstract linguistic knowledge. More specifically, tracing back to the *verb-island hypothesis* proposed by Tomasello (1992), these approaches assume that, early on, children maintain an inventory of lexically specific and item-based constructions. These are then gradually generalized on a verb-specific basis, i.e. patterns correspond to concrete verbs, by replacing concrete lexical items with slots which can be filled by (a restricted group of) words or short sequences of words (Tomasello et al., 1997), yielding verb-specific predicate structures, i.e. *verb-islands*. It is not known in detail how children induce such slots, but one hypothesis is that they observe type variation in a position of otherwise identical utterances (Tomasello, 2000a). Moreover, new linguistic qualities are also assumed to emerge in the sense that more complex structures can emerge from simpler ones (Behrens, 2009).

It is still under debate to what extent human language skills are indeed innate (e.g. Eisenbeiß, 2009). Yet, several studies investigating child language acquisition provide support for an item-based nature of children’s early linguistic knowledge (e.g. Tomasello, 2000b, 2003; Olguin & Tomasello, 1993; Lieven et al., 1997) as well as evidence for the existence of domain-independent learning abilities in children (e.g. Saffran et al., 1996; Saffran, 2003; Aslin et al., 1999). Furthermore, recent findings suggest that statistical learning is implicated in the acquisition of grammar (Kidd, 2012). However, a detailed and precise description of the underlying learning mechanisms and representations is missing. For instance, it remains rather unclear how statistical learning mechanisms are implicated in the acquisition of syntactic patterns, i.e. how they interact with other learning mechanisms such as rule learning, and how exactly the involved generalization processes may operate.

Further, recent studies with children provide novel insights into the emergence of verb-general constructions and the representation and refinement of early verb entries. In particular, psycholinguistic findings suggest i) that children can set up initial verb entries based on syntactic information alone (Arunachalam & Waxman, 2010), and ii) that they store information about possible referents and co-occurrence statistics with verb entries and update this information incrementally over time (Scott & Fisher, 2012). However, again a detailed and precise description of the underlying learning mechanisms and representations is missing. For instance, it remains relatively unclear how verb-general constructions emerge, how they are represented, and how they can guide attention to establish verb entries based on syntactic information alone.

In this thesis, we explore how ideas proposed within usage-based approaches to language acquisition and construction grammar can be formalized and modeled computationally. In addition, we investigate how the explored learning mechanisms can be utilized and extended for application in a spoken language understanding task, aiming at the design of more flexible and adaptive systems. In the following, we will first outline work on developing a computational model for child language acquisition (Section 1.1) and subsequently address computational language learning in spoken language understanding systems (Section 1.2).

1.1. Computational modeling of child language acquisition

Computational models are an important tool in language acquisition research since they can be applied to test hypotheses concerning child language acquisition and verify the plausibility of theoretical models by performing simulations with the implemented model (Räsänen, 2012). More specifically, while psycholinguistic theories can be rather vague, computational models provide a precise implementation which can be tested in order to verify its workings and thus plausibility. For instance, a computational model can be applied to corpora of child-directed speech to test its language learning abilities or to simulate findings from psycholinguistic studies with children.

In this thesis we aim to provide a usage-based computational model for the gradual acquisition of syntactic constructions. Similar to a child, our model learns language through exposure to language in some environment. To illustrate our learning scenario, consider a child who observes an utterance “Mia eats pizza” while several actions take place concurrently. For example, Mia might actually be eating pizza, but moreover, further actions may take place concurrently. For instance, another person might also be eating pizza and a dog might be barking while the utterance is being uttered. Similar to such natural learning situations, in this thesis we attempt to learn language from the input of two temporarily paired channels: a language channel and a visual channel. For the sake of simplicity and since we attempt to model a stage in learning where syntactic constructions emerge gradually, we assume that at the modeled stage of learning the child is already able to extract words from the speech signal and structured representations for actions from the visual context. Visual information is given in the form of symbolic representations of actions by means of predicate logic formulas.

In line with natural settings, the input is ambiguous in the sense that an utterance is presented together with many different actions and it is not clear if the utterance refers to any of the actions and if so to which one. This problem is also known as *referential uncertainty* (Quine, 1960). A great challenge is hence to induce the appropriate meanings starting from a set of ambiguous contexts. In order to establish correspondences between form and meaning, we rely on the principle of *cross-situational learning*. Many researchers (e.g. Quine, 1960; Pinker, 1989; Gleitman, 1990; Siskind, 1996; Smith & Yu, 2008) assume that this mechanism enables children to establish correct form-meaning mappings in the presence of referential uncertainty. The assumption is that form – words in particular – and the meaning they refer to co-occur frequently enough so that mappings between meaning and form can be derived based on co-occurrence statistics. In contrast to previous models investigating cross-situational learning (e.g. Fazly et al., 2010; Frank et al., 2007; Yu, 2005), which typically focused on word-referent mappings, we explore how the same cross-situational learning mechanism can be applied consistently to establish form-meaning pairings beyond such simple mappings.

Importantly, in our model language learning proceeds online, i.e. each observed example directly causes an update of the model. This is an important aspect with respect to modeling human language acquisition skills because the resulting models may not only account for why a certain behavior emerges, but also address the question of how the behavior may be learned given constraints on the infant learner, e.g. on memory (Pearl et al., 2011). In particular, just like Pearl et al. (2011) we assume that a learner can only process one utterance at a time, in our case presented together with concurrent information derived from the visual context. This contrasts with models storing a whole dataset in memory and processing over the data in batch mode, often even iterating over the data several times during learning.

Due to the fact that the verb-island hypothesis assumes that generalization is (initially) performed on a verb-specific basis, in this thesis we first present a computational model for the gradual acquisition of an inventory containing verb-specific constructions. Since recent work also provides novel insights concerning the emergence of verb-general constructions, afterwards we present an extension of the model to also learn verb-general constructions.

1.2. Learning from speech without word transcriptions

State-of-the-art Spoken Language Understanding (SLU) systems are typically based on predefined linguistic resources, e.g. lexicons and/or grammars. Building such resources usually requires extensive manual effort and/or large amounts of (labeled) training data, making them costly to produce. The resulting systems are also often out-dated rather quickly during application, since one cannot know at design time which linguistic knowledge is needed during applications, e.g. which words a user may utter. Moreover, natural languages simply do not have fixed vocabularies. By contrast, children are able to learn linguistic structures by being exposed to language in some context or environment, and they continuously adapt their knowledge to novel input, e.g. they acquire novel lexical entries over time. Thus, computational models for child language acquisition may also inform SLU research on the design of self-adaptive systems. Further, exploring algorithms which i) learn language similarly to children directly from examples of spoken language utterances coupled with non-linguistic information and ii) rely on as few predefined resources as possible can yield not only self-adaptive, but also low resource systems.

In fact, making use of ambiguous context representations has already been explored in the context of the Natural Language Processing (NLP) field Semantic Parsing, i.e. the task of mapping natural language utterances to their corresponding formal meaning representations. Traditionally, data-driven approaches to semantic parser induction have been investigated in a supervised setting, i.e. these parsers were learned from examples consisting of utterances annotated with their correct meanings (e.g. Wong & Mooney, 2006; Zettlemoyer & Collins, 2007). Because such annotations are time-consuming and costly to produce, research has also focused on learning parsers using ambiguous context representations instead of annotations (e.g. Chen et al., 2010; Börschinger et al., 2011; Chen & Mooney, 2008) as a step towards building machines which can learn language – analogous to children – through exposure to language in some environment (Chen & Mooney, 2008). These parsers were, however, trained on written text.

With respect to application in SLU, in this thesis we present an approach which makes the semantic parsing task based on ambiguous context information applicable to speech data, instead of written text, without assuming any predefined linguistic resources other than a task-independent phoneme recognizer. Thus, contrasting with previous approaches to semantic parsing, we also address lexical acquisition. That is, since words are not given, the learning scenario is extended in that lexical units must be segmented out of a continuous stream of phonemes. Compared to applying a word-based speech recognizer, applying a phoneme recognizer makes it easy to adapt the system to novel tasks and supports the acquisition of a potentially unrestricted vocabulary. The learned parser is represented in the form of a lexicon and an inventory of syntactic constructions and is applicable to spoken utterances. Learning a semantic parser in this setting is much more challenging compared to learning from text due to recognition errors and different pronunciations of the same written word and due to the additional segmentation task. Thus, in the case of spoken utterances we do not focus on modeling child language acquisition, but explore how learning mechanisms introduced within the framework of our computational model can be extended and applied to tackle the increased complexity of the learning setting with respect to performance on a SLU task. In particular, we will not explore online learning. Instead, we assume that a system has the capability to log observed utterances, for instance, to update its linguistic knowledge at certain time intervals by applying the proposed approach.

1.3. Contributions

In the first part of this thesis, we present a usage-based computational model for the early acquisition of verb-specific constructions under referential uncertainty, including a mapping between lexical units and their corresponding meanings. In doing so, we formalize ideas proposed within usage-based approaches to language acquisition and construction grammar and combine them with a cross-situational learning mechanism. In contrast to previous computational models exploring cross-situational learning, we investigate how the same cross-situational learning mechanism can be applied consistently to establish form-meaning pairings at different levels. While the principles of item-based generalization and cross-situational learning have been discussed extensively in the literature, we believe that we present the first comprehensive computational model that combines cross-situational learning beyond word-referent mappings with a formalization of a generalization mechanisms based on an item-based induction of slots in order to learn a lexicon and syntactic constructions in an online fashion. In the design of the model, we particularly address the following research questions concerning usage-based approaches to child language acquisition:

1. How can linguistic knowledge be represented in the form of an interrelated network of constructions, and what mechanisms enable their retrieval?
2. How exactly do the underlying learning mechanisms operate, i.e. how can utterances be generalized in an item-based fashion by inducing slots?
3. How exactly can rule-based learning mechanisms be combined with statistical learning mechanisms?
4. Can the same cross-situational learning mechanism be applied beyond establishing simple word-object mappings?
5. How do more complex linguistic structures emerge from simpler ones?

In the second part of this thesis, we present an extension of our computational model, which allows it to learn verb-general constructions by exploiting the same learning mechanisms used for learning verb-specific constructions. We show how the model can simulate children's behavior observed in psycholinguistic studies on the acquisition of verbs and verb-general constructions, thus providing one possible, formal explanation for the observed behavior. In particular, we address the following research questions:

1. How do verb-general constructions emerge and how are they represented?
2. How can these constructions guide attention to establish verb entries based on syntactic information alone?
3. How can information about possible referents and co-occurrence statistics be stored with verb entries?
4. How can this information be updated incrementally over time, thus allowing for learning of verb meanings across situations?

In the third part of this thesis, we explore how a semantic parser can be learned and applied to spoken utterances, rather than written text, without assuming any predefined linguistic resources other than a task-independent phoneme recognizer. While learning linguistic structures of rather low complexity from speech without word transcriptions has been addressed previously, e.g. learning (novel) words (e.g. Roy & Pentland, 2002; Taguchi et al., 2009; Yu et al., 2005) or semantically meaningful sequences (e.g. Gorin et al., 1999; Levit et al., 2002; Cerisara, 2009), we are not aware of other approaches to learning syntactic constructions using ambiguous non-linguistic contexts. During the design and evaluation of our system, we will especially focus on the following research questions:

1. Does a top-down step in which knowledge of previously acquired syntactic constructions is used to refine segmentations improve segmentation accuracy and/or language learning performance?

2. Does the proposed method yield state-of-the-art performance when applied to written text, as explored previously in NLP?

3. Can we expect similar performance by applying a phoneme-based speech recognizer compared to applying a word-based one?

4. Is our weakly supervised approach useful for inducing recognition grammars applicable as a language model for a speech recognizer, which have been typically created manually or induced in supervised settings?

5. Is it possible to determine (the boundaries of) semantically meaningful sequences accurately, and what is the effect of using several different sequences for the same written word for parsing instead of a single “best” one?

1.4. Outline

The remainder of this thesis is organized as follows. In the next chapter, we will present background information concerning existing techniques relevant to this thesis and discuss related work. More specifically, in this thesis we investigate learning from spoken utterances, thus yielding a spoken language understanding task. We will hence provide an overview of automatic speech recognition and spoken language understanding. Further, since we explore learning from (spoken) natural language utterances coupled with ambiguous context information, several learning tasks need to be addressed within our approach at the same time. In particular, these learning tasks are i) segmenting a continuous stream of discrete units, ii) detecting semantically meaningful sequences, iii) mapping words to meanings and iv) acquiring (semantic-)syntactic constructions. These learning tasks have been addressed previously, though often independently of each other, and we will describe existing approaches in the following chapter. Moreover, making use of context information with respect to learning from text has been explored previously in NLP and we will review relevant work in this area.

In Chapter 3, we present our computational model for early syntactic acquisition by focusing on the acquisition of verb-specific syntactic constructions. We will first describe several ideas proposed within usage-based approaches to language acquisition and construction grammar which are relevant for computational modeling. Subsequently, we will present the design of the model. We will then present experimental results concerning the model's learning capabilities, and afterwards discuss the model with respect to psycholinguistic theories and findings as well as its limitations and possible extensions.

In Chapter 4, we will present an extension of our computational model, which learns verb-general constructions. We will then present empirical results by replicating findings from psycholinguistic studies with children, and subsequently discussing the results' implications for learning mechanisms which may be at play.

In Chapter 5, we will focus on learning from spoken utterances without word transcriptions. We will first present our system for learning lexical units and syntactic constructions. We will then present several experiments. For instance, we will compare the performance of our system, which works with phoneme transcriptions, to the expected performance of systems which work with word transcriptions made by a speech recognizer. Moreover, we will investigate the role of syntactic information in segmentation.

Finally, in Chapter 6 we will summarize the research presented in this thesis.

Chapter 2

Background

In this chapter, we provide a brief overview of existing techniques relevant to this thesis and discuss related work. As we focus on speech data in Chapter 5 of this thesis, we briefly discuss the main paradigms for Spoken Language Understanding (SLU). Spoken language understanding systems are typically based on automatic speech recognizers. Thus, in the following we will first present a brief overview about existing techniques in Automatic Speech Recognition (Section 2.1) and Spoken Language Understanding (Section 2.2), focusing on aspects relevant to this thesis.

In this thesis, we address the acquisition of syntactic constructions from examples of (spoken) natural language utterances coupled with ambiguous context information. In general, learning language from a speech signal mainly comprises acoustic, lexical and syntactic acquisition, where lexical acquisition may comprise segmentation of a continuous stream, detecting meaningful sequences and/or the acquisition of word-to-meaning mappings. Further, learning may be addressed with or without establishing meanings for learned structures. In cases where semantic acquisition is addressed, current approaches typically explore learning using concurrent non-linguistic context information, e.g. describing the visual context a learner observes. Most of these learning tasks have been addressed with respect to both modeling child language acquisition and application in (spoken) natural language processing systems and/or robotics. For instance, with respect to automatic speech recognition, where systems typically rely on linguistic knowledge provided by an expert, such as transcriptions of speech, it is of interest how the required linguistic knowledge can be learned automatically from speech. While in applied settings, the main focus for evaluating such learning algorithms lies on performance, in case of cognitive modeling, the cognitive plausibility of the developed model becomes an (additional) important criterion.

In this thesis we focus on learning from speech using a subword speech recognizer, in particular, a phoneme recognizer. Thus, we will not address phonetic acquisition. For a recent review of phonetic and lexical acquisition from speech with respect to modeling child language acquisition, please see Räsänen (2012). A summary of a recent workshop focusing on phonetic and lexical discovery from speech with respect to application in automatic speech recognition is provided by Jansen et al. (2013).

In the following, we will first describe approaches to lexical acquisition which attempt to segment a continuous stream of discrete units into word-like units with the goal of determining all word boundaries (Section 2.3). Subsequently, we will present approaches aiming to detect (semantically meaningful) linguistic structures in speech transcribed by a subword speech recognizer (Section 2.4). We will then discuss approaches to word-to-meaning mapping (Section 2.5) as well as syntactic acquisition, focusing on unsupervised learning (Section 2.6) and on determining a mapping from words to the corresponding semantics (Section 2.7); such approaches learn from sequences of words, and thus exclude the learning tasks described above. Finally, we will give an overview of approaches in NLP which attempt to make use of non-linguistic context information for computational language learning (Section 2.8).

2.1. Automatic Speech Recognition

Automatic speech recognizers (ASR) (Schukat-Talamazzini, 1995) are applied to transcribe input speech, i.e. an acoustic signal, into symbolic form reflecting the signal's content. In most cases the output corresponds to a sequence of words. A typical ASR is based on a pronunciation lexicon, acoustic models and a language model. The lexicon comprises the units, usually words, which the speech recognizer can recognize. Further, one or more pronunciations are associated with each lexical entry, e.g. represented as sequences of phonemes.

A first step for transforming an acoustic signal into symbolic form is the extraction of feature vectors, for example Mel Frequency Cepstral Coefficients (MFCCs), from the continuous signal. Starting from the feature vectors, the goal is then to infer the spoken sequence of words (or other units). In statistical ASR this corresponds to finding the most likely sequence of words given a sequence of observations:

$$\hat{W} = \arg\max_{W} P(W|O) = \arg\max_{W} \frac{P(O|W)P(W)}{P(O)} \tag{2.1}$$

where $P(W|O)$ denotes the probability of the occurrence of a sequence of words W given feature vectors O. Since W is not dependent on $P(O)$ the formula resolves to

$$\hat{W} = \arg\max_{W} P(W|O) = \arg\max_{W} \underbrace{P(O|W)}_{\text{acoustic modeling}} * \underbrace{P(W)}_{\text{language modeling}} \tag{2.2}$$

and thus comprises two problems: acoustic modeling and language modeling. Model parameters can be estimated automatically from suitable (labeled) training data. Acoustic modeling requires spoken language corpora, while language modeling requires (often large amounts) of textual data, preferably transcriptions of speech. $P(O|W)$ represents the acoustic similarity of the word sequence and the signal. A common approach for estimating probabilities for underlying units of the word sequence such as (tri)phones are Hidden Markov Models (HMM) (Rabiner, 1989). However, in this thesis we are not concerned with acoustic modeling.

With respect to language modeling, mainly n-gram models are applied (Lamel & Gauvain, 2003). An n-gram model (Schukat-Talamazzini, 1995) estimates the occurrence probability for each word – or other unit – in the ASR's vocabulary based on the $n-1$ preceding words. Hence, the probability for a sequence of words $W = w_1 \dots w_n$ is given by

$$P(W) = P(w_1, w_2, \dots, w_n) = P(w_1)P(w_2|w_1) \dots P(w_n|w_1, \dots, w_{n-1}). \tag{2.3}$$

Due to insufficient data, a word w_n cannot be predicted by taking an arbitrary number of preceding words into account. Applying the Markov Assumption, a word's history is thus estimated based on its local context which is given by n. For instance, with respect to a trigram model this yields

$$P(w_n|w_1 \dots w_{n-1}) \approx P(w_n|w_{n-2}, w_{n-1}). \tag{2.4}$$

Thus, in this case $P(W)$ is given by

$$P(W) = P(w_1, w_2, \dots, w_n) \approx P(w_1)P(w_2|w_1) \dots P(w_n|w_{n-2}, w_{n-1}). \tag{2.5}$$

N-gram models can be estimated automatically from large amounts of training data. For instance, the probability for a trigram can be estimated based on its frequency and the frequency of its bigram prefix as:

$$P(w_i|w_{i-2}, w_{i-1}) = \frac{freq(w_{i-2}, w_{i-1}, w_i)}{freq(w_{i-2}, w_{i-1})}, \tag{2.6}$$

where $freq(x)$ denotes the frequency of n-gram x in the training data.

However, in order to build a word-based language model (LM) that yields good recognition performance, the data used for model training must also match the data to be recognized with respect to topic/domain and language use, and thus the performance of word-based n-gram models and word-based automatic speech recognizers is – at least to a certain extent – topic/domain-dependent. For instance, in order to model phone conversations, using an LM trained on two million words of transcribed phone conversations is better suited than a language model trained on 140 million words of transcribed broadcast news (Rosenfeld, 2000). Moreover, by only taking into account the local context, that is, the immediately preceding words, n-gram models cannot capture long-range linguistic dependencies. More specifically, the identity of a word is not only dependent on its directly preceding words. For instance, with respect to a trigram model, after a sequence "a few", basically any noun or adjective may follow. Thus, all nouns and adjectives may only be assigned a low probability by the model.

Even when using a large corpus for model training, not all n-grams may appear. Based on the formula shown previously, these would receive a probability of 0 by the language model, hence making sentences incorporating such an n-gram statistically impossible. Since in theory each word sequence may appear, n-gram models are usually smoothed, i.e. a portion of the probability mass for observed occurrences is reserved for n-grams not seen during training. Several smoothing methods have been explored; for a comparison of common methods, see Chen & Goodman (1998).

Instead of applying stochastic models like n-gram models, speech recognition can also be performed by applying a speech recognition grammar as the language model. A speech recognition grammar may be defined, for instance, according to the W3C standard Speech Recognition Grammar Specification (SRGS) Version 1.0[1] (Hunt & McGlashan, 2004). While n-gram models are applied to estimate probabilities of word sequences, in such a grammar one can explicitly define which words and patterns a user may utter. Further, semantic information can be directly specified within the rules, for instance, according to the W3C standard Semantic Interpretation for Speech Recognition (SISR) Version 1.0[2] (Van Tichelen & Burke, 2007). Thus, when applied with an ASR, spoken utterances can be transformed into a corresponding semantic representation instead of a sequence of words. Fig. 1 shows an example for such a grammar with respect to application on a robot. The grammar matches utterances of the form "Go, fetch me X" where X refers to one of the spec-

[1] http://www.w3.org/TR/speech-grammar/

[2] http://www.w3.org/TR/semantic-interpretation/

ified drink or snack names, and the output is the semantic tag of the name matched in an input utterance. For instance, given an input utterance "Go, fetch me a piece of cake", the grammar would return the semantic tag *cake*.

Figure 1.: An example for an SRGS grammar including semantic information in SISR.

```
<grammar version="1.0" xmlns="http://www.w3.org/2001/06/grammar"
xml:lang="en-US" tag-format="semantics/1.0-literals" root="fetch">
   <rule id="fetch" scope="public">
      go fetch me
      <ruleref uri="#object"/>
   </rule>
   <rule id="object" scope="private">
      <one-of>
         <item><ruleref uri="#drink"/></item>
         <item><ruleref uri="#snack"/></item>
      </one-of>
   </rule>
   <rule id="drink" scope="private">
      <one-of>
         <item>coca cola<tag> coke </tag></item>
         <item>the coke<tag> coke </tag></item>
         <item>apple juice<tag> juice </tag></item>
      </one-of>
   </rule>
   <rule id="snack" scope="private">
      <one-of>
         <item>a sandwich<tag> sandwich </tag></item>
         <item>an apple<tag> apple </tag></item>
         <item>a piece of cake<tag> cake </tag></item>
      </one-of>
   </rule>
</grammar>
```

While n-gram models are typically estimated automatically from large amounts of suitable training data, semantic speech recognition grammars are typically created manually. Thus, even though the process of grammar creation requires no (large amounts of) suitable training data, it does require manual effort and often extensive knowledge of the underlying semantic domain and/or linguistic knowledge. Applying a manually created semantic grammar directly with a speech recognizer is the main approach to SLU in commercial systems which contrasts with SLU research, where mainly data-driven approaches are investigated (Wang et al., 2011).

A common measure to evaluate the performance of an ASR is the word error rate (WER), which is based on the number of words substituted, deleted and inserted by an ASR compared to a reference transcription. It is defined as follows (Lamel & Gauvain, 2003):

$$WER = \frac{\#\text{substitutions} + \#\text{deletions} + \#\text{ insertions}}{\#\text{ reference words}}. \tag{2.7}$$

2.2. Spoken Language Understanding

Spoken Language Understanding (Tur & Mori, 2011) systems aim to extract semantic information from speech utterances. In this thesis, we explore learning from and applying a semantic parser to spoken utterance, thus also investigating a SLU task. In general, several SLU tasks have been investigated, for instance, Spoken Question Answering (Rosset et al., 2011) or Speech Summarization (Liu & Hakkani-Tür, 2011). In the following, we will focus on Semantic Frame-based Spoken Language Understanding (Wang et al., 2011), since the task is similar to the one investigated in this thesis.

Semantic Frame-based Spoken Language Understanding is typically explored with respect to a restricted semantic domain which can be represented by means of a set of *semantic frames* (Fillmore, 1976). A semantic frame is a template-based representation which contains a number of slots where the type of slot defines the kind of elements the slot can be filled with. In general, in this thesis we represent semantic representations in line with approaches to semantic parsing in NLP, i.e. by means of predicate logic formulas, and our semantics are defined in a broader sense based on thematic relations such as $AGENT$ or $PATIENT$ rather via domain-specific slots. However, because the investigated formulas are rather simple, i.e. comprising a predicate along with a set of thematic relations or arguments each, the scenario is somewhat similar in that the predicate logic formulas might be converted into semantic frames where the predicate denotes the frame's name and the predicate's arguments denote the slots.

Several measures have been applied for the evaluation of frame-based SLU systems. Two commonly used measures are the Sentence Level Semantic Accuracy (SALSA) and the Slot Error Rate (SER, also referred to as Concept Error Rate (CER)) which are defined as follows (Wang et al., 2011):

$$SALSA = \frac{\#\text{ of sentences assigned the correct semantic representation}}{\#\text{ of sentences}} \tag{2.8}$$

and

$$SER = \frac{\text{\# of inserted/deleted/substituted slots}}{\text{\# of slots in the reference semantic representation}}. \tag{2.9}$$

In the previous section, we have already mentioned that SLU can be performed by creating a semantic speech recognition grammar, which can then be applied as the language model with an ASR, where the ASR can be applied to transform spoken utterances directly into their corresponding semantic representations. While this approach is typically chosen for building commercial applications (Wang et al., 2011), SLU research mostly explores cascading systems (Deoras et al., 2013). In this case, an ASR is applied – for instance, using an n-gram model as the language model – in order to transcribe a given speech utterance into a sequence of words. Subsequently, the resulting transcriptions are fed to an understanding component to transform them into their corresponding semantics, for instance, to detect slots. While this understanding task is related to the Natural Language Understanding task, which mainly focuses on written text, it raises further challenges. More specifically, systems need to be designed with respect to robustness concerning speech recognition errors and phenomena of spoken language such as disfluencies and extra-grammaticality (Wang et al., 2011).

Due to their robustness to noise, several approaches exploring probabilistic models and machine learning techniques have been proposed in SLU research for conceptual tagging. For instance, Conditional Random Fields (Lafferty et al., 2001) have been applied (e.g. Wang & Acero, 2006a; Dinarelli et al., 2012), and He & Young (2005) present an approach based on Hidden Markos Models. Such models are typically based on rather local features, e.g. on n-grams. While these methods are somewhat robust to noise, typically they cannot capture long-range linguistic dependencies within an utterance. Moreover, evaluations have shown that even when applying machine learning techniques or probabilistic models, semantic parsing of ASR transcriptions yields many more errors compared to parsing of correct transcriptions (De Mori, 2011).

Further, approaches to Spoken Language Understanding based on (semantic) grammars have been proposed. In general, knowledge-based and data-driven approaches as well as combinations thereof have been explored. In Section 2.1 we have already mentioned that spoken language understanding in knowledge-based approaches used for commercial systems is typically performed using manually created (semantic) speech recognition grammars. Due to the manual effort, and often also linguistic knowledge needed for grammar creation, some work in SLU research has focused on data-driven approaches for the induction of grammars and on combining knowledge-based with data-driven approaches. In doing so, data-driven approaches have typically been explored in a supervised setting, e.g. based on word-level semantic annotations, thus still requiring a large manual effort (e.g. Wang & Acero, 2006b, 2005, 2003). For instance, Wang & Acero (2006b) propose a supervised approach for the automatic induction of grammars which can be represented according to the SRGS (cf. Section 2.1) and are applicable for both speech recognition and understanding. In the training data, slots of semantic frames must be labeled. The authors present a combination of knowledge-based and data-driven approaches in that they also provide a tool for annotating utterances which requires little linguistic knowledge. This tool is also able to suggest annotations based on its current knowledge. These can then be modified by a user if necessary (Wang & Acero, 2006b, 2005). The tool uses a HMM/CFG Composite Model (Wang & Acero, 2003) for automatic grammar induction starting from labeled data. In particular, their grammar is based on template rules modeled by an HMM. States corresponding to slots are modeled as sequences of the form *preamble-filler-postamble*, where fillers are modeled by PCFGs and pre- and postambles are modeled by n-gram models. These are learned from the labeled data.

In addition, several approaches have addressed the (semi-)automatic induction of grammars from unlabeled corpora, particularly aiming at the automatic induction of semantic classes (e.g. Wong & Meng, 2001; Siu & Meng, 1999; Meng & Siu, 2002). Such approaches typically explore the linguistic context to infer regularities that imply semantic similarity. For example, Meng & Siu (2002) present an approach based on spatial and temporal clustering, where spatial clustering is based on the Kullback-Leibler distance (Kullback, 1959), and aims to group words having similar left and right contexts since these typically have similar semantics. They perform temporal clustering based on Mutual Information (Cover & Thomas, 1991), aiming to detect phrasal structures. Resulting grammars were then manually postprocessed, thus yielding a semi-supervised approach. For example, sets of terminals were completed and some where replaced by semantically meaningful symbols. In general, approaches which induce syntactic structures and semantic classes automatically from unlabeled corpora do not infer their corresponding meanings, thus needing manual postprocessing in order to be applicable for SLU. In sum, the induction of semantic grammars with respect to application for SLU has been investigated with respect to both (semi-)automatic approaches and automatic approaches explored in a supervised setting, both of which require manual effort.

In general, while data-driven approaches to SLU have been explored for showcase problems, including grammar-based ones, they are dependent on the availability of large amounts of labeled training data and are hence rather impractical with respect to real-world applications (Wang et al., 2011). To reduce annotation costs, recent work has, for instance, focused on exploring supervised learning in combination with active learning (Wu et al., 2010) or, as mentioned previously, on providing annotation tools (Wang & Acero, 2006b, 2005). Further, gaining additional training data, e.g. from the Web using queries generated from a (small) existing grammar (Klasinas et al., 2013), has been explored. Notice though that all of these approaches still require some manual effort.

In SLU research performances of ASR and parsing components are often optimized independently of each other, in particular with respect to low error rates such as WER and SER. However, research has shown that a lower error rate of an ASR can in fact yield worse understanding performance (Wang et al., 2003; Bayer & Riccardi, 2012). Moreover, joint approaches to recognition and understanding can yield improved parsing performance (Wang & Acero, 2006b; Deoras et al., 2013), even though the WER may be higher (Bayer & Riccardi, 2012; Wang & Acero, 2006b). In particular, applying a SLU model as the language model for the ASR is beneficial in SLU – notice, however, that not all SLU models can be used with an ASR (Wang et al., 2011). Specifically, Wang & Acero (2006b) have shown that applying the same grammar for speech recognition and understanding can yield improved understanding performance compared to applying a standard n-gram model with the ASR, since dependencies between acoustics and semantics can be captured.

SLU systems typically apply word-based recognizers for speech recognition. However, recent research has also investigated performing SLU on phoneme sequences, indicating that with respect to conceptual tagging, SLU performed on phoneme sequences can yield performance similar, or even slightly higher, compared to working with words (Svec et al., 2013). By contrast, the focus of this thesis is not on spotting sequences referring to concepts in speech, but on learning more complex linguistic structures capturing long-range linguistic dependencies, i.e. syntactic constructions. However, – in line with semantic grammar-based approaches to SLU – these can be applied for spoken language understanding tasks as well. Further, we investigate a weakly supervised learning setting rather than a fully supervised one.

2.3. Segmentation of a continuous sequence of discrete units into word-like units

Several models and algorithms have addressed the segmentation of a continuous stream of discrete units, e.g. characters or phonemes, into word-like units with the goal of determining a complete segmentation of the input stream, i.e. to detect all word boundaries (e.g. Cohen & Adams, 2001; Hewlett & Cohen, 2011; Brent, 1999; Pearl et al., 2011; Goldwater et al., 2009; Johnson & Goldwater, 2009). In doing so, segmentation may be performed either on a single input stream or with respect to a number of individual sentences, where the latter setting is semi-supervised in that some information concerning boundaries is provided.

One approach to the segmentation problem is Voting Experts (VE) (Cohen & Adams, 2001; Cohen et al., 2006). Roughly, the intuition behind the algorithm is that word-like units are i) rather frequent, i.e. the entropy or unpredictability of comprised elements is rather low, and ii) the following character is rather uncertain, i.e. the entropy or unpredictability is rather high. In particular, in the VE algorithm, two so-called experts vote for segmentation points: one votes for segmentation points after chunks/sequences having low internal entropy and one votes for segmentation points after sequences/chunks having high boundary entropy. Boundaries are then introduced at positions for which the number of votes reaches a local maximum. The algorithm has been extended several times. One such extension is Bootstrap Voting Experts (BVE) (Hewlett & Cohen, 2009, 2011) which refines segmentations iteratively by reusing precise segmentations.

Particularly with respect to modeling human language acquisition, research has mainly focused on utilizing statistics concerning syllable and phoneme regularities (Pearl et al., 2011), e.g. by applying Bayesian methods. Further, making use of knowledge about (linguistic) context information in statistical segmentation models has been investigated. In particular, Goldwater et al. (2009) explore unsupervised segmentation in a Bayesian framework. They compare computational models, assuming that words are either independent units or useful for predicting other units. Thus, word sequences are assumed to be generated by a unigram or bigram language model, respectively. They evaluate their models on grapheme-to-phoneme converted transcriptions[3] of child-directed speech, and report that the bigram model produced

[3] Grapheme-to-phoneme converted transcriptions/data refers to transcriptions/data where one has transformed each word in a text automatically into a corresponding phoneme sequence. This might be done by using a lexicon comprising words coupled with their corresponding transcriptions or by applying a tool for automatic conversion; such tools are, for instance, available with text-to-speech systems.

more accurate segmentations compared to the unigram model. Further, Johnson & Goldwater (2009) also investigate the utility of higher order features, i.e. (linguistic) context information, by applying Adaptor Grammars (Johnson et al., 2007). In their case, the underlying assumption is that a word sequence is generated by a sequence of collocations; these may in turn comprise one or more words. Like Goldwater et al. (2009), they evaluate performance on grapheme-to-phoneme converted transcriptions and report an improvement in accuracy over applying a unigram model.

In general, this segmentation task has been mainly investigated in NLP and/or with respect to cognitive modeling, and typically non-linguistic contextual information has been neglected, though one may assume that meanings can be attached afterwards. However, work has also shown that performing segmentation and word meaning acquisition jointly can improve segmentation performance. In particular, Jones et al. (2010) present a Bayesian model for segmentation/lexical acquisition. They tested their model on grapheme-to-phoneme converted orthographic transcriptions annotated with some objects and found that, by making use of contextual information, segmentation accuracy can be improved, at least for semantically meaningful sequences, i.e. those referring to presented objects.

To date, in NLP and/or with respect to cognitive modeling the segmentation task has mainly been evaluated with respect to segmenting either written text or transcriptions of speech for which words were converted into sequences of phonemes by applying grapheme-to-phoneme conversion. Typically, for evaluating such algorithms all word boundaries are removed for learning a segmentation. Subsequently, it is inspected how accurately the algorithm can recreate boundaries. However, when working with text or grapheme-to-phoneme converted transcriptions, each word is represented by a single sequence of characters or phonemes, which does not reflect spoken language where the same written word typically has different pronunciations. Notably, segmentation has recently also been explored with respect to cognitive modeling by taking into account phonetic variability, i.e. different pronunciations of the same word (Elsner et al., 2013).

Generally, algorithms proposed in NLP and cognitive modeling may also be applied to segment the output of an automatic speech recognizer, in particular, a phoneme recognizer. However, in this case performance can degrade rapidly, since recognizers introduce noise in the form of recognition errors, thus resulting in a large number of different sequences for each written word (Jansen et al., 2013). Specifically, recent work (Jansen et al., 2013) has investigated the performance of two of the algorithms described previously, i.e. those of Johnson & Goldwater (2009) and Goldwater et al. (2009), when applied to automatic transcriptions of speech. Besides a drop in segmentation accuracy, with increasing noise the models exploring a bigram model or Adaptor Grammar presented by Goldwater et al. (2009) and Johnson & Goldwater (2009) performed worse compared to the unigram model. This is the case because these models are based on estimating recurring sequences, and by not taking sequence variation into account, detection of lexical units and thus the ability to make use of them concerning contextual information degrades rapidly (Jansen et al., 2013).

Recently, lexical segmentation and learning from continuous speech in an unsupervised fashion has also gained increasing interest due to applications such as, for instance, bootstrapping resources, e.g. (pronunciation) lexica, for under-resourced languages; we will discuss learning from speech in the following section.

While in this thesis we are not concerned with unsupervised segmentation and focus on determining semantically meaningful sequences rather than a complete segmentation into word-like units, we apply the BVE algorithm during our experiments. In our case, the segmentation task is semi-supervised in that utterance boundaries are given (and only these utterances must be segmented further), while BVE has been typically applied for segmenting a single input stream.

2.4. Learning linguistic structures from speech without word transcriptions

To date, the acquisition of linguistic structures from speech without word transcriptions has focused on detecting units of rather low complexity, e.g. on learning (novel) words or semantically meaningful sequences. The task has been explored both by making use of a subword speech recognizer (e.g. Gorin et al., 1999; Levit et al., 2002; Cerisara, 2009; Taguchi et al., 2009; Yu et al., 2005) and by working with feature vectors extracted from the signal, such as MFCCs (e.g. Brandl, 2009; McInnes & Goldwater, 2011; Räsänen et al., 2009; Räsänen, 2011; Muscariello et al., 2012). Since in this thesis we work with the output of a phoneme recognizer, in the following we will focus our discussion on approaches working with a subword recognizer.

As mentioned in the previous section, learning from a phoneme stream may comprise determining a complete segmentation into word-like units, and recent work has addressed this segmentation task with respect to speech. Specifically, lexical and language model acquisition starting from phoneme lattices generated by an automatic speech recognizer have been addressed (e.g. Neubig et al., 2010, 2012; Heymann et al., 2014). For instance, Neubig et al. (2012) present a corresponding

Bayesian approach that simultaneously learns lexical units and an n-gram model starting from phoneme lattices. They present results indicating that this can reduce the phoneme error rate of an ASR.

Further work has focused on detecting semantically meaningful sequences, so-called *acoustic morphemes* (Gorin et al., 1999), rather than determining all word boundaries. Gorin et al. (1999) and Levit et al. (2002) extract semantically meaningful sequences from the output of a speech recognizer. A sequence is rated based on the mutual information of its components and its salience for an understanding task. The latter is in turn based on the maximum of the a posteriori distribution of a semantic class given the sequence. Cerisara (2009) explores the unsupervised creation of lexica comprising acoustic morphemes by detecting sequences based on their recurrence. The underlying assumption by the author is that sequences appearing with a certain frequency correspond to coherent linguistic units. Due to recognition errors and different pronunciations of the same word, he applies approximate string matching based on an edit distance to detect recurring sequences.

In contrast to the described approaches, we are concerned with learning from speech coupled with concurrent visual information. Making use of concurrent visual information for learning novel words has, for instance, been explored in the field of robotics. In particular, Taguchi et al. (2009) explore how a novel keyword can be detected in a phonemically transcribed utterance. Learning is investigated in a tutoring scenario where visual information is provided by showing a single object. Since the authors assume that the robot can identify the object correctly, the setting corresponds to providing visual information in symbolic form. Lexical learning is based on deriving a statistical model capturing the joint probability of an object and an utterance where the statistical model incorporates a phoneme acoustic model, a word meaning model as well as a word-based bigram model (Taguchi et al., 2009).

Lexical acquisition from speech using a phoneme recognizer and concurrent visual information has also been explored with respect to modeling child language acquisition. In particular, Yu et al. (2005) and Yu & Ballard (2002) present a computational model which learns words and their meanings. Their model also shows how social cues, in particular information concerning eye-gaze, can be used to facilitate the establishment of correspondences between words and their corresponding visually grounded meanings.

In contrast to the approaches to learning from speech described in this section, we focus not only on the acquisition of lexical units or acoustic morphemes, but also on learning syntactic constructions. While we also make use of concurrent visual information for lexical acquisition, we additionally incorporate previously acquired syntactic knowledge for bootstrapping a parser.

2.5. Word-to-meaning mapping

Some previous research has addressed mapping words to their corresponding semantic referents given contextual information in symbolic form, and some has exploited cross-situational statistics for this purpose.

An early, formal model which learns word-to-meaning mappings given ambiguous contexts has been proposed by Siskind (1996). Besides relying on cross-situational statistics, the model also exploits several inference rules. For example, partial knowledge acquired for meanings of words in a given utterance can be utilized to infer meanings of different words in the same utterance. The author performs several computational simulations, showing that the model can induce word meanings given artificially generated data.

Horst et al. (2006) and McMurray et al. (2012) apply a Normalized Recurrence Network to capture correspondences between words and referents. The network comprises two input layers comprising a set of predefined neurons each: one for auditory and one for visual input. In addition, Horst et al. (2006) explore possible relations between word learning and fast mapping; fast mapping refers to children's ability to quickly set up an initial word-to-meaning mapping when observing a novel word in the presence of referential uncertainty (Carey & Bartlett, 1978). The authors perform experiments using a small set of words and objects. They report that on being presented with a novel word along with a novel object and two known ones, the model can – in line with children (Horst & Samuelson, 2008) – correctly map the novel word onto the novel object. The authors further report that the model cannot retain these newly established connections, suggesting that fast mapping and word learning are related but different processes (in the proposed model) (Horst et al., 2006).

Frank et al. (2007) present a Bayesian model for cross-situational word learning and test their model on data extracted from child-directed speech from the CHILDES database (MacWhinney, 2000), coupled with concurrent visual information. In particular, they annotated utterances with objects which were visible to the child. Moreover, they annotated social cues, such as the mother's hands. They show that by integrating social cues into their model for cross-situational learning, word acquisition performance is improved. Further, besides acquiring word meanings, their model is also able to mimic the fast mapping ability observed in children. Similarly, Yu & Ballard (2007) also integrate social cues, e.g. joint attention, into a model for cross-situational learning and report that this improves performance in establishing correspondences between words and their meanings.

Fazly et al. (2010) introduce a probabilistic model for incremental/online word learning under referential uncertainty, also building on the idea of cross-situational learning. They test their model on child-directed utterances from the CHILDES database (MacWhinney, 2000) coupled with corresponding scene representations, i.e. sets of semantic features corresponding to the words in the utterance. In order to model referential uncertainty, they also add distractor semantic features, i.e. features taken from the scene representation of the following utterance. They show that their model is able to successfully acquire word meaning under referential uncertainty. The model has also been tested on fast mapping experiments and has successfully simulated referent selection (Alishahi et al., 2008). Further, the model has been extended to include syntactic categories (Alishahi & Fazly, 2010). In particular, assuming that the child has already acquired a number of lexical categories grouping sets of word forms, Alishahi & Fazly (2010) integrate a categorization function into their existing model, which can determine the category of any given word. They present empirical results indicating that including lexical categories can improve word learning performance. In further work, Alishahi & Chrupala (2012) show how these categories can be learned automatically.

Kachergis et al. (2012a) present an associative model for cross-situational learning which incorporates competing familiarity and uncertainty biases. In further work, they compared this model, which maintains approximately all co-occurrence statistics, to a model maintaining a single "best" hypothesis for observed words (Kachergis et al., 2012b). They fit the two models to data obtained from a cross-situational learning task with human subjects and report that the human learning curves are better fitted by the associative model.

In this thesis, we also explore word-to-meaning mapping, and we explore a cross-situational learning mechanism for this purpose. However, our main focus is on the extension to also acquire syntactic constructions, which is not addressed by the described approaches.

2.6. Syntactic acquisition from raw text

Several algorithms (e.g. Wong & Meng, 2001; Siu & Meng, 1999; Meng & Siu, 2002; Solan et al., 2005; Zaanen & Adriaans, 2001b,a; Elman, 1990) and computational models for child language acquisition (e.g. Bannard et al., 2009; Waterfall et al., 2010; Bod, 2009) have explored the unsupervised induction of syntactic structures from text. Since only raw text is given, syntactic patterns are typically detected by exploring regularities for appearing words. While semantic acquisition is not addressed, it is possible to (manually) attach semantic information to induced patterns. As already described in Section 2.2, this has been explored with respect to creating semantic grammars for spoken language understanding systems (e.g. Wong & Meng, 2001; Siu & Meng, 1999; Meng & Siu, 2002).

Further algorithms for the automatic induction of syntactic structures from raw text include ADIOS (Solan et al., 2005), ABL (Zaanen & Adriaans, 2001b) and EMILE (Zaanen & Adriaans, 2001a). ADIOS (Automatic Distillation Of Structure) works on corpora containing sentences over a lexicon L of smaller units, e.g. words. Based on this input, it induces syntactic patterns where patterns may have different levels of generalization and may contain equivalence classes or other patterns. These patterns are represented as a directed graph. The graph's vertices are initially all lexicon entries, augmented by two further vertices *begin* and *end*, representing the start and end of utterances, respectively. While training proceeds, new nodes are inserted which may represent equivalence classes or (sub)patterns. Each (induced) syntactic pattern is then represented as an indexed path through the graph. The idea for establishing equivalence classes in the ADIOS algorithm is to iteratively define a slot at all positions (except for the first and the last) j of a context window sliding over a pattern. Subsequently, it searches for sequences which have identical prefixes (ending at position $j-1$) and identical suffixes (starting at position $j+1$). All patterns that appear between the suffix and the prefix in the resulting patterns might constitute an equivalence class if some criterion – MEX – is satisfied. Given the graph, induced patterns and equivalence classes, one can test whether a given utterance is captured by the grammar induced by ADIOS by searching for a matching path in the graph (Solan et al., 2005).

Aiming to model child language acquisition, Waterfall et al. (2010) present a learning strategy which, according to the authors, is much simpler than ADIOS. In their algorithm, distributional statistics for (sequences of) words are determined based on their local context, i.e. the surrounding words. The authors report that in their experiments generative grammars could be induced in an unsupervised fashion based on data taken from the CHILDES database (MacWhinney, 2000).

Bannard et al. (2009) present a computational model based on techniques from unsupervised grammar induction and use it to extract probabilistic, item-based grammars from transcribed speech of children. These grammars are in turn used to parse utterances later produced by the children, and are compared to more abstract grammars. According to the authors, the findings provide support for usage-based

approaches to language acquisition (Bannard et al., 2009).
Bod (2009) also proposes a model for language acquisition working with raw sequences of words. Given such input data, the model acquires a grammar represented in the form of phrase-structure trees. The authors present results with respect to syntactic parsing of child-directed speech and show that their model can also capture a move from item-based to abstract linguistic knowledge.
Contrasting with the described approaches, in this thesis we explore syntactic and semantic acquisition jointly. More specifically, in our case the induction of syntactic constructions is driven by concurrent semantic information, and we focus not only on syntactic acquisition but also on determining a corresponding semantic mapping. However, an unsupervised algorithm for the induction of syntactic patterns has inspired the computational model proposed in this thesis, i.e. ADIOS. Moreover, we make use of ADIOS in experiments concerning the induction of speech recognition grammars as an unsupervised baseline.

2.7. Models for the acquisition of syntactic constructions

Different models have been proposed concerning the acquisition of constructions (e.g. Alishahi & Stevenson, 2008; Dominey & Boucher, 2005; Chang & Maia, 2001; Hinaut & Dominey, 2013).
Chang & Maia (2001) explore the induction of verb-specific constructions. The authors present an approach based on Bayesian model merging, where more complex grammatical structures are induced based on previously acquired simple lexical mappings. These mappings are given, facilitating the task of inducing verb-specific constructions by reducing the referential uncertainty.
Further research has addressed the acquisition of verb argument structure constructions (e.g. Alishahi & Stevenson, 2008; Parisien & Stevenson, 2010; Perfors et al., 2010). These approaches typically attempt to cluster individual verb uses into argument structure constructions, for instance, based on syntactic features. However, unlike the model presented in this thesis, they usually do not address lexical acquisition and often make further simplifying assumptions concerning what has been learned previously (by the child). For instance, Alishahi & Stevenson (2008) introduce an incremental/online Bayesian model exploring the representation and acquisition of abstract verb argument structure constructions (modeled in form of probabilistic correspondences between syntactic and semantic features) by assuming that the relevant words have already been acquired. Parisien & Stevenson (2010)

present a hierarchical Bayesian model which clusters specific verb uses based on syntactic features, assuming that the child is already able to detect syntactic arguments in observed utterances. In particular, the input to their model is generated by a dependency parser; lexical acquisition is not addressed. Perfors et al. (2010) also address the acquisition of verb argument constructions based on a hierarchical Bayesian framework but rely on built-in knowledge about constructions.

Yu (2006) presents a model which learns both word meanings and syntax-semantics mappings in an offline fashion to investigate the role of syntactic information on word learning. For instance, if a word is grouped into a syntactic class referring to object names, then it likely also refers to an object. This kind of information is explored in the model. Specifically, in the model word-to-meaning mapping is based on co-occurrence frequencies, and syntactic structure acquisition is performed using the ADIOS algorithm (Solan et al., 2005, cf. Section 2.6). Results from the two learning processes are then integrated, i.e. words are grouped based on induced syntactic roles, and meanings for syntactic categories are inferred based on the word learning process. The author tested the model on narrations of picture books made by parents to their children. In case of the visual input, a list of objects was presented, i.e. the objects the parent was attending to when producing an utterance. By comparing word learning with and without making use of syntactic cues, the author found that the integration of syntactic cues yields improved word learning performance (Yu, 2006). Maurits et al. (2009) also explore the utility of performing learning tasks jointly. In particular, they investigate the acquisition of both word meanings and word order, i.e. verb-argument structure orderings, given examples of utterances coupled with their corresponding meaning representations. They explore learning in a very simple setting, i.e. in a simulated world, and present results suggesting that performing both learning tasks jointly facilitates learning.

Dominey & Boucher (2005) propose a system which learns a small lexicon and an inventory of grammatical constructions based on narrated video. The form of constructions is represented as a sequence of closed-class words and slots corresponding to open-class words, which in turn map to semantic referents in the associated meaning, e.g. (agent *verb*-ed object to recipient, *verb(agent, object, recipient)*). Such pairings are derived from (sentence, meaning) pairings and directly stored in the inventory. In more recent work (Hinaut & Dominey, 2013), they also explore the acquisition of grammatical constructions using a Recurrent Neural Network. This network is able to learn in an online fashion, and the authors report that their model possesses the capability of generalizing to novel constructions (Hinaut & Dominey, 2013). Further, they also transferred their system to a robotic platform, allowing the robot to learn grammatical constructions through human tutoring (Hinaut et al., 2014). They also explore learning from speech. However, in line with traditional spoken language understanding systems, they make use of a word-based speech recognizer for this purpose. Moreover, they assume that the meaning corresponding to an utterance is available to the learner, while our learning system is able to handle ambiguous input, i.e. sentences coupled with several competing meanings.

Taken together, approaches to the acquisition of syntactic constructions often assume that lexical mappings have been acquired previously and/or do not address learning from ambiguous contexts. However, learning words and grammatical constructions starting from ambiguous contexts has also been explored (Kwiatkowski et al., 2012; Beekhuizen et al., 2014).

Kwiatkowski et al. (2012) propose a probabilistic model for syntactic and semantic acquisition. Specifically, they utilize the Combinatory Categorial Grammar (CCG) framework (Steedman, 2000) to learn both a lexicon and a parsing model in an online fashion. Given an input example, i.e. an utterance and an (ambiguous) scene representation, their approach roughly works by extracting all possible parses and updating the parsing model accordingly. Since this is somewhat memory-intense, the authors acknowledge that it is unlikely that children indeed generate all parses consistent with an input example, at least once they have already acquired some of the language. They evaluate their approach on child-directed data taken from the Eve corpus (Brown, 1973) coupled with automatically created potential meaning representations (i.e. one actually corresponding to the utterance and additional distractor meanings). They show that their model can be applied to parse unseen utterances, and that it can mimic fast mapping. However, while the learning task is similar, our approach contrasts with their probabilistic approach by explicitly formulating ideas proposed within usage-based approaches and construction grammar. In particular, our model is represented as a network where syntactic constructions are learned by gradually inducing slots. This allows our model to also pose fewer constraints on memory, as it does not create a large number of possible parses but learns a compact model of the input data, where the number of induced rules is much smaller than the number of observed examples (cf. Section 3.4.4).

Beekhuizen et al. (2014) explore a usage-based model which starts learning from minimal linguistic representations. In particular, the authors propose several learning mechanisms which can be applied to incrementally acquire constructions based on previous parses. That is, given an utterance and ambiguous scene representation, the model attempts to determine the correct meaning along with the most probable parse, and then uses this information to update its current inventory of constructions. Their work is similar with respect to computational modeling to the work presented in this thesis in that they also explore a usage-based model. However, the concrete learning mechanisms differ. In particular, while they also incorporate a simple cross-situational learning mechanism, this mechanism is not applied at different levels of complexity, and the proposed generalization mechanisms differ. Further, just like Kwiatkowski et al. (2012), they test their model on data artificially generated based on (distributional information obtained via) child-directed data, and, contrasting with the work presented in this thesis, the correct meaning is always among the competing meanings for a given utterance, which does not correspond to natural settings.

The word-based scenario investigated in this thesis has also been investigated in NLP with respect to inducing semantic parsers, and, in general, the utility of context information for situated language learning has been explored. We will give an overview of such approaches in the following section.

2.8. Context information in computational language learning

In NLP, data-driven approaches to semantic parsing have traditionally been investigated in a supervised setting, i.e. by learning from examples consisting of utterances annotated with their correct meaning representations (e.g. Wong & Mooney, 2006; Zettlemoyer & Collins, 2007). Because such annotations are time-consuming and costly to produce, research has also focused on exploring unsupervised (e.g. Poon & Domingos, 2009; Goldwasser et al., 2011) and weakly supervised (e.g. Chen et al., 2010; Börschinger et al., 2011; Chen & Mooney, 2008) methods for parser induction. One weakly supervised learning setting is the one also investigated in this thesis. In particular, research has focused on using ambiguous context representations instead of annotations as a step towards building machines which can learn language through exposure to language in some environment (Chen & Mooney, 2008). Addressing this issue, Chen et al. (2010) explore several semantic parsing systems, and they extend systems previously proposed with respect to supervised parser induction to handle ambiguous data. In particular, their experiments were based on the existing approaches KRISP (Kate & Mooney, 2006), KRISPER (Kate & Mooney, 2007) and WASP (Wong & Mooney, 2006). KRISP is an approach based on Support Vector Machines (SVM) supporting supervised learning, and KRISPER is an extension of KRISP which can handle ambiguous data by exploiting an approach based on expectation maximization (EM) (Dempster et al., 1977), i.e. the current parser is applied to score potential meanings and the parser is then retrained on disambiguated data. Using EM-like training has also been investigated by Chen et al. (2010) to extend further parsers, in particular to extend WASP to handle the ambiguous data; WASP is an approach based on statistical machine translation techniques which has been previously applied in a supervised setting. That is, the task of semantic parser induction can be seen as a machine translation problem where one translates from a natural language into a formal meaning language (Wong & Mooney, 2006). In WASP, word alignments are created first using the tool GIZA++ (Och & Ney, 2003), yielding a lexicon comprising *NL* substrings coupled with a mapping to their semantics. Complete meanings are constructed based on the lexicon using a synchronous context-free grammar (SCFG) (Aho & Ullman, 1972). Further, Chen et al. (2010) performed experiments in which they included a preprocessing step, where they applied a system (Liang et al., 2009) to disambiguate the ambiguous training data, and subsequently applied their approach to learn a semantic parser. In addition, Börschinger et al. (2011) tackle the learning task by inducing a Probabilistic Context Free Grammar (PCFG).

Since this learning task is similar to the one also investigated by Kwiatkowski et al. (2012) with respect to computational modeling of child language acquisition, their model might also be applied for semantic parser induction. Notice, however, that unlike Kwiatkowski et al. (2012), Chen et al. (2010) do not assume that the correct meaning is always among the competing ones given an utterance. Notice further that the systems proposed by Chen et al. (2010) and Börschinger et al. (2011) did not aim to model child language acquisition. In particular, their algorithms work by iterating over the full training dataset for several times in batch mode which is both cognitively implausible and computationally expensive.

Acquiring language by utilizing context information has also been explored in the context of games and virtual worlds (e.g. Qu & Chai, 2010; Reckman et al., 2010; Gorniak & Roy, 2005). For example, Qu & Chai (2010) utilized eye-gaze data, but focused only on improving automatic acquisition of words. By contrast, Reckman et al. (2010) attempted to derive both (sequences of) words and grammatical constructions from game logs obtained from the Restaurant Game (Orkin & Roy, 2007). These logs provided concurrent information of player's actions and chats. They first derived expressions referring to food-items. Then, complete patterns were derived by replacing these expressions by a slot. However, the meaning for complete constructions was not determined out of an ambiguous context: all patterns were assumed to refer to ordering a food-item. Gorniak & Roy (2005) introduced a system which mapped utterances to actions by using data collected in a game environment. However, they did not learn language from scratch but utilized existing parsers.

Several approaches have been proposed which attempt to learn language in the context of some environment (e.g. Branavan et al., 2009; Vogel & Jurafsky, 2010; Goldwasser & Roth, 2011; Branavan et al., 2010), e.g. for establishing mappings from natural language instructions to sequences of executable computer actions (Branavan et al., 2009, 2010) or interpreting navigation instructions (Vogel & Jurafsky, 2010). In contrast to utilizing parallel data from the visual or situational context, these approaches learn language by directly interacting with the environment, e.g. by exploring provided feedback in the framework of reinforcement learning.

2.9. Summary

In this chapter, we have reviewed existing techniques and related work relevant to this thesis. To date, several computational models addressing the acquisition of word meanings and/or syntactic acquisition have been proposed. Specifically, cross-situational learning has been explored by many computational models. However, these models have typically focused on learning simple mappings, particularly on establishing mappings between words and objects. Several models have addressed the acquisition of syntactic constructions, but often assume lexical mappings as given and/or do not take ambiguous contexts into account. However, our word-based learning scenario explored with respect to computational modeling of child language acquisition has also been explored before, albeit assuming that given an input example the correct meaning is always among the competing ones and by exploring different representations and learning mechanisms. Further, some work has already investigated learning language using context information with respect to computational language learning rather than cognitive modeling. In particular, the word-based learning scenario explored in this thesis has been explored previously in the NLP field semantic parsing.

Some work has investigated learning from speech without word transcriptions, but mainly focused on the acquisition of linguistic structures of rather low complexity such as words or acoustic morphemes. By contrast, in this thesis we aim at the induction of syntactic constructions under ambiguous context information. In particular, to the best of our knowledge, the spoken language learning scenario investigated in this thesis, which works with a phoneme recognizer, has not been explored previously.

Further, we have presented relevant techniques and research with respect to Frame-based Spoken Language Understanding. In research, this task is typically performed by first applying a word-based ASR and subsequently parsing the resulting transcriptions into their corresponding meaning representations. Moreover, the task can also be performed by applying a semantic speech recognition grammar with the ASR; in this case the ASR can be directly applied to transform spoken utterances into their corresponding semantic representations. In general, while data-driven and knowledge-based approaches as well as combinations of both have been proposed, knowledge-based approaches require extensive human effort for hand-crafting the system, while data-driven approaches typically require labeled training data. Notably, while SLU has typically been performed on ASR word transcriptions, recent research has shown that performing SLU on phoneme sequences can yield comparable, or even slightly improved, performance. However, in contrast with the work presented in this thesis, previous work did not explore syntactic acquisition, but focused on conceptual tagging of speech.

Chapter 3

A computational model for the acquisition of verb-specific constructions

In this chapter, we explore how language can be learned by formalizing and implementing learning mechanisms which are assumed as being implicated in early child language acquisition by psycholinguistic theories. In particular, we investigate early syntactic acquisition, i.e. the emergence of verb-specific constructions, including word-to-meaning mapping.
Work presented in this chapter has been published previously in Gaspers & Cimiano (2014a) with an early version of the model being presented in Gaspers et al. (2011); for a version of the model working with phoneme sequences please see Gaspers & Cimiano (2012).

3.1. Introduction

As mentioned before, there are two major competing theoretical approaches to language acquisition: generativist or nativist approaches (see Eisenbeiß (2009) for an overview) and usage-based or emergentist approaches (see Behrens (2009) for an overview). While the traditional generative approach to language acquisition originally proposed by Chomsky (1965) assumes that language ability is hard-wired/innate in form of a "universal grammar", usage-based approaches assume that linguistic knowledge is learned through interaction with and exposure to language in some environment. Specifically, usage-based approaches typically assume linguistic knowledge as being represented in terms of *constructions* as proposed within the framework of construction grammar. According to Goldberg (Goldberg & Suttle, 2010; Goldberg, 2003), construction grammar assumes linguistic knowledge to be represented in terms of constructions at varying degree of complexity and abstraction, ranging from words over morphemes to fully generalized, productive linguistic patterns. These constructions are captured by an interrelated network which comprises both item-specific knowledge and generalized patterns. Based on the network, novel utterances can be generated by combining existing constructions. In particular, many constructions contain slots which can be filled by other constructions, words for example. The network of constructions is assumed to be learned on the basis of positive input coupled with domain-independent learning mechanisms. Further, learning is assumed to be early on item-based in nature, and to begin with concrete examples, proceeding only later on to developing productive syntactic patterns (Goldberg & Suttle, 2010; Goldberg, 2003).

In fact, the assumption that children's representation of linguistic knowledge is early on item-based is not specific to construction grammar, but a key concept of usage-based approaches to language acquisition. Specifically, it is assumed that, from early on, children, unlike adults, maintain an inventory of lexically specific and item-based constructions which are gradually generalized by replacing concrete lexical items by slots which can be filled by (a restricted group of) words or short sequences of words (Tomasello et al., 1997). The resulting patterns are also referred to as *slot-and-frame patterns* (Pine & Lieven, 1997). It is not known in detail how children induce such slots, but one hypothesis is that they observe type variation in a position of otherwise identical utterances (Tomasello, 2000a). In general, in usage-based theories type frequencies are assumed to be involved in the generalization of linguistic knowledge along with token frequencies. While type frequencies guide the productivity of a construction and thus abstraction (e.g. Bybee, 1995), high token frequencies yield entrenchment of utterances (e.g. Bybee & Scheibman, 1999), and hence learning of constructions as a whole. It is not known what amount of type variation is required in order to achieve productivity/generalization of (a particular kind of) constructions, and the required amount may decrease over time, that is, less type variation in slots may be needed later on (Tomasello, 2000a). Patterns are assumed to be (initially) induced on a verb-specific basis – i.e. patterns correspond to concrete verbs –, yielding verb-specific predicate structures which are also referred to as *verb-islands* (Tomasello, 1992). Moreover, new linguistic qualities are also assumed to emerge in the sense that more complex structures can emerge from simpler ones

(Behrens, 2009).

It is still under debate to what extent human language skills are indeed innate (e.g. Eisenbeiß, 2009). Yet, several studies investigating child language acquisition provide support for an item-based nature of children's early linguistic knowledge (e.g. Tomasello, 2000b, 2003; Olguin & Tomasello, 1993; Lieven et al., 1997) as well as evidence for the existence of domain-independent learning abilities in children (e.g. Saffran et al., 1996; Saffran, 2003; Aslin et al., 1999). In particular, several studies suggest that children are able to detect statistical regularities within different domains, and in particular at different levels within the auditory domain (e.g. Saffran et al., 1996; Saffran, 2003; Aslin et al., 1999; Romberg & Saffran, 2010) as well as across different domains, for example between the auditory and the visual domain (e.g. Scott & Fisher, 2012; Smith & Yu, 2008). Further, there is empirical evidence that children are able to utilize the output of statistical learning mechanisms in turn as the basis for bottom-up learning mechanisms, for instance, tracking co-occurrence statistics about syllables to find words and in turn utilize these to track word order (e.g. Saffran & Wilson, 2003). Such statistical learning processes are assumed to be implicated in language acquisition, and there is now growing evidence for this assumption even in the case of syntax as, for instance, several studies suggest that input frequency and diversity shapes syntactic knowledge (e.g. Rowland, 2007; Huttenlocher et al., 2010). However, the direct relationship between statistical learning abilities and the acquisition of syntax in children has been investigated only recently by Kidd (2012). The author reports results suggesting that statistical learning is indeed implicated in the acquisition of grammar.

Taken together, usage-based and emergentist approaches assume that linguistic knowledge emerges over time where i) more complex structures can emerge from simpler ones, and ii) generalization yields gradual abstraction over seen input. Further, psycholinguistic findings support usage-based approaches to language acquisition by providing empirical evidence for the item-based nature of children's early linguistic knowledge and for the implication of domain-independent statistical learning mechanisms in language acquisition. However, what remains rather unclear is

1. how statistical learning mechanisms are implicated in the acquisition of syntactic patterns, i.e. how they interact with other learning mechanisms such as rule learning,

2. how exactly the involved generalization processes may operate, and

3. how more complex linguistic structures may emerge based on simpler ones.

Addressing these questions, in this chapter we explore how statistical learning processes can be combined with an item-based generalization method in order to learn rudimentary syntactic structures within the framework of a computational model. In particular, we investigate how the gradual emergence of an inventory containing verb-specific slot-and-frame patterns by an item-based induction of slots can be modeled computationally. In doing so, our model captures linguistic knowledge by an interrelated network of constructions at varying degree of abstraction without assuming pre-coded linguistic knowledge. As we are interested in modeling the emergence of slot-and-frame patterns, we consider two types of constructions: short sequences of words and lexically anchored slot-and-frame patterns.

Like a child, the model learns by observing natural language utterances – sequences of words – in a noisy and ambiguous context. As we explore different types of constructions, meaning must be associated with linguistic structures at different levels of complexity and abstraction. We propose uniform mechanisms for establishing and rating such associations at different levels of generalization. In particular, in order to establish correspondences between form and meaning we rely on the principle of cross-situational learning. As we attempt to model knowledge by means of a network, in our model all correspondences between form and meaning are modeled by associative networks (Rojas, 1993), where – as proposed by Hebb (1949) – connections between neurons which are active concurrently (i.e. between neurons representing form and meaning being observed concurrently) are strengthened, capturing their co-occurrence frequencies. We thus propose a uniform approach to learning based on the principle of cross-situational learning implemented on the basis of Hebbian-style learning to acquire constructions at different levels of abstraction.

We explore an incremental approach to language learning in the sense that linguistic structures of small complexity are learned first, followed later by more complex constructions acquired by bootstrapping on the simpler ones. In particular, the model starts by learning the structures of low complexity, i.e. (sequences of) words and their meanings. Once a learner is sufficiently confident in the linguistic knowledge it has obtained about (sequences of) words, it proceeds to learning more abstract constructions that abstract from specific words, yielding lexically anchored and partially productive slot-and-frame patterns through to fully productive constructions. This seems also cognitively plausible given the fact that children typically learn first the meaning of (proper) nouns and afterwards of more complex syntactic constructions (Bloom, 2000).

At the beginning, the model starts with an empty network. While learning proceeds, the network is continuously augmented and refined, dynamically adapting the model with new input. An important aspect of our model is thus given by the fact that learning proceeds online, i.e. each example directly causes an update of the network. This is an important aspect as the ability to learn online is a crucial capability of humans, enabling them to dynamically adapt to changes in their environment. Online learning is particularly important with respect to modeling human language acquisition, because the resulting models may not only account for why a certain behavior emerges but also address the question of how the behavior may be learned given constraints on the infant learner, e.g. on memory (Pearl et al., 2011). In particular, just like Pearl et al. (2011) we assume that a learner can only process one utterance – in our case presented together with concurrent information derived from the visual context – at a time, which contrasts with models storing a whole dataset in memory and process over contained utterances simultaneously. In this sense our model poses less requirements on memory and should a priori be preferable as a model for language acquisition compared to models that do not learn online but need to store examples explicitly.

The model we present in this chapter is composed of the following components:

1. **Representation:** An interrelated network for the **representation** of linguistic knowledge in the form of constructions of different levels of abstraction and complexity.
2. **Confidence:** Mechanisms for the **assessment of the confidence** in the learned structures and mappings as a basis for the **retrieval of knowledge** captured in the network.
3. **Learning:** A **language learning algorithm** which starts by incorporating item-specific knowledge into the network and proceeds to draw generalizations.

In sum, we present a computational model for early syntactic acquisition, including the acquisition of word meanings, which – in line with usage-based approaches and construction grammar – possesses the following properties:

1. Linguistic knowledge is represented in the form of a single **network** comprising constructions at different levels of abstraction and generalization
2. Patterns are induced on a verb-specific basis, yielding verb-specific predicate structures, i.e. **verb-islands**
3. Generalization is performed in an **item-based** manner and proceeds by gradually replacing concrete lexical items by slots

4. No **pre-coded linguistic knowledge** is assumed

5. Learning is performed in an **unsupervised** fashion in that no explicit tutoring is provided. However, since ambiguous information describing the visual context a learner observes is provided, the approach can also be seen as **weakly supervised** with context information being seen as ambiguous supervision (cf. Chen & Mooney (2008))

6. Learning proceeds **incrementally** in the sense that the model first learns the meanings of linguistic structures of low complexity and then uses these to learn the meanings of more complex constructions

Moreover, the model possesses the following important properties:

1. The model learns in the presence of **referential uncertainty**, i.e. by observing utterances while several actions are taking place concurrently where it is unclear which action – if any – is expressed by the utterance

2. **Cross-situational learning** is explored at different levels, and in particular beyond establishing mappings between words and objects

3. The model learns **online** in that it processes examples one-by-one, each directly yielding an update of the network structure

4. The model is capable of both **language understanding and production**, though in this thesis we focus on language understanding

5. The **fast mapping** ability (Carey & Bartlett, 1978) observed in children is explicitly build into the model by modeling a **disambiguation bias** (Merriman & Bowman, 1989)

While the principles of item-based generalization and cross-situational learning have been discussed extensively in the literature, we believe that we present the first comprehensive computational model that combines cross-situational learning beyond word-referent mappings with a formalization of a generalization mechanisms based on an item-based induction of slots in order to learn a lexicon and syntactic constructions in an online fashion.

We provide empirical results showing how the model is able to learn from positive linguistic input only, producing a compact construction grammar the size of which is much smaller than the number of examples observed, thus generalizing over the examples observed. We rely on a standard cross-fold validation scenario on a reference dataset to demonstrate the generalization abilities of our model.

The remainder of this chapter is organized as follows. In the next section, we describe the learning problem by presenting the model's input and the desired output, i.e. the construction grammar to be learned. Then, we present our model in more detail, including the network structure, measures and retrieval of knowledge, and the language learning algorithm. Afterwards, we will present empirical results obtained on a semantic parsing task. Before concluding, we discuss our model with regard to psycholinguistic findings and outline limitations and possible extensions.

3.2. Learning problem

In this chapter we propose a usage-based computational model of early language acquisition which assumes that linguistic knowledge is captured by an interrelated network of constructions that are acquired on the basis of positive input, and from which constructions can be retrieved at any time. The network in this sense encodes a grammar that can be used to process (unseen) utterances. As mentioned previously, the input to the model consists of two temporarily paired channels: a language channel and a visual channel. In the following, we will describe the learning problem in more detail by precisely defining the input and output of the model.

3.2.1. Input

Similar to a child, the model learns by observing natural language utterances (*NL*) in a noisy and ambiguous context (*MR*, represented by formulas in predicate logic *mr*). The context is ambiguous in the sense that several actions formalized as predicate logic formulas might be observed, and only at most one of these actions is expressed by the utterance. More specifically, the input to our model consists of a list of examples comprising *NL* utterances, each coupled with a set of meaning representations $\{NL, \{MR = mr_1, ..., mr_n\}\}$. For each example, *NL* is represented in symbolic form in the form of a sequence of words $w_1, \ldots, w_k$. Each $mr_i \in MR$ consists of a predicate ξ along with a list of semantic referents and their thematic relations (which might be empty). We distinguish between an observed *mr* and its corresponding semantic frame $[\![mr]\!]$. The semantic frame does not contain concrete semantic referents, but only an abstract signature of thematic relations representing the argument slots. For instance, for an *mr* *see(AGENT:mia,THEME:pizza)* the semantic frame is given by *see(AGENT,THEME)*, and we also say that *mr* instantiates the semantic frame $[\![mr]\!]$.

A concrete example for an (*NL,MR*) pair is given by:

(3.1)

NL:	Tim sees candy
mr_1:	*see(AGENT:tim, THEME:candy)*
mr_2:	*eat(AGENT:tim, THEME:pizza)*
mr_3:	*see(AGENT:mommy, THEME:candy)*
mr_4:	*sleep(AGENT:dog)*

Notice that there are no direct correspondences between *NL* utterances and their corresponding *mr*s; these correspondences must be learned by the model instead. Notice further that semantic referents are only arbitrary symbols to the model. We denote them by their corresponding natural words for reasons of clarity.

We define the underlying vocabulary of the *MR* portion of the data V_{MR} as containing all predicates ξ and arguments occurring in the input data. Thus, it incorporates all semantic entities – actions, actors, etc. – which might appear in a scene visually. For instance, with respect to Example 5.7 V_{MR} would contain the semantic entities *tim*, *candy*, *pizza*, *mommy* and *dog*.

We define the underlying vocabulary of the *NL* portion of the data V_{NL} as comprising all observed bi- and unigrams. In the following, we also refer to them as *(atomic) lexical units*. For instance, with respect to Example 5.7 V_{NL} would contain the lexical units “Tim”, “sees”, “candy”, “Tim sees” and “sees candy”.

3.2.2. Goal

Given a set of (possibly ambiguous) examples $\{NL, \{mr_1, ..., mr_n\}\}$ as described previously, our goal is to propose a model for the induction of a network capturing a construction grammar. In doing so, we consider two types of constructions, both comprising a form $\hat{NL}$ and a meaning $\hat{mr}$:

1. Constructions at the word level CON_{Word} where $\hat{NL}$ corresponds to a (short sequence of) word(s), e.g. “Vincent”’
2. More complex constructions at the level of slot-and-frame patterns $CON_{S\&F}$ where $\hat{NL}$ corresponds to an *NL* (pattern), e.g. “*X* sees *Y*”

In case of constructions belonging to CON_{Word}, the form $\hat{NL}$ constitutes a lexical unit $v_{nl} \in V_{NL}$, and $\hat{mr}$ corresponds to exactly one semantic entity $v_{mr} \in V_{MR}$. For $CON_{S\&F}$ constructions, the form $\hat{NL}$ constitutes an *NL* (pattern). Patterns may have different levels of generalization and may contain slots in which lexical elements can be inserted, and we distinguish two types of such elements:

- *Sets of slot-filling elements*: groups of lexical units which are required by an associated predicate, i.e. these sets represent slots in *NL* syntactic patterns which correspond to argument slots in an associated semantic frame, and different lexical elements imply different semantics (if the elements are not synonyms). In this sense these elements represent minimal units of semantic variation. For example, in case of two utterances "Mia eats" and "Tim eats" a set of slot-filling elements [Mia, Tim] might be established which maps to the $AGENT$ role in an associated predicate *eat(AGENT)*.

- *Sets of linguistically optional elements*: groups of lexical units which are optional with regard to an associated predicate, i.e. the exchange of elements contained in a set of linguistically optional elements causes no change in an associated meaning with respect to a given domain or semantic language. For example, a set of linguistically optional elements SL_1 = [huge, large] in a pattern "X eats a SL_1 pizza" does not account for changes in an associated meaning *eat(AGENT)*. Notice though that these elements may not be meaningless in general, but simply do not have a semantic counterpart in a given semantic vocabulary or inspected domain.

The meaning $\hat{mr}$ in case of a $CON_{S\&F}$ construction is represented by exactly one semantic frame $[\![mr]\!]$. If $\hat{NL}$ contains sets of slot-filling elements $SEs(\hat{NL})$, the argument slots $ARGs(\hat{mr})$ in $\hat{mr}$ are associated with them by a one-to-one mapping $\Phi : SEs(\hat{NL}) \rightarrow ARGs(\hat{mr})$.

As a concrete example, consider the following $(NL, \{mr_1, ..., mr_n\})$ pair:

(3.2)

NL:	Tim sees candy
mr_1:	*see(AGENT:tim, THEME:candy)*
mr_2:	*eat(AGENT:tim, THEME:pizza)*
mr_3:	*see(AGENT:mommy, THEME:candy)*
mr_4:	*sleep(AGENT:dog)*

and the $(NL, \{mr_1, ..., mr_n\})$ pair:

(3.3)

NL:	Mia sees pizza
mr_1:	*see(AGENT:mia, THEME:pizza)*
mr_2:	*sleep(AGENT:dog)*

At CON_{Word} our goal is to induce the constructions

(3.4)

$\hat{NL}$	Mia
$\hat{mr}$	*mia*

$\hat{NL}$	Tim
$\hat{mr}$	*tim*

$\hat{NL}$	pizza
$\hat{mr}$	*pizza*

$\hat{NL}$	cake
$\hat{mr}$	*cake*

and at $CON_{S\&F}$ our goal is to induce the construction

(3.5)

$\hat{NL}$	SE_1 sees SE_2
$\hat{mr}$	*see(AGENT,THEME)*
Φ	$SE_1 \rightarrow$ *AGENT* $SE_2 \rightarrow$ *THEME*

together with the sets of slot-filling elements below, along with their mapping to semantic referents:

SE_1 = [Mia → *mia*, Tim → *tim*],
SE_2 = [pizza → *pizza*, cake → *cake*],

where the elements contained in the sets of slot-filling elements are in turn constructions contained in the lexical network CON_{Word}.
Note that the mapping between sets of slot-filling elements and argument slots in semantic frames – Φ – has to be learned by the model, and its acquisition process is not dependent on the order of sets of slot-filling elements. For instance, the model is able to establish a link between the first set of slot-filling elements in an $\hat{NL}$ pattern and any arguments slot in an associated $\hat{mr}$. This is an important aspect because it enables the model to acquire both active and passive constructions in this way.

3.3. The computational model

Because our goal is to encode a construction grammar by means of a network, induced constructions are not stored directly as pairings consisting of form and meaning as might be suggested by the examples presented in the previous section. Instead, we propose an approach in which linguistic knowledge is stored in a network architecture and evolves over the course of time, thus continuously adapting the network structure to novel input. Yet, at each developmental step, constructions in the form as specified in Section 3.2.2 can be retrieved from the network by applying retrieval mechanisms which rate and reassemble the linguistic knowledge captured by the network. In the following, we will first introduce basic components included in our model (Section 3.3.1). Then, we will discuss the representation of constructions in the network (Section 3.3.2), and subsequently the employed generalization processes (Section 3.3.3). We will then present confidence measures which assess particular parts of the linguistic knowledge captured by the network (Section 3.3.4). Finally, we will explain the language learning algorithm employed in our model (Section 3.3.5), and show how constructions can be retrieved from the network (Section 3.3.6).

3.3.1. Basic components

In this chapter, in the process of construction grammar induction, the following learning steps play an important role:

1. Establishing correspondences between lexical units and simple semantic referents
2. Inducing generalized syntactic patterns by generalizing over observed *NL* utterances
3. Establishing associations between sets of slot-filling elements in syntactic patterns and argument slots in semantic frames
4. Associating semantic frames with syntactic patterns

where subtasks 1, 3 and 4 all correspond to the establishment of associations between form and meaning. In order to model these three subtasks, we consistently apply **associative networks** which model associations between form and meaning. Moreover, in order to address subtask 2, we utilize a **directed graph** which captures the word order of observed *NL*s and induced patterns by representing them as indexed paths, and the graph also provides mechanisms for the merging of paths which is important with respect to generalization of observed *NL*s. In the following, we describe these representational devices in more detail.

Associative networks

To model all correspondences between form and meaning we consistently apply associative networks as suggested by Rojas (1993), where – as proposed by Hebb (1949) – connections between neurons which are active concurrently are strengthened, capturing co-occurrence frequencies between form and meaning. An associative network A comprises two layers of neurons, x and y, which are fully connected by a matrix W of learnable weights. Based on the network, associations are retrieved by

$$y = Wx \tag{3.6}$$

and

$$x = W^T y. \tag{3.7}$$

To train the weights, we use the adjusted learning rule suggested by Schatten (2003)[1]

$$\Delta w_{i,j} = \eta(x_i - x'_i)(y_j - y'_j) \tag{3.8}$$

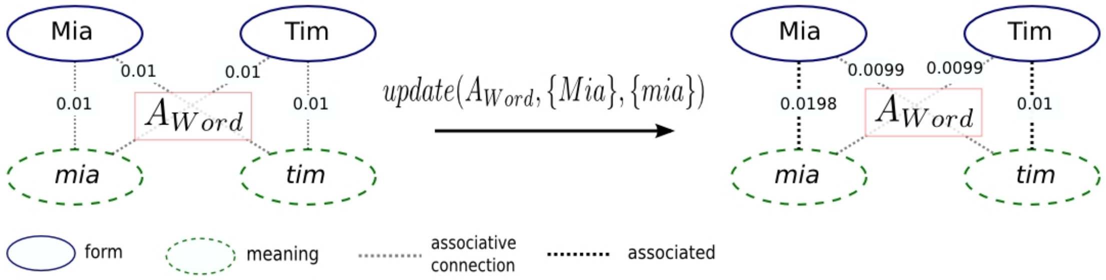
where x'_i and y'_j is the network's current value of x_i and y_j after processing the input y and x, respectively, and η denotes the learning rate. The change of a weight w_{ij} is then computed as

$$w_{i,j} = w_{i,j} + \Delta w_{i,j}. \tag{3.9}$$

Definition 1 (Update)**.** We also refer to the update of all weights in A according to equations 3.8 and 3.9 as $update(A, a_x, a_y)$ where a_x and a_y denote the sets of neurons which are active when observing stimuli x and y concurrently. We set their activation to 1 and the activation of all other neurons to zero.

Example 1. For instance, imagine that we attempt to model correspondences between lexical units and simple semantic referents with an associative network A_{Word} where the neurons in x correspond to lexical units and the neurons in y correspond to simple semantic referents as illustrated on the left side of the arrow in Fig. 1. Imagine further that the word "Mia" and the semantic referent *mia* are observed concurrently. Then, the update of the associative network $update(A_{Word}, \{\text{Mia}\}, \{mia\})$ ($\eta = 0.01$) yields a change in the weights contained in A_{Word} as depicted on the right side of the arrow in Fig. 1. As can be seen, the update step yields a strengthening of the connection between "Mia" and *mia*, which were observed together, while the weights between these two and the competing alternatives ("Mia" and *tim*, "Tim" and *mia*) are decreased slightly.

Figure 1.: Example of a network capturing associations between form and meaning at the lexical level, and the execution of an update step changing the contained weights.



[1] Hebb's rule states that the simultaneous activation of neurons results in a strengthening of the connections between them; it is given by $\Delta w_{ij} = \eta x_i y_j$ (Hebb, 1949). We apply Schatten (2003)'s rule because it prevents the continuous growing of weights (even in a fully trained network). Further, in contrast to Hebb's rule, weights can be decreased, i.e. (incorrectly) acquired words can become "forgotten"/"unlearned".

Definition 2 (Association)**.** We say that $j \in y$ is *associated* with $i \in x$ if it maximizes the value of the weights between neuron i and all neurons in y:

$$associated(i) = \underset{j \in y}{\operatorname{argmax}}\ w_{i,j} \tag{3.10}$$

Example 2. In the network assembling presented on the right side of the arrow in Fig. 1, *mia* is associated with "Mia" and *tim* is associated with "Tim".

Typically, associative networks are applied with a predefined set of neurons. However, because we explore a growing network, additional nodes and connections must be incorporated into the network. Weights of these connections may be initialized by zero and subsequently trained by the learning rule given in equation 3.8.[2] However, in the case of the associative network which captures correspondences between lexical units $v_{nl} \in V_{NL}$ and referents $v_{mr} \in V_{MR}$, we explore a different strategy for initialization. Our attempt here is to equip our model with the ability to quickly set up an initial mapping when observing a novel word in the presence of referential uncertainty, i.e. to model the *fast mapping* ability observed in children (Carey & Bartlett, 1978). In particular, in order to initialize weights for novel word-object mappings, we model a phenomenon observed in children which is also referred to as the *disambiguation effect* (Merriman & Bowman, 1989): children – at least at the age of two and older – who are presented with a novel object along with one or more known objects and are asked for the referent of a novel word, consistently choose the novel object (e.g. Golinkoff et al., 1992; Markman & Wachtel, 1988; Horst & Samuelson, 2008; Bion et al., 2013). In order to equip our model with such a disambiguation bias, we adapted a formula from a framework proposed by Vogt & Divina (2007)[3]. In particular, the weight $w_{nl,j}$ for a connection between a new n_{nl} representing a $v_{nl} \in V_{NL}$ observed for the first time and a neuron n_j is initialized as

$$w_{nl,j} = \frac{(1 - \max_i(w_{i,j}))\eta}{\#\text{ new } n_{nl} \in CON_{\mathit{Word(NL)}}}, \tag{3.12}$$

[2] This is the case in this model if not stated otherwise. Empty networks are always initialized in this way.

[3] The proposed model included several artificial agents maintaining lexica which contained association scores between words and meanings based on probabilistic cross-situational learning and feedback. Associations scores σ_{nj} for a newly observed word w_n were initialized building on the idea of the *principle of contrast* (Clark, 1993) as

$$\sigma_{nj} = (1 - max_i(\sigma_{ij}))\sigma_0 \tag{3.11}$$

where $i \neq n$ and $max_i(\sigma_{ij})$ denotes the maximum score meaning m_j has with other words w_i, $i \neq n$ (Vogt & Divina, 2007).

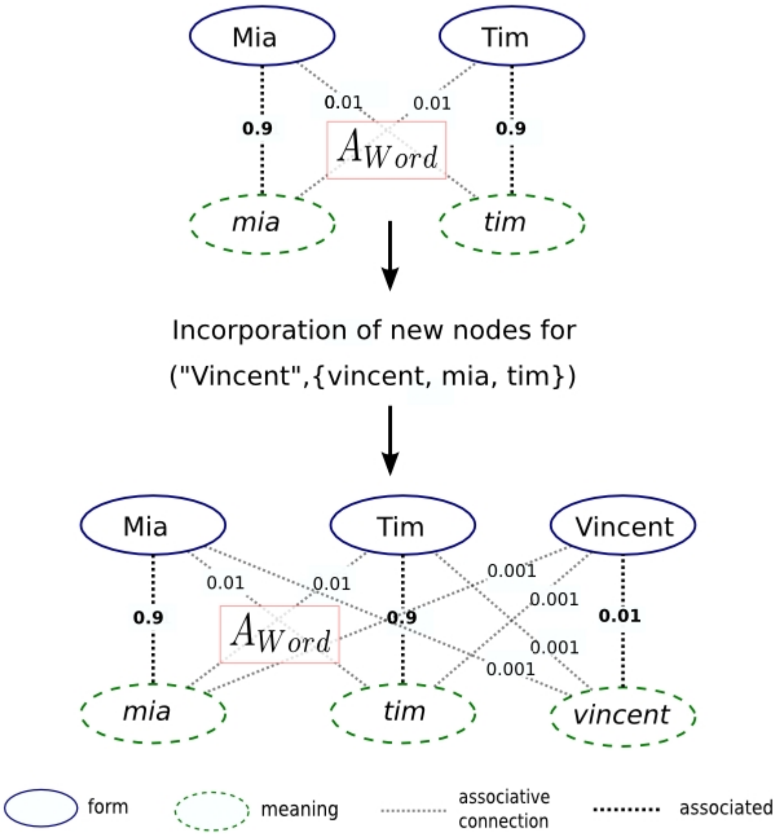
and the weight $w_{i,mr}$ for a connection between a new n_{mr} representing a $v_{mr} \in V_{MR}$ observed for the first time and a neuron n_i is defined analogously as

$$w_{i,mr} = \frac{(1 - \max_j(w_{i,j}))\eta}{\#\text{ new } n_{mr} \in CON_{\mathit{Word(MR)}}}, \tag{3.13}$$

where η – as in Equation 3.8 – denotes the learning rate. The underlying intuition of these formulas is that new lexical units/referents should preferably be associated with referents/lexical units which have not yet been associated with other lexical units/referents.

Example 3. Imagine for instance that the current state of A_{Word} is of the form as depicted above the arrow in Fig. 2 ($\eta = 0.01$). After observing the word-meaning pair ("Vincent", $\{$*vincent, mia, tim*$\}$) A_{Word} is updated as illustrated below the arrow in Fig. 2. Note that the illustration depicts the state of the network after the new nodes have been incorporated, but before *update*$(A, \text{Vincent}, \{$*vincent,mia,tim*$\})$ is

Figure 2.: Example for the incorporation of new nodes at the word level.



executed. As can be seen in the example, the model is biased to associate *vincent* with "Vincent" because all other referents (lexical units) are already associated with other lexical units (referents), mimicking the disambiguation ability observed in children.

Within the scope of the generalization process employed in our model, two procedures operating on associative networks are explored. We included these procedures in order to initialize the associations between form and meaning for generalized patterns by utilizing previously acquired information of the associations for the subsumed *NL*s and sets of slot-filling elements. Imagine for instance that two patterns "Mia sees SE_1" associated with *see(AGENT,THEME)*, $SE_1 \rightarrow \mathit{THEME}$ and "SE_2 sees pizza" associated with *see(AGENT,THEME)*, $SE_2 \rightarrow \mathit{AGENT}$ are identified by a merging process to represent the same pattern "SE_2 sees SE_1". Then, the operations introduced in the following will allow the model to associate the induced pattern directly with *see(AGENT,THEME)*, $SE_1 \rightarrow \mathit{THEME}$, $SE_2 \rightarrow \mathit{AGENT}$ by accumulating rows in an associative network (in case of the association with *see(AGENT,THEME)*) and by combining two associative networks (in case of the mappings $SE_2 \rightarrow \mathit{AGENT}$ and $SE_1 \rightarrow \mathit{THEME}$).

Definition 3 (Combining weights in an associative network)**.** Let A be an associative network comprising a layer x, a layer y and a matrix of weights W. Let furthermore $\vec{y}(i)$ be a vector comprising the weights $w_{i,j}$ between neuron $i \in x$ and each neuron $j \in y$ (as provided by W). Summing up the weights[4] in A for a given set of neurons $N \subseteq x$ results in a vector $\vec{y}(N)$, and is computed by adding up the vectors $\vec{y}(n)$ for all neurons in N, i.e

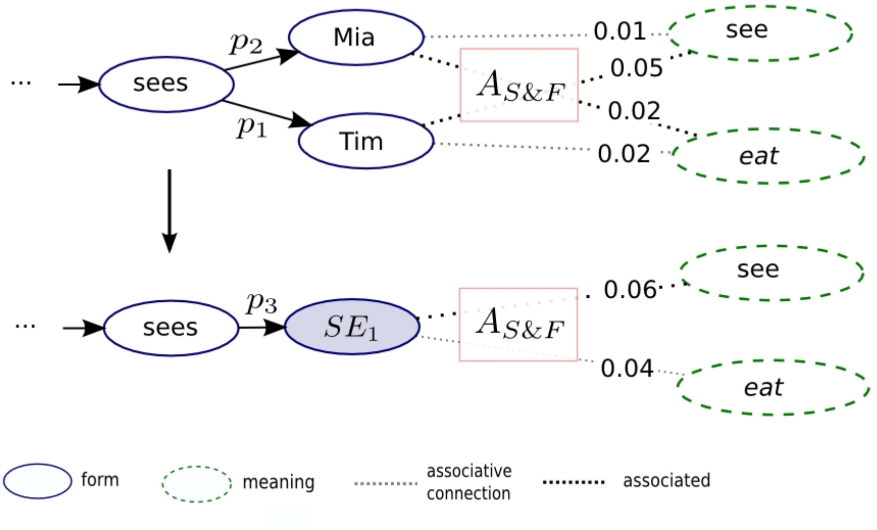
$$\vec{y}(N) = \sum_{n \in N} \vec{y}(n), \ n \in N. \tag{3.14}$$

The resulting vector may then be used to initialize the weights between a new neuron and all neurons in the opposite layer.

Example 4. Imagine for instance that we attempt to model correspondences between two *NL*s at the *S&F* construction level (represented as paths p_1 and p_2) and *mr*s using an associative network $A_{S\&F}$, where the two layers x and y of neurons correspond to paths and *mr*s, respectively, as illustrated in Fig. 3. The figure shows two paths (sees, Tim) (p_1) and (sees, Mia) (p_2) encoded in the network $A_{S\&F}$ as nodes, together with their weights for the connections to predicates *see* and *eat*,

[4] In our current implementation, weights are restricted to values between 0 and 1. Greater values are set to 1, smaller values to 0.

Figure 3.: Example for the combination of weights in an associative network.

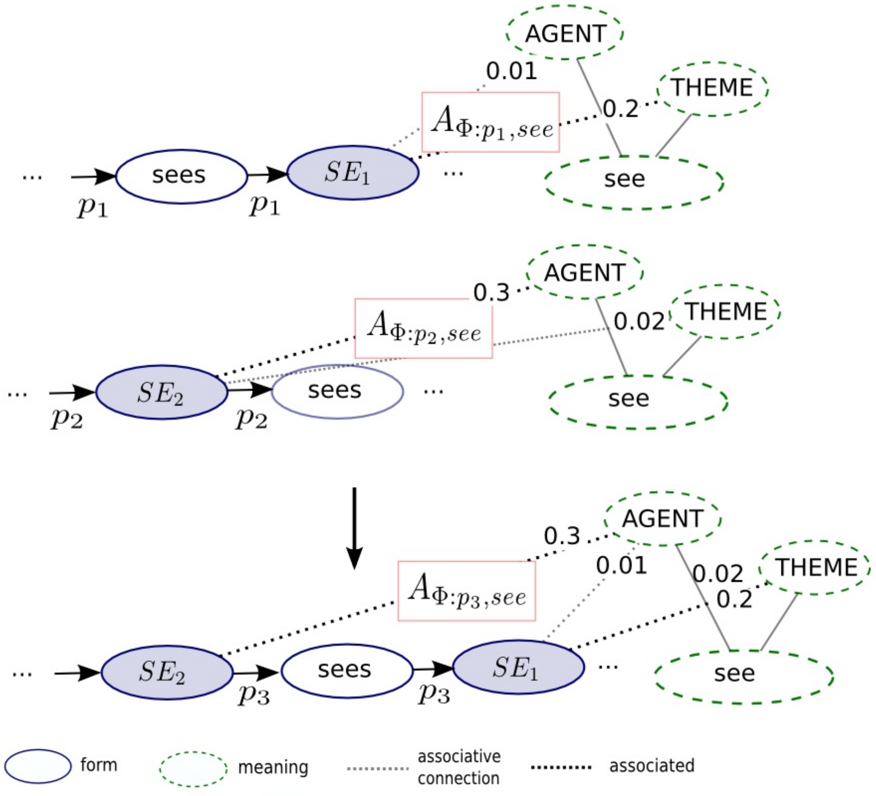


thus modeling the association strengths of the paths with the predicates. Merging both paths and combining the corresponding rows in the network – i.e. summing up the individual weights for all connections from the concerned nodes in layer x, i.e. p_1 and p_2, for each node in y, i.e. *see* and *eat* – yields a path (sees, SE_1) (p_3). The initialization based on the summed up weights clearly prefers the meaning *see*.

Definition 4 (Combining associative networks)**.** Two associative networks A_1 and A_2 are combined by $A_1 \oplus A_2$ into a single associative network A' composed of a layer x, a layer y and a matrix of weights W where $neurons(A') = neurons(A_1) \cup neurons(A_2)$, and $neurons(A_z)$ denotes all neurons contained in network A_z. The weights for connections between neurons $i \in x$ and $j \in y$ in A' are then initialized by $A'_{w_{i,j}} = A_{1_{w_{i,j}}} + A_{2_{w_{i,j}}}$.

Example 5. Imagine for instance that we want to model the correspondences between sets of slot-filling elements in syntactic patterns and argument slots in semantic frames – i.e. mappings – using associative networks where x and y represent sets of slot-filling elements and argument slots, respectively. Because sets of slot-filling elements appear at certain positions in patterns and argument slots are specific to semantic frames, such associative networks are hence specific to both a pattern and a semantic frame. Consider for example the first two graphs depicted in Fig. 4 showing a path p_1 (p_2) containing a set of slot-filling elements SE_1 (SE_2) for which co-occurrence frequencies regarding the argument slots in the *see* predicate are captured by an associative network $A_{\Phi:p_1,see}$ ($A_{\Phi:p_2,see}$). Imagine further that p_1 and p_2

Figure 4.: Example for combining the information of two associative networks.



are identified by a generalization process to represent the same pattern "SE_2 sees SE_1". By applying $A_{\Phi:p1,see} \oplus A_{\Phi:p2,see}$ we can directly model a mapping for the new syntactic pattern containing both sets of slot-filling elements and the semantic frame *see(AGENT, THEME)*, which is of the form as illustrated by the last graph in Fig. 4, associating SE_2 with the *AGENT* and SE_1 with the *THEME* argument slot in the *see* predicate.

Word order graph

The second representational device we utilize is a directed graph which captures the word order of *NL*s, which corresponds to the structure of the graph proposed within the ADIOS algorithm (Solan et al., 2005, cf. Section 2.6). Specifically, we represent the *NL*s of constructions at $CON_{S\&F}$ as indexed paths. Each contained node corresponds either to a lexical unit, a set of (linguistically optional or slot-filling) elements, or marks the start or end of a sequence. In the following, we will refer to this assembling as *word order graph*.

In our model, as a byproduct of generalization, the *NL*s of concrete examples at

$CON_{S\&F}$ are merged into generalized patterns. This is important as:

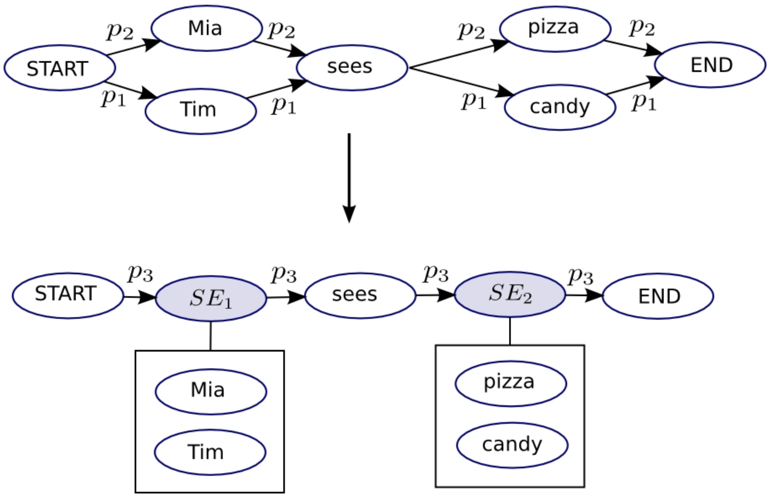
1. It keeps the network size – and therefore the corresponding grammar – small (because specific examples subsumed by generalized patterns are removed)

2. It enables the model to use constructions in a compositional manner, yielding generalization beyond examples seen, and thus understanding/generation of novel sentences

Because *NL*s of constructions at $CON_{S\&F}$ are represented as paths, a mechanism for merging paths is needed.

Definition 5 (Merging paths)**.** Merging a set of mergeable (see Section 3.3.3 for a definition) paths P with $|P| = m$ represented on a word order graph W yields a single path $p_{com} = p_1 \oplus p_2 \oplus \cdots \oplus p_m$ with $p_i \in P$ representing the merge of patterns in P. The combined path p_{com} is computed by iterating over the nodes for all paths in P concurrently. If all paths are alike at a position pos, the node at position pos in p_{com} is set to that node. Otherwise, it is set to a new node n_{se} representing a set of elements. Furthermore, for each path in P, the node at position pos is added to the set and subsequently replaced by n_{se} for each path in W. Finally, all paths in P are deleted from the graph. During the merging procedure, new sets of elements are induced. If such a newly induced set has an element (v_{nl} and/or in case of sets of slot-filling elements a v_{mr}) in common with at least one of the already existing sets, the corresponding sets are merged into a single set ec (note that according to the definitions of slot-filling and linguistically optional sets it is never possible to merge a linguistically optional and a slot-filling set). The nodes corresponding to subsumed sets of elements are then replaced by the node corresponding to ec. In the model, associations between paths and semantic frames are modeled by an associative network $A_{S\&F}$ with layer x corresponding to paths and layer y corresponding to semantic frames. The weights for p_{com} are initialized by combining the weights of connections between subsumed paths and semantic frames (cf. Definition 3), i.e. by summing up the weights of the subsumed paths in $A_{S\&F}$, i.e $\vec{y}(p_{com}) = \vec{y}(P) = \sum_{p \in P} \vec{y}(p)$. If p_{com} contains sets of slot-filling elements, an associative network representing the mapping $A_{\Phi:p_{com},[\![mr]\!]}$ is included for each $[\![mr]\!]$ in $CON_{S\&F}$ which contains slots. Each $A_{\Phi:p_{com},[\![mr]\!]}$ is initialized by combining the mapping associative networks for subsumed paths (cf. definition 4), i.e. each mapping between p_{com} and an semantic frame $[\![mr]\!]$ is initialized as $A_{\Phi:p_{com},[\![mr]\!]} = A_{\Phi:p_1,[\![mr]\!]} \oplus \cdots \oplus A_{\Phi:p_m,[\![mr]\!]}$.

Figure 5.: Example for the induction of sets of elements by merging paths.



Example 6. For instance, merging two paths p_1 and p_2 as depicted in the first graph illustrated in Fig. 5 by $merge(\{p_1, p_2\})$ results in a path p_3 as depicted in the second graph of the figure. The procedure induces the sets of slot-filling elements

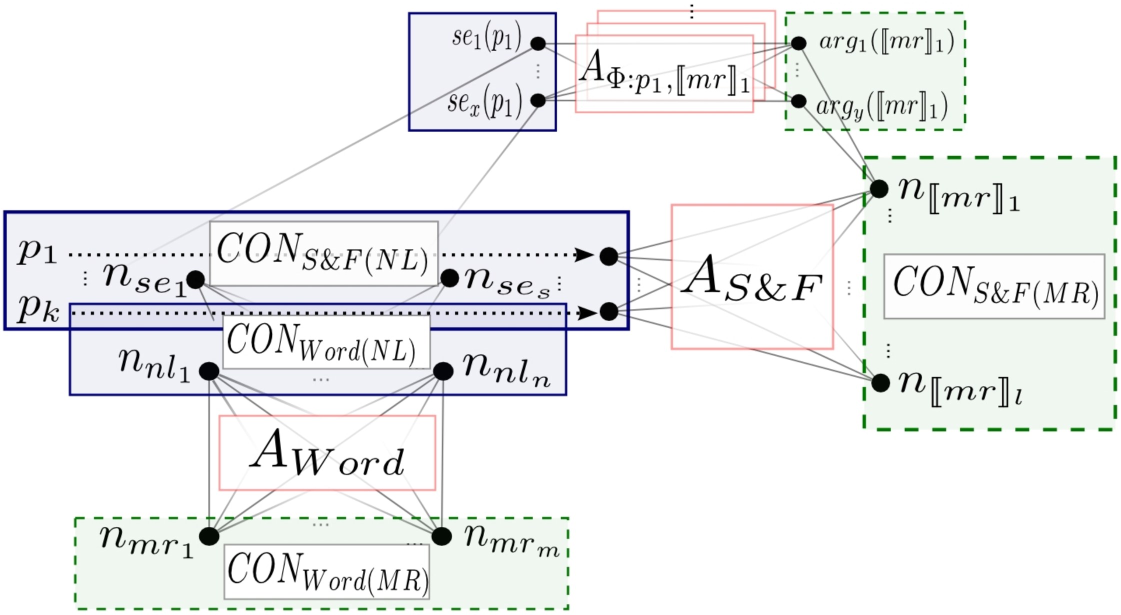
- SE_1 = [Mia → *mia*, Tim → *tim*]
- SE_2 = [pizza → *pizza*, candy → *candy*]

and deletes p_1 and p_2 from the graph, leading to a reduction in the number of paths, thus decreasing the network's size. As can be seen, the generalization furthermore enables the model to understand/generate two novel utterances (not observed in the input): "Mia sees candy" and "Tim sees pizza".

3.3.2. Network structure

The proposed network structure is depicted in Fig. 6. Recall from Section 3.2.2 that our aim is to include both constructions at the word level CON_{Word} as well as at the *S&F* level of S&F patterns $CON_{S\&F}$ into our network. Hence, the network is divided into two subnets representing constructions at the word level CON_{Word} and constructions at the level of slot-and-frame patterns $CON_{S\&F}$, where $CON_{S\&F}$ builds on CON_{Word}. Both subnets consist of a layer representing the form – $CON_{Word(NL)}$ and $CON_{S\&F(NL)}$, respectively – and a layer representing the meaning – $CON_{Word(MR)}$ and $CON_{S\&F(MR)}$, respectively. All correspondences between form and meaning are modeled by associative networks A_{Word} and $A_{S\&F}$. During learning, all observed

Figure 6.: Network modeling three levels of association: lexical units (captured in layer $CON_{Word(NL)}$) and single semantic referents ($CON_{Word(MR)}$), patterns represented as paths ($CON_{S\&F(NL)}$) and semantic frames ($CON_{S\&F(MR)}$), sets of slot-filling elements in patterns and slots in predicates.



linguistic input is incorporated into the form layers, while the action input, which is represented in the form of predicate logic formulas, is incorporated into the meaning layers. In CON_{Word}, each observed lexical unit is modeled as a single node n_{nl}, and semantic referents are modeled as single nodes n_{mr}. Constructions in $CON_{S\&F}$ are modeled as paths through a word order graph ($CON_{S\&F(NL)}$). The word order graph incorporates nodes from $CON_{Word(NL)}$, nodes representing the start n_{START} and the end n_{END} of a sequence, as well as a node n_{se} for each induced set of elements (these nodes group in turn sets of token nodes from $CON_{Word(NL)}$). $CON_{S\&F(MR)}$ contains a node $n_{[\![mr]\!]}$ for each semantic frame $[\![mr]\!]$ derived from mrs observed in the input. In $CON_{S\&F}$, constructions may include a mapping which maps sets of slot-filling elements to argument slots. These mappings are construction-specific, i.e. specific to both a path p and a semantic frame $[\![mr]\!]$. Thus, they are each modeled by an individual associative network $A_{\Phi:p,[\![mr]\!]}$.

In this thesis we only attempt to establish correspondences between NL expressions and observations from the visual domain. Yet, several words and expressions do not refer to observations from the visual domain, e.g. expressions being grounded in the time domain or referring to inner states, e.g. emotions. Because *NL* expressions may exist which have no semantic analogy in a particular domain, we include a special node $n_\perp$ in each associative network that allows to capture the fact that a certain linguistic construction has no correspondence at a meaning layer.

3.3.3. Generalization

The generalization mechanisms in our model support the emergence of generalized patterns beyond specific utterances observed that allow to understand and produce novel sentences. Generalization is performed in essence by i) inducing sets of elements and ii) merging paths to yield more general and productive ones (as described in Section 3.3.1). In order to identify paths that can be potentially merged, we introduce a *mergeability* condition that roughly states that patterns are mergeable if they differ in less than k positions. Recall that we distinguish between two types of sets of elements: *sets of slot-filling elements* and *sets of linguistically optional elements*. Sets of slot-filling elements represent sets of lexical units required by an argument of an associated predicate, while sets of linguistically optional elements group lexical units which do not cause a change in the meaning of an associated predicate. In order to differentiate between these two cases, we define a notion of *slot-driven mergeability* and and a notion of *syntactic mergeability*, leading to two different generalization steps in our model: a slot-driven generalization step and a syntactic generalization step which differ in the conditions specifying when the grouping of varying elements into a set of elements is reasonable.

As mentioned before, sets of slot-filling elements group elements the exchange of which induces a meaning difference. On this account, they are identified by searching for differences in patterns which lead to corresponding differences in the corresponding meanings. In particular, we assume that sets of elements in a group of *NL*s represent a slot-filling set if they can account for a slot in the corresponding predicate. In the following, we will first illustrate the intuition behind the employed generalization mechanisms and then present the model more formally. Given for instance the following two observed form-meaning pairings depicted first, one can easily infer the correspondences shown on the right side of the arrow.

(3.15)

NL:	Mia sees	*NL*:	Tim sees
mr:	*see(AGENT:mia)*	*mr*:	*see(AGENT:tim)*

→

NL:	X (= {Mia,Tim}) sees
mr:	*see(AGENT)*
Φ:	X → *AGENT*

The inference that “Mia” and “Tim” are substitutable and that the grouping accounts for the slot in the corresponding predicate *see* can be performed based on two observations:

1. The observed *NL*s differ in one position/slot *pos* and the corresponding *mr*s differ in one argument position/slot *ARG*, i.e. the one denoting the *AGENT*

2. The meanings of the observed words at position *pos* occur in argument slot *ARG* for both examples

Note that the second condition is crucial because our goal is to develop a model which learns from noisy input in which the meaning corresponding to an utterance may not be observed with it. Instead, the context representations for an observed utterance may comprise one or more distractor meanings only. Consider for example the following two *NL*s.

(3.16)

NL:	Mia sees a big cake	*NL*:	Mia sees a small cake
mr:	*sleep(AGENT:dog)*	*mr*:	*sleep(AGENT:cat)*

The first condition would lead to the grouping of “big” and “small” into a slot-filling set representing the argument slot in the *sleep* predicate. Yet, one can easily see that this is incorrect because “big” and “small” do not mean *dog* and *cat*, respectively. In order to avoid generalization errors, it is thus essential to acquire the meanings of individual lexical units in an *NL* before generalizing it. In particular, we only consider lexical units in the slot-filling generalization process the meaning of which has been learned in the lexical associative network (see the following section for a definition).

In our model, we compare a (possibly ambiguous) example (*NL*, $\{mr_1, ..., mr_n\}$) with a path p in the network.

Definition 6 (Slot-driven mergeable)**.** Given an example (*NL*, $\{mr_1, ..., mr_n\}$) and a path p, we first retrieve a semantic frame $\llbracket mr \rrbracket_p$ associated with p from the associative network $A_{S\&F}$ (which captures associations between paths and semantic frames) as the corresponding meaning. p is *slot-driven mergeable* with *NL* if *NL* and p differ in at most k positions and the element at each differing position in both *NL* and p corresponds either to i) a lexical unit with a learned meaning observed in a slot of some $mr_i \in \{mr_1, ..., mr_n\}$ which instantiates $\llbracket mr \rrbracket_p$[5] or ii) a set of slot-filling elements.

[5] Note that we cannot determine whether the lexical units actually map to the same slot in $\llbracket mr \rrbracket_p$ because we store only semantic frames in $A_{S\&F}$

The condition stating that an element at a differing position may correspond to a set of slot-filling elements models the case where a newly observed sequence is an instance of a generalized pattern already included in the graph. According to the described criteria a newly observed example ("Vincent sleeps",*sleep(AGENT:vincent)*) is mergeable with a path corresponding to a pattern "SE_1 sleeps" associated with the semantic frame *sleep(AGENT)* under the condition that *vincent* is the learned meaning of "Vincent".
In contrast to sets of slot-filling elements, sets of linguistically optional elements group lexical units which do not lead to a change in the meaning of a construction. Hence, the criterion in the syntactic generalization step is based on the meaning of paths.

Definition 7 (Syntactically mergeable)**.** A given path p_1 with a learned meaning $[\![mr]\!]_{p_1}$ is *syntactically mergeable* with a path $p_2 \in CON_{S\&F(NL)}$ if it differs from p_2 in at most k positions and $[\![mr]\!]_{p_1}$ is associated with p_2.

3.3.4. Confidence

In a learning system, it is crucial to compute how confident the learner is in its acquired knowledge, i.e. in our case in the acquired constructions. A notion of confidence is thus required to estimate how accurate the mappings learned at different levels actually are on the basis of the weights stored in the corresponding associative network. We apply a uniform approach to confidence assessment across levels, relying on the notion of entropy. The main idea is to measure the reduction in entropy as an estimator of how confident a learner can be in its acquired linguistic knowledge, in particular the mappings between meaning and form. The intuition behind exploiting reduction in entropy as a main criterion is to regard a construction as learned if a certain amount of information is acquired about it, i.e. if the uncertainty about its possible meanings has been sufficiently reduced. Specifically, the reduction in uncertainty is measured by the current entropy compared to the "maximum entropy" as follows. Given an $nl \in x$ and the weights $w_{nl,j}$, $j \in y$ we first normalize these weights to sum up to one (because the entropy is defined on a probability mass function). We then compute the current entropy as

$$H(nl) = -\sum_{j=1}^{|y|} w'_{nl,j} \log w'_{nl,j} \tag{3.17}$$

where $w'_{nl,j}$, $j \in y$ denotes the normalized weights. $H(nl)$ is then compared to the maximum entropy where each meaning is equally probable, i.e. we have no information about the correct meaning, by

$$H_{max}(nl) = -\sum_{j=1}^{|y|} \frac{1}{|y|} \log \frac{1}{|y|}. \tag{3.18}$$

An *nl* is considered as learned according to the entropy criterion if the proportion of the current entropy and the maximum entropy is below some threshold θ_E:

$$\frac{H(nl)}{H_{max}(nl)} < \theta_E. \tag{3.19}$$

While the reduction in entropy measures the amount of information captured by the network concerning the meaning of a given *nl*, i.e. it measures whether the number of possible alternative meanings has been reduced, it does not provide a criterion for selecting its best meaning. In addition, we therefore incorporate a rating for ranking all possible meanings for a given *nl*. In case of *nl*s without sets of slot-filling elements, we simply utilize the weights provided by A_{Word} and $A_{S\&F}$ for that purpose. For patterns containing sets of slot-filling elements, the rating additionally covers the quality of the learned mappings. Given an $nl \in x$ without sets of slot-filling elements, the corresponding rating for each $j \in y$ is computed as

$$rating(nl, j) = A_{w_{nl,j}}. \tag{3.20}$$

Recall from Section 3.2.2 that in case of a pattern $nl \in CON_{S\&F(NL)}$ containing sets of slot-filling elements, a one-to-one mapping between those sets of slot-filling elements and the argument slots in the appropriate semantic frame j is required. That is, a different argument slot $arg \in ARGs(j)$ must be associated with each set of slot-filling elements *se*, $se \in SEs(nl)$ according to the associative network $A_{\Phi:nl,j}$ which specifies the mapping between *nl* and j. Given an $nl \in x$ containing sets of slot-filling elements, the corresponding rating for each $j \in y$ is computed by augmenting the weight provided by the associative network $A_{S\&F_{w_{nl,j}}}$ with the association scores between each $se \in SEs(nl)$ and its associated argument slot $associated(se)$ in j as

$$rating(nl, j) = A_{S\&F_{w_{nl,j}}} + \sum_{se \in A_{\Phi:nl,j}} A_{\Phi:nl,j_{w_{se,associated(se)}}} \tag{3.21}$$

if a one-to-one mapping between the sets of slot-filling elements $SEs(nl)$ and the argument slots $ARGs(j)$ exists. Otherwise, the *rating* is set to zero.

Definition 8 (Learned Meaning)**.** Let A be an associative network comprising a form layer x, a meaning layer y and a matrix of weights W. Let $\hat{mr} \in y$ be the only meaning which maximizes $rating(nl, j), j \in y$ as defined by equations 3.20 or 3.21 for a given form $nl \in x$. Then, $\hat{mr}$ it is said to be the *learned meaning* of *nl* if the entropy criterion provided by equation 3.19 holds and $rating(nl, \hat{mr}) > \theta_R$.

3.3.5. Language learning algorithm

Slot-driven generalization as defined in Section 3.3.3 requires learned knowledge at the lexical level and syntactic generalization in turn requires learned knowledge at the level of slot-and-frame patterns. Hence, our approach to language learning is incremental in the sense that results obtained at a previous learning step are exploited within subsequent learning steps. The language learning algorithm is therefore roughly divided into three basic learning steps:

1. Training of the word layer CON_{Word}, acquisition of word level constructions, i.e. lexical units and their meaning

2. Training of the slot-and-frame pattern construction layer $CON_{S\&F}$, generalization step which searches for sets of slot-filling elements

3. Training of the slot-and-frame pattern construction layer $CON_{S\&F}$, generalization step which searches for sets of linguistically optional elements

Step 1 yields item-specific knowledge about individual lexical units, while step 2 und 3 yield generalized *NL* patterns which are derived from *NL*s or patterns based on knowledge acquired in prior learning steps. Because we propose an online algorithm, the three basic learning steps are in principle applied to each observed example. In particular, for each example, it is inspected if slot-driven and/or syntactic generalization is possible and if so performed accordingly. Because generalization in step 2 and 3 requires learned knowledge acquired within prior learning steps, slot-driven generalization builds on the acquisition of lexical units and in turn syntactic generalization builds on slot-driven generalization. The individual learning steps are composed of the methods described in the previous sections and are detailed in Algorithm 1.

In the first step, knowledge about lexical units is acquired, i.e. the corresponding associative network A_{Word} is updated. In the process, all semantic referents and lexical units are extracted from a given example and – if not yet present in the network – incorporated into the word layer CON_{Word}. New connections are initialized by applying the strategy described in Section 3.3.1. Subsequently, co-occurrence frequencies at the word level are captured by executing an update of the weights in A_{Word} with the referents and lexical units extracted from the example.

Algorithm 1 Language learning algorithm

```
Input: A list of examples $E = \{(NL_1, MR_1), \dots, (NL_k, MR_k)\}$
Output: A network $N$ representing constructions

  $N$ = an empty network
  for all examples $(NL_i, MR_i) \in E$ do
    1. update the word layer $CON_{Word}$
    ○   $units \leftarrow$ extract all lexical unit types from $NL_i$
    ○   $referents \leftarrow$ extract all semantic referent types from $MR_i$
    ○   add new lexical units and referents, incorporate and initialize connections
    ○   update associations between $units$ and $referents$ by $update(A_{Word}, units, referents)$
    2. update the S&F construction layer $CON_{S\&F}$, slot-driven generalization step
    ○   $s$ = preprocess $NL_i$
    ○   $P_m = \{p_1, \dots, p_{|P_m|}\} \leftarrow \{p \mid p \in CON_{S\&F(NL)}$ and $p$ is slot-driven mergeable with $s\}$
    ○   if $P_m \neq \emptyset$
            $p' = p_1 \oplus \cdots \oplus p_{|P_m|} \oplus s$
        else
            $p'$ = new path corresponding to $s$
        end if
    ○   $mrs \leftarrow$ extract all semantic frames from $MR_i$
    ○   incorporate each $mr \in mrs, mr \notin CON_{S\&F(MR)}$ into $CON_{S\&F(MR)}$
    ○   incorporate $p'$ into $CON_{S\&F(NL)}$, add and initialize connections
    ○   update associations between $p'$ and $mrs$ by $update(A_{S\&F}, p', mrs)$
    ○   update all mappings between $p'$ and semantic frames
    ○   merge identical paths
    3. update the S&F construction layer $CON_{S\&F}$, syntactic generalization step
    ○   $P_l \leftarrow p' \bigcup \{p \mid p \in CON_{S\&F(NL)}$ and $p$ is syntactically mergeable with $p'\}$
    ○   while $P_l$ contains paths $p \neq p'$
            $p' = p_1 \oplus \cdots \oplus p_{|P_l|}$
            $P_l \leftarrow p' \bigcup \{p \mid p \in CON_{S\&F(NL)}$ and $p$ is syntactically mergeable wit $p'\}$
          end while
    ○   merge identical paths
  end for
```

In the second step, the *NL* of the example is first preprocessed by replacing sequences of tokens in a set of elements by the set identifier. Afterwards, it is checked whether the preprocessed sequence is slot-driven mergeable with paths already contained in the network. If this is the case, it is merged with them as described in Section 3.3.1, and the resulting path is incorporated into the network. Otherwise, the sequence is incorporated as a new path. In any case, a new path p' corresponding to the sequence is incorporated into $CON_{S\&F(NL)}$. Then, all semantic frame types are extracted from the example and incorporated into the meaning layer at the *S&F* construction level $CON_{S\&F(MR)}$.

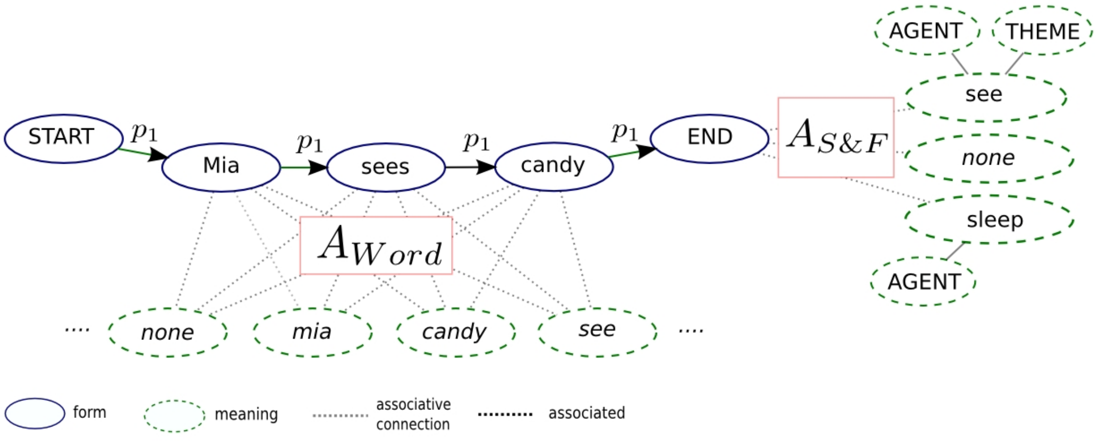
Example 7. For instance, imagine that the following example is observed as the first input example when starting with an empty network:

(3.22)

NL:	Mia sees candy
mr_1:	*see(AGENT:mia, THEME:candy)*
mr_2:	*sleep(AGENT:dog)*

Processing this example would yield the network state depicted in Fig. 7 (for reasons of clarity, only an excerpt of the nodes is shown in case of $CON_{Word(NL)}$).

Figure 7.: Example illustrating the incorporation of nodes and connections when starting with an empty network.

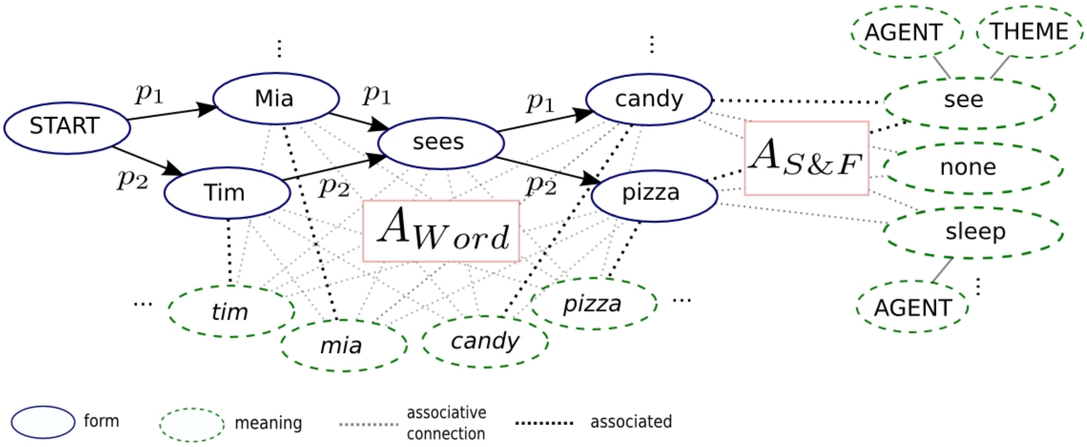


Subsequently, an update of the associative network $A_{S\&F}$ is executed, capturing the co-occurrence of p' and the semantic frame types appearing in the example. If p' contains sets of slot-filling elements, an associative network modeling the mapping is incorporated into the network – if not present already – for all observed semantic frames containing argument slots. All mappings between p' and the semantic frame types observed in the example containing argument slots are then updated.

Specifically, for each semantic frame $[\![mr]\!]$ derived from the input example, it is determined if the lexical unit's meaning is observed in an argument slot ARG of $[\![mr]\!]$. If so, the correspondence between pos_{se} and ARG is captured by an update of the corresponding associative network by executing $update(A_{\Phi:p',[\![mr]\!]}, pos_{se}, ARG)$. Otherwise the fact that the lexical unit's meaning is not observed in an argument slot is captured by $update(A_{\Phi:p',[\![mr]\!]}, pos_{se}, \perp)$. Due to the fact that as a result of path merging paths may become identical by replacing nodes with newly induced sets of slot-filling elements, as a final step identical paths are merged (as described in Section 3.3.1). This procedure includes combining their corresponding rows in $A_{S\&F}$ as well as combining the mapping associative networks.

Example 8. Consider for instance the network state depicted in Fig. 8.

Figure 8.: Example of a network representing two concrete natural language utterances along with semantic correspondences.



Imagine further that the following example is observed:

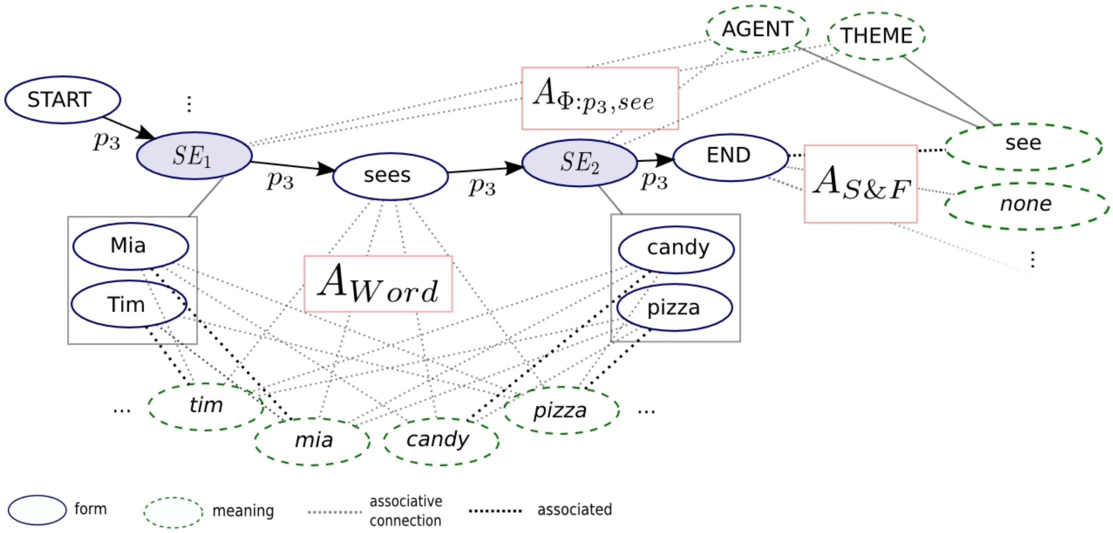
(3.23)

NL:	Tim sees candy
mr_1:	*see(AGENT:tim, THEME:candy)*

Processing the example would yield the network state depicted in Fig. 9 because the *NL* of the input example is slot-driven mergeable with both paths and is hence merged with them into a new path p_3.

Since the path contains sets of slot-filling elements and the *see* predicate observed with the input *NL* contains argument slots, an associative network modeling the mapping between the sets of slot-filling elements and the argument slots in *see* is incorporated into the network. The associative network is then updated by strengthening the connections between the first and second set of slot-filling elements in p_3

Figure 9.: Example of a network representing a syntactic pattern along with semantic correspondences.



and the first and second argument slot in the *see* predicate, respectively. This update step results from the fact that the meaning of the lexical unit appearing at the position in the input *NL* corresponding to that of the position of the first and second set of slot-filling elements in p_3, i.e. "Tim" and "candy", is observed in the first and second argument slot of the predicate, respectively, e.g. "Tim" is observed at the position corresponding to SE_1 and its meaning *tim* is observed in the first argument slot of *see* and thus their co-occurrence is captured by strengthening the corresponding connection in the mapping associative network $A_{\Phi:p_3,see}$.

In the third and final step, the model searches for paths which are syntactically mergeable with p'. If such paths exist, they are merged with p' yielding a new path, and the procedure is repeated until no more syntactic merging is possible. The whole learning procedure is then repeated for each observed input example and, while learning proceeds, the model's generalization capacity increases. Specifically, observed *NL*s may also be generalized without the presence of mergeable paths in the network due to the replacement of lexical units by sets of elements, and in doing so it is possible to directly acquire the correct meaning even in the case where an *NL* is presented in an ambiguous context; this will be illustrated by an example in the following.

Example 9. Imagine for instance that at the network state depicted in Fig. 9 the model observes the input example

(3.24)

NL:	Mia takes candy
mr_1:	*take(AGENT:mia, THEME:candy)*
mr_2	*see(AGENT:mia, THEME:dog)*

As a result of preprocessing which mainly replaces lexical units contained in sets of elements by the set IDs, the *NL* "Mia takes candy" is directly incorporated into the network as a generalized path (START, SE_1, takes, SE_2, END). As a result of updating the associative network representing the mapping Φ, in the case of *take* both sets of slot-filling elements are associated with appropriate slots in the predicate. In the case of *see*, however, only the first set of slot-filling elements is mapped to a slot. This is the case because "candy" does not fill any argument slot of the observed semantic frame *see(AGENT:mia, THEME:dog)*. A one-to-one mapping between sets of slot-filling elements and argument slots can thus only be induced for the *take* predicate, making it the only valid meaning for the new pattern. By this, the model is able to associate a unique meaning with the generalized pattern "SE_1 takes SE_2".

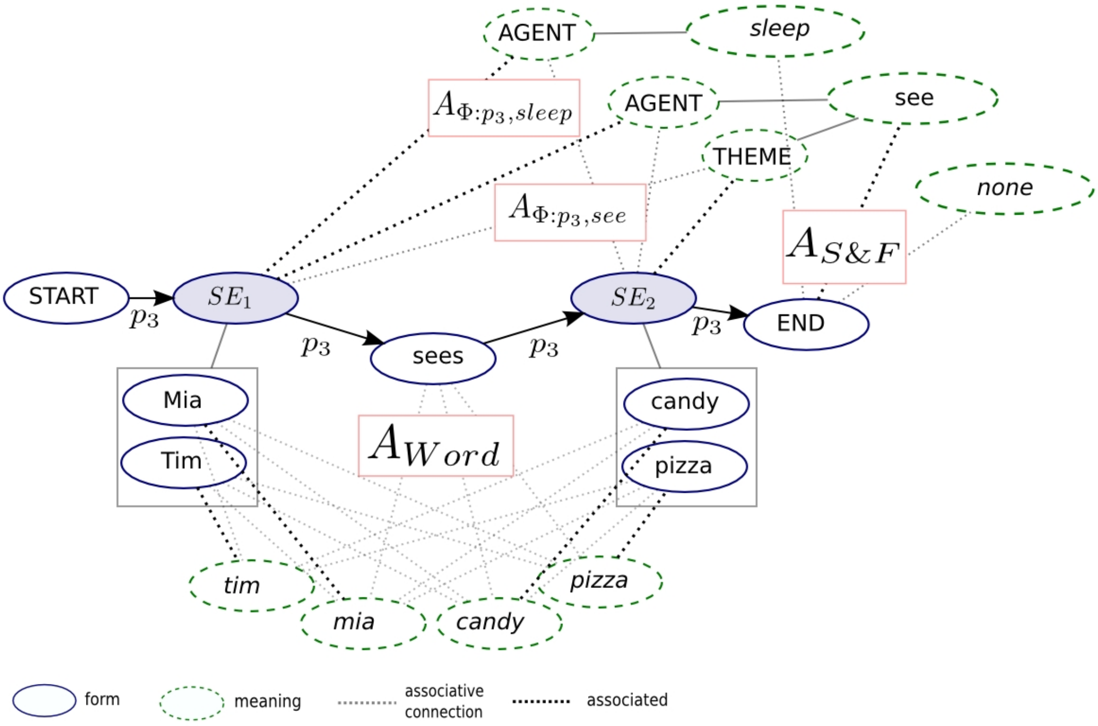
3.3.6. Retrieval of constructions

The model is able both to understand and produce language. In this chapter we focus on language understanding (since the model is composed of linear networks, production can be performed analogously). Given $nl \in CON_{\mathit{Word(NL)}}$, we can easily determine whether a (learned) meaning for *nl* exists as defined in Section 3.3.4. If this is the case, we can retrieve the meaning using the confidence measures defined there.

In order to retrieve the meaning of a complete *NL* utterance, we first preprocess it as described in the previous section, i.e. the utterance is converted into a sequence of lexical units where units contained in sets of elements are replaced by the set id. Note that a preprocessed utterance can match at most one path in the graph $CON_{\mathit{S\&F(NL)}}$ because each lexical unit can be only contained in one set of elements according to definition 5 and hence replacing lexical units by set IDs is non-ambiguous. Moreover, identical paths are merged in the graph during language learning and thus the preprocessed sequence can match at most one of the paths contained in the graph. If the preprocessed pattern is contained in $CON_{\mathit{S\&F(NL)}}$, we can determine whether a meaning exists and if so retrieve it. If no meaning with a *rating* score greater then zero exists, the meaning is set to $\bot$. If the meaning of an utterance is $\bot$ or if no corresponding syntactic pattern can be found in the network, the utterance cannot be understood (parsed) by the model. Otherwise, the meaning of each lexical unit at a position corresponding to a set of slot-filling elements in *nl* is retrieved from CON_{Word} and inserted into the argument slot associated with the set (via the associative network modeling the corresponding mapping) in the retrieved semantic frame. Whether the retrieved meaning is a learned meaning can again be measured as defined in Section 3.3.4.

Example 10. As an example, consider the small network depicted in Fig. 10 based on which we want to derive a meaning for the utterance "Mia sees pizza". First, the utterance is preprocessed, replacing the concrete elements "Mia" and "pizza" by the set IDs SE_1 and SE_2, respectively, yielding the sequence (START, SE_1, sees, SE_2, END), which matches path p_3. Subsequently, it is determined if a meaning corresponding to p_3 exists, and for that purpose the rating for p_3 and all semantic frames is computed. In case of *sleep(AGENT)*, the rating is zero because a one-to-one mapping cannot be established. By contrast, in case of *see(AGENT, THEME)* such a mapping can be established because ARG_1 is associated with SE_1 and ARG_2 is associated with SE_2. Thus, *see(AGENT, THEME)* is identified and returned as the corresponding meaning. Finally, the meanings for lexical units in the utterance which have been replaced by set IDs are retrieved from the lexical associative network A_{Word} and inserted into the appropriate argument slots (according to the mapping associative network) in the semantic frame, yielding *see(AGENT:mia, THEME:pizza)*.

Figure 10.: Example of a small network.



3.4. Experimental evaluation and discussion

We evaluated the performance of our model for the slot-and-frame pattern layer $CON_{S\&F}$ on a reference dataset. The reference dataset was chosen because it contains natural settings where natural language utterances occur in the context of several observed actions or events with many of these being unrelated to the utterance. More specifically, we used a semantic parsing task for evaluation. As mentioned previously, semantic parsing is the task of mapping *NL* sentences to $\hat{mr}$s and has been explored by learning from examples of *NL* utterances coupled with ambiguous contextual information. This setting is hence well suited to measure the system's language understanding abilities. In the following, we will first present information regarding the utilized dataset (Section 3.4.1). Subsequently, we will present the model's performance regarding language understanding for a fixed number of observed examples (Section 3.4.2). We will then provide a qualitative analysis of the acquired constructions (Section 3.4.3). Afterwards, we will examine the model's behavior over time, with a focus on performance in language understanding as well as its generalization abilities (Section 3.4.4). Subsequently, we will discuss the contribution of parameters incorporated in order to rate acquired knowledge (Section 3.4.5). Finally, we will verify the modeled behavior concerning fast mapping (Section 3.4.6).

3.4.1. Dataset

We used the RoboCup Soccer corpus (Chen & Mooney, 2008) for evaluation, which has been used widely for evaluating approaches to the induction of semantic parsers with ambiguous context information. The RoboCup Soccer corpus comprises four RoboCup games, i.e. the RoboCup finals from 2001-2004[6]. In this corpus, game events are represented by predicate logic formulas, constituting the meaning representations. The games were commented by humans, yielding the *NL* utterances. More specifically, in the corpus, each *NL* comment is coupled with a set of meaning representations – *MR* – comprising a set of possible $mr_i \in MR$, where the *NL* comment corresponds to at most one $mr_i \in MR$. For instance, *pass(purple10,purple7)* represents an *mr* for a passing event which might be commented as "purple10 kicks to purple7". However, there is no direct correspondence between the *NL* comments and their corresponding *mr*s. These correspondences must be learned by the model

[6] RoboCup is an international initiative that promotes research in the field of intelligent robotics (www.robocup.org). Games were taken from the Soccer Simulation League (www.robocup.org/robocup-soccer/simulation). In the Simulation League two simulated teams of agents play soccer against each other.

instead. The situation corresponds to natural settings in which an utterance occurs in the context of several observed actions or events with many of those being unrelated to the utterance. An example for an $(NL, \{mr_1, ..., mr_n\})$ pairing is given by

(3.25)

NL:	purple10 kicks to purple7
mr_1:	*ballstopped*
mr_2:	*badPass(pink1,purple10)*
mr_3:	*turnover(pink1,purple10)*
mr_4:	*playmode(play_ on)*
mr_5:	*kick(purple10)*
mr_6:	*pass(purple10,purple7)*

Notice that in this dataset thematic roles are inherent in the argument order for a given predicate, and hence we will denote them as $ARG_1, \ldots, ARG_n$ where n is the number of argument slots in the predicate. For example, in *pass*(ARG_1, ARG_2), ARG_1 denotes the actor while ARG_2 denotes the recipient.

While the training data is ambiguous – i.e. the mapping between utterance and corresponding action is not given – the corpus also contains a gold standard. The gold standard is a subset of the training data and comprises one meaning representation for each comment. Some statistics for the RoboCup training dataset are presented in Table 3.1.[7]

Table 3.1.: Some statistics for the RoboCup training dataset

Total number of comments	1,872
Comments having correct *mr*	1,539
Average number of actions per comment	2.5
Maximum number of actions per comment	12
SD in number of actions per comment	1.8
Mean utterance length	5.7
Number of tokens	10,700
Vocabulary size	443

3.4.2. Semantic parsing

We evaluated our model on a semantic parsing task on the RoboCup soccer corpus in order to estimate its language learning abilities by applying the evaluation schema

[7] Numbers for mean utterance length, number of tokens and vocabulary size are in contrast to Chen et al. (2010) only computed for comments included in the training dataset because only these are presented to the model. Regarding the total number of comments we use one more per game than Chen et al. (2010) in line with Börschinger et al. (2011).

explored in Chen et al. (2010). The authors evaluated their systems using 4-fold cross-validation on the four RoboCup games. In doing so, training was done on the ambiguous training data (always three games in our case), while the gold standard for a fourth game was used for testing. Results are presented by means of the F_1 score which is the harmonic mean of precision and recall. Precision and recall were computed as the percentage of $\hat{mr}$s produced by the system that were correct and the percentage of $\hat{mr}$s that the system produced correctly, respectively. A parse was considered as correct only if it matched the gold standard exactly (Chen et al., 2010). We used the same evaluation schema using $\eta = 0.01$ (learning rate) and $k = 1$ (maximum number of varying positions in mergeable paths).

Recall that our algorithm includes measures to determine whether a CON_{Word} or a $CON_{S\&F}$ construction should be regarded as learned. These measures are in turn based on two thresholds – Θ_E concerning the reduction in entropy, Θ_R concerning the rating –, yielding four parameters altogether since both thresholds are applied in case of CON_{Word} constructions and in case of $CON_{S\&F}$ constructions. In order to optimize these parameters, for each fold we trained the model with varying sets of parameters on the ambiguous training data. Subsequently, we measured its performance by means of the achieved F_1 score on the gold standard games corresponding to the games used for training. Note that disambiguated data was never used during training, and test data was used neither during training nor during parameter optimization. The parameters were then optimized with respect to the F_1 score. Since we explore an online algorithm, the number of examples necessary to yield a satisfying result is not known in advance. During parameter optimization, for each fold we used the incorporated training games three times since our algorithm is based on three basic learning steps, and hence an offline algorithm performing each step one after another may use each fold three times. The determined values found during the optimization process averaged over the four folds are: Θ_E at $CON_{Word} = 0.74$ (SD: 0.02), Θ_R at $CON_{Word} = 0.04$ (SD: 0.01), Θ_E at $CON_{S\&F} = 0.44$ (SD: 0.08), Θ_R at $CON_{S\&F} = 0.02$ (SD: 0.0). Notice that the parameters are rather consistent across folds with the exception of the entropy criterion at the *S&F* construction level.

Our goal is the induction of a construction grammar given ambiguous examples, i.e. to generalize concrete examples to slot-and-frame patterns. Without performing generalization, a learner may at most understand *NL*s which were presented during training. However, since the data are ambiguous rote learning is not actually possible. Rather, in several cases the learner further needs to choose one out of several competing alternative meanings. Thus, we created a simple "rote learning" strategy as a baseline. For this, we computed F_1 by parsing an utterance in the test data – if it had also been observed in the training data – by choosing one of the meanings observed with it randomly.[8] Thus, the baseline indicates the number of examples which can be parsed without performing generalization. The result is presented in Table 3.2 along with the results achieved by our model on the semantic parsing task.

Table 3.2.: F_1 scores for different construction grammars using different datasets and with varying times training data was seen.

Grammar	**#times training data was seen**	F_1 (%)
Examples ("rote learning")	1, 2 or 3	7.5
Our model	1	77.5
Our model	2 or 3	84.3
Our model without syntactic generalization	3	81.6

As can be seen the value is quite low with 7.5%. Further, the percentage of utterances appearing in the test data which also appear in the training data averaged over all folds is 16.3%, indicating that a large proportion of the *NL*s contained in the test datasets are novel.

We tested our model on the semantic parsing task using the training data for each fold between one and three times (Table 3.2). The results reveal that even after the first run the model is already able to perform a good deal of generalization, yielding $F_1 = 77.5\%$. After the second run, generalization is further increased, yielding $F_1 = 84.3\%$. The value then settles at 84.3% (precision: 96.6%, recall: 75%), indicating that (up to) two runs are already sufficient as an additional run does not further improve semantic parsing results. In order to investigate the impact of the individual generalization steps on the F_1 score, we also tested the model using three training runs without performing the syntactic generalization step. The result of 81.6% indicates that the large increase in F_1 achieved by our model is mainly due to the slot-driven generalization mechanisms. While performing syntactic generalization improves the results further, its contribution to the overall F_1 score is comparatively little.

The comparatively small contribution of syntactic generalization may be due to the fact that syntactic generalization requires previous establishment of meanings at the level of *S&F* patterns. For example, performing slot-driven generalization in case of two patterns "SE_1 passes to pink6" and "pink5 passes to SE_2" into a general-

[8] Note that the low percentage is to some extent due to varying capitalization for player names. While player names where typed starting with a capital letter in case of two of the games, they did not in the remaining two games. We did not lowercase input data because we neither did so during model training.

ized pattern "SE_1 passes to SE_2" may yield the additional understanding of all *NL*s instantiating "SE_1 passes to SE_2" (assuming that its correct meaning has been established). By this, a high increase in F_1 may be achieved. In contrast, merging two patterns "SE_1 shoots toward the goal" and "SE_1 shoots for the goal", both already associated with a meaning $kick(ARG_1)$ as required for syntactic generalization, into a pattern "SE_1 shoots SL_1 the goal" would not yield an increase in F_1 with respect to this pattern. This is the case because subsumed patterns could have been parsed already before performing syntactic generalization. Yet, it yields a more compact grammar. Further, in case of other patterns an increase may be achieved because the concrete lexical units "toward" and "for" would be replaced in all patterns by the induced set. For instance, in a pattern "SE_1 kicks for the goal" contained in the graph, "for" would be replaced by SL_1, yielding the additional understanding of instantiations of the pattern "SE_1 kicks toward the goal". However, this effect may be moreover less efficient in case of syntactic generalization – and thus an additional reason for its comparatively little contribution with respect to F_1 – because less grouping of elements took place in case of linguistically optional sets compared to slot-filling sets. In particular, in case of seeing the training data three times, the average number of elements (averaged over all folds) contained in a linguistically optional set was 3.1. By contrast, the average number of elements contained in a slot-filling set was 28.1 and hence about nine times higher. Thus, less productivity in case of syntactic merging was observed.

3.4.3. Qualitative analysis of acquired constructions

The *mr*s contained in the RoboCup soccer corpus conform to a simple CFG. In the following, we will present a qualitative analysis of constructions induced with respect to this CFG. The CFG incorporates only two non-terminals referring to arguments: $*PLAYER$ mapping to all players and $*PLAYMODE$ mapping to different playmode types. The grammar further incorporates nine predicates; Table 3.3 lists these predicates along with their frequencies in the training dataset and the number of comments annotated with them in the gold standard.

Corresponding to the **PLAYER* non-terminal, our model induced sets of slot-filling elements grouping players. More specifically, in three out of four folds players were grouped into one set of slot-filling elements while two distinct sets of slot-filling elements grouping different players were established in the remaining fold for the observed *NL* data. In turn, these sets of slot-filling elements were typically mapped to appropriate argument slots, i.e. to argument slots in predicates requiring an instance of **PLAYER* as an argument. In doing so, the learning algorithm was also

Table 3.3.: Predicates in the RoboCup corpus along with their counts in the training dataset and the number of comments annotated with them in the gold standard, and the five most frequent verb forms present in the training data.

Predicate	**#occurrences**	**#comments**
*playmode(*PLAYMODE)*	234	53
*turnover(*PLAYER,*PLAYER)*	694	121
*kick(*PLAYER)*	829	71
ballstopped	940	1
*pass(*PLAYER,*PLAYER)*	1,278	1,068
*steal(*PLAYER)*	101	46
*block(*PLAYER)*	44	9
*badPass(*PLAYER,*PLAYER)*	505	160
*defense(*PLAYER,*PLAYER)*	50	10

Verb form	**#occurrences**
passes	603
kicks	390
makes (as in “makes a pass”)	153
picked (as in “picked up the ball”)	109
intercepts	58

able to find lexical variants for the same entity, for example grouping “Pink goalie” and “Pink1”, both mapping to *pink1* in the same set of slot-filling elements.

The second non-terminal – $*PLAYMODE$ – appears only in the *playmode* predicate. This predicate can not be represented correctly by slot-and-frame patterns in our model due to the fact that instances of $*PLAYMODE$ are composed of several individual parts which in turn correspond to both the predicate and an argument. For instance, two examples describing a *playmode* event taken from the gold standard are given by

(3.26)

NL:	freekick from the purple team	*NL*:	pink team scores
MR:	*playmode(free_kick_l)*	*MR*:	*playmode(goal_r)*

As can be seen, in both cases the whole *NL* maps to the complex argument. Therefore, a correct slot-and-frame pattern cannot be derived. In two out of the four folds a set of slot-filling elements was induced (incorrectly) which mapped to the slot in the *playmode* predicate. For instance, in case of the second example “pink team scores“ a set of slot-filling elements $SE = \{$“pink”,“Purple”$\}$ and a pattern “*SE* team scores” was established along with its corresponding meaning $playmode(ARG_1)$, $SE \rightarrow ARG_1$.

Averaged over all folds, our model extracted 377.5 patterns. Table 3.4 shows the averaged number of patterns grouped by the predicates they refer to.

Table 3.4.: Number of patterns associated with the individual meanings averaged over all folds. An example for an extracted $\hat{NL}$ is provided for each meaning. In case of *playmode* *SE* refers to a set of slot-filling elements grouping the lexical units "pink" and "Purple". In all other $\hat{NL}$s *SE* refers to a set of slot-filling elements grouping players.

Associated meaning	**Avg #patterns**	**Example of an extracted $\hat{NL}$**
pass	77.25 (SD: 22.4)	*SE* fires a pass to *SE*
kick	41 (SD: 7.4)	*SE* dribbles the ball
badPass	40.25 (SD: 11.6)	*SE* makes a bad pass that was intercepted by *SE*
turnover	20.25 (SD: 4.0)	*SE* turns the ball over to *SE*
steal	7.75 (SD:2.2)	*SE* steals the ball
block	5.5 (SD: 2.2)	*SE* blocked the ball
playmode	3 (SD: 1.7)	*SE* team scores
defense	0	–
ballstopped	0	–
$\perp$	182.5 (SD: 40.1)	The shot was just a bit wide of the goal

Averaged over all folds, our model measured 182.5, i.e. about 50%, of the extracted patterns as meaningless. In fact, about one fifth of the comments in the corpus actually cannot be expressed correctly by the given predicates (Chen & Mooney, 2008), for instance, comments like "this way the game is going" which do not describe any action. Yet, the model also regarded various patterns as meaningless for which a corresponding meaning could be in principle determined. For instance, *playmode* events such as "free kick by the purple team" were often associated with $\perp$. This seems to be due to the fact that the model is not able to induce correct slot-and-frame patterns for *playmode*, as already discussed above. Token frequency may be a further reason, but, for instance, – relating back to Table 3.3 – *playmode* events were commented on more frequently than *steal* and *block* events. Yet, fewer patterns were established. In order to further investigate to what extent the result in case of *playmode* is dependent on the structure of the predicate, we performed the computations again by modifying the logical vocabulary such that, e.g. *playmode(free_kick_l)* would be represented as *free_kick(l)*. On average, using this modification 8.5 patterns were established referring to modified *playmode* events. Thus, the mapping of *playmode* events to $\perp$ is indeed to some extent due to its predicate structure. However, modifying *playmode* did not result in a higher F_1 score (note that an additional pattern can only improve F_1 if *NL* instantiations of it appear in both the training and the test data for a fold). Furthermore, patterns containing more sets of slot-filling elements than required by the appropriate predicate were associated with $\perp$ since no one-to-one mapping could be extracted. For example, "SE_1 tries to pass to SE_2 but was intercepted by SE_3" was mapped to $\perp$ since three sets of slot-filling elements were established but the appropriate predicate *badPass* contains only two argument slots.

As can be seen in Table 3.4, for each predicate between 0 and 77.25 patterns were extracted averaged over all folds. This may be the case because some events in the games were commented on more frequently than others (cf. Table 3.3). For example, *pass* events were typically commented on while *ballstopped* events were usually not. As indicated by the high precision of 96.6%, the derived meanings for *NL*s that were parsed by our model were mostly correct. The few erroneous parses basically included a confusion of *badPass* and *turnover* on the one hand and *turnover* and *steal* on the other hand because these events often follow each other in the games.

The sets of linguistically optional elements induced by our model were mostly appropriate with respect to the predicate logic formulas, i.e. typically the exchange of their elements actually did not cause a change in the meaning of *NL* patterns. Averaged over all folds, 12 sets of linguistically optional elements were induced by the model. To illustrate the nature of induced sets of linguistically optional elements, two constructions derived by our model and their included sets of linguistically optional elements are depicted in the following

(3.27)

$\hat{NL}$	SE_1 SL_1 SL_2 to SE_2
$\hat{mr}$	pass(ARG_1,ARG_2)
Φ	$SE_1 \rightarrow ARG_1$ $SE_2 \rightarrow ARG_2$

$\hat{NL}$	SE_1 makes a SL_3 pass to SE_2
$\hat{mr}$	pass(ARG_1,ARG_2)
Φ	$SE_1 \rightarrow ARG_1$ $SE_2 \rightarrow ARG_2$

where

SL_1 = [passes, kicks],
SL_2 = [forward, backward, off, back, up, out],
SL_3 = [short, quick, dangerous, cross],

and in both cases SE_1 and SE_2 correspond to the same set of slot-filling elements grouping players.

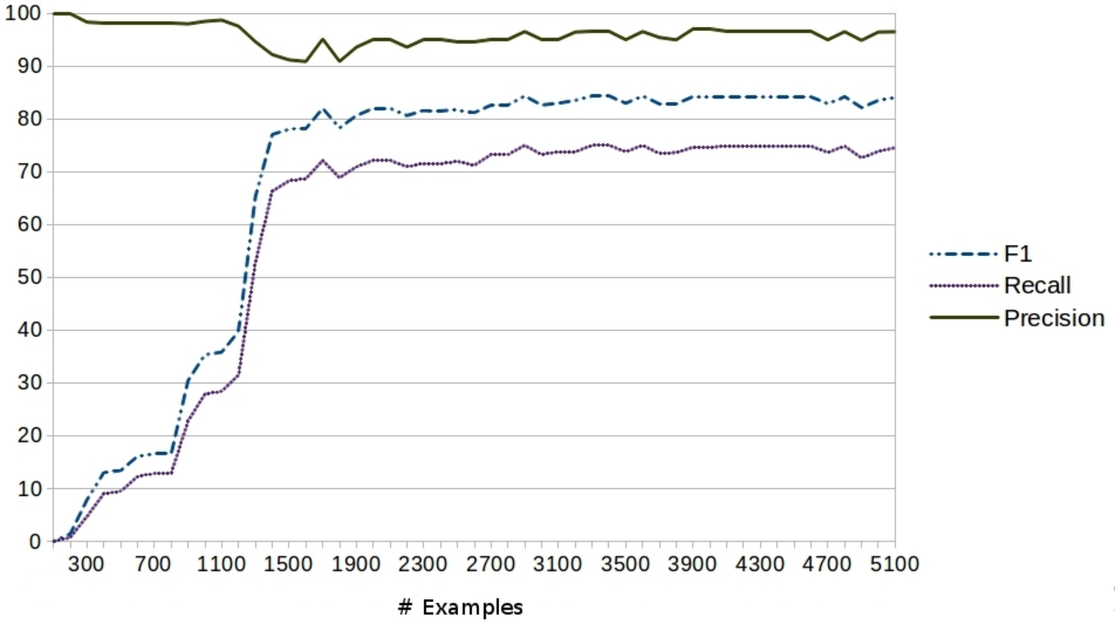
Altogether, our model thus accounted for the input data successfully in that it produced a compact grammar, allowing precise parsing.

3.4.4. Learning over time

In order to study the learning behavior over time, we monitored the performance of our model at different numbers of examples seen. Fig. 11 shows F_1, precision and

recall over the number of examples observed in steps of 100. The diagram reveals that the precision remains quite constant at levels between 90% and 100%, while the recall steadily increases over time, showing a jump at around 1,200 examples observed and coming to a plateau at about 1,300 examples observed. The reason for this jump can be seen in Fig. 12, which plots the size of the grammar compared to the number of *NL* types seen over the number of examples observed. While the number of observed *NL* types increases steadily, the number of stored grammar patterns increases less steadily. In fact, from 100 examples on, generalization is occurring, leading to a much slower increase in the number of patterns compared to the increase in the number of *NL* types observed. This shows that generalization is effectively occurring, yielding more generalized patterns which in turn yield an increased recall (see Fig. 11).
During further processing of the first 1,200 examples, the number of stored *NL* patterns increases up to slightly less than 400 patterns which is only about half of the number of observed *NL* types so far.

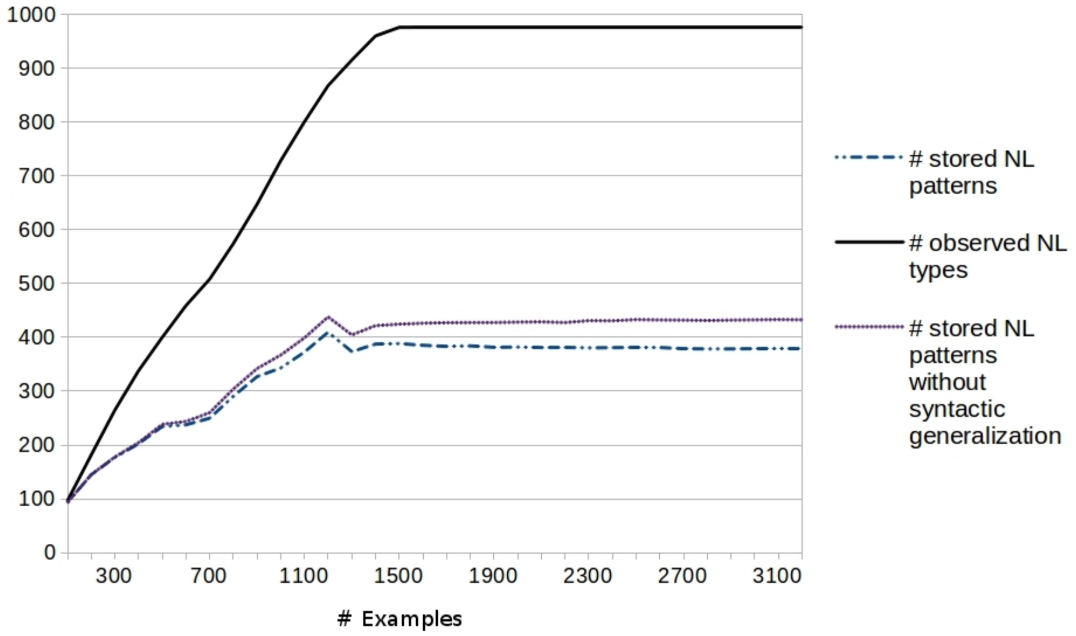
Figure 11.: Change in F_1, precision and recall over the number of observed examples.



After the first 1,400 examples seen, the model already achieves a high F_1 score of approximately 77%. After 1,700 examples, the model achieves an F_1 score of 80%, while the highest result of 84.3% is first achieved after 2,900 example are observed. While processing examples, F_1 fluctuates slightly but does not drop beyond 80%, indicating that already acquired knowledge is still refined but not lost (severely) during further processing of examples.

Overall, generalization yielded a large reduction in the grammar size; the final

Figure 12.: Number of observed and stored *NL* patterns over time.



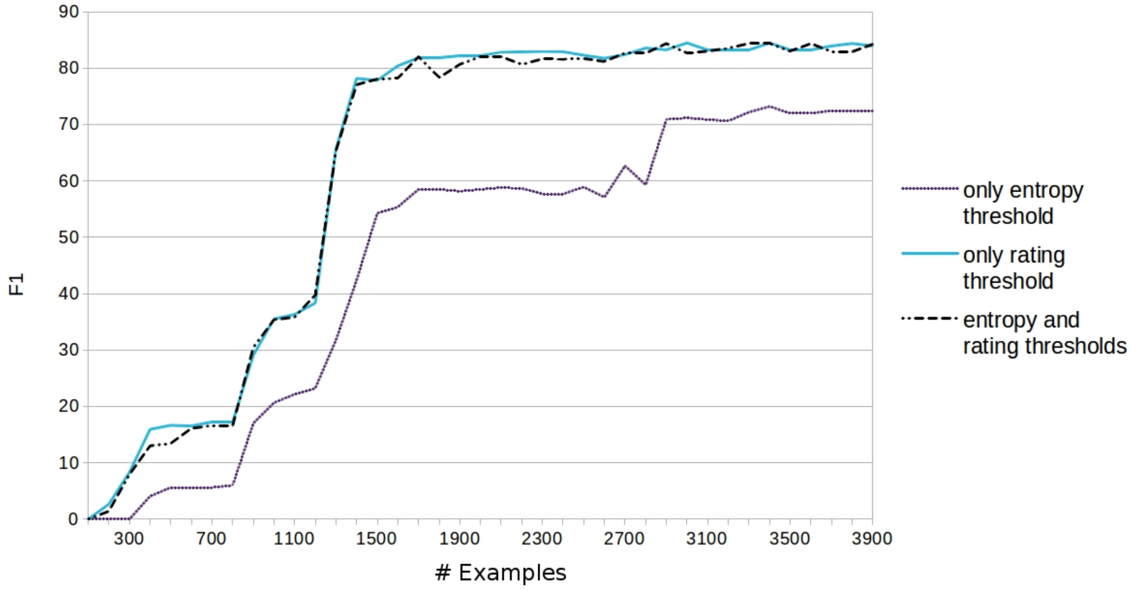
number of *NL* patterns is less then 40% of the number of observed *NL* types. The main effect in the reduction in grammar size is due to slot-driven generalization, while syntactic generalization yields a comparatively small effect as can be seen in Fig. 12. This is in line with the previously described results regarding their effect on the F_1 score: the slot-driven generalization step, which yields a larger decrease in grammar size also leads to an important increase in F_1. By contrast, syntactic generalization, which yields only a comparatively small effect on grammar size, leads to a comparatively small effect on the obtained F_1 score.

3.4.5. Parameters and confidence

In this section we discuss the suitability of our entropy and rating criteria as indicators of the confidence of the model in its acquired knowledge, and investigate how the employed parameters influence model performance. For this, we first optimized the parameters for the entropy and rating criteria on training data as described in Section 3.4.2. Fig. 13 shows F_1 over the number of examples observed for three conditions used to determine whether the learner is confident about some acquired knowledge: i) applying only the entropy threshold, ii) combining the entropy threshold with the rating criterion and iii) using the rating criterion only.

The results reveal that using only the entropy as confidence criterion, the perfor-

Figure 13.: Number of observed and stored *NL* patterns over time.



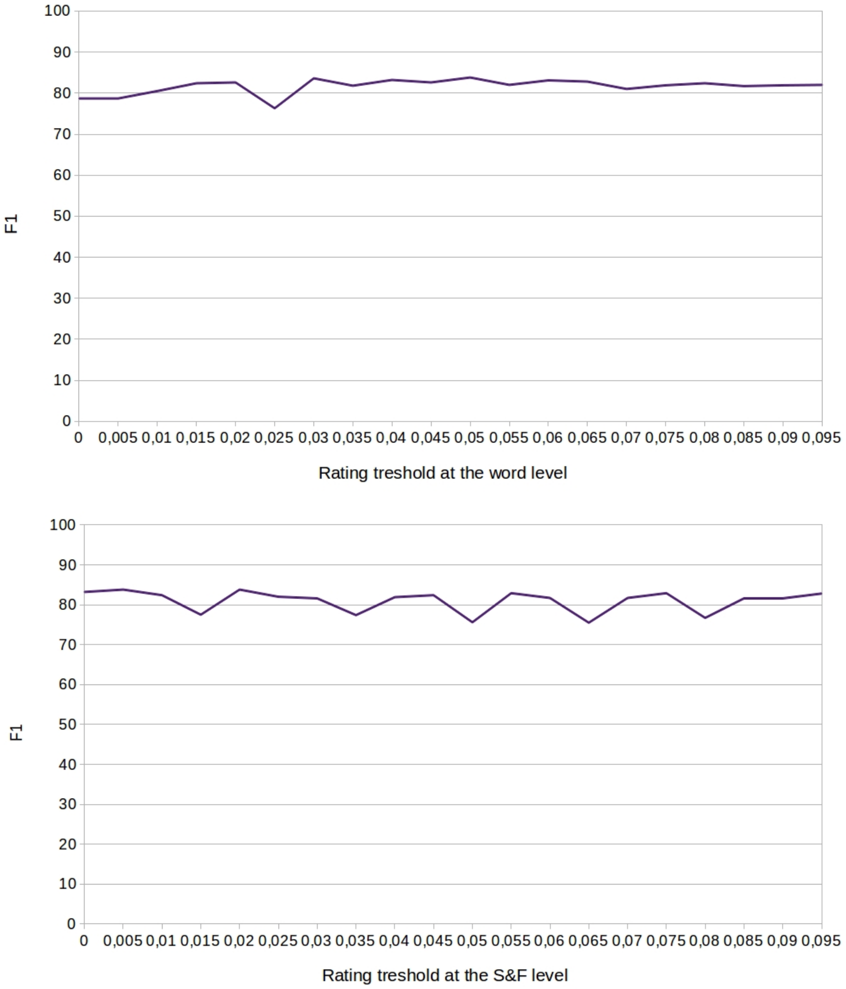
mance in terms of F_1 is suboptimal, peaking at around 70%. By contrast, computing confidence on the basis of the rating criterion as well as the combination of rating and entropy criteria yields F_1 scores of around 84%. Interestingly, the results show that there seems to be no significant contribution from the entropy-based criterion since the results of the combined measure is comparable to the condition using the rating criterion only.

In order to investigate the influence of each parameter with respect to its impact on the model, we study the influence of each parameter by varying it while holding the value of the other parameters constant by using the average values obtained in Section 3.4.2. The results for varying the rating criterion in case of both word and *S&F* constructions are illustrated in Fig. 14; they show that while there is some fluctuation, the impact of varying the rating threshold is rather low.

The results for varying the entropy threshold in case of both word and *S&F* constructions are shown in Fig. 15.

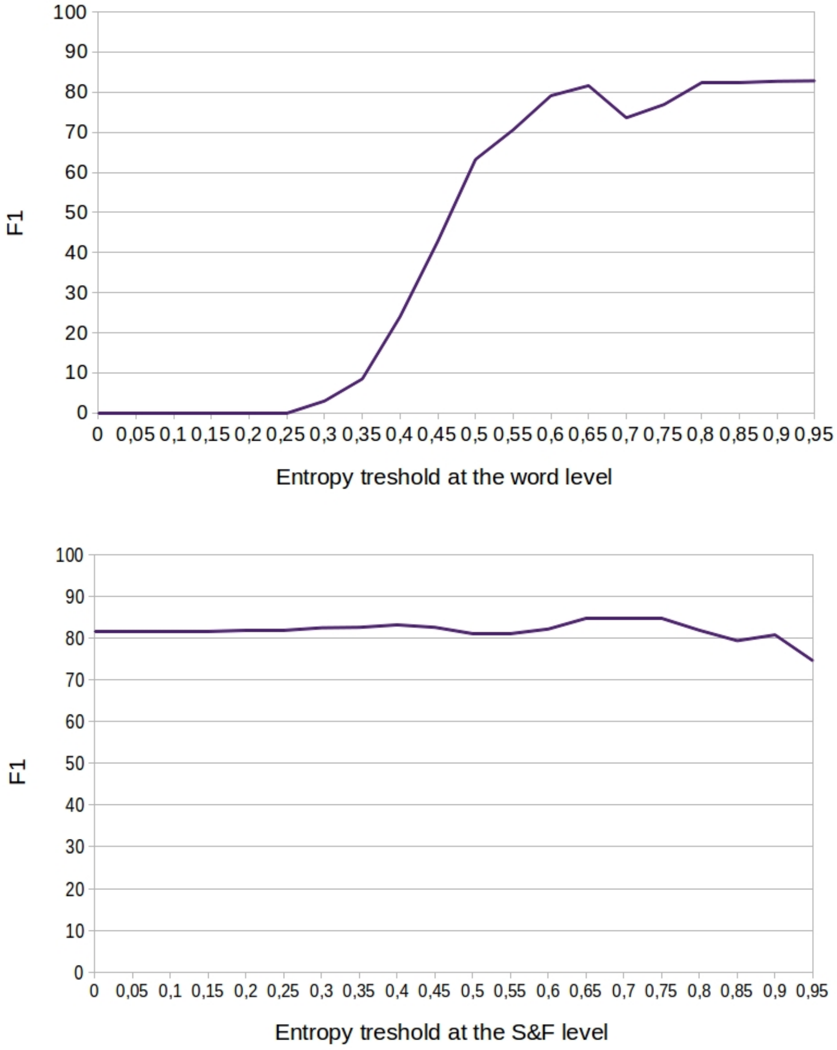
In case of CON_{Word}, using very low thresholds results in very low F_1 scores. This is the case because – due to referential uncertainty – the entropy concerning word meanings cannot be reduced sufficiently and hence the model cannot perform generalization. By decreasing the required reduction in entropy for a word to be measured as learned, i.e. using a higher threshold, the F_1 score increases to values of above 80%. In case of the *S&F* construction level, using very low thresholds again

Figure 14.: Change in F_1 when varying a rating threshold while keeping the remaining parameters constant



prevents generalization from being performed at all. However, since the threshold at $CON_{S\&F}$ level only guides syntactic generalization, the model is still able to perform slot-driven generalization, such that a low threshold basically shows how the model would perform without applying syntactic generalization at all. As in case of word level constructions, decreasing the required reduction in entropy for *S&F* constructions to be measured as learned, i.e. using a higher threshold, allows the model to perform generalization. Then, F_1 increases, reaching a top value of 84.8% by using thresholds of 0.65 and 0.7. Yet, F_1 fluctuates and even drops below the F_1 score achieved without performing syntactic generalization, indicating that existing knowledge has been deteriorated. That is, the model merged patterns which shouldn't have been merged. However, the highest amount of deterioration yields a

Figure 15.: Change in F_1 when varying an entropy threshold while keeping the remaining parameters constant



drop of about 8% absolute in F_1 compared to performing no syntactic generalization, and it only takes place by using a very high threshold. In particular, F_1 drops to about 74% only by using 0.95 as the threshold, i.e. by requiring only very little reduction in the entropy concerning the meanings of *S&F* constructions in order to perform syntactic generalization.

Taken together, while the model's performance depends on the employed criteria, varying one of them – at least in combination with the other parameters – did not prevent the model from learning language. In particular, all resulting F_1 scores were at least above 74%, which is – relating back to Section 3.4.2 – a high increase in F_1 when compared to the naive rote learning strategy which yielded an F_1 score of

7.5%.

3.4.6. Fast mapping

Relating back to section 3.3.1, we incorporated a disambiguation bias into our model, i.e. our model relies on a strategy for initializing weights in A_{Word} which biases the model to associate novel words with novel objects. This bias was incorporated to model a behavior observed in psycholinguistic studies with children within the scope of a so-called referent selection task. That is, when children – at least at the age of two and older – are presented with a novel object along with one or more known objects, and are asked for the referent of a novel word, they consistently choose the novel object (e.g. Golinkoff et al., 1992; Markman & Wachtel, 1988; Horst & Samuelson, 2008; Bion et al., 2013). However, while this disambiguation mechanism allows children to map a novel word correctly to a novel object under referential uncertainty, its relation to word learning is not completely understood. In contrast to referent selection tasks, retention tasks investigate how much a child actually learns about a novel word from a single or few exposure(s) as experienced within referent selection tasks. In order to investigate retention, typically a number of referent selection tasks are performed first. Subsequently, several of the previously introduced novel objects – and sometimes furthermore a completely novel foil object – are put together, and the child is then asked to select the object corresponding to a previously introduced novel word. Several studies suggest that retention occurs directly after referent selection tasks were performed (e.g. Golinkoff et al., 1992; Dollaghan, 1985; Wilkinson & Mazzitelli, 2003). In particular, Golinkoff et al. (1992) investigated retention in 30-month old children. Experiments included three blocks, each in turn consisting of four trials. In each trial, subjects were asked for both a familiar and a novel object. In trial 1 children were presented with four objects out of which three were familiar and one was a novel object. The goal was to test whether children would attach a novel name to the unnamed novel object rather than to familiar objects. Subsequently, in trial 2 children were presented again with four objects: two familiar objects, a novel exemplar of the new object presented in trial 1 and a completely novel object. Children were asked to select the object the name already presented in trial 1 refers to. The goal of trial 2 was to investigate whether children had learned the novel name well enough to extend it to another exemplar. In trial 3 children were presented with two familiar objects along with the novel object utilized in trial 1 and a completely novel object. They were then asked to select the completely novel object by asking for a novel name. The purpose of this experiment was to investigate whether children would attach the novel name rather to the novel unnamed object than to the previously named one, i.e. whether they would act as if the previously named object already had a name. Finally, in trial 4 a novel exemplar of the novel object from trial 3 was put together with two familiar and a novel foil object, and children were asked to select the object named in trial 3. The question was – as in trial 2 – whether children would make an extension to a new exemplar of a previously named object. Children performed correctly well above chance in all trials, suggesting that children can disambiguate the meaning of several novel objects – six in this experiment – in a short period of time, and that name-object links which were established by only two indirect exposures were strong enough to block further novel names from being attached to the object (Golinkoff et al., 1992). However, while these results may provide further insights concerning the character of novel word-objects mappings, they did not investigate whether the established links were stable enough to be retained after some delay. Further studies addressed this issue by investigating retention over a longer period of time, indicating that fast mapped words are forgotten rapidly (e.g. Horst & Samuelson, 2008; Vlach & Sandhofer, 2012). In particular, Horst & Samuelson (2008) investigated retention in 24-month old children with a delay of five minutes between referent selection and retention tasks. The main focus of their work was to investigate whether retention as found in studies without a (long) delay between referent selection and retention tasks was simply a short-time effect or due to retrieval from long-term memory, where according to the authors only the latter would suggest that the word-object pairs were actually learned. While children performed well in the referent selection tasks, they performed poor in the retention tasks (Horst & Samuelson, 2008). Yet, it must be noted that the ages of the children tested within the Horst & Samuelson (2008) and the Golinkoff et al. (1992) studies differed, and that children's ability in case of both disambiguation and retention appears to be age-related (Bion et al., 2013). In particular, in the experiments of Bion et al. (2013) children at the age of 18 months were not able to perform either one reliably. By contrast, children at the age of 24 months were able to perform disambiguation reliably, and children at the age of 30 months furthermore showed fragile retention skills. The authors concluded that children's disambiguation skill evolves between the age of 18 and 30 months, and that while this skill is related to word learning, word learning does not depend on it (Bion et al., 2013).

In the following, we will present experiments aiming to verify that the modeled disambiguation bias can indeed yield the desired behavior. Addressing this issue, we perform experiments with the word layer CON_{Word} employed in our model regarding fast mapping. Importantly, training of A_{Word} continued during these experiments due to the fact that children do not stop learning during referent selection or retention experiments either. Specifically, we investigated the initialization of weights by replicating the Golinkoff et al. (1992) task described above. Since in their experiments retention was investigated after a very short period of time, the experiments basically give insights concerning the characteristics of newly established word-object mappings, and we thus replicate them in order to investigate whether our model is able to initialize mappings accurately.

In order to perform the experiments, we first needed an initial vocabulary of familiar words and referents. For that purpose we trained CON_{Word} using 100 dummy words $word_x$ along with their corresponding dummy meanings $object_x$, $1 \leq x \leq 100$. In particular, we performed 10,000 update steps of the form $update(A_W, nl, mrs)$ (η = 0.01) where nl was a randomly selected word w_x, and mrs contained the object o_x corresponding to w_x along with two competing randomly selected objects other than o_x. The model was then evaluated on three blocks of referent selection and retention trials analogous to those previously described as performed by Golinkoff et al. (1992) (except for a slight modification: we did not use a new exemplar of an object in trial 2 and 4 but the object again. This was done because currently types of objects are not modeled within the underlying predicate logic). In all cases, an update of A_{Word} was first executed with the presented word and objects, and in case of novel words and objects weights for newly incorporated connections were initialized as described in Section 3.3.1. Subsequently, it was checked whether the correct object was retrieved as the meaning of the presented word, i.e. if the weight between the word and the object was higher than the weights between the word and all competing objects. In doing so, we did not apply the learning criteria introduced in Section 3.3.4 but simply tested whether the model showed a preference for the correct meaning. Our model was able to select the intended referent correctly in all trials and furthermore performed all retention tasks correctly, showing the same pattern for choosing objects as the children in the experiments. These results indicate that by explicitly building a disambiguation bias into our model it is able to utilize previously acquired knowledge on the word layer efficiently: very much like the children in the study our model is able to i) determine the meaning of several novel words in a row presented in an ambiguous context, and ii) establish name-object links from one or two indirect exposures which are strong enough to (temporarily) block the name from being attached to another novel object as well as another name from being attached to the object.

However, while the results indicate that the disambiguation bias can yield the desired behavior in initializing novel word-object connections, we cannot make direct predictions concerning relations between fast mapping and word learning. This is the case because we do not distinguish between working memory and long-term memory (processes). Hence, testing the model on a retention task with a delay as explored by Horst & Samuelson (2008) would not be comparable to behavior observed in children since waiting for some time would not yield a change in the model's behavior. Thus, fast mapping would again simply be tested with respect to short-time retention. However, the model might be extended addressing these issues, and then experiments concerning long-time retention may yield further implications concerning relations between fast mapping and word learning. Notice that in the performed experiments the results do not indicate that actual word learning occurred. In particular, the establishment of correct correspondences between novel words and novel objects does not mean that they were actually *learned* by the model according to the employed criteria (cf. Section 3.3.4). Recall that we did not apply these in the previously described experiments but tested whether the model showed a preference regarding the correct meanings. Compared to the weights for associations established during the vocabulary acquisition phase, weights for newly established associations during the referent selection trials were comparatively low, indicating that while the model is able to map novel words correctly to novel objects, the resulting new connections are rather weak and cannot be measured as *learned* word-object pairs, which is in line with the claim of Horst & Samuelson (2008). However, recall that the disambiguation bias is designed by initializing novel connections based on weights contained in the network, and these weights can increase over time. Thus, after longer learning periods weights for newly established connections might already by weighted such that they might be measured as learned by the employed criteria. Yet, here it must be noted again that we do not distinguish between short-time and long-time learning processes, i.e. even if our disambiguation bias would yield high initial weights for newly established connections, this would rather affect working memory and activations would likely decay rapidly, probably yielding little word learning with respect to long-term memory. However, the fact that weights for newly established connections can be greater at a greater number of examples observed, i.e. the disambiguation bias may become more effective, is somehow in line with the finding that disambiguation and retention appear to be age-related and evolve over time (Bion et al., 2013).

3.5. General discussion

In this chapter we have presented a formal, computational model for the gradual emergence of slot-and-frame patterns. The model unifies learning mechanisms proposed within usage-based approaches as being implicated in language acquisition and construction grammar with the idea of cross-situational learning in order to enable language learning in the presence of referential uncertainty. We have presented empirical results showing how the model was able to acquire a simple grammar starting from ambiguous input examples. In the following, we first relate the performance and functionality of our model to that of other algorithms and models learning from ambiguous context information (for a detailed overview of related computational models please see Section 2). Afterwards, we discuss the design of our model with respect to learning mechanisms proposed within psycholinguistic theories and cross-situational learning. Subsequently, we discuss some possible experiments as well as limitations of our model, possible extensions and future work.

3.5.1. Semantic parsing with ambiguous context information

In Section 3.4.2, we have shown that our model can be applied to a semantic parsing task. In particular, we evaluated the language learning capability of our model on the RoboCup soccer corpus which has been utilized by several work in order to evaluate approaches to semantic parsing in NLP. The evaluation scenario has been designed with respect to measuring performance, i.e. the goal is to achieve a high value in F_1. However, with respect to evaluating a computational model, cognitive plausibility becomes an important (additional) criterion. In fact, if children are not able to solve a certain task (at a certain age), then this should also be the case for a computational model for this learning task. Hence, a model showing worse performance on a certain learning task may in fact capture child language acquisition more plausibly.

In contrast to our model, the parsers introduced by Börschinger et al. (2011) and Chen et al. (2010) are induced by iterating over the full training dataset for several times in batch mode. Specifically, our model processes each example directly, i.e. adapts its knowledge directly and thus an incorrect decision may yield errors that cannot be corrected during further learning steps. However, processing examples online, our model achieved an F_1 score of 84.3% by processing the training data twice, requiring only a small portion of training data to achieve its maximum results in contrast to the system of Börschinger et al. (2011), which needed on average 76 iterations for each fold. Notice, however, that the performance of our model depends on the incorporated parameters and that unlike Börschinger et al. (2011) and Chen et al. (2010) we used disambiguated training data to optimize the employed parameters. Hence, direct comparisons of the achieved F_1 scores are rather difficult. Yet, the F_1 scores achieved by our model show that – depending on the chosen parameters – it can be successfully applied to induce a semantic parser from ambiguous input. The fact that our model learns in an online fashion is a relevant feature of our model compared to systems which work by iterating over the full training dataset for several times in batch mode as this is both cognitively implausible and computationally expensive. In particular, as we – in contrast to Börschinger et al. (2011) and Chen et al. (2010) – attempt to model human language acquisition skills, taking into account constraints regarding memory and processing of the infant learner is important (Pearl et al., 2011). Furthermore, the ability to learn online is interesting in case of several real-word applications, e.g. in the context of robotics, by aiming at the development of adaptive systems which are able to perform "life-long" learning. Similar to the work presented in this chapter, Kwiatkowski et al. (2012) addressed a semantic parsing task from the perspective of cognitive plausibility. In particular, they proposed a probabilistic model which was able to induce a semantic parser by processing examples one by one, and the authors report that their model outperformed a state-of-the-art semantic parser. However, they didn't use the RoboCup dataset for evaluation and therefore we cannot compare the models' performances directly.

In Gaspers & Cimiano (2012), we introduced an adapted version of the model presented in this work, which derived syntactic patterns from unsegmented phoneme sequences instead of sequences of words. In this work, we did not investigate learning from speech but applied grapheme-to-phoneme conversion. As we were only interested in how the model could be augmented to handle unsegmented phoneme sequences, we applied only the slot-filling generalization step without the weight initialization that mimics fast mapping, and utilized only the threshold concerning the weights. The model achieved an F_1 score of 81.1% on grapheme-to-phoneme converted RoboCup data in case of a single pass over the dataset. Thus, the mechanisms introduced within our model may yield a useful basis for further extensions, e.g. extending the model to work with a speech signal instead of sequences of words where words are not given, but have to be segmented out of the continuous speech stream.

3.5.2. Usage-based language acquisition

Both construction grammar and usage-based approaches assume language learning to proceed gradually from item-based and formulaic to abstract linguistic knowledge. Specifically, tracing back to the verb-island hypothesis proposed by Tomasello (1992) and several supporting studies with infants (e.g. Pine & Lieven, 1997; Tomasello, 2000b; Lieven et al., 1997), such approaches assume that children early on maintain an inventory of lexically specific and item-based constructions. These are then gradually generalized on a verb-specific basis by replacing concrete lexical items with slots which can be filled by (a restricted group of) words or short sequences of words (Tomasello et al., 1997), yielding verb-specific predicate structures, i.e. verb-islands. It is not known in detail how children induce such slots, but one hypothesis is that they observe type variation in a position of otherwise identical utterances (Tomasello, 2000a). In general, in usage-based theories type frequencies are assumed to be involved in the generalization of linguistic knowledge along with token frequencies. While type frequencies guide the productivity of a construction and thus abstraction (e.g. Bybee, 1995), high token frequencies yield entrenchment of utterances (e.g. Bybee & Scheibman, 1999), and hence learning of constructions as a whole. It is not known what amount of type variation is required in order to achieve productivity/generalization of (a particular kind of) constructions, and the required amount may decrease over time, that is, less type variation in slots may be needed later on (Tomasello, 2000a). In this chapter, we have formalized these ideas and presented a precise algorithm for an item-based induction of slots. Specifically, in line with usage-based approaches, in our model *NL* utterances are first incorporated into the network as a whole. Later on, already incorporated *NL*s and (partially) generalized patterns as well as newly observed *NL*s may be merged into patterns by inducing slots. In our model, induced slots are at first likely restricted to very few lexical units since our model can induce a slot-filling set based on type variation encompassing only two lexical units, thus creating highly pattern-specific slot-filling sets. While learning proceeds, slot-filling sets are extended and merged, yielding more general slot-filling sets and more general and more productive patterns. As already illustrated by example 9, at later learning steps the model is able to generalize over observed *NL*s directly without the additional observation of another *NL* showing variation in the surface structure. The generalization processes employed in our model allow for the establishment of an inventory of (partially) generalized slot-and-frame patterns and *NL*s by gradually merging and generalizing *NL*s and patterns with increasing productivity.

While type and token frequency are strongly involved in the employed generalization processes, we furthermore rely on information regarding the meaning derived from the context based on cross-situational statistics. In particular, in our model high token frequency coupled with (high) type variation regarding a position is not sufficient for a group of *NL*s in order to be generalized. In fact, merging and generalization are rather conservative in the sense that two *NL*s are only merged if a consistent meaning can be identified. In our case, a consistent meaning is identified if either those *NL*s vary in one position and this position corresponds to a slot in a corresponding predicate (slot-driven condition) or both *NL*s vary in one position and both correspond to exactly the same meaning at the *S&F* construction layer (syntactic condition). Otherwise, no generalization is performed. In particular, during slot-driven generalization, only such slots are induced which correspond to semantic roles required by the corresponding meaning, and this in turn depends on the ability of extracting this meaning from an ambiguous context. Specifically, knowledge concerning the meaning both of the corresponding predicate as well as the concerned lexical units is required, and thus – in line with emergentist approaches to language acquisition (Behrens, 2009) – more complex units, i.e. *S&F* patterns, emerge from simpler ones, i.e. words. As already discussed and illustrated by Example 3.16, we additionally utilize non-linguistic context information and require learned knowledge in order to avoid generalization errors.

Token frequency of *nl*s (complete *NL*s or lexical units) is incorporated into the generalization process in the sense that an *nl* has to be observed several times in order to be estimated as learned, and token frequency coupled with the observation of meanings yields entrenchment of the correspondences between *nl* and the concurrently observed meaning(s) and therefore learning. However, high token frequency is not sufficient in order to derive a meaning for an *nl* because this requires additionally that it must be possible to extract a corresponding meaning out of the ambiguous context. In particular, the emergence of a meaning *mr* for an *nl* relies on the frequency of co-occurrence of *nl* and *mr* compared to the frequency of the concurrent observation of *nl* and competing meanings. That means that an *nl* and its meaning must be seen multiple times together to yield entrenched pairings of form and meaning.

While – as suggested within the framework of usage-based approaches – type variation is utilized in our model in order to determine positions in *NL*s which may correspond to a slot, the amount of variation needed in order to induce a slot is quite low: merging is already possible if two different types are observed at a position (under the condition that further constrains concerning the meanings are satisfied). Higher type frequency may yield comparatively higher productivity in resulting slots since more lexical units are grouped initially, but an increase in productivity of a slot may also be achieved through merging of slot-filling sets.

3.5.3. Cross-situational learning

Because we attempted to enable our model to – like a child – learn language under referential uncertainty, we combined learning mechanisms proposed within the usage-based account to language acquisition with the idea of cross-situational learning, and we have shown how the model was able to establish correct form-meaning pairings in the presence of referential uncertainty. To model cross-situational learning, we applied associative networks and hence – in line with supporting evidence from psycholinguistic experiments (e.g. Yu & Smith, 2007; Smith et al., 2011) and several previous computational models addressing cross-situational learning (cf. Section 2.5) – our model implements the learning mechanism basically by tracking co-occurrence statistics between all *nl*s and *mr*s in a given ambiguous scene. Moreover, it incorporates a disambiguation bias. However, recently some research has also reported evidence that human learners may form a single "best" hypothesis only, i.e. they track one referent per word, which they test until it is disconfirmed, hence indicating a "fast mapping" procedure rather than a gradual, statistical one (Medina et al., 2011; Trueswell et al., 2013). While evidence has been found with respect to both accounts it remains rather difficult to unify results since studies differ along several dimensions such as methodology or stimuli. Moreover, cross-situational learning studies are often performed with adult participants who may apply learning mechanisms children do not (yet) have access to. In particular, at least very early on it is almost necessary that learning is associative since there is little lexical knowledge available which may facilitate other learning mechanisms (McMurray et al., 2012). For instance, recall from Section 3.4.6 that children's disambiguation skill appears to evolve between the age of 18 and 30 months (Bion et al., 2013). Investigating the underlying learning mechanisms can also be approached from a modeling perspective. In particular, as mentioned in Section 2.5, Kachergis et al. (2012b) fitted two different models, i.e. an associative one maintaining approximately all co-occurrences which incorporates competing familiarity and uncertainty biases (Kachergis et al., 2012a) and one maintaining a single hypothesis only, to data obtained from a cross-situational learning task with human subjects, and found that the human learning curves were better fitted by the associative model.

An interesting point for future work would be a more detailed analysis of the model's abilities to use cross-situational statistics, in particular by comparing its behavior to that observed for human learners in cross-situational learning tasks. While with respect to modeling child language acquisition applying an associative mechanism appears to be appropriate, further cross-situational learning mechanisms could be incorporated and compared as well as further learning mechanisms, e.g. making use of social cues. Moreover, it would be interesting to see how the amount of referential uncertainty affects the model's speed of learning because, for instance, evidence exists that a higher degree of referential uncertainty amounts to a slower rate of word learning in humans (Smith et al., 2011). However, with respect to the current work the focus is not on comparing different cross-situational learning mechanisms. Rather, our goal is to explore how such a mechanism can be combined with a rule learning mechanism and on showing how the same mechanism can be applied consistently at different levels. In particular, in contrast to previous computational models (cf. Section 2.5), which formalized the idea of cross-situational learning typically by focusing on modeling correspondences between single words and referents, we explored how cross-situational statistics can additionally be utilized in the process of learning syntactic patterns. Applying this mechanism beyond exploiting such simple mappings appears to be particularly interesting since recent work (Scott & Fisher, 2012) has shown that 2.5-year-old children are able to use cross-situational statistics to infer verb meanings under referential uncertainty, even if this requires abstraction across different actors and objects, and the findings thus suggest that children attach information about possible referents to novel verb entries along with their co-occurrence statistics and refine this information across trials. This behavior can be reflected by our model in the sense that after observing an *NL* utterance containing a novel verb, the model can set up an entry for the verb which captures some information about possible referents and co-occurrence statistics, and it is able to refine this entry over time. Such an entry would be of the form of an *NL* (pattern) with information about possible referents being stored by means of sets of slot-filling elements. In the following chapter, we will hence explore cross-situational verb-learning as well as the establishment and representation of early verb entries with the model.

3.5.4. Limitations and possible extensions

In this chapter, due to focusing on modeling early language acquisition – and in particular on the emergence of verb-specific slot-and-frame patterns –, we excluded and simplified several aspects and learning mechanisms assumed to be involved in language acquisition. Therefore, the model's language abilities – like those of an infant – are limited, and several aspects of language cannot be learned by the model in its current version but may be addressed in extended versions of the model. For instance, in our model, the meaning of word and *S&F* constructions always corresponds to actions or referents grounded in the visual domain, and therefore several words and utterances cannot be learned by our model (recall, however, that our model also possesses the ability to learn that a certain form does not have a meaning in an examined domain), e.g. expressions being grounded in the time domain or referring to inner states, e.g. emotions. However, the model may be extended in order to handle language which is not grounded in the visual domain, and it would be interesting to investigate whether similar learning mechanisms can be applied. Furthermore, as already discussed previously, it would be interesting to transfer the incorporated learning mechanisms to the induction of further types of constructions, and in particular to more abstract ones. Additionally, aiming at further improvement of the model's ability to learn language under referential uncertainty, it could be equipped with the ability to evaluate social cues such as pointing gestures.

According to the measures for rating the model's knowledge defined in Section 3.3.4, we consider only a single semantic referent as the learned meaning of an *nl*. This reflects the RoboCup corpus which we used for evaluation. That is, in this corpus a word's meaning has at most one corresponding referent. While due to this fact we did not consider homonyms, these can be captured by the associative networks simply by not requiring a single best meaning, but taking all referents over a certain threshold into account; the rating threshold may be utilized for this purpose. By contrast, synonyms were taken into account during language understanding, e.g. the model can map both "pink goalie" and "pink1" onto the the same referent *pink1*.

Our strategy for merging sets of slot-filling or linguistically optional elements may be somewhat greedy because sets having just one element in common are merged. One may argue that by applying such a greedy strategy the model may learn several erroneous patterns in the sense that patterns can contain slots in which elements appear that yield semantically ill-formed utterances. Imagine for instance that the model has learned two form-meaning pairings ("the SE_1 reads a book", *read(AGENT)*) and ("the SE_1 sleeps", *sleep(AGENT)*) with $SE_1 = [boy, girl]$, and then observes a form-meaning pairing ("the dog sleeps", *sleep(AGENT:dog)*). The model may then group "dog" into SE_1 because it appears at the position of SE_1 in pattern "the SE_1 sleeps", thereby learning the meaning of an utterance "the dog reads a book" which is semantically ill-formed in the sense that typically, i.e. in the real world, dogs cannot read. Notice, however, that this is to some extent a desired behavior because we only ask the model to parse utterances into their corresponding meaning representations and not whether the resulting *mr*s are semantically well-formed (in the real world or some domain). In particular, like a human, our model is able to derive a meaning *read(AGENT:dog)* for the presented utterance. Humans are furthermore able to decide whether *read(AGENT:dog)* is semantically well-formed which depends on the underlying domain, e.g. in the real world dogs cannot read but in a cartoon a dog may be able to read. While our model currently does not incorporate such world or domain knowledge, it would be interesting to extend the model in this direction. For instance, instead of modeling objects by single words they may furthermore be specified by sets of attributes such as "human". In case of the previous example, the model could then learn that the pattern "the SE_1 reads a book" only yields a semantically well-formed mr in the real world in the case where SE_1 is filled with an element possessing the attribute "human". However, using a greedy strategy is – at least to some extent – advantageous because it may lead to the emergence of more abstract knowledge and in particular parts of speech. For instance, in the long run, among others, a set grouping all nouns may be established by applying greedy merging. An interesting aspect for future work may be the comparison of different strategies for merging sets of elements. A promising starting point may be to utilize a greedy strategy – because this may yield the emergence of parts of speech – and moreover incorporate a strategy based on semantic attributes which either groups elements having certain attributes in common that are required by a slot to yield a semantically well-formed meaning, or which simply identifies the relevant attributes and associates them with a slot in order to model selectional restrictions of predicates.

The employed generalization processes may also be used to further induce more abstract constructions. In particular, in the current implementation slot-and-frame patterns are specific to individual sets of slot-filling elements, i.e. it is restricted which lexical units can occur in a slot of a pattern. As stated previously, this limitation may be overcome in the long run because parts of speech may emerge by applying a greedy strategy for merging sets of elements. However, this may also be achieved faster by further generalizing the learned *S&F* constructions to a form where slots are not represented by sets of slot-filling elements but only by variables mapping to the corresponding argument position in the associated predicate. These could in turn be grouped into more abstract constructions, e.g. all constructions of the form ("SE_1 VE_1 SE_2",$ACTION(ARG_1, ARG_2)$), $SE_1 \rightarrow ARG_1$, $SE_2 \rightarrow ARG_2$, $VE_1 \rightarrow ACTION$ might be grouped into an abstract verb-construction representing the SVO sentence structure. Moreover, words might be modeled as sequences of morphemes (by a morpheme order graph) with sets of slot-filling elements representing modifications such as the inflection of verbs. While in this chapter we focused on verb-specific construction due to taking the verb-island hypothesis into account, in the following chapter we will present an extension of the model to learn verb-general constructions.

While the focus of this thesis is on language understanding, the model was designed to both understand and generate language. Further work may evaluate language generation, particularly in combination with language understanding concerning the model's behavior in dialogues. The linear design of our model in case of establishing associations between form and meaning may allow the development of user-adaptive dialogue systems, specifically with an additional extension of a working memory component. A working memory component may be designed by rapidly increasing the weights for recent observations with a subsequent decay. Due to the linear design, the model would then be biased to prefer lexical units and frames recently used by the user to express meanings for which competing alternatives exist. This may yield a user-adaptive artificial agent in the sense that the agent would mimic alignment (Pickering & Garrod, 2004, 2006), i.e. adapt to the user's way of speaking in case of lexical units and frames, and thus the conversation with the agent might be experienced as more pleasant by the user. Moreover, as already mentioned in Section 3.4.6, exploring different memories and their interaction may also be interesting with respect to computational investigations concerning short- vs. long-time fast mapping effects on word learning and modeling forgetting processes.

Unlike children, our model is trained using symbolic input only (i.e. sequences of words and formulas in predicate logic) because we attempted to model a stage in language acquisition in infants where slot-and-frame patterns emerge gradually. As a simplification for our model, we assumed that at this stage of learning the child is already able to extract words from the speech signal as well as information in some structured form, i.e. formulas in predicate logic, from the visual context. The main direction for future work is therefore getting rid of the symbolic input in both cases. In particular, in contrast to the current symbolic semantic representations which are not grounded in the sense of Harnad (1990), perceptually grounded representations of meaning, such as image or cognitive schemas, may be derived from the visual context and utilized in the model instead. Moreover, the speech signal may be used to acquire words, possibly by first deriving smaller units, e.g. phonemes and/or syllables, and then using these along with the information derived from the visual context to bootstrap words. Towards this end, addressing this issue we have already collected a multimodel corpus designed with the main goal to allow the evaluation of computational models addressing the acquisition of rather complex grounded linguistic structures, i.e. syntactic patterns, from sub-symbolic input (Gaspers, Panzner, et al., 2014). Further, we will address learning from speech and concurrent context information in symbolic form, albeit not with respect to modeling child language acquisition, in Chapter 5 of this thesis.

3.6. Summary

In this chapter we have presented a computational and formal model for the gradual emergence of verb-specific slot-and-frame patterns. In the model, linguistic knowledge is represented in form of an interrelated network which comprises constructions at varying degrees of complexity and abstraction. Constructions are induced based on observing ambiguous input, i.e. natural language utterances (presented as sequences of words) observed in an ambiguous context represented symbolically using first-order predicate logic formulas.

We modeled the acquisition of two types of constructions: (short sequences of) words and their meanings and bottom-up induced verb-specific slot-and-frame-patterns. For all levels of constructions, our model proposes uniform representational devices and learning mechanisms to determine appropriate meanings out of the ambiguous contexts. Specifically, all correspondences between form and meaning are modeled by associative networks; measurement of the linguistic knowledge captured by the model is determined based on the weights of connections contained in those networks. In the scope of our language learning algorithm, observed *NL* utterances are first incorporated into the network as a whole. Once sufficient knowledge is regarded as learned, the model starts to gradually induce slot-and-frame patterns. Specifically, the model searches for *NL* utterances and already (partially) generalized patterns representing the same pattern. Roughly speaking, this is the case if the *NL*s under consideration show minimal variation in the surface structure, i.e. varying elements in one position, and these elements are represented as a set of elements corresponding to a slot.

Our proposed model is in line with usage-based psycholinguistic theories stating that in early language acquisition children maintain an inventory of lexically-specific and item-based constructions which are gradually generalized by replacing concrete lexical items by slots which can be filled by (a restricted group of) words or short sequences of words. In particular, it is represented as an interrelated network of constructions at varying degrees of complexity and abstraction without assuming precoded linguistic knowledge. Knowledge emerges gradually from specific words over partially productive slot-and-frame patterns through to fully productive patterns.

We provided empirical results on the RoboCup dataset showing that the employed learning and generalization mechanisms are appropriate in order to i) generalize beyond specific examples seen, while ii) not overgeneralizing and iii) assess confidence in the acquired knowledge accurately. In our experiments, the model was shown to be highly precise, and it achieved a large reduction in the number of stored patterns compared to the number of individual *NL* utterances observed in the input data. This in turn yielded understanding of several novel utterances, i.e. utterances not observed in the input. In our model, In line with findings from psycholinguistic studies with infants in the framework of usage-based theories, language learning proceeds gradually. Initially, in our experiments the model's generalization abilities – i.e. understanding of novel utterances – were limited, but increased over the time course and finally converged, suggesting that the employed mechanisms allow accurate learning without (severe) deterioration of knowledge already captured by the network during further processing of examples. Taken together, our model thus yielded a compact and precise representation of the input data which generalized well to unseen data. The model provides an interesting framework for future research because it can be utilized for experiments aiming at investigations concerning the mechanisms at play during language acquisition.

Chapter 4

From verb-specific to verb-general constructions

In the previous chapter we have presented a computation model based on the verb-island hypothesis for the early acquisition of verb-specific constructions. Since recent psycholinguistic findings provided novel insights concerning the representation and emergence of early verb entries, in this chapter we extend our computational model to also learn verb-general constructions and investigate how psycholinguistic findings concerning the acquisition of verbs can be captured by the model.
Work presented in this chapter has been presented previously in Gaspers, Foltz, & Cimiano (2014).

4.1. Introduction

Unlike nouns, verbs describe actions that involve a number of participants who play certain (thematic) roles in the event. Hence, sentence structure, i.e. syntactic frames, may serve as a "zoom lens" to guide the child's attention to relevant aspects of verb meaning, in particular to thematic relations during verb learning (e.g. Gleitman & Fisher, 2005). In line with this assumption, Arunachalam & Waxman (2010) showed that 27-month-old children can create an initial verb entry based on information from the syntactic context without access to any corresponding visual information, and retrieve this information when encountering the verb later on. For instance, when hearing a verb in transitive syntax they can establish an initial entry based on syntactic information, and they can retrieve this entry on encountering a candidate causative event. Thus, children are able to do something like fast mapping verbs, i.e. to quickly set up an initial entry, based on syntactic information alone, which may, however, be incomplete (Arunachalam & Waxman, 2010).

As mentioned previously, Scott & Fisher (2012) provide evidence that 2.5-year-old children are also able to use cross-situational statistics to infer verb meanings under referential uncertainty. In particular, the authors showed that children can abstract across different actors and objects, suggesting that they can attach information about possible referents to novel verb entries along with their co-occurrence statistics and refine these entries over time.

However, what remains unclear is

1. how verb-general constructions emerge and how they are represented,
2. how they can guide attention to establish verb entries based on syntactic information alone,
3. how information about possible referents and co-occurrence statistics might be stored with verb entries, and
4. how this information is updated incrementally over time, thus allowing for learning of verb meanings across situations.

In order to shed light on the potential learning mechanisms involved in early verb acquisition, in this chapter we extend the computational model presented in the previous chapter to also acquire verb-general constructions by exploiting the same basic learning mechanisms as those explored earlier for the induction of verb-specific constructions. In particular, verb-general constructions are learned bottom-up based on verb-specific constructions only once verb-specific knowledge has been derived with sufficient confidence. Again, generalization occurs in an item-based fashion (albeit with respect to more complex structures/mappings) by searching for variation at the linguistic layer which has corresponding variation at the meaning layer.

We present empirical results replicating psycholinguistic experiments performed by Arunachalam & Waxman (2010) and Scott & Fisher (2012) with our model. Depending on its "age", the model behaves very similarly to the children in these studies. Thus, the results suggest possible learning mechanisms implicated in the early acquisition and representation of verbs and verb-general constructions.

The remainder of this chapter is organized as follows. Next, we will present how the learning problem and thus the model are modified in order to induce verb-general constructions. Subsequently, we will present empirical results, replicating findings from psycholinguistic studies with children and discuss our results with respect to these studies.

4.2. Learning problem

In this chapter, we aim to extend the existing model, which is able to learn verb-specific constructions, to also learn verb-general constructions. The input data remains the same as for the model learning verb-specific constructions; for a detailed description see Section 3.2.1. The goal also remains the same in that we attempt to induce a constructions grammar which is represented in the form of an interrelated network and this network is acquired incrementally over time. However, we additionally address the acquisition of a further type of construction, i.e. verb-general slot-and-frame patterns. These are learned bottom-up based on verb-specific constructions at the level of slot-and-frame patterns, thus yielding a further generalization of linguistic knowledge contained in the network. With respect to the representation of verb-general constructions in the network, we introduce another type of sets of elements, i.e. *sets of predicate-filling elements*, and we define them as groupings of lexical units mapping to the predicate $ACTION$ of an associated verb-general semantic frame. Analogous to verb-specific constructions, the form $\hat{NL}$ constitutes an *NL* (pattern), and the meaning $\hat{mr}$ is represented by exactly one semantic frame $\llbracket mr \rrbracket$. However, in case of verb-general constructions we also require that $\hat{NL}$ contains a set of predicate-filling elements which maps to the $ACTION$ predicate in the semantic frame. Verb-general semantic frames are derived from verb-specific ones by replacing the concrete predicate by $ACTION$ during generalization. Moreover, since verbs map to actions taking nouns, another goal is to learn "preferred"/correct syntactic frames for given verbs and hence possible verb argument structures.

To illustrate how verb-general constructions emerge from verb-specific ones, consider the following example which has been presented previously when introducing the learning problem for the model learning verb-specific constructions (Section 3.2.2):

(4.1)

$\hat{NL}$	SE_1 sees SE_2
$\hat{mr}$	*see(AGENT,THEME)*
Φ	$SE_1 \rightarrow AGENT$ $SE_2 \rightarrow THEME$

where

SE_1 = [Mia → mia, Tim → tim],
SE_2 = [pizza → $pizza$, cake → $cake$].

Now consider the following input example:

(4.2)

NL:	Tim takes pizza
mr_1:	*take(AGENT:tim, THEME:pizza)*
mr_2:	*fetch(AGENT:mia, THEME:cookie)*
mr_3:	*see(AGENT:mia, THEME:dog)*

Then, we would like to induce the verb-general construction:

(4.3)

$\hat{NL}$	$SE_1\ VE_1\ SE_2$
$\hat{mr}$	$ACTION(ARG_1, ARG_2)$
Φ	$SE_1 \rightarrow AGENT$ $SE_2 \rightarrow THEME$

along with SE_1 and SE_2 as listed previously and the following set of predicate-filling elements

VE_1 = [takes → *take*, sees → *see*]

mapping to *ACTION*. Moreover, we would like to capture the fact that "takes" and "sees" where observed within the syntactic pattern "$SE_1\ VE_1\ SE_2$", i.e. to model the fact that "takes" and "sees" can be expressed using this verb-general syntactic pattern.

For reasons of clarity and since it is not relevant for the experiments presented in the following, we do not explore syntactic generalization in the framework of the extended model. However, the syntactic generalization step is implemented in the extended model and can in principle be applied with it.

4.3. The extended computational model

In order to extend our computational model, we investigate the same basic learning mechanisms and representations of knowledge as explored for the induction of verb-specific constructions. In extending the model, all of its properties listed in the previous chapter are retained.

4.3.1. Network structure

In order to address argument structure acquisition, we need to model correspondences between verbs and verb-general syntactic patterns. To do this, we augment the network structure by including an additional associative network. Specifically, an associative network A_{Verb} is incorporated into the network structure, where the

form layer $A_{Verb(NL)}$ corresponds to all lexical units which were observed at a position of a set of predicate-filling elements – i.e. elements which were observed as expressing a predicate; these are assumed by the model as being verbs. The meaning layer $A_{Verb(MR)}$ corresponds to all verb-general syntactic patterns, modeling associations between specific verbs, i.e. lexical units, and syntactic frames. In this way, the network captures the syntactic frames in which a verb is likely to appear.
In the network, sets of predicate-filling elements are modeled analogously to sets of slot-filling elements as nodes which, in turn, group nodes referring to lexical units.

4.3.2. Generalization and confidence

In order to induce verb-general constructions based on verb-specific ones, a further criterion for merging paths is needed. We address this issue in a similar manner as we did when inducing verb-specific constructions: patterns are potentially mergeable if they differ in at most one position and exchange of elements observed at this position yields a corresponding change in an associated meaning. However, since our goal is to induce verb-general constructions, we consider variation with respect to predicates instead of variation with respect to argument slots. To illustrate the intuition behind the proposed generalization mechanisms, consider the following two examples:

(4.4)

$\hat{NL}$	SE_1 $\underline{\text{eats}}$ SE_2
$\hat{mr}$	$\underline{\textit{eat}}$*(AGENT,THEME)*
Φ	$SE_1 \rightarrow AGENT$ $SE_2 \rightarrow THEME$

(4.5)

$\hat{NL}$	SE_1 $\underline{\text{takes}}$ SE_2
$\hat{mr}$	$\underline{\textit{take}}$*(AGENT,THEME)*
Φ	$SE_1 \rightarrow AGENT$ $SE_2 \rightarrow THEME$

One can easily infer that these two verb-specific constructions can be merged into the following verb-general construction:

(4.6)

$\hat{NL}$	SE_1 VE_1 SE_2
$\hat{mr}$	$ACTION(AGENT,THEME)$
Φ	$SE_1 \rightarrow AGENT$ $SE_2 \rightarrow THEME$

assuming that “eats” and “takes” mean *eat* and *take*, respectively. We thus define a mergeability criterion with respect to predicates as follows.

Definition 9 (Predicate-driven mergeable)**.** Two paths p_1 and p_2 are *predicate-driven mergeable* if both differ in at most one position and both already have a learned (verb-specific or verb-general) meaning including the same mapping between slots in syntactic patterns and argument slots in semantic frames. In the case of a position with varying elements, each of these elements must either i) correspond to a set of predicate-filling elements or ii) have a learned meaning which, in turn, must be the predicate of an associated semantic frame.

The condition that elements at varying positions may also correspond to sets of predicate-filling elements allows the model to directly merge verb-specific with verb-general paths. Hence, for instance, a path "SE_1 VE_1 SE_2" can be merged with another path "SE_1 sees SE_2", assuming that a meaning for "sees" has previously been learned and that the corresponding mappings are alike.
We explore the same measures of confidence in the acquired linguistic knowledge as we did for the induction of lexical and verb-specific constructions (see Section 3.3.4). As previously done, measures are again described based on the weights stored in the associative networks. We do not apply the entropy criterion in the experiments presented in this chapter because of its small contribution to performance in the experiments presented in the previous chapter (cf. Section 3.4.5). This is again done for reasons of clarity; the entropy criterion is implemented in the extended model and can in principle be applied analogously to the model presented previously. With respect to rating knowledge in A_{Verb}, for a given verb, semantic frames are rated just like meanings are rated for given *nl*s without sets of slot-filling elements. That is, we utilize the weights provided by A_{Verb} (cf. Equation 3.20) for this purpose. Verb-general constructions are rated just like verb-specific constructions by applying Equation 3.21, which is based on the weights stored in $A_{S\&F}$ and the weights in a corresponding mapping associative network.

4.3.3. Language learning

Recall that the language learning algorithm proposed within the model presented in the previous chapter was roughly composed of three different learning steps, where the learning steps built onto each other, and were applied to each observed example. In order to extend language learning towards the induction of verb-general constructions a further learning step is introduced, yielding the following four learning steps:

1. Training of the word layer CON_{Word}, acquisition of word level constructions, i.e. lexical units and their meaning

2. Training of the slot-and-frame pattern construction layer $CON_{S\&F}$, generalization step which searches for sets of slot-filling elements
3. Training of the slot-and-frame pattern construction layer $CON_{S\&F}$, generalization step which searches for sets of linguistically optional elements
4. Training of the slot-and-frame pattern construction layer $CON_{S\&F}$, generalization step which searches for sets of predicate-filling elements

Again, the learning steps are applied to each example. That is, whether generalization is possible is inspected for each case, and if so, it is performed accordingly. Because predicate-filling generalization requires learned knowledge at the level of verb-specific constructions, our approach to language learning remains incremental in the sense that results obtained at a previous learning step are exploited within subsequent learning steps. Since we do not explore syntactic generalization in this chapter, three basic learning steps are explored in the following. They are detailed in Algorithm 2.

Recall that the slot-driven generalization step yields a path p' as a result which corresponds either to i) a path representing a preprocessed input utterance (i.e. a sequence where lexical units contained in sets of slot-filling elements are replaced by the set IDs), or ii) a new generalized path – if slot-driven generalization has been possible and performed accordingly. Whichever applies, the resulting path is the input to the predicate-driven generalization step. Given p', the model searches for paths which are predicate-driven mergeable with it. If such paths exist, they are merged with p' into a novel path which is inserted into the network structure. Elements at a varying position are replaced by a node referring to a set of predicate-filling elements, and – if not yet present – a verb-general semantic frame is inserted into the network where the concrete predicate is replaced by *ACTION*. Similar to the other generalization steps, merging of paths includes merging of weights in corresponding associative networks as well as merging of corresponding mapping associative networks. For instance, imagine that the construction depicted in Fig. 1 is represented in the network corresponding to the network state already presented in Fig 9 in Section 3.3.5 with the exception that associations with respect to the mapping have been acquired. Imagine further that the construction presented in Fig. 2 is also stored in the network. Merging those two verb-specific constructions into a verb-general one should yield the network state shown in Fig. 3.

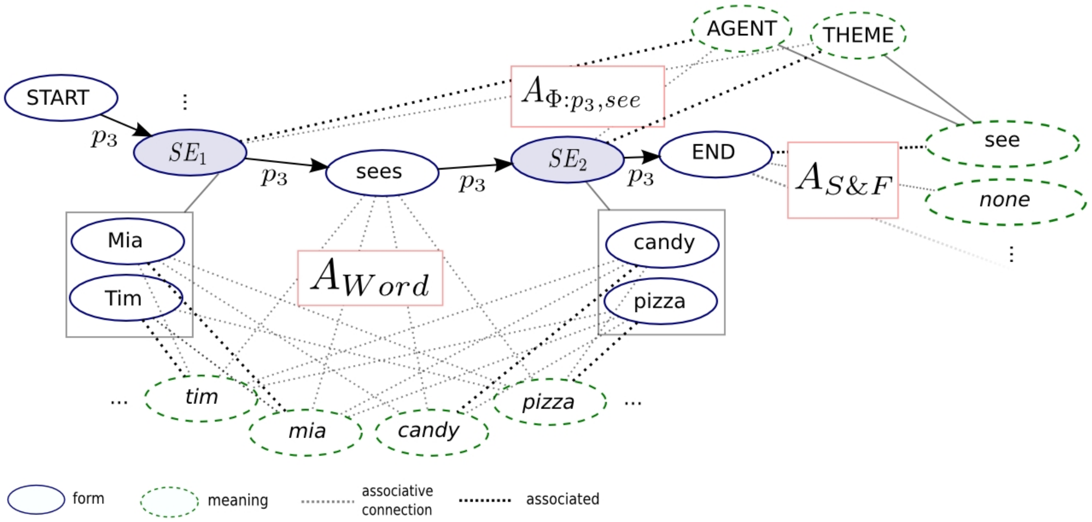
Subsequently, A_{Verb} is updated, independently of whether verb-general merging is possible. This procedure is similar to the one updating the mappings. An *NL* is first preprocessed in that lexical units contained in sets of slot-filling elements are

Algorithm 2 Extended language learning algorithm

```
Input: A list of examples $E = \{(NL_1, MR_1), \ldots, (NL_k, MR_k)\}$
Output: A network $N$ representing constructions

  $N$ = an empty network
  for all examples $(NL_i, MR_i) \in E$ do
    1. update $\boldsymbol{CON_{Word}}$
    ○   $units \leftarrow$ extract all lexical unit types from $NL_i$
    ○   $referents \leftarrow$ extract all semantic referent types from $MR_i$
    ○   add new lexical units and referents, incorporate and initialize connections
    ○   update associations between $units$ and $referents$ by
        $update(A_{Word}, units, referents)$
    2. update $\boldsymbol{CON_{S\&F}}$, slot-driven generalization step
    ○   $s$ = preprocess $NL_i$
    ○   $P_m = \{p_1, \ldots, p_{|P_m|}\} \leftarrow \{p \mid p \in CON_{S\&F(NL)}$ and $p$ is slot-driven mergeable
        with $s\}$
    ○   if $P_m \neq \emptyset$
              $p' = p_1 \oplus \cdots \oplus p_{|P_m|} \oplus s$
        else
              $p'$ = new path corresponding to $s$
        end if
    ○   $mrs \leftarrow$ extract all semantic frames from $MR_i$
    ○   incorporate each $mr \in mrs, mr \notin CON_{S\&F(MR)}$ into $CON_{S\&F(MR)}$
    ○   incorporate $p'$ into $CON_{S\&F(NL)}$, add and initialize connections
    ○   update associations between $p'$ and $mrs$ by $update(A_{S\&F}, p', mrs)$
    ○   update all mappings between $p'$ and $mr$ templates
    ○   merge identical paths
    3. update $\boldsymbol{CON_{S\&F}}$, predicate-driven generalization step
    ○   $P_m = \{p_1, \ldots, p_{|P_m|}\} \leftarrow \{p \mid p \in CON_{S\&F(NL)}$ and $p$ is predicate-driven merge-
        able with $p'\}$
    ○   if $P_m \neq \emptyset$
              - $p'' = p_1 \oplus \cdots \oplus p_{|P_m|} \oplus s$
              - incorporate $p''$ into $CON_{S\&F(NL)}$
              - incorporate verb-general semantic frame
              - add and initialize connections, update associative networks
        end if
    ○   merge identical paths
    ○   Update $A_{Verb}$
  end for
```

Figure 1.: Example of a verb-specific construction referring to the predicate *see* stored in the network.

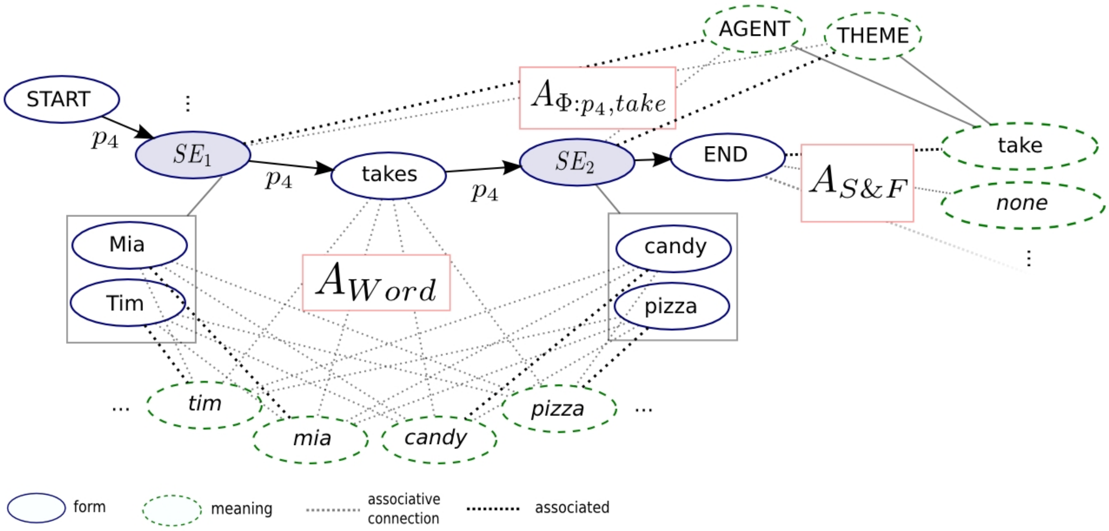


replaced by the set IDs. Given the resulting sequence, the model searches for paths with a learned verb-general meaning which differs from the observed sequence only at the position of a set of predicate-expressing elements, for instance, a path $p =$ "SE_1 VE_1 SE_2" in case of a modified NL "SE_1 eats SE_2". Then, the appearance of the lexical unit – which is regarded as a verb because it appears at a verb position – within the syntactic pattern represented by the path, is captured by training the associative network A_{Verb} using the lexical unit and the path, for instance, "eats" with path p in case of the above example. This establishes a correspondence between the lexical unit and the syntactic frame.

4.3.4. Retrieval of constructions and syntactic frames

A meaning for a given NL utterance corresponding to a verb-general construction is retrieved analogous to the case of verb-specific constructions. Just like described previously, as a first step the NL utterance is preprocessed, i.e. all lexical units contained in sets of elements – including lexical units contained in sets of predicate-expressing elements – are replaced by the set IDs. Afterwards, a corresponding path can be determined in the graph and – if existent – possible meanings can be rated based on Equation 3.21. If a corresponding semantic frame exists, the final meaning is constructed by retrieving meanings for lexical units at positions of sets of elements from A_{Word}. In case of sets of predicate-expressing elements, the meaning is inserted

Figure 2.: Example of a verb-specific construction referring to the predicate *take* stored in the network.



at the predicate position *ACTION* of the associated verb-general semantic frame. Given a concrete verb, i.e. a lexical unit, an associated syntactic pattern – if existent – can be retrieved based on the weights provided by A_{Verb} according to Equation 3.20 just like in case of finding a meaning for lexical units or *NL*s without sets of slot-filling elements. A verb-general semantic frame associated with this pattern can in turn be found as described previously by rating all possible semantic frames according to Equation 3.21. By this, the model is able to determine corresponding argument structures for given verbs.

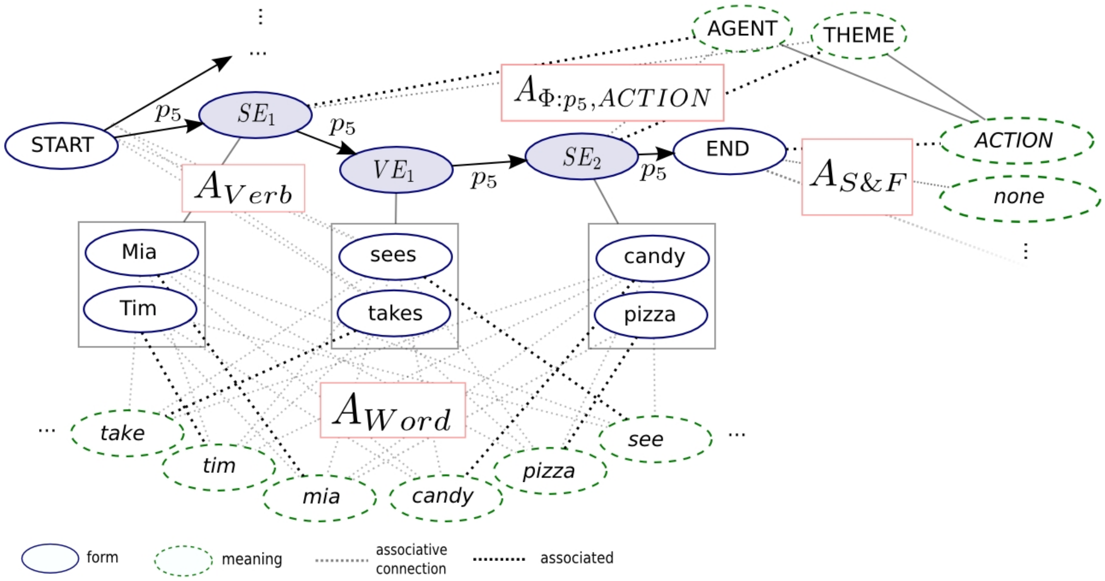
4.4. Experimental evaluation and discussion

In order to allow the evaluation with respect to psycholinguistic studies, our model needed some initial linguistic knowledge, just like the children in the studies. Next, we describe how input data were generated. Then, we present experiments with respect to psycholinguistic findings.

4.4.1. Input data

Input data were generated in a similar manner as described by Alishahi & Stevenson (2008), using the Eve corpus from the CHILDES database (Brown, 1973). The corpus contains transcriptions of interactions with the child Eve. We considered

Figure 3.: Example of a verb-general construction stored in the network.



utterances spoken by Eve's mother. Both of the studies we will consider took into account transitive and intransitive structures only, one including conjoined subjects. Hence, we extracted all patterns of the form "*AGENT verb*" and "*AGENT verb THEME*" from the corpus. We considered the same verbs as Alishahi & Stevenson (2008) but since two of them did not appear in the considered form, only 11 out of the 13 verbs were included into our experiments: *come*, *eat*, *fall*, *get*, *go*, *look*, *make*, *put*, *see*, *sit* and *take*. All patterns along with their occurrence frequencies were inserted into an input generation lexicon. Moreover, with respect to each verb (concrete) nouns appearing at the positions of *AGENT* and *THEME* along with their occurrence frequencies were inserted into the lexicon. Two nouns conjoined by "and" were also included; "me", "you" and "we" were annotated/treated as "Eve", "Mom" and "Mom and Eve", respectively. Based on the lexicon input data, *NL* examples were created by choosing patterns and their referents probabilistically according to the occurrence frequencies stored in the lexicon. Semantic representations *mr* were created automatically, using words appearing in a generated *NL* to denote the corresponding semantic referents. Notice that semantic referents are only arbitrary symbols to the model; it still has to establish connections between words and referents, for instance, learn that a word "eve" refers to the semantic referent *eve*. In the case of an *NL* including a thematic relation consisting of two referents conjoined by "and", these referents were treated as separate arguments in an *mr* having the same thematic relation. Such an example may be: ("mom and eve see",

see(AGENT1:mom,AGENT2:eve)). In the experiments presented in the following, we do not address learning morphology. Hence, all words appear in their root form only.

Ten different datasets containing 500 examples of the form (*NL*,*mr*) were created. These were used for the experiments presented in the following. Presented results are averaged over the ten datasets, simulating ten different learners. Model parameters were optimized on an independent dataset. In particular, a training dataset and a test dataset were created by applying the process also used for creating the ten datasets for the experiments. Parameters were optimized by training the model with varying parameters on the training data and evaluating on the test data. Optimization was performed with respect to the F_1 score, where precision and recall were computed as specified in Chen et al. (2010) and applied previously during the evaluation of the model learning verb-specific constructions.

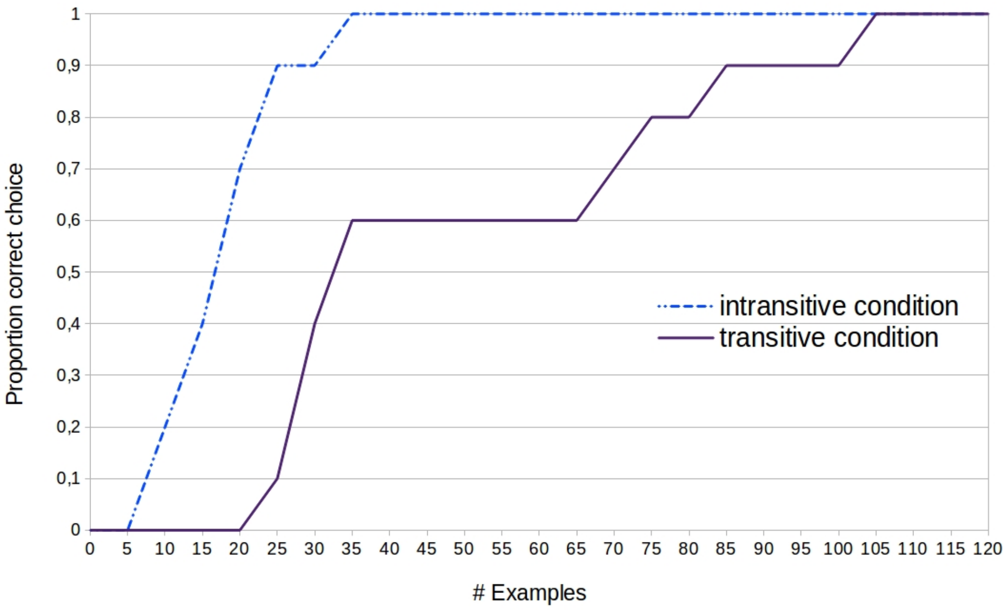
4.4.2. Cross-situational verb learning

As mentioned before, Scott & Fisher (2012) investigated cross-situational verb learning and found that 2.5-year-old children can use cross-situational statistics to infer verb meanings under referential uncertainty, even if this requires abstraction across different actors and objects. This suggests that children attach information about possible referents to novel verb entries along with their co-occurrence statistics and refine this information across trials.

The study performed by Scott & Fisher (2012) investigated learning of both transitive and intransitive verbs. During each of the 12 experimental trials, children heard two transitive or intransitive sentences, each containing a different novel verb, while watching two videos showing two different actors, each performing a novel action. In the transitive condition, the action was performed with different objects. Transitive and intransitive sentences had the structures “she’s pimming her toy” and “she’s pimming”, respectively. Children in the intransitive condition were significantly above chance in choosing the target actions over the distractor actions. Performance in the transitive condition depended on children’s vocabulary size: Only children with large vocabularies performed significantly above chance.

We tested whether our model can infer meanings for novel verbs without receiving unambiguous label trials for any of the verbs. Thus, we tested whether the model can set up verb entries that contain information about possible referents and update co-occurrence frequencies over time. Notice that since we use symbolic input, we cannot investigate the influence of abstraction over different actors and objects at the visual level. We used the same verbs, i.e. “pim”, “nade”, “rivv”, and “tazz”, and pairings of verbs as Scott & Fisher (2012). Referents for verbs were selected from the input generation lexicon (i.e. “mom” and “celery”). Since the model processes one sentence at a time, each input example contained one sentence and two possible mrs. For example, the first two intransitive input examples (which correspond to one trial in the study) were *NL:* “mom pim”; mr_1 *:pim(AGENT:mom)*; mr_2: *nade(AGENT:mom)* and *NL:* “mom nade”; mr_1 *:pim(AGENT:mom)*; mr_2: *nade(AGENT:mom)*. After receiving the examples, the model was asked to retrieve the semantic representations for the novel verbs, e.g. for “mom nade”, and we counted how often the model returned the correct representation, e.g. *nade(AGENT:mom)*. Results were computed for different numbers of examples observed prior to the experimental trials, corresponding to different “ages” of the model. Fig. 4 shows the results. In both conditions, – in line with the children in the experiments – the model can solve the task from a certain “age” on. In addition, it can solve the task earlier in the intransitive condition compared to the transitive condition. Similarly, children had more problems with the transitive condition: several of the 2.5-year-old children failed in the transitive condition. Since typically even 12-to-14-month-old children can master such a task when it involves mapping nouns to objects (Smith & Yu, 2008), Scott & Fisher (2012) concluded that in cross-situational learning the same learning mechanisms may not apply uniformly for words of different categories. However, with respect to possible learning mechanisms at play, our model shows that a behavior similar to the observed one can be produced by applying the

Figure 4.: Proportion of the model’s choice of the correct semantic representation for the novel verbs in the transitive and intransitive sentences.



same mechanism for tracking co-occurrence statistics in case of nouns and verbs – recall that we applied associative networks in all cases –, albeit with respect to more complex structures.

In our model, sentence-/verb-to-action mapping lags behind word-to-object mapping because it involves more complex structures whose acquisition depends on the prior acquisition of less complex structures, i.e. nouns. In particular, in order to establish a mapping for a verb “pim” in a sentence “mom pim celery”, an NL pattern like “SE_1 pim SE_2” must have been derived prior, and “mom” and “celery” must be contained in the sets of slot-filling elements SE_1 and SE_2, respectively. Moreover, a necessary condition for deriving the pattern is that meanings for “mom” and “celery” have been learned. Hence, similar to the children, the model’s ability to solve the task depends on vocabulary, though not on the absolute vocabulary size, but rather on whether the meanings for the words observed at argument positions have already been learned (though, of course, the probability that the needed lexical units have already been learned may be higher for larger vocabularies).

The model learns faster in the intransitive compared to the transitive condition because it must have acquired only one word instead of two words for referents. In addition, patterns containing fewer sets of slot-filling elements are in general learned earlier because the model generalizes based on type variation observed in one position. Notice, however, that we do not claim that children learn “mom” or “celery” at a specific age; these words were chosen arbitrarily for our experiments because they appear in our input data. Notice further that in contrast to the following experiment, the model can solve the above task even without the proposed extension.

4.4.3. Syntax as a zooming lens into semantics

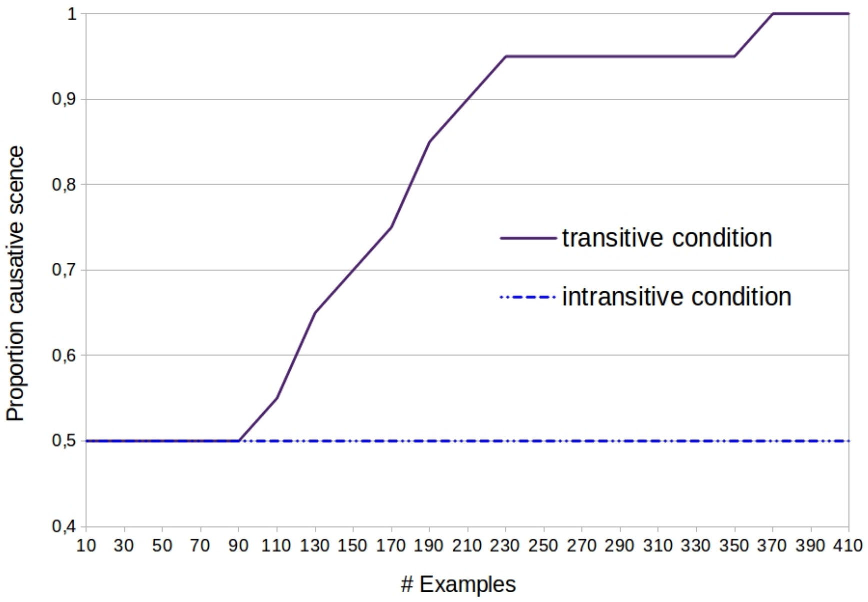
As mentioned before, experimental findings suggest that children at the age of 27 months can set up an initial verb entry based on a purely syntactic context and retrieve this entry when encountering the verb later on (Arunachalam & Waxman, 2010). We tested the model’s ability to do so by replicating the experiment performed by Arunachalam & Waxman (2010).

In the study, toddlers performed two training trials involving known verbs to get familiar with the task and four experimental trials involving different novel verbs. Each verb was presented in the framework of a dialogue and appeared eight times without accompanying visual information. Experiments were performed for using verbs in one of two conditions at a time: in transitive sentences (e.g., “the lady mooped my brother”) or in conjoined-subject intransitive sentences (e.g., “the lady and my brother mooped”). On each experimental trial, toddlers viewed two different scenes side-by-side depicting the same two participants: one synchronous scene (i.e. two persons were performing the same action) and one causative scene. Toddlers were then asked to find the novel verb without providing syntactic information, e.g. they were asked to "find moop". The results revealed that toddlers in the transitive condition were more likely to choose the causative scene than those in the intransitive condition, and moreover they performed significantly above chance. By contrast, those in the intransitive condition did not differ significantly from performing at chance.

We tested the model in a similar manner both in a transitive and intransitive condition. Training trials were omitted, since there was no need to make the model familiar with the task, resulting in four experimental trials. Each trial featured a different novel verb. Just like in the study, each verb was presented to the model eight times in either a transitive or subject-conjoined intransitive sentence (depending on the experimental condition); referents for verbs were chosen from the input data, and the same referents were used in both conditions. Since the experiments did not provide children with concurrent visual information, the model was trained using these sentences without accompanying *mr*s (notice that for utterances presented prior to the experimental simulations, i.e. for the acquisition of initial linguistic knowledge, accompanying *mr*s were provided). Subsequently, for each trial, the model was asked to "find *new-verb*" in the presence of two *mr*s, a causative and a synchronous one. Since toddlers do not stop learning during test periods of experiments either, a learning step was executed with the test input, e.g. with an example (*NL:* "find moop", (mr_1: *moop(AGENT1:mom,AGENT2:eve)*, mr_2: *moop(AGENT:mom,THEME:eve)*. Subsequently, we asked the model to retrieve the *mr* associated with the syntactic frame for the novel verb. Again, we computed results for different numbers of examples observed, corresponding to different "ages" of the model; they are presented in Fig. 5.

As can be seen, the model's behavior corresponds to that of the children in that from a certain "age" on, it picks the causative scene (significantly) above chance in the transitive condition. By contrast, in the conjoined-subject intransitive condition, the model performs at chance, which is also in line with the behavior observed for the children in the study. In the case of our model, the ability to perform this experiment depends on whether a suitable verb-general syntactic frame has been learned prior to the experimental trials. If this is the case, the model behaves similarly to children in that it can create an initial verb entry based on syntactic information alone. The model can also retrieve this information when encountering the verb later on and infer the verb's concrete meaning by relying on its disambiguation bias.

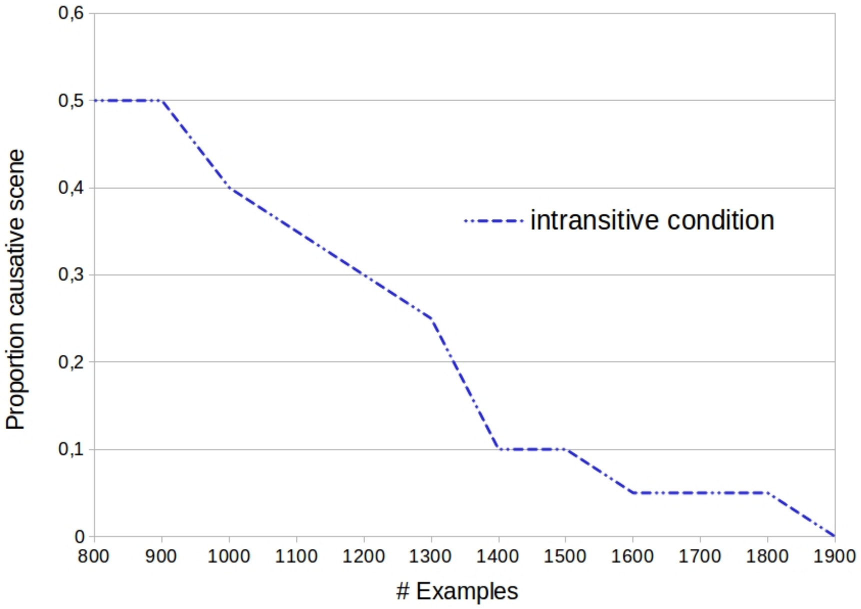
Figure 5.: Proportion of the model's choice of the causative scene in the transitive and conjoined-subject intransitive conditions.



In the case of the model, the explanation for performing at chance in the conjoined-subject condition is that it has not yet derived a corresponding verb-general syntactic pattern based on the given number of examples. In order to test whether the model is in general able to solve the task, we ran the experiments again for the conjoined-subject intransitive condition, albeit over a larger number of input examples. Results are presented in Fig. 6.

The diagram reveals that with a greater number of input examples, the model also performs above chance in the conjoined-intransitive condition, i.e. it chooses the causative scene less often in that case. Thus, the model can solve the task in both conditions, but associates conjoined-subject intransitive with non-causal events at a later "age" than transitive with causal events. This is the case because the model learns the verb-general construction corresponding to conjoined-subject intransitives later than that corresponding to transitives. Similarly, children do not succeed in the conjoined-subject intransitive task until the age of 3;4, even though they can succeed in the transitive task at the age of two (Nobel et al., 2011, but see Sheline et al. (2013)). Since the model acquires both types of verb-general constructions in the same manner, the same proposed learning mechanisms can account for the different results. For the model, this result is due to the input data: The model acquires the conjoined-subject intransitive later because conjoined-subject intransitive sentences

Figure 6.: Proportion of the model's choice of the causative scene in the conjoined-subject intransitive conditions.



appear in the data much less frequently than transitive sentences.

4.5. General discussion

In this chapter, we have presented a computational model for the acquisition of verb-general constructions under referential uncertainty by extending our model for the acquisition of verb-specific constructions. All in all, the model captures the acquisition of different types of constructions, i.e. lexical, verb-specific and verb-general constructions, including verb argument structure acquisition, and learning proceeds in an online fashion in the presence of referential uncertainty. While several models that acquire constructions have been proposed (cf. Section 2.7), including models that address the acquisition of verb-general constructions and verb argument structure (e.g. Alishahi & Stevenson, 2008), they often assume that words or lexical mappings have already been learned and/or do not address learning from ambiguous contexts. However, such learning is relevant for the simulations presented in this chapter since they address the acquisition of verb entries, including the establishment of lexical mappings under referential uncertainty. Yet, it must be noted that our experiments only addressed the acquisition of a few different structures, i.e. transitive and (conjoined-subject) intransitive ones, whereas previous work (Alishahi &

Stevenson, 2008) addressed argument structure acquisition through comprehensive experimental analyses concerning different structures, including recursive ones. This is the case because we model the early emergence of verb-general constructions and representation of early verb entries, and attempt to model some recent psycholinguistic findings regarding this issue which only addressed transitive and intransitive structures. However, in general the model may be able to acquire further structures in the same manner. Future work may reveal whether this is indeed the case.

Several computational models can also use cross-situational learning to establish form-meaning mappings under referential uncertainty (cf. Section 2.5). However, these models have mainly focused on establishing mappings between words and referents, while our model applies the same cross-situational learning mechanism consistently to establish correspondences between form and meaning beyond simple word-referent mappings, in particular, between *NL* patterns/syntactic frames and actions, including thematic relations. Hence, our model can represent verb entries in the framework of these *NL* patterns, allowing it to store information about possible referents with verb entries in addition to associations with possible meanings. Both, information concerning possible referents and co-occurrences with different semantic frames are updated incrementally over time, allowing the acquisition of verb meanings and verb-general constructions. In the learning process, the model first acquires verb-specific constructions, which is in line with the verb-island hypothesis. For instance, early on a verb-specific pattern "*verb* SE_1" might be derived where SE_1 groups possible referents for *verb*, and co-occurrence frequencies for possible predicates/semantic frames such as $verb_1$*(AGENT)* are captured by the associative network $A_{S\&F}$. At this stage of learning, the model is already able to solve the cross-situational verb learning task based on the study performed by Scott & Fisher (2012). Thus, our model indicates that the behavior observed in children can occur if co-occurrence frequencies for verbs are – at least early on – represented and updated in the framework of complete (partially) generalized structures such as *NL* patterns. The patterns' applicability for the cross-situational verb learning task might be dependent on its productivity and on previously acquired nouns appearing as referents with novel verbs.

While the findings from Scott & Fisher (2012) can be replicated by the model without the proposed extension for learning verb-general constructions, this is not the case for the experiments presented by Arunachalam & Waxman (2010). The model can solve the task only once a suitable verb-general syntactic frame has been established. Hence, in case of the Arunachalam & Waxman (2010) experiment, our model indicates that the task can be solved if a verb-general representation has been learned prior to the experimental trials, thus associating the verb with a syntactic frame, e.g. with a causative one in case of a transitive verb. If the verb is encountered later on, a word-to-meaning mapping can be directly established by applying the disambiguation mechanism, thus yielding both a word-to-meaning mapping and a corresponding syntactic frame for the verb. According to our model, a verb-general construction can only be learned if suitable verb-specific constructions have already been acquired previously and can then be merged into a verb-general one.

In both experiments, the model's behavior is in line with usage-based approaches in that it is dependent on the input data, where both token frequency and type variation are taken into account. Similar to the model learning verb-specific constructions, it is not only absolute size of input data that matters. In addition, i) data must provide lexical variation and ii) lexical units and patterns must appear frequently enough to establish form-meaning mappings. To illustrate that it is not only size of input data which matters, consider diagrams 5 and 6. Recall that the diagrams present values averaged over ten different learners, i.e. the model trained on ten different datasets, which have been created using the same input lexicon. The diagrams show that, for the transitive condition, the first learner is able to solve the task at about 110 examples observed, whereas the latest learner needs about 370 examples, which is about three times as much, even though the applied learning mechanisms are exactly the same.

Overall, our results suggest that enough suitable input data in combination with the model's learning mechanisms can model the behavior observed in children, and the model hence provides one possible formal explanation for the observed behavior. While several models that acquire constructions and/or word-to-meaning mappings have been proposed (cf. Section 2), we are not aware of other computational investigations that relate to the findings resulting from Arunachalam & Waxman (2010)'s and Scott & Fisher (2012)'s studies. Experiments with children may establish whether or not children indeed apply learning mechanisms that are similar to those implemented in the model. For instance, cross-situational verb learning can be explored through more detailed analyses of children's vocabularies and by testing children with novel vs. known nouns as referents for verbs.

In this chapter, we have extended our model by exploring learning mechanisms similar to those already investigated for inducing verb-specific constructions. In particular, in both cases generalization is performed based on determining linguistic variation with respect to a slot which yields corresponding variation at a meaning layer based on previously acquired knowledge. In this thesis, we have specified the detection of slots with respect to both inducing verb-specific and verb-general constructions. However, as already mentioned in the previous chapter, similar learning mechanisms may also be applicable to induce further types of constructions, e.g. morphemes. An interesting point for future work would be to investigate whether it is possible to develop a generic learning mechanism which can be applied to induce several types of constructions, one after another. That is, instead of incorporating explicit mechanisms into the model which detect variation with respect to a certain type of construction, a generic mechanism may detect such variation first for a construction type of rather low complexity, generalize accordingly, detect variation with respect to the newly induced more complex constructions, and so on.

4.6. Summary

In this chapter we have presented a computational model for the acquisition of verb-general constructions which builds on our previously described model for the induction of verb-specific constructions. The extended model exploits the same basic learning mechanisms that were explored for the induction of verb-specific constructions. Verb-general constructions are learned bottom-up based on verb-specific constructions only once verb-specific knowledge has been derived with sufficient confidence. Generalization occurs in an item-based fashion (albeit with respect to more complex structures/mappings) by searching for variation at the linguistic layer which has corresponding variation at the meaning layer.

Our model infers form-meaning mappings under referential uncertainty by applying the same cross-situational learning mechanisms at different levels, implemented via associative networks. In particular, in contrast to previous models exploring cross-situational learning, we apply the same cross-situational learning mechanism beyond simple word-referent mappings, i.e. between *NL* patterns/syntactic frames and actions, including thematic relations. Hence, our model can represent verb entries in the framework of these *NL* patterns, allowing it to store additional information about possible referents with verb entries. Both, information concerning possible referents and co-occurrences with different semantic frames are updated incrementally over time, allowing the acquisition of verb meanings and verb-general constructions under referential uncertainty.

We have presented empirical results, showing how the model can establish verb meanings under referential uncertainty. Moreover, we have shown how the model can learn verb-general constructions, and how it can use this knowledge to create initial verb entries based on syntactic information alone, thus suggesting possible learning mechanisms at play concerning the emergence of verb-general constructions and the representation of early verb entries.

Chapter 5

Learning a semantic parser from speech without word transcriptions

In the previous two chapters we have presented and extended a computational model for the early acquisition of syntactic constructions based on a symbolic learning setting. In particular, because we attempted to model a stage where verb-specific constructions emerge gradually, for the sake of simplicity we have assumed that at the modeled stage of learning the child is already able to extract words from a speech signal. Thus, we have explored learning from *NL* utterances in the form of sequences of words coupled with ambiguous context information. In this chapter, we extend this learning setting and address spoken utterances, instead of a symbolic representation of words. We do not assume any predefined lexical knowledge, thus extending the task towards lexical acquisition. To the best of our knowledge, this learning setting has not yet been investigated but is of interest with respect to the design of spoken language understanding systems. Addressing the learning problem, in this chapter we do not focus on modeling child language acquisition, but explore how the learning mechanisms introduced within the framework of the computational model can be extended and applied to tackle the increased complexity of the learning setting with respect to application in spoken language understanding systems. In particular, in contrast to modeling child language acquisition where cognitive plausibility is a main criterion for evaluation, in this chapter we focus on performance. Specifically, with respect to semantic parsing of speech we do not explore online learning. Instead, we assume that a system has the capability to log observed utterances, for instance, to update its linguistic knowledge at certain time

intervals by applying the proposed approach.
Work presented in this chapter has been published previously in Gaspers & Cimiano (2014b).

5.1. Introduction

State-of-the-art SLU systems are typically based on predefined linguistic resources, e.g. lexicons and/or grammars. Building such resources typically requires extensive manual effort, linguistic knowledge and/or (labeled) in-domain training data, making them costly to produce. Further, such systems are often out-dated rather quickly during application, since one cannot know at design time which linguistic knowledge is needed during applications, e.g. which words a user may utter. Moreover, natural languages simply do not have fixed vocabularies. By contrast, children are able to learn linguistic structures by being exposed to language in some context or environment, and they continuously adapt their knowledge, e.g. acquire novel lexical entries over time. Exploring systems which i) learn language similarly to children directly from examples of spoken language utterances coupled with non-linguistic information (e.g. describing the environment) and ii) rely on as few predefined resources as possible can inform SLU concerning the design of self-adaptive and low recourse systems.
In the previous chapters, we have already explored learning language from natural language utterances coupled with non-linguistic context information. In doing so, we have investigated a semantic parsing task (cf. Chapter 3) which, as mentioned previously, has been explored in NLP in order to reduce manual effort and costs for training semantic parsers as a step towards building machines which can learn language – analogous to children – through exposure to language in some environment (Chen & Mooney, 2008). While several approaches have addressed this learning setting (e.g. Chen et al., 2010; Börschinger et al., 2011; Chen & Mooney, 2008), including our computational model presented previously, these have addressed learning from text, not speech. In this chapter, we explore the same learning setting with respect to SLU, i.e. we explore how a semantic parser applicable to spoken utterances can be learned directly from spoken utterances coupled with ambiguous context information without assuming any predefined linguistic knowledge bases other than a task-independent phoneme recognizer.
While a word-based ASR may be applied in order to handle spoken utterances, just like it is typically done in spoken dialogue systems, working with a phoneme recognizer yields several advantages; the main advantages compared to applying a

word-based ASR are:

- There are **no a-priori lexical restrictions by the ASR**. This supports the acquisition of a potentially unrestricted vocabulary by the parser. Further, parsing can be performed at the whole-sentence level without a priori restrictions concerning possible words and hence meanings.
- It yields **low costs for training.** Compared to training a word-based ASR, training a phoneme-based one usually requires much less training data. In particular, in case of a word-based ASR, typically large amounts of in-domain training data are needed to train a language model in order to yield satisfying performance. This makes our approach also interesting with respect to **application for under-resourced languages** for which large amounts of suitable training data to build word-based language models may be simply not available (at least with respect to several domains).
- Applying a task-independent phoneme recognizer makes it **easy to adapt** the system to novel tasks.

Due to recognition errors and since a segmentation task must be tackled additionally, learning a parser from speech data without word transcriptions is much more challenging compared to learning from text. Our system performs segmentation in the presence of noise, i.e. recognition errors and different pronunciations of the same word, and semantic ambiguity by inducing alignments between *NL* utterances and context representations. A parser – represented in the form of a lexicon and an inventory comprising syntactic constructions – is then estimated based on co-occurrence frequencies, thus building on learning mechanisms already explored in the computational model presented previously (cf. Chapter 3). Alignments are computed both bottom-up by first determining structures of rather low complexity and top-down by including syntactic information. While learning linguistic structures of rather low complexity from speech without word transcriptions has been addressed previously, e.g. learning (novel) words (cf. Section 2.4), we are not aware of other algorithms learning syntactic constructions using ambiguous non-linguistic contexts. We present empirical results indicating that:

- When applied to textual input, using the proposed learning mechanisms a parser achieving state-of-the-art performance can be induced straightforwardly.
- When applied to speech, in spite of noise and contextual ambiguity, a parser can be learned which can be successfully applied to understand several unseen spoken utterances.

- In fact, in line with work investigating phoneme-based SLU with respect to determining concepts in speech (Svec et al., 2013), our results suggest that application of a phoneme recognizer can yield performance similar to the performance which can be expected for the application of an in-domain word-based recognizer.
- Top-down knowledge of syntactic patterns can yield useful segmentation cues, improving both boundary detection and language learning.
- Parsing performance can be improved by taking different phoneme sequences for syntactic patterns and lexical units into account, even if several of them are incorrectly segmented and do not correspond to actual words.
- In cases where training data in the form of text or manual transcriptions are available, the approach can be successfully applied to induce semantic speech recognition grammars – which are typically created manually or learned in a supervised setting –, allowing semantic parsing of speech with a rather low loss in performance compared to parsing of input without recognition errors.

The remainder of this chapter is organized as follows. First, we will present the learning problem and subsequently describe the proposed approach. Afterwards, we will present empirical results, i.e. we will first compare our proposed learning mechanisms to the state-of-the-art and subsequently explore learning from speech. Finally, we will discuss our approach, implications of the results as well as future work.

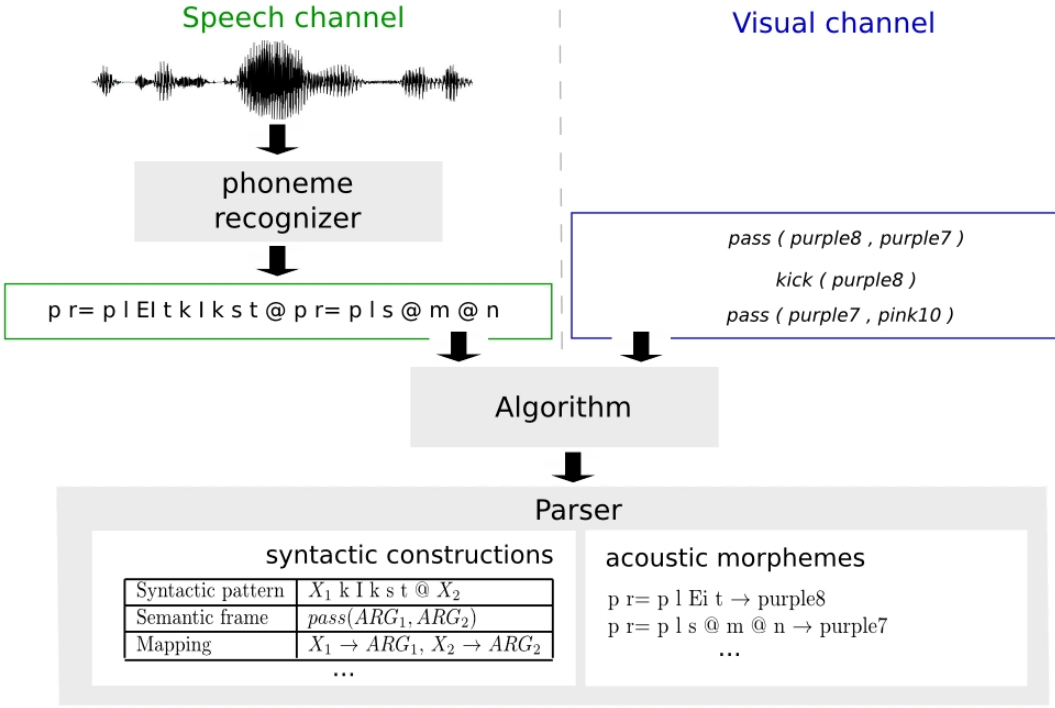
5.2. Learning problem

In this chapter, we explore learning from spoken instead of written utterances, and we assume no predefined linguistic resources other than a task-independent phoneme recognizer. The input to the learning system is the same as in case of the computational model presented previously, with the exception that spoken utterances are presented instead of written ones. Spoken utterances are transcribed using a phoneme recognizer yielding natural language utterances in the form of phoneme sequences (*NL*) as the input to the learning algorithm. The learning process is illustrated in Fig. 1[1].

Given a set of input examples, the goal is to estimate a parser P consisting of a lexicon V_P and a set C_P of syntactic constructions, both containing meanings for

[1] Again, we work with the RoboCup corpus and hence examples are taken from the RoboCup domain.

Figure 1.: Overview of the learning process.


Speech channel
Visual channel
phoneme recognizer
p r= p l EI t k I k s t @ p r= p l s @ m @ n
pass (purple8 , purple7)
kick (purple8)
pass (purple7 , pink10)
Algorithm
Parser
syntactic constructions
Syntactic pattern | X_1 k I k s t @ X_2
Semantic frame | $pass(ARG_1, ARG_2)$
Mapping | $X_1 \rightarrow ARG_1$, $X_2 \rightarrow ARG_2$
...
acoustic morphemes
p r= p l Ei t → purple8
p r= p l s @ m @ n → purple7
...

entries. We represent entries in both cases similarly as in the computational model in the form of constructions, albeit sequences of phonemes are considered instead of sequences of words. The parser's lexicon consists of semantically meaningful sequences $a_i \in V_{NL}$; following Gorin et al. (1999) we also call such sequences *acoustic morphemes*. Each syntactic construction in C_P consists of a syntactic pattern which can contain syntactic slots. In this case, syntactic slots are positions where a $v \in V_P$ may be inserted. The meaning is represented by a semantic frame, and a one-to-one mapping which maps slots in the syntactic pattern to argument slots in an associated semantic frame is required.

An example of an input pair is given by:

(5.1)

NL:	p r= p l EI t k I k s t @ p r= p l s @ m @ n
mr_1:	$playmode(play\_on)$
mr_2:	$pass(purple8, purple7)$
mr_3:	$pass(purple2, purple5)$

Given this example, V_P should contain the following two entries:

(5.2)

$\hat{NL}$	p r= p l EI t	$\hat{NL}$	p r= p l s @ m @ n
$\hat{mr}$	*purple8*	$\hat{mr}$	*purple7*

C_P should contain the following syntactic construction:

(5.3)

$\hat{NL}$	X_1 k I k s t @ X_2
$\hat{mr}$	$pass(ARG_1, ARG_2)$
Φ	$X_1 \rightarrow ARG_1$ $X_2 \rightarrow ARG_2$

Notice that both V_P and C_P contain different entries for different pronunciations of the same (sequence of) word(s), which may be erroneous due to recognition errors.

The definition of the underlying vocabulary of the *MR* portion of the data V_{MR} remains the same as containing all semantic entities. We define the vocabulary of the *NL* portion of the data V_{NL} as containing all potential acoustic morphemes, i.e. all observed sequences of length 5 to 13. While this may be rather arbitrary, restricting the sequence length reduces computational costs for our experiments, and we assume that sequences of such length already cover most "good" candidates for acoustic morphemes.

Notice that while we again represent knowledge in the form of constructions, in this chapter it is not our goal to model claims posed within the framework of construction grammar, in particular we do not represent linguistic knowledge by means of a network. As mentioned previously, we also do not attempt to model child language acquisition. Thus, we will not explicitly design the system with respect to cognitive plausibility, even though we will explore similar learning mechanisms as those explored within the computational model.

As mentioned before, learning from *NL* utterances coupled with ambiguous context information has been previously investigated with respect to textual utterances, not phonemically transcribed speech. The learning setting investigated in this chapter is more challenging since a segmentation task must be tackled additionally and because noise in the form of different pronunciations of the same word(s) and recognition errors must be taken into account. To illustrate the additional challenges of the phoneme-based learning setting consider the following two word-based input examples which illustrate the learning problem from the system's perspective, i.e. words and semantic referents are replaced by arbitrary symbols:

(5.4)

NL	w_1 w_2 w_3 w_4 w_5
mr_1	$pred_1(ref_1, ref_2)$
mr_2	$pred_2(ref_2, ref_3)$
mr_3	$pred_1(ref_1, ref_3)$
mr_4	$pred_3(ref_4)$

(5.5)

NL	w_1 w_5 w_6 w_7 w_8
mr_1	$pred_4(ref_1, ref_5)$
mr_2	$pred_5(ref_6)$

Learning across these two situations we may assume that w_1 refers to the semantic referent ref_1 due to their co-occurrence.

Now let us consider the phoneme-based setting in which the previously presented examples may be of the following form:

(5.6)

NL	p_1 p_2 p_1 p_3 p_4 p_5 p_6 p_7 p_8 p_7 p_9 p_6 p_{10} p_1 p_2 p_1 p_3 p_9 p_{10} p_{11} p_{12}
mr_1	$pred_1(ref_1, ref_2)$
mr_2	$pred_2(ref_2, ref_3)$
mr_3	$pred_1(ref_1, ref_3)$
mr_4	$pred_3(ref_4)$

(5.7)

NL	p_1 p_{13} p_8 p_1 p_3 p_4 p_5 p_1 p_{14} p_7 p_8 p_7 p_9 p_6 p_1 p_2 p_1 p_3 p_{15} p_4 p_{16}
mr_1	$pred_4(ref_1, ref_5)$
mr_2	$pred_5(ref_6)$

Across those two situations several candidate sequences, e.g. "p_1 p_2 p_1 p_3", "p_1 p_2 p_1" and "p_1 p_3 p_4 p_5", co-occur with ref_1 and none of them actually corresponds to w_1. Thus, in contrast to the word-based setting in which it may be possible to determine lexical units and their meanings solely based on co-occurrence frequencies, i.e. by applying cross-situational learning, this strategy is unlikely yielding satisfying results in the phoneme-based setting.

5.3. Algorithm

Due to the increased complexity of the learning scenario we modified the learning mechanisms explored within the computational model towards enabling faster language learning. Relating back to Example 9 in Chapter 3, we observed for the computational model that, at later learning steps, language learning performance increased in that utterances could be incorporated directly as generalized paths (corresponding to syntactic patterns) into the network, and that it was also possible to directly infer the correct meaning out of an ambiguous context. In the approach presented in the following, we attempt to capture this behavior. In particular, we attempt to directly generalize utterances based on lexical units and their semantics. Minimal variation in the surface structure of different utterances is not taken into account. Instead, we introduce alignments between form and meaning.

The main idea of the proposed system is to compute alignments between *NL*s and ambiguous context representations for the given input examples. Given such alignments, a parser is then estimated by computing co-occurrence frequencies at different levels just like it has been previously explored in the computational model. An overview of the algorithm's work flow is illustrated in Fig. 2.

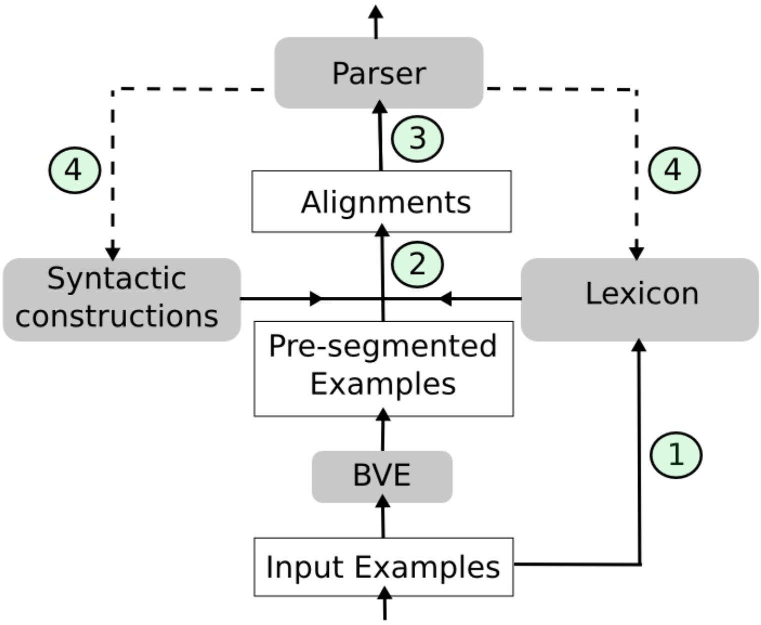


Figure 2.: The algorithm's work flow.

It is roughly divided into the following four steps:

1. Acquisition of an initial lexicon comprising initial knowledge about acoustic morphemes

2. Bottom-up computation of alignments based on the initial lexicon

3. Estimation of a parser by computing co-occurrence statistics on the alignments

4. Top-down re-estimation of alignments using the learned parser.

Steps 3 and 4 are then repeated until some criterion is met.

In order to restrict possible segmentations and computational costs – notice that the number of occurring sequences can be very high (cf. Section 5.4) –, we apply an unsupervised algorithm, i.e. Bootstrap Voting Experts (Hewlett & Cohen, 2009, cf. Section 2.4)[2], to segment all *NL*s into (sub)word-like units and then utilize the

[2] We utilized the Java implementation available online at http://code.google.com/p/voting-experts/ with parameter optimization via minimum description length; for each fold parameters were optimized on the training data.

segmented utterances for further processing.
Given a pre-segmented (*NL*,*MR*) pair, an alignment is computed by measuring possible segmentations for *NL* along with a hypothesized mapping to semantics for each $mr_i \in MR$. For instance, consider the previously shown input example:

(5.8)

NL:	p r= p l EI t k I k s t @ p r= p l s @ m @ n
mr_1:	$playmode(play\_on)$
mr_2:	$pass(purple8, purple7)$
mr_3:	$pass(purple2, purple5)$

Then, the following alignment should be created:

(5.9)

NL	X_1 k I k s t @ X_2
mr	$pass(ARG_1, ARG_2)$ $\phi : X_1 \rightarrow ARG_1,\ X_2 \rightarrow ARG_2$
$nl \rightarrow ref$	p r= p l EI t $\rightarrow purple8$ p r= p l s @ m @ n $\rightarrow purple7$

Notice that when creating alignments we also attempt to make use of lexical knowledge to disambiguate the data, that is, to directly rule out *mr*s which the *NL* does not correspond to. For instance, if no sequences expressing the semantic referent at position ARG_1 in $playmode(ARG_1)$ appear in *NL* this semantic frame cannot be aligned with *NL* and is directly ruled out, i.e. not considered during the computation of co-occurrence frequencies (step 4).
Given a list of alignments, a parser is estimated by computing co-occurrence statistics at different levels. In particular, we compute association scores at three levels:

1. Lexical L: $nl \rightarrow ref$: between all $v_{nl} \in V_{NL}$ (L_{NL}, e.g. "p r= p l EI t") and $v_{mr} \in V_{MR}$ (L_{MR}, e.g. $purple8$) appearing in alignments.

2. Pattern P: $NL \rightarrow mr$: between all patterns (P_{NL}, e.g. "X_1 k I k s t @ X_2") and semantic frames (P_{MR}, e.g. $pass(ARG_1, ARG_2)$) appearing in alignments. The induction of such patterns will be described later on.

3. Mapping M: between all variable positions (M_{NL}, e.g. X_1) and argument slots (M_{MR}, e.g. ARG_1) for each pattern and semantic frame.

Then, $nl \rightarrow ref$ yields V_P, while $NL \rightarrow mr$, each coupled with its individual mapping, yields C_P.
The association score is computed as follows.

Definition 10 (Association score)**.** Let $freq(z_y)$ be the number of observations z_y appears in (at least once). The association score $assoc(z_{nl}, z_{mr})$ between a $z_{nl} \in Z_{NL}$ and a $z_{mr} \in Z_{MR}$, $Z \in \{L, P, M\}$ is given by:

$$assoc(z_{nl}, z_{mr}) = P(z_{nl}|z_{mr}) \times P(z_{mr}|z_{nl}), \tag{5.10}$$

$$P(z_{nl}|z_{mr}) = \frac{freq(z_{nl}, z_{mr})}{freq(z_{mr})},\ P(z_{mr}|z_{nl}) = \frac{freq(z_{nl}, v_{mr})}{freq(z_{nl})}.$$

Based on the association score we determine meanings for referents and, in turn, expressions for referents as follows.

Definition 11 (Meaning and expression)**.** A $z_{mr} \in Z_{MR}$ is said to be a *meaning* of $z_{nl} \in Z_{NL}$ and z_{nl} *expresses* z_{mr} if

$$P(z_{nl}|z_{mr}) = \underset{z_i \in Z_{MR}}{\text{argmax}}\ assoc(z_{nl}, z_i). \tag{5.11}$$

Thus, a meaning is computed similarly to the computation of associations in the computational model (cf. Equation 2). However, since we do not explore online learning in this chapter, we opted to simply compute association scores over the whole dataset in an offline fashion.

Due to different pronunciations and recognition errors, an algorithm for approximate matching is needed in order to map different phoneme sequences onto each other. We compute the similarity between phoneme strings following Yu et al. (Yu & Ballard, 2002; Yu et al., 2005) by first transforming phonemes into vectors of (articulatory) distinctive features (Ladefoged, 1993) and subsequently determining the similarity between two strings based on the dynamic programming principle (Kruskal, 1999). In doing so, a positive reward is given to matching phonemes and negative scores are assigned otherwise, depending on the number of differing features. In the following, we call the phonetic similarity between two phoneme strings sp_1 and sp_2 $sim(sp_1, sp_2)$, and only strings having at least a certain number of phonemes in common are considered as (potentially) similar. In particular, we set a threshold by multiplying the maximal sequence length with a fraction of the reward set for matching phonemes. Detailed information concerning the computation of phonetic similarities can be found in Appendix A.

The algorithm is detailed in Algorithm 3. In the following, the four steps of the algorithm will be explained in more detail.

Algorithm 3 Learning algorithm

```
Input: A list of examples $E = \{(NL_1, MR_1), \ldots, (NL_k, MR_k)\}$
Output: A lexicon $V_P$ and an inventory of syntactic constructions $C_P$

  1. Acquisition of an initial lexicon
     ◦ compute association scores between all $v_{nl} \in V_{NL}$ and all $v_{mr} \in V_{MR}$ in $E$
     ◦ $L_a \leftarrow$ select for each $v_{mr} \in V_{MR}$, $v_{mr}$ refers to an argument, sequences having highest association score(s) with $v_{mr}$ along with their score

  2. Bottom-up computation of alignments
     ◦ $E \leftarrow$ pre-segment all $NL_i \in E$ using the BVE algorithm
     ◦ $A \leftarrow$ an empty list
     ◦ for all examples $(NL_i, MR_i) \in E$ do
         - create and score initial alignments using $L_a$ for all $mr \in MR_i$
         - $max\_score \leftarrow$ best score obtained for an alignment created for any $mr \in MR_i$
         - store all alignments with $max\_score$ in $A$
     ◦ end for

  repeat as long as the cumulative alignment score increases

     3. Estimation of a parser
     ◦ compute association scores at three levels $(L, P, M)$ on $A$
     ◦ $L_a \leftarrow$ select for each $v_{mr} \in V_{MR}$, $v_{mr}$ refers to an argument, sequences having highest association score(s) with $v_{mr}$ along with their score
     ◦ $L_p \leftarrow$ select for each $v_{mr} \in V_{MR}$, $v_{mr}$ refers to a predicate, sequences having highest association score(s) with $v_{mr}$ along with their score and mapping

     4. Top-down computation of alignments
     ◦ $A \leftarrow$ an empty list
     ◦ for all examples $(NL_i, MR_i) \in E$ do
         - create and score initial alignments using $L_a$ and $L_p$ for all $mr \in MR_i$
         - $max\_score \leftarrow$ best score obtained for an alignment created for any $mr \in MR_i$
         - store all alignments with $max\_score$ in $A$
     ◦ end for

  end repeat

  Estimation of the final parser
     ◦ compute association scores at three levels $(L, P, M)$ on $A$
     ◦ $V_P \leftarrow$ select for each $v_{mr} \in V_{MR}$, $v_{mr}$ refers to an argument, all sequences expressing $v_{mr}$
     ◦ $C_P \leftarrow$ select for each $v_{mr} \in V_{MR}$, $v_{mr}$ refers to a predicate, all sequences expressing $v_{mr}$ along with their mapping

  return $V_P$ and $C_P$
```

5.3.1. Acquisition of an initial lexicon

As illustrated before, due to recognition errors and different pronunciations, several different phoneme sequences may exist for the same word. Thus, finding meanings based on co-occurrence statistics is much more challenging compared to working with words. In particular, several sequences referring to semantic referents typically occur infrequently, i.e. once or twice only (cf. Section 5.4.6), and thus several expressions may not be found by applying cross-situational learning. However, we assume that at least some sequences co-occur frequently enough with their corresponding semantic referents to bootstrap the parser, i.e. to establish initial form-meaning mappings. Notice that these sequences are likely only subsequences of expressions for the corresponding semantic referents but we assume that these are still useful for bootstrapping the parser. Thus, we compute association scores between all $v_{nl} \in V_{NL}$ and $v_{mr} \in V_{MR}$. For each semantic referent $v_{mr} \in V_{MR}$, we then select a number of "good" candidates, i.e. sequences having highest association score(s) with v_{mr}, as acoustic morphemes for the initial lexicon. These are inserted into the lexicon along with their meanings and the corresponding association scores, and are utilized for bootstrapping the parser.

5.3.2. Bottom-up creation of alignments

Given an example (*NL,MR*), an alignment is created and scored for each $mr_i \in MR$. The parser is then only trained on alignments with maximal score. Given an (*NL,mr*) pair, possible alignments are created by segmenting *NL* such that segments express semantic referents observed in *mr* according to the initial lexicon.

Definition 12 (Alignment)**.** An alignment is a hypothesized mapping between form and meaning given a phoneme string *NL* and a meaning *mr*. It includes i) a segmentation of *NL* such that all referents appearing in *mr* are expressed by individual sequences, i.e. a potential syntactic pattern, ii) a hypothesized mapping for each of the segments to their corresponding semantics, and iii) a hypothesized mapping between syntactic slots in the *NL* pattern and argument slots in the *mr*.

Example 11. An example of an alignment between a phoneme string "p I n k @ p A r p { s I s t @ t EI k @ t EI t" and an *mr* *pass(pink4,pink8)* is illustrated in Fig. 3. In the alignment, "p I n k @ p A r " and "t EI k @ t EI t" are aligned with the referents *pink4* and *pink8* using the entries "p I N k f O r" and "p I N k EI t" in the initial lexicon, respectively. The pattern "X_1 p { s I s t @ X_2" is hypothesized to express the *pass* predicate, and the first and second slot are hypothesized to express the first

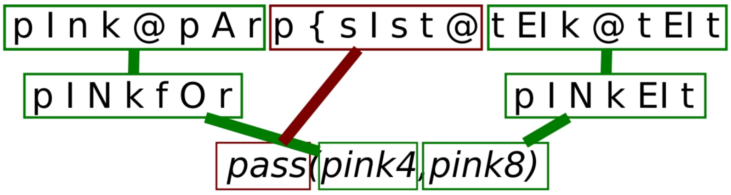


Figure 3.: Example of an alignment between form and meaning

and second argument slot in the predicate, respectively. In sum, the alignment is composed of the following hypothesized correspondences:

(5.12)

NL	X_1 p { s I s t @ X_2
mr	$pass(ARG_1, ARG_2)$ $\phi : X_1 \to ARG_1,\ X_2 \to ARG_2$
$nl \to ref$	p I n k @ p A r → *pink*4 t EI k @ t EI t → *pink*8

As mentioned before, alignments are created using the initial lexicon. More specifically, the meaning of a segment s is computed as the meaning mr_s of an entry having maximal similarity score with s (if existent) $e_s^{L_i}$ in the initial lexicon L_i. The similarity score is computed by applying the procedure for approximate string matching as described previously and detailed in Appendix A.
The alignment score between s and mr_s is then computed as

$$align^{L_i}(s, mr_s) = \frac{sim(s, e_s^{L_i})}{MAXSIM_{mr_s}} * assoc(e_s^{L_i}, mr_s), \tag{5.13}$$

where $MAXSIM_{mr_s}$ is the maximal similarity which has been obtained for any segment s_i with meaning mr_s and lexicon entry $e_{s_i}^{L_i}$ in one of the segmentations inspected for $(NL{,}mr)$. Thus, an alignment is measured by inspecting whether i) a sequence likely corresponds to a lexicon entry, and ii) whether this entry is a good expression for an observed argument. For instance, in Fig. 3 this corresponds to inspecting i) whether “p I k @ p A r” likely corresponds to “p I n k f O r”, and in turn ii) whether “p I n k f O r” actually means *pink*4. The alignment score for a complete alignment $align(NL{,}mr)$ is then computed as the sum of the alignment scores for segments expressing the arguments, i.e.

$$align_{arg}^{L_i}(NL{,}mr) = \sum_{arg \in ARGs(mr)} align^{L_i}(s, arg). \tag{5.14}$$

We use the accumulated score over arguments because it prefers alignments in which more arguments are expressed in the segmentation. Notice that according to the definition of alignments only segmentations are considered in which all arguments in mr are indeed expressed by individual segments.

5.3.3. Creating a parser

As described previously, given a list of alignments, association scores are computed at the three levels (Lexical, Pattern, Mapping) as defined by equation 5.10.

5.3.4. Top-down creation of alignments

If a sequence co-occurs with a referent n-times, then all of its subsequences do so at least n-times and may thus yield better candidates for acoustic morphemes and subsequent segmentation errors. For instance, a sequence "p I N k I l @ v a n k I k s" might be incorrectly segmented as "p I N k X_1 k I k s" because "I l @ v a n" is a "better" expression for $pink11$ than "p I N k I l @ v a n". Due to such potential errors we apply a top-down step to refine alignments based on previously learned knowledge of syntactic patterns. For instance, once the system has learned that "X_1 k I k s" is a likely expression for $kick(ARG_1)$ while "p I N k X_1 k I k s" is not, it can utilize this information to correct the error described previously. In general, in the top-down step alignments are computed as in step 2, but in addition a score for segments expressing the predicate is added. In particular, in addition to a lexicon containing acoustic morphemes, a lexicon containing patterns is utilized; both are extracted from the parser. As in case of creating the initial lexicon, they are created by taking a number of "good" candidates according to the association score. Specifically, a number of acoustic morphemes and patterns are selected for all semantic referents referring to arguments and predicates/semantic frames and stored in lexicon L_a and L_p, respectively. Given an (NL,mr) pair, the alignment score is computed as defined in Equation 5.14 as $align_{arg}^{L_a}(NL,mr)$. The score for segments sp instantiating the pattern $align^{L_p}(sp, mr)$ is computed as defined in Equation 5.13 if the meaning of the lexicon entry for sp matches the observed mr and summed up with $align_{arg}^{L_a}(NL,mr)$. Based on the re-estimated alignments, the parser is then induced again (step 3). This procedure is then repeated (steps 3 and 4) as long as the cumulative alignment score increases. Taking only a number of "good candidates" into account for refining alignments is again mainly done in order to reduce computational costs for our experiments since the number of occurring sequences can be very high (cf Section 5.4). Notice, however, that "good candidates" are chosen for re-estimation of alignments only; the final parser comprises all acoustic morphemes and syntactic constructions learned in the last run.

5.3.5. Parsing

Parsing is performed in a similar manner as in case of the computational model, albeit taking approximate matching into account. Given an *NL*, the system inspects whether this utterance is an instantiation of one (or more) of the stored syntactic constructions, i.e. it searches for a matching pattern $p \in C_P$. All matches are rated by summing up the similarity scores for p and the acoustic morphemes observed in its slots with the corresponding segments in *NL*, and – if existent and a meaning has been determined – one having maximal score is selected. The meaning *mr* is the semantic frame associated with p in which meanings of acoustic morphemes appearing in syntactic slots are inserted into the corresponding argument slots according to the mapping. In contrast to parsing with the computational model (cf. Section 3.3.6), parsing can be ambiguous, i.e. different matches with maximal score may be found, especially due to applying approximate matching. For instance, given an *NL* "t I n k I l @ v a n k i k s" two matching derivations having maximal score may be returned: "X_1 k I k s" with X_1 = "p I N k I l @ v a" and "X_1 k I k s" with X_1 = "p I N k I @ v a n". However, assuming that meanings have been obtained correctly, both would yield the same semantics, i.e. *kick(pink11)*, and hence the choice of the resulting derivation does not affect the final parsing result. In general, it may be rather often the case that if different matches exist, these correspond to different phoneme sequences or segmentations actually referring to the same (sequences of) word(s) and hence semantics. Thus, assuming that different matching derivations often yield the same semantics, we simply choose one in cases where different derivations with maximal score exist.

5.4. Experimental evaluation and discussion

The task of learning language from examples comprising natural language utterances coupled with ambiguous perceptual context information has been previously addressed with respect to learning from written text, not speech. Previous algorithms have been mainly evaluated on the RoboCup soccer corpus. In order to compare the learning mechanisms proposed in this chapter to the state-of-the-start, we evaluate our system on textual RoboCup data in addition to an evaluation on speech data. In the following, we first provide information about the input data. We then investigate the system's performance when applied to the typical learning scenario, i.e. written text. Further, we compare the system's performance when applied to phonemic transcriptions of speech to the expected performance when applied to ASR word transcriptions, investigate the role of syntactic information on segmentation, and explore effects of phonemic variation on parsing and learning. Afterwards, we show how the system can be applied to induce semantic speech recognition grammars. We compare the performance of applying such grammars as a language model with an ASR to the performance of applying a standard n-gram model and to the performance of applying purely syntactically motivated grammars.

5.4.1. Datasets

For evaluation we use the RoboCup soccer corpus which has been used previously for evaluating the cognitive model for the induction of verb-specific constructions. This corpus contains written *NL*s. Because we explore learning from spoken utterances, all *NL* utterances in the RoboCup training data were read by a native American speaker. Out of them, 23 were excluded due to an error made by the speaker, yielding 1849 spoken input examples. All spoken utterances were then transcribed using a phoneme recognizer. In particular, we applied Sphinx-3 (Placeway et al., 1997) with the configuration and resources available online[3]. The applied acoustic models were trained on the HUB4 dataset (Fiscus et al., 1998) which contains broadcast news speech which matches our spoken RoboCup data with respect to acoustics in that in both cases read speech is addressed. Silence was removed from the transcriptions, and transcriptions were converted from ARPABET into X-SAMPA, allowing comparison to MaryTTS output. No effort was made in order to improve recognition performance.

Furthermore, for evaluating the system without recognition errors and different pronunciations of the same word, we applied grapheme-to-phoneme conversion to the written RoboCup comments using MaryTTS (Schröder & Trouvain, 2003). All markers of syllable and word boundaries were removed. By comparing the ASR transcribed data to the grapheme-to-phoneme converted data, a phoneme error rate (PER) of 34.2% averaged over all four games was obtained.[4]

Some statistics for the RoboCup soccer corpus as well as the grapheme-to-phoneme converted data and the spoken utterances transcribed by the phoneme recognizer are shown in Table 5.1; the statistics for the written data have been presented previously

[3] http://cmusphinx.sourceforge.net/wiki/phonemerecognition

[4] Notice that the determined PER can only provide a rough approximation of the actual PER because reference datasets were created automatically. Automatically transforming words into a sequences of phonemes may not yield the sequences actually spoken in several cases and, in addition, automatic transformations may be erroneous.

but are shown again for direct comparison.

Table 5.1.: Some statistics for the RoboCup training dataset, the grapheme-to-phoneme converted data, and the speech data transcribed by a phoneme recognizer.

RoboCup dataset, written text	
Total number of comments	1,872
Comments having correct *mr*	1,539
Average number of events per comment	2.5
Maximum number of events per comment	12
SD in number of events per comment	1.8
Mean utterance length	5.7 words
Number of tokens	10,700
Vocabulary size	443
Speech data, ASR transcribed phonemes	
Total number of comments	1,849
Mean utterance length	25.67 phonemes
Number of tokens (sequence length 5–13)	294,390
Vocabulary size (sequence length 5–13)	186,708
Grapheme-to-phoneme converted data	
Mean utterance length	27.03 phonemes
Number of tokens (sequence length 5–13)	321,088
Vocabulary size (sequence length 5–13)	68,781

In this chapter, we apply the same evaluation schema as introduced by Chen et al. (2010) and already described in Section 3.4.2. In the following, we apply it with respect to the different datasets, i.e. the written dataset, the grapheme-to-phoneme converted dataset and the speech dataset.

5.4.2. Application to written text

In order to compare the employed learning mechanisms to the state-of-the-art, we evaluate our system on the written RoboCup data. As mentioned previously, to the best of our knowledge the best performing system on this dataset so far has been proposed by Börschinger et al. (2011).

When applied to text, using our proposed learning mechanisms a parser can be induced straightforwardly. In particular, we computed an initial lexicon by taking all uni- and bigrams as the vocabulary V_{NL} and computed alignments only once by applying a single bottom-up step. Approximate matching, while not needed when computing alignments, was applied during parsing of *NL*s for which no pattern could be found otherwise. Specifically, we explored three strategies for approximate matching: i) matching with a Levenshtein distance of 1 (LD, e.g. “Pink1 makes a cross pass” can be matched with “X_1 makes a pass”), ii) matching partially (partial, e.g. “Pink1 passes to Pink2 near midfield” can be matched with “X_1 passes to ”X_2”), and iii) matching with both of them (LD+partial). Notice that similarly to parsing phoneme sequences, parsing textual utterances can be ambiguous in that different matching derivations may be found. While, in general, in this case we simply choose one of them, in case of parsing sequences of words we always choose a derivation yielding a maximum number of semantic referents. This is done because we apply partial matching and attempt to determine a matching subsequence containing as much semantic information as possible.[5]

Table 5.2.: Semantic parsing results for written text

Our system			
Parsing strategy	$\mathbf{F_1}$	**Precision**	**Recall**
Complete	83.54	95.97	74.19
LD	86.82	93.62	81.14
Partial	86.46	94.53	79.71
LD+partial	**89.09**	93.38	85.23

Börschinger et al (2011)			
	$\mathbf{F_1}$	**Precision**	**Recall**
	86.0	86.0	86.0

Results are presented in Table 5.2. As can be seen, our system outperforms Börschinger et al. (2011) with respect to F_1 when approximate matching is applied, yielding its best result of 89.09% when both matching with a Levenshtein distance of 1 and partial matching are applied. By contrast, (expectedly) the highest precision is achieved when approximate matching is not applied, yielding a result of 95.97%. Still, when approximate matching is applied the system's precision remains rather high with values of above 93%. With respect to recall, our system achieves values of up to 85.23% which is only slightly below the 86% achieved in Börschinger et al. (2011)'s case. The results show that when applied to written text, using the proposed learning mechanisms a parser yielding state-of-the-art performance can be induced straightforwardly.

5.4.3. Application to simulated ASR word transcriptions

In this section we evaluate the system's performance with respect to the typical scenario in SLU, i.e. application of a word-based ASR to transcribe speech into words

[5] Due to this fact the results presented in the following differ slightly from those presented in Gaspers & Cimiano (2014b) where we did not prefer derivations yielding as many semantic referents as possible during parsing.

and subsequent semantic parsing. In this scenario, parsing performance depends heavily on the performance of the applied ASR. SLU systems typically include an ASR with an in-domain language model in order to yield good performance. An in-domain language model was not available for our experiments since no suitable in-domain data were available. Yet, one can be built if we assume the availability of in-domain data by using data from the written RoboCup soccer corpus. The experiments presented in the following are twofold: i) we explore recognition performance on the spoken RoboCup data when assuming the availability of in-domain training data, and ii) from these experiments we determine statistics concerning recognition errors to simulate data containing different amounts of errors made by an in-domain ASR for investigating parsing performance with respect to different word error rates. Recall, however, that a lower WER does not necessarily yield better parsing performance which is rather dependent on the type of errors made (Bayer & Riccardi, 2012; Wang et al., 2003, cf. Section 2.2). With respect to the dataset at hand, for instance, a spoken utterance “pink two passes backward to pink seven” in which “seven” is incorrectly recognized as “eleven” likely yields a parsing error while a recognition error which substitutes “backward” by “forward” may not prevent correct parsing. This is the case because “forward” and “backward” do not carry any semantics in the dataset at hand while correct identification of numbers is in most cases essential for detecting the correct semantic referents. However, the results presented in the following will serve as a rough estimate towards what WER may yield reasonable results, and as a basis for a rough comparison between application using a phoneme- and a word-based speech recognizer. In these experiments, we assume that the ASR is applied with a standard n-gram language model.

In order to investigate speech recognition performance when assuming the availability of in-domain training data we performed 4-fold cross-validation on the four RoboCup games. For each fold, written training data for three games were used to train a trigram language model; trigram models were created using SRILM (Stolcke, 2002)[6]. For application with the ASR, data were normalized beforehand which mainly comprised lowercasing and replacing numbers in player names, e.g. “pink4” → “pink four”. The trigram language model was then applied with a speech recognizer to transcribe the spoken training data for the remaining game. Specifically, we applied Sphinx-4 (Walker et al., 2004) using lexicon and acoustic models trained on the HUB4 dataset (Fiscus et al., 1998). As already mentioned with respect to phoneme recognition, this dataset contains broadcast news speech which matches

[6] Notice that n-gram models are typically build using large amounts of data. Because the vocabulary size of the RoboCup dataset is small, the training data of three games appear to be already sufficient as indicated by the resulting recognition results.

our spoken RoboCup data with respect to acoustics in that in both cases read speech is addressed. The applied resources are available online. We added transcriptions for out of vocabulary (OOV) words to the lexicon. Only two words were OOV along with some typos. Averaged over the four folds a WER of 9.4% on the spoken training data and 7.1% on the spoken gold standard data, which is a subset of the training data, was obtained.
Starting from the written, normalized data, ASR errors were simulated roughly following Jung et al. (2009). The authors presented a system for user simulation which can be utilized to evaluate spoken dialogue systems; the system also includes ASR channel simulation. In order to simulate an erroneous utterance, a correct input sequence is transformed by applying the following four steps:

1. Error positions are determined randomly.

2. For each error position the error type – substitution, deletion or insertion – is determined based on some error distribution.

3. The corresponding errors are generated.

4. Steps 1–3 are repeated several times, and all simulated utterances are ranked using a language model. Finally, one of the top-ranked utterances is chosen randomly as the resulting erroneous utterance (Jung et al., 2009).

We used the recognition results from the previous experiment for simulation. In particular, the error distribution found averaged over all folds was used in case of step 2. Insertion and substitution errors were generated probabilistically according to the errors made by the ASR. In step 4, we applied a trigram model trained on the complete written RoboCup training data. We repeated steps 1–3 20 times for each utterance and chose on of the top 5 ranked candidate sequences randomly. We created datasets representing error rates of 5%, 10% and 15% and used these to train and test semantic parsers. Results are presented in Table 5.3.

The results reveal that, expectedly, performance degrades when the system is applied to data containing recognition errors. Even with a rather low WER of 5%, performance already degrades about more than 15% absolute with respect to F_1. Here it must be noted that due to the evaluation schema a single recognition error can yield a completely incorrect parse. Recall that evaluation is performed on the basis of completely correctly determined *mr*s, that is, all referents and the predicate must be determined correctly in order to yield a correct parse. For instance, if any of the words “purple”, “pink”, “two” or “five” is deleted or substituted in an utterance “purple two passes to pink five”, this might prevent the correct identification of one

Table 5.3.: Semantic parsing results for learning from and testing on data containing different amounts of simulated recognition errors.

Simulated WER of 5%			
Parsing strategy	**F$_1$**	**Precision**	**Recall**
Complete	69.29	85.98	58.33
LD	**71.8**	80.67	65.01
Partial	63.04	64.28	61.85
Partial-LD	68.47	69.40	67.58
Simulated WER of 10%			
Parsing strategy	**F$_1$**	**Precision**	**Recall**
Complete	57.05	74.91	46.74
LD	**61.29**	70.83	54.57
Partial	51.59	52.5	50.72
Partial-LD	58.02	58.86	57.21
Simulated WER of 15%			
Parsing strategy	**F$_1$**	**Precision**	**Recall**
Complete	48.15	66.46	38.22
LD	**49.13**	58.94	42.44
Partial	41.90	42.99	40.87
Partial-LD	45.15	46.06	44.3

of the referents. Similarly, deleting or substituting “passes” may yield an incorrect predicate or no parse at all. Moreover, since the training data contain errors as well, compared to working with textual input, less (correct) training data are available for parser induction and the system may learn (more) erroneous patterns, lexical units and semantics, yielding subsequent parsing errors.

5.4.4. Application to ASR phoneme transcriptions

To the best of our knowledge, the task of learning language from examples of natural language utterances coupled with ambiguous context information has to date been evaluated with respect to written text, not speech. Hence, we cannot compare the performance of our system to that of other systems. Therefore, in order to evaluate the amount of language learned by our system we computed a simple “rote learning” baseline. In particular, as in case of evaluating the cognitive model, we computed the F_1 score that would have been achieved if the system would have performed “rote learning” of input examples. In the baseline, an *NL* in the test data was parsed – if it had also been observed in the training data – by choosing one of the *mr*s observed with it randomly. Results for both, applying the system to ASR transcriptions of speech and grapheme-to-phoneme converted data, along with their corresponding

baseline values are presented in Table 5.4. Notice that for ASR output the baseline is very low as due to recognition errors it is the case that only a single *NL* appears in both the training and the test data for two folds, in one case together with 8 and in the other case together with 3 possible *mr*s. By applying approximate matching as described in Appendix A the baseline can be increased to $F_1 = 19.6\%$.
In case of unsegmented phoneme sequences without recognition errors (grapheme-to-phoneme) still a high F_1 of 82.8% is obtained, indicating that the proposed segmentation mechanisms are appropriate. However, notice that in this case expressions for referents can also be found by coupling co-occurrence frequencies with a length bias (Gaspers & Cimiano, 2012).

Table 5.4.: Results for the application to ASR transcribed phoneme sequences.

Input	**Parser**	$\mathbf{F_1}$	**Precision**	**Recall**
Grapheme-to-phoneme	Baseline	18.9	43.6	12.2
Grapheme-to-phoneme	System	82.8	84.1	81.4
ASR phoneme	Baseline	0.3	50.0	0.2
ASR phoneme	System	64.2	66.0	62.6

Expectedly, when applied to ASR output, performance degrades. Yet, the results are promising, showing that in spite of recognition errors and ambiguity at the semantics level, it is still possible to learn a semantic parser which can be successfully applied to understand several unseen utterances as indicated by the large increase in F_1 compared to the baseline. In fact, relating back to the previous section, the resulting F_1 of 64.2% is even higher than the best value achieved in case of a simulated WER of 10% for word transcriptions. Thus, the results indicate that in the SLU task at hand, a speech recognizer yielding a WER of less than 10% is needed in order to yield results in the word-based setting which are comparable to those obtained in the phoneme-based setting. Recall from the previous section that a WER of 9.4% averaged over all folds can be achieved when applying a word-based in-domain ASR. Hence, results for the phoneme-based setting are comparable to those which can be expected when applying an in-domain word-based ASR which, however, in contrast to the phoneme recognizer, was built using in-domain training data. A main reason for this result may be the fact that in case of phonemes no restrictions concerning possible words – and thus meanings – are given by the ASR. Instead, a meaning can be determined at the whole-sentence level. In particular, while when working with words – as described in the previous section – a single recognition error may yield an incorrect parse for a given utterance, when working with phonemes a given utterance containing several errors (the PER is much higher than 10%) might still be parsed correctly due to the fact that often subsequences and sequences containing errors are still sufficient for determining a correct meaning (cf. Section 5.4.6). This finding is also in line with recent research showing that SLU (with respect to determining concepts in speech) performed on phoneme lattices can indeed yield comparable or even slightly better results than on word lattices (Svec et al., 2013, cf. Section 2.2).

5.4.5. On the role of knowledge about syntactic patterns in segmentation

Our system determines alignments and segmentations given a list of utterances both bottom-up based on knowledge about acoustic morphemes and top-down by including information of previously learned syntactic patterns. In this section we investigate the influence of top-down information of syntactic patterns on segmentation performance and language learning.
Recall that the number of runs for re-estimating alignments using top-down information and subsequently inducing the parser again is determined by the system based on the accumulative alignment score (cf. Section 5.3). In the previous experiments, for three folds the number of re-estimation steps determined and performed by the learning algorithm was four, while three were performed in case of the forth fold. Fig. 4 illustrates the change in F_1 over the number of re-estimation steps; Step 0 corresponds to performing the bottom-up learning step only.

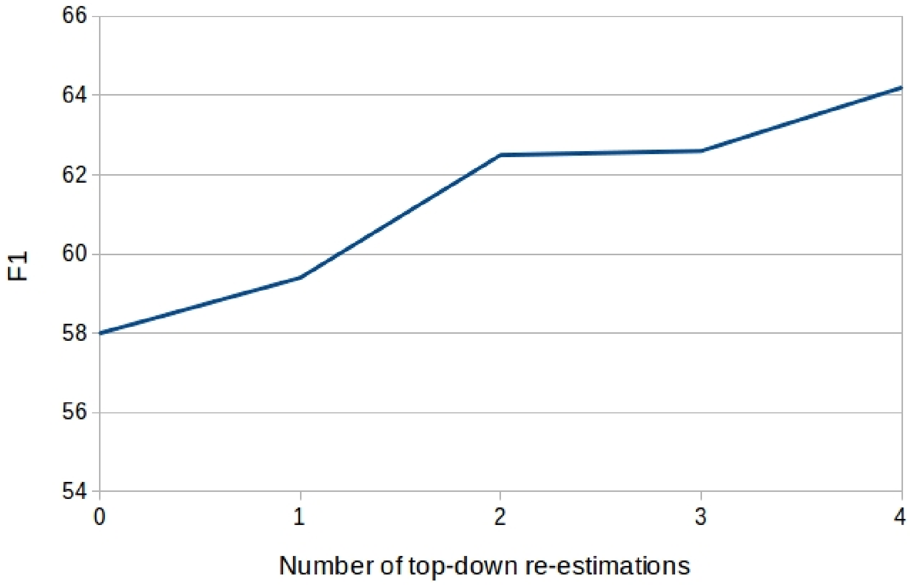


Figure 4.: F_1 over the number of top-down re-estimations.

The diagram reveals that utilizing syntactic knowledge for re-estimating alignments yields improved parsing performance, indicating that knowledge of syntactic patterns can indeed yield useful segmentation cues and improve language learning performance. With a value of 64.2% applying the top-down step (repeatedly) yields an improvement of about 6% absolute over 58.0% which are achieved when applying the bottom-up step only.

However, while the improvement in F_1 indicates that information of syntactic patterns can yield improved segmentation performance, it does not directly imply that more boundaries were detected (completely) precisely. For instance, segmenting a sequence “p r= p @ l @ l @ v a n k I k s” as “p r= – p @ l @ l @ v a n – k I k s” instead of “p r= p @ l – @ l @ v a n – k I k s” might allow correct parsing even though in both cases “p r= p @ l @ l @ v a n” has not been segmented correctly out of the utterance. Therefore, we also computed the percentage of correctly segmented sequences referring to players (see Section 5.4.6 for the computations) over the number of re-estimation steps. Results are presented in Fig. 5.

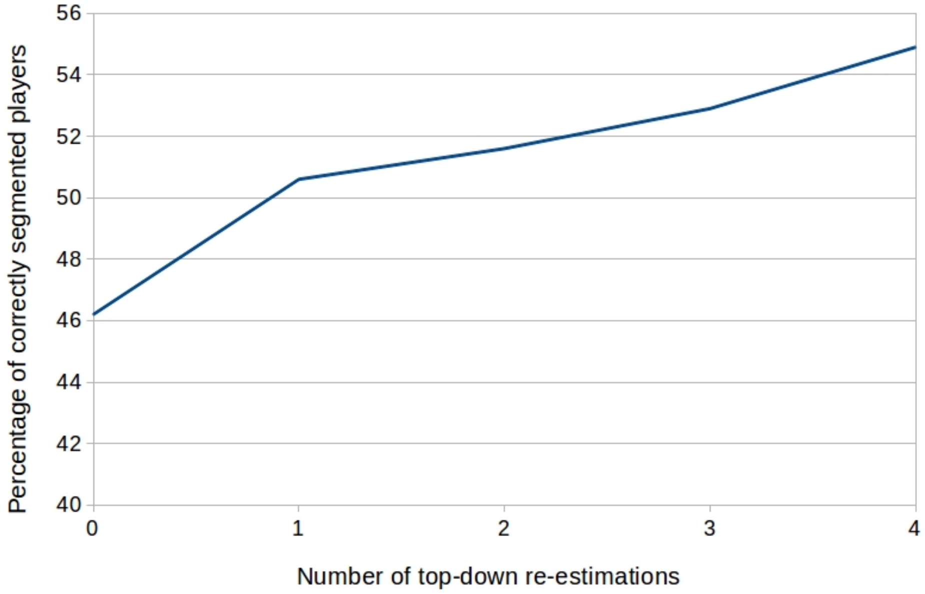


Figure 5.: Percentage of correctly segmented players over the number of top-down re-estimations.

As can be seen, repeated re-estimation of alignments using top-down information improves not only F_1 but also yields the detection of a larger percentage of sequences referring to players precisely.

Taken together, our results thus suggest that knowledge of previously learned syntactic patterns at the whole-utterance level, which is typically not utilized in algorithms for segmentation, can indeed provide useful segmentation cues, enabling improved segmentation and language learning performance.

5.4.6. Phonemic variation and its effects on parsing and learning

In contrast to working with written text, several sequences exist which express the same word or sequence of words when working with phonemes due to recognition errors and different pronunciations. In the following, we will investigate phonemic variation and its effects on parsing and learning. In particular, we performed experiments with respect to a subset of the semantic referents, i.e. the players, with the exception of *pink1* and *purple1*. We focused on these referents because in case of textual input they are always referred to by concatenating their team color with their number and thus correspond to the same spoken words, allowing us to explore how many different phoneme sequences exist in the data for the same spoken word(s) in their case.
In order to extract which phoneme sequences referring to players appear in the ASR transcribed data, start and end times for words referring to player names were annotated manually in the spoken RoboCup data using ELAN (Sloetjes & Wittenburg, 2008). By time-aligning the ASR transcriptions and the ELAN annotations we then extracted the ASR transcriptions for each of the referents and counted their frequencies (the first and the last phoneme for each sequence correspond to those phonemes having an overlap with the annotated start or end point in cases where annotation boundaries do not correspond to phoneme boundaries). The resulting number of different phoneme sequences for the same spoken words is quite high with an average of 84.5 different phoneme sequences for each player, ranging from 33 (*pink3*) to 161 (*purple11*) sequences for a single player with several sequences appearing only infrequently, i.e. once or twice. The most frequent phoneme sequence for each player is given in Table 5.5.

The high number of different phoneme sequences referring to the same player highlights the challenge when working with ASR transcribed phonemes compared to written text for which the same player is always expressed by the same word(s) in case of the inspected referents. The high amount of variation is especially an issue when estimating co-occurrence frequencies for establishing initial form-meaning mappings. When applied to textual input, the system directly associates the words referring to players correctly with their corresponding semantic referents. By contrast, when applied to phonemic input, the system typically associates several sequences with each referent, with boundaries often not corresponding to actual word boundaries. To illustrate this behavior, we computed association scores on the complete training dataset and present a sequence having maximal association score for

Table 5.5.: Sequences appearing most frequently in the data for a number of semantic referents, along with the top entry in the initial lexicon and the sequence extracted for the referent most frequently by the system.

Semantic Referent	**Sequence most frequently annotated manually**	**Initial top lexical entry**	**Sequence most frequently aligned by the system**
pink2	p I N k @ t u	u p { s I	p I k j u
pink3	p I N k T r i	N k T r i	p I N k T r i
pink4	p I N k f A r	I N k f A r	p I N k f A r
pink5	p I N k f AI v	N k f AI v	p I N k f AI v
pink6	p I N k s I k s	k s I k s	p I k s I k s
pink7	p I N k s E v @ n	k s E v @	p I N k s E v @ n
pink8	p I N k EI t	I N k EI t	p I N k EI t
pink9	p I k s n AI n d	n AI n d k	p I k s n AI n d
pink10	p I N k s t E n	k s t E n	p I N k s t E n
pink11	p I N k @ l E v @ n	k @ l E v	p I k @ l E v @ n
purple2	p r= p @ l t u	p @ l t u	p r= p @ l t u
purple3	p r= p @ l T r i	p @ l T r i	p r= p @ l T r i
purple4	p r= p @ l f A r	p @ l f A r	p r= p @ l f A r
purple5	p r= p @ l f AI v	p @ l f AI	p r= p @ l f AI v
purple6	p r= p l s I k s	l s I k s	p r= p l s I k s
purple7	p r= p l s E v @ n	l s E v @	p r= p l s E v @ n
purple8	p r= p l EI t	p r= p l EI	p r= p l EI t
purple9	p r= p U l AI n	r= p @ l AI	p r= p @ l AI n
purple10	p r= p @ l t E n	p @ l t E	p r= p @ l t E n
purple11	p r= p @ l E v @ n	p @ l E v	p r= p @ l E v @ n

each of the referents in Table 5.5. As can be seen, these sequences do not correspond to actual words or sequences of words. However, they typically correspond to subsequences of bigrams actually referring to the corresponding player and are hence useful for bootstrapping the parser because the correct *mr* can be found and aligned.

Recall that entries in the initial lexicon are not directly inserted into the parser's lexicon. The initial lexicon is only used along with BVE to compute a segmentation for a given utterance by aligning subsequences and their potential meanings at the whole-sentence level. Then, subsequences aligned with semantic referents are taken as entries for the parser's lexicon. In order to investigate what sequences were aligned with the players most frequently by the system, we computed the frequency of sequences expressing players in alignments summed up over all folds. The most frequent sequence for each player is presented in Table 5.5, showing that there is an impressive match between the sequences referring to players which appear most frequently in the data and the sequences which were most frequently aligned with the players by the system. Further, as an example concerning the alignment of different sequences for the same player, Table 5.6 lists the top five most frequent sequences for player *purple10* in the data and as aligned by the system, showing that the system can determine several sequences for a player matching those sequences actually appearing frequently in the data.

Table 5.6.: Top five most frequent sequences in the data and aligned by the system exemplary for *purple10*.

	System	**Data**
1	p r= p @ l t E n	p r= p @ l t E n
2	p r= p @ l t E m	p r= p @ l t E n t
3	p r= p @ l t E n t	p r= p @ l t AI m
4	p r= p @ l t AI m	p r= p @ l t E n d
5	p @ l t E n	p r= p @ l t E m

On average, the system aligned 121.6 sequences with each referent which is more than the number of sequences actually expressing players in the data. Moreover, only about 31% of the sequences appearing in the data are also found by the system. This is the case because especially sequences appearing only infrequently are not always detected or detected only partially. In particular, – especially with respect to infrequent sequences – the system often does not determine the word boundaries exactly. In order to investigate boundary detection performance in more detail, we computed the percentage of correctly segmented sequences referring to players (of the number of all sequences that the system segmented as mapping to players). Because annotation boundaries in the ELAN-files often do not correspond to actual phoneme sequences we allowed a differing phoneme at the start and end of sequences for matches for these computations. The resulting value of 54.9% averaged over all folds indicates that boundaries are indeed often not detected completely precisely. For instance, "t @" is often added in front of player names. This may, at least to some extent, be an artifact of the dataset. Since all player names start with a "p" and they are often preceded by the word "to", BVE frequently pre-segments phoneme sequences expressing "to p" and hence "t @" might be detected as belonging to a sequence referring to a player or "p" might be omitted. However, not determining boundaries exactly is in general no problem with respect to parsing because such errors may not prevent correct parsing as long as both, pattern and referents, can still be identified correctly which is often the case given only subsequences. In fact, even if sequences are not segmented (exactly) correctly they may still provide an additional benefit in parsing compared to parsing with a number of top entries only. Notice that by parsing with a number of top entries only, still different sequences can be found due to applying approximate matching, but approximate matching may also yield incorrect results (cf. Section 5.4.4). In order to investigate the potential benefit of taking all sequences into account during parsing (as we do), we run parsing using different numbers of sequences for semantic referents. In particular, we performed parsing by choosing 1, 10, 20, and 30 top sequences (according to the association score) for each referent only; all further sequences expressing these referents were omitted. For predicates all sequences were used for parsing, because in their case even the written data often includes a large number of different patterns referring to them. Thus, omitting sequences would not just leave out different sequences expressing the same words but different sequences of words. Notice that this experiment includes all semantic referents, not just the players. Results are presented in Table 5.7.

Table 5.7.: F_1, precision and recall when taking a different number of top sequences into account for each semantic referent during parsing.

Number of sequences	$\mathbf{F_1}$	**Precision**	**Recall**
1	48.4	53.0	44.7
10	58.9	61.0	57.0
20	62.1	64.1	60.3
30	62.8	64.6	61.0
all	64.2	66.0	62.6

As can be seen, by using the single “best” sequences only, F_1 decreases about 16% absolute compared to including all sequences. By including more different sequences for parsing, F_1 increases but even when taking the 30 top sequences into account, performance is still slightly lower compared to taking all sequences into account, even though the additional sequences are typically infrequent and incorrectly segmented. Thus, while the system can determine the best/most frequent sequences (cf. Table 5.5 and Table 5.6), parsing performance benefits from taking further sequences into account. This may be the case because the system may learn how the speaker actually pronounces words and what errors are added by the ASR. This may be especially an advantage with respect to phonetically similar bigrams such as “purple eleven” and “purple seven”, i.e. knowing the different sequences referring to each of them may prevent mapping them incorrectly onto each other by applying approximate matching. Thus, even if several incorrectly segmented sequences are learned, these may still help in determining the correct meanings.

5.4.7. Application for the induction of semantic speech recognition grammars

In Section 5.4.3 we have explored word-based speech recognition of the spoken RoboCup data under the assumption that written data for training a standard trigram model were available. For each fold, we used written training data for that purpose and performed speech recognition on the spoken test data for the remaining game. However, – relating back to Section 2.2 – an ASR can also be applied by using a grammar as the language model, and Wang & Acero (2006b) have shown that applying the same grammar for speech recognition and understanding can yield improved understanding performance compared to applying a standard n-gram model with the ASR, since dependencies between acoustics and semantics can be captured. Yet, their grammars were learned in a supervised fashion. In general, semantic speech recognition grammars are typically created manually or induced automatically in a supervised learning setting (cf. Section 2.2). Grammar creation in the former case requires human effort and often extensive domain and/or linguistic knowledge, while in the latter case typically large amounts of labeled training data are needed, making this approach rather impractical for several (real-world) applications.

Addressing this issue, in this section we explore the utility of weak supervision in the form of perceptual context information for the induction of speech recognition grammars. Assuming the availability of training data in the form of textual utterances or manual transcriptions of speech coupled with ambiguous context information, we transform grammars learned by our system into a grammar format applicable with a speech recognizer, and we explore recognition and subsequent semantic parsing for applying these grammars compared to applying standard n-gram models and purely syntactically motivated grammars as the language model. We compare performance of our semantically motivated grammars to that of purely syntactically motivated ones to explore whether contextual information can be beneficial for the induction of rules, i.e. whether it is useful to ground grammars for speech understanding. It must be noted that while applying n-gram models as LMs is common, this is not the case for grammars learned in an unsupervised fashion. Relating back to Section 2.2, for application in SLU, i.e. with respect to semantic parsing, automatically induced grammars are typically postprocessed manually which we refrain from doing.

Figure 6.: A subset of weighted speech recognition grammar rules.

```
public <utt> = /6/ <ref> again passes to <ref> | /199/ <ref> kicks to <ref> | ...
<ref> =  /15/ pink goalie | /132/ pink nine | /10/ pink one | ...
```

Creation of speech recognition grammars

Semantic speech recognition grammars were built given a semantic parser by transforming all rules with an occurrence > 1 into JSpeech Grammar Format (JSGF)[7]. Resulting grammars consisted of rules representing the parser's inventory of syntactic constructions as well as its lexicon. In case of the inventory of syntactic constructions, alternative expansions of learned syntactic patterns were defined, and in case of the lexicon alternative expansions of learned lexical units were defined. In particular, with respect to the lexicon we defined a rule <ref> which comprised the learned lexical units. With respect to syntactic constructions we defined a rule <utt> which comprised the patterns. Slots in syntactic patterns were replaced by <ref>, allowing lexical units to appear at those positions. In grammar creation, we also investigated the influence of occurrence frequencies of syntactic patterns and lexical units to enhance grammatical rules with weights. To examine the influence of weights, we created both weighted and unweighted grammars. When using weights, rules were weighted by using occurrence frequencies, i.e. the frequency which was observed for a pattern or lexical unit in the alignments created by the system. Hence, weights for patterns and lexical units observed less frequently in alignments were smaller, indicating that they were less likely to be spoken. These weights were used by the speech recognizer during recognition; using an unweighted rule corresponds to using a weight of 1.
An example illustrating a subset of two weighted rules is provided by Fig. 6.

Notice that resulting JSGF grammars did not explicitly contain semantic information but their induction was driven by semantic information. This is the case because a mapping to semantics was not needed during recognition since we explore a two-stage approach where parsing is performed after recognition, allowing the inclusion of further LMs during recognition. However, because both parsing and understanding are performed using the same grammar – where semantic information is ignored by the LM – it is also conceivable to induce a semantic grammar that directly maps ASR output into semantic representations. In that case we would define

[7] http://www.w3.org/TR/jsgf/

Figure 7.: A subset of a purely syntactic speech recognition grammar rules obtained via ADIOS

```
public <utt> = ( <P36> | <P37> shoots | pink <P2> three | pink <P22> |...
<E0> = eleven | goalie | four | ten | five | nine | two | eight | one | ...
<E1> = ten | six | three | seven | eleven | five | four | nine;
<E2> = backward | back | out | laterally | forward;
...
<P0> = <E0> kicks to pink;
<P1> = <E1> passes <E2> to purple;
...
```

individual rules for different predicates comprising patterns learned as referring to them and define semantic tags corresponding to the predicates. Similarly, semantic tags would be defined for lexical units, and the mapping would be incorporated by defining positions. For instance, "<ref> again passes to <ref>" would be defined as "<ref1> again passes to <ref2>" with both <ref1> and <ref2> referring to a rule grouping players.

To induce syntactically motivated grammars in an unsupervised fashion we applied the ADIOS algorithm (Solan et al., 2005, cf. Section 2.6) to the raw training data, i.e. we did not make use of perceptual context information. Recall that ADIOS is an unsupervised algorithm which induces syntactic patterns which are represented in the form of a graph, patterns and equivalence classes. We transformed the output by ADIOS into JSGF. Resulting grammars contained rules comprising paths which were generalized by ADIOS. Further, for each induced equivalence class and pattern a rule was added to the grammar. An example illustrating a subset of a grammar induced using ADIOS is presented in Fig. 7.

Results

We explored recognition i) using grammars, ii) using standard trigram models, and iii) using both jointly. In the latter case a trigram model was applied in case of out of grammar (OOG) utterances, as these might still be parsed subsequently by applying approximate matching; in these experiments we always apply matching with a Levenshtein distance of 1. Notice, however, that for our experiments we did not apply both LMs at a time but combined the output of two recognizers for further processing. Notice further that most speech recognizers can only be applied using either a recognition grammar or an n-gram model at a time, but one can assume that two recognizers might be configured to run in parallel. Specifically, we explored seven different (combinations of) LMs for speech recognition, and investigated semantic parsing performance on their resulting transcriptions, in particular:

1. Trigram
2. Rule-based semantic speech recognition grammar without weights
3. Rule-based semantic speech recognition grammar without weights and back off to trigram for OOG utterancess
4. Rule-based semantic speech recognition grammar including weights
5. Rule-based semantic speech recognition grammar including weights and back off to trigram for OOG utterances
6. Rule-based syntactic speech recognition grammar
7. Rule-based syntactic speech recognition grammar and back off to trigram for OOG utterances

As in Section 5.4.3, we performed 4-fold cross-validation on the four RoboCup games. For each fold, learning semantic parsers and creation of language models was performed using the ambiguous *written* training data for three games and the *spoken* gold standard for the forth game for testing.

Speech recognition was performed using each of the seven (combinations of) LMs listed before individually; lexicon and acoustic models were always the same, i.e. the ones already applied in Section 5.4.3. Speech recognition results (on the test data) with respect to the WER averaged over all folds are presented in Table 5.8. As can be seen, with a rather low error rate of 7.1% applying trigram language

Table 5.8.: Speech recognition results using different language models.

Applied language model(s)	**WER**
Trigram	**7.1**
Semantic grammar w/o weights	15.55
Semantic grammar w/o weights + trigram back off	12.63
Semantic grammar inc. weights	17.15
Semantic grammar inc. weights + trigram back off	10.88
Syntactic grammar	18.98
Syntactic grammar + trigram back off	13.98

models yields the best results. In case of applying semantically motivated recognition grammars the WER increases. It must be noted that in cases in which no back off models were applied, this is to some extent due to OOG utterances (as these yield several deletions compared to the reference data). Yet, the OOG-rate is rather low, i.e. averaged over all folds 8.6% and 4.1% when using grammars with and without weights, respectively. However, even in cases where OOG utterances are recognized by applying trigram language models the WER is higher compared to applying trigram language models only. Notably, these results were not consistent across folds. In case of two folds – i.e. one half of the folds – the WER actually decreased when combing a semantically motivated grammar including weights with a trigram language model compared to applying the trigram language model only, thus indicating that combining semantically motivated grammars learned with weak supervision with trigram models can also yield improved recognition performance over applying trigram models only in some cases. In these experiments, applying syntactically motivated grammars yields the worst results with a WER of 13.98% when combined with a trigram model.

For each fold, ASR transcriptions were parsed using the semantic parser learned on the training data for that fold. For comparison, parsing performance was also determined on normalized gold standard data. Results are presented in Table 5.9.

Table 5.9.: Semantic parsing results on written text and on speech transcribed using different language models

Written text (reference)			
	F$_1$	**Prec.**	**Recall**
Normalized text	87.26	94.28	81.42
Speech			
Applied language model(s)	**F$_1$**	**Prec.**	**Recall**
Trigram	78.36	90.34	69.4
Semantic grammar inc. weights	84.18	88.7	80.18
Semantic grammar inc. weights + trigram back off	**84.46**	87.53	81.64
Semantic grammar w/o weights	82.24	84.83	79.84
Semantic grammar w/o weights + trigram back off	82.37	84.67	80.21
Syntactic grammar	70.86	76.09	66.35
Syntactic grammar + trigram back off	71.27	75.37	67.65

As can be seen, when applying trigram language models F_1 degrades about 9% absolute compared to parsing written text (reference), yielding 78.36%, even though the WER is rather low with a value of 7.1%. By contrast, when applying a semantically motivated recognition grammar including weights, performance degrades only about 3% absolute, even though the WER is higher in this case. Moreover, including occurrence frequencies as weights in the recognition grammars yields improved performance compared to using unweighted grammars.

Notably, the decrease in performance when applying a weighted semantically motivated recognition grammar (+ trigram back off) compared to performance on the reference data is mainly due to a decrease in precision. Here it must be noted that the high values in F_1 were achieved without performing any optimization of (recognition) parameters. In the performed experiments, the probability for OOG utterances was rather low, and thus utterances were matched incorrectly by the ASR which were actually not covered by the grammar, yielding both recognition and subsequent parsing errors. However, these parameters can be tuned, likely increasing precision and F_1 even further (and probably also speech recognition performance, i.e. WER). Again, in these experiments performance is worst when syntactically motivated grammars are applied for recognition, yielding at most 71.27% in F_1 when applied together with a trigram model.

In case of both types of grammars, applying a back off trigram model yields only little improvement in parsing performance, though this may be also – at least to some extent – due to not tuning recognition parameters. If the ASR would be tuned to reject more OOG utterances correctly, these utterances might instead be recognized by a trigram model and probably parsed correctly by applying approximate matching.

The results show that, in line with previous research (Wang et al., 2003; Bayer & Riccardi, 2012), a lower WER does not necessarily yield better understanding results, i.e. in our experiments parsing performance is not directly dependent on the WER but rather on the type of errors made. In particular, as mentioned previously, for semantic parsing it is important that words carrying semantics are recognized correctly. Applying the semantically motivated grammars may have been beneficial in recognizing the semantic referents correctly because the system can explicitly learn them and their appearances in certain patterns in contrast to the trigram model. In particular, if an utterance "pink nine passes the ball to pink seven" appears during recognition and "pink seven" has not been observed in the context of the preceding words during training, then the n-gram would assign a low probability, probably yielding a recognition error such as "pink eleven". By contrast, in case of semantic grammars the system can learn that the utterance is an instantiation of a pattern "*player* passes the ball to *player*" and that all players can appear in the contained slots, thus making the appearance of the example utterance more likely. In order to detect and generalize over semantic referents correctly, using weak supervision in the form of perceptual context information appears to be beneficial. In particular, the purely syntactically motivated grammars induced by ADIOS did not, for instance, induce semantic classes grouping players but rather induced several equivalence classes grouping numbers and team colors. While a different algorithm for the unsupervised induction of syntactic patterns may be explored as well, we assume that the results would be similar.

We have also investigated weighting rules for semantically meaningful lexical units, i.e. the probability for the occurrence of player names can be increased according to their occurrence frequencies in alignments, thus making their recognition more likely. Our results indicate that by weighting semantically meaningful sequences, performance is improved, possibly because more words carrying semantics are recognized correctly, even though words carrying no semantics like “forward” or “backward” might be confused which, however, may not prevent correct parsing.

In general, while in SLU research data-driven approaches typically explore cascading systems (Deoras et al., 2013), in line with previous work (Wang et al., 2003; Bayer & Riccardi, 2012), our results indicate that joint models yield improved parsing performance, even though word recognition performance may decrease. Yet, our results also indicate that combination of a semantic grammar with a standard trigram model during speech recognition can reduce the word error rate in some cases compared to applying the trigram model only. Further, in line with Wang et al. (2003) and Bayer & Riccardi (2012) the results emphasize that capturing semantic information in a language model applied during speech recognition is beneficial for subsequent semantic parsing, since by this the ASR can be tuned towards recognizing words carrying semantics more precisely which is important with respect to parsing performance.

5.5. General discussion

In this chapter, we have presented an approach which learns lexical units and syntactic constructions using ambiguous non-linguistic context information directly from speech without word transcriptions. Relating back to Section 2.8, while several approaches have addressed learning with ambiguous context information (e.g. Chen et al., 2010; Börschinger et al., 2011; Kwiatkowski et al., 2012), and learning from speech without word transcriptions has been explored as well (e.g. Roy & Pentland, 2002; Taguchi et al., 2009; Yu et al., 2005; Cerisara, 2009), in the former case utterances have been represented by sequences of words and in the latter case learning has focused on linguistic units of rather low complexity. To the best of our knowledge, the learning scenario addressed in this chapter has not been investigated before but is of great interest with respect to the development of adaptive systems which are able to perform “life-long” learning and developing low-cost language understanding capabilities directly from speech. The latter is, for instance, of interest with respect to building spoken language understanding systems (and/or (pronunciation) lexica) for under-resourced languages, since much less training data – and in particular no in-domain data – are needed for building an ASR. Here it must be noted that while in this chapter we have utilized perceptual context information for parser estimation, a spoken language understanding component can of course also be built by applying the system to examples of spoken utterances which are manually annotated with their correct meaning representations (i.e. by investigating a supervised learning scenario as traditionally investigated for data-driven semantic parser induction, albeit with respect to speech).

The ability to learn language from spoken utterances coupled with ambiguous context information appears to be particularly interesting with respect to developing language understanding components for artificial agents which ideally should be grounded in the real world, such as robots. Assuming that an artificial agent is equipped with the ability to extract structured representations from its environment, it could log the utterances it hears along with the actions and objects it observes. Based on these data, lexical units and syntactic constructions could then be extracted by applying an approach as the one presented in this chapter, adapting the robot’s vocabulary and syntactic constructions over time, also covering unseen items. Especially with respect to this application scenario, exploring online learning appears to be an interesting point for future work.

Due to the increased complexity in the learning scenario, in this chapter we did not attempt to explicitly model child language acquisition but focused on solving the learning task with respect to maximizing performance (with respect to F_1 which is the main measure for evaluating algorithms on the RoboCup corpus). Cognitive plausibility and modeling ideas from psycholinguistic theories was not explicitly addressed. However, we investigated a learning task also faced by children who learn language by being exposed to *spoken* language in some environment, and besides exploring offline learning, the proposed learning mechanisms may also be regarded as cognitively plausible. In fact, with respect to application to words, the learning mechanisms explored for the system are similar to those investigated in the cognitive model presented in Chapter 3. In particular, in both cases we applied the same cross-situational learning mechanism at different levels (though different mechanisms are explored in the model and the system) and in both cases generalization is performed based on previously acquired lexical knowledge. However, the generalization mechanisms differ: while in the cognitive model the induction of slots (initially) requires utterances showing minimal variation with respect to the corresponding position, this is not the case in the system presented in this chapter. Further, in the system we do not consider sets of elements but simply store all lexical units and allow their appearance in all syntactic slots. This was done because for the computational model it was observed that at later learning steps language learning performance of the model increased in that utterances could be incorporated directly as generalized paths, and it was also possible to directly infer the correct meaning (cf. example 9). Due to the increased complexity in the learning scenario we enabled fast language learning and hence captured the model's behavior observed later on by directly generalizing utterances based on lexical units and their semantics without requiring utterances showing variation. Instead, we introduced alignments between form and meaning.

Further, in the phoneme-based system explored in this chapter we applied an unsupervised algorithm, i.e. BVE, to pre-segment utterances into (sub)word-like units. Relating back to Section 2.4, the algorithm is based on two so-called experts which vote for segmentation points: one votes for segmentation points after chunks having low internal entropy and one votes for segmentation points after chunks having high boundary entropy. With respect to modeling child language acquisition, applying such an algorithm would not be implausible since research suggests that infants are able to utilize predictability statistics to determine word-like units, at least from artificial languages (Saffran et al., 1996) and modeling the use of predictability statistics based on the entropy for learning lexical segmentation has also been addressed previously (Çöltekin & Nerbonne, 2014). In fact, it appears to be beneficial to take further cues into account for segmentation since combining the information of distributional cues to word boundaries can improve predictions (Jarosz & Johnson, 2013). With respect to child language acquisition, research has shown that children are sensitive to a number of different cues to word boundaries such as phonotactics (Mattys et al., 1999), coarticulation (Johnson & Jusczy, 2001) or prosodic stress patterns (Gladfelter & Goffman, 2013). Such cues may also be beneficial in order to increase the segmentation accuracy of our system further. As a first step, we will focus on the integration of prosodic cues.

However, while when applied to text the system yields improved language learning capabilities compared to the cognitive model, precision decreases. In case of the computational model values for F_1 of up to 84.3% with 96.6% in precision were obtained, while in case of the system values of up to 89.09% in F_1 with 93.38% in precision were obtained. However, values of up to 95.97% in precision were achieved for lower values of F_1.

When working with ASR phoneme transcriptions precision is even lower compared to working with words, yielding a value of only 66%. Here it must be noted that the RoboCup corpus may be complicated in that several sequences expressing referents have subsequences in common. That is, most expressions for players start with either the prefix "purple" or "pink" followed by a number. Thus, what mainly distinguishes referents are the numbers. Due to recognition errors sometimes only subsequences expressing numbers were associated, yielding segmentation and parsing errors because the prefix is needed for determining the correct referent. Furthermore, ASR errors yielded sequences which were phonetically most similar to lexicon entries expressing different referents. For instance, if "I" is deleted and "l" substituted in "p r= p @ l I l E v @ n", then "p r= p @ l s E v @ n" may be returned as lexical entry. Yet, because we explore learning by utilizing perceptual context information, one may assume that perceptual context information is also available during application and utilized to correct such errors, e.g. by increasing probabilities for referents observed during parsing. Moreover, further types of contextual information might be utilized. For instance, relating back to Section 2.8 systems have been explored which learn language by directly interacting with the environment, e.g. by exploring provided feedback in the framework of reinforcement learning. This kind of learning appears to be particularly interesting with respect to embodied systems which are able to interact with their environment. In general, with respect to possible application on an embodied system, increasing precision appears to be a relevant aspect for future work since in that case it is important that the system can assess its confidence in the acquired knowledge and its interpretation of an utterance accurately. This might, for instance, be also approached by incorporating measures similar to those explored within the computational model presented in Chapter 3.

In this chapter we evaluated induced grammars on read speech, and hence spoken utterances mainly corresponded to well-formed sentences containing only few disfluencies such as hesitations. Thus, our input does not reflect spoken language faithfully. In particular, spoken language is in general not as well-formed as written language since people may follow syntactic constraints less strictly (Wang et al., 2011). However, with respect to a potential application on a robot as described previously, this may not be an issue. In that case we would expect an operator to use instructions in the form of (typically) well-formed sentences, thus corresponding to the spoken utterances investigated in this chapter. Further, it has been shown that in robot-directed speech higher pitch, hyper-articulation and more loudness can be identified, indicating that its acoustic characteristics are also more similar to read speech than to spontaneous speech, as the latter clearly has a tendency towards hypo-articulation (Kriz et al., 2010). However, future work may also focus on making the induced grammars more robust to noise, in particular to unexpected input. For example, fillers modeling garbage words can be incorporated (automatically) into an existing grammar (Yu et al., 2006). Moreover, semantic parsing might be improved. In particular, in this chapter we have only taken the best result of the speech recognizer into account. However, automatic speech recognizers can also output an n-best list of recognition results. Parsing performance might be improved by applying the parser to further results in this list, at least in cases where the best result cannot be parsed by the system (notice that making use of n-best lists or lattices of ASR hypotheses in order to improve spoken language understanding performance has been explored before (De Mori, 2011)). In doing so, further parsing strategies could be explored, in particular more sophisticated ones. For instance, with respect to the word-based setting, phonetic similarity could be taken into account. Specifically, the entries in the word-based learned parser and words contained in the alternatives in the n-best list returned by the ASR could be transformed into sequences of phonemes by applying grapheme-to-phoneme conversion. Subsequently, a phonetically (most) similar match, i.e. a syntactic pattern instantiated by one of the ASR transcribed alternative utterances, can be determined and returned as the result, at least in cases where the phonetic similarity is high.

With respect to Sections 5.4.3 and 5.4.4, our results indicate that the proposed method, which works with a task-independent phoneme recognizer, can yield results comparable to the application of an in-domain word-based speech recognizer. However, – as mentioned previously – it has several advantages. In particular, besides low costs for training, it supports the development of systems which are adaptive in that they can acquire novel lexical units and syntactic patterns during application. This would not be the case for the application of a word-based speech recognizer working with a predefined lexicon. Since ASRs can only recognize units stored in their lexicon, in that case the parser would be restricted to acquire meanings of words stored in the lexicon. However, if suitable training data in the form of textual input or manual transcriptions of speech coupled with concurrent context information are available, better parsing performance of spoken utterances might by achieved by applying our system to this input (cf. Section 5.4.7). In fact, our experiments suggest that this even enables parsing of spoken utterances with a rather low loss in performance compared to parsing of text or manual transcriptions. Thus, initially building a system in this manner would be reasonable if suitable training data are available. The resulting system could then be combined with a phoneme-based component which could acquire lexical units and patterns not covered by the initial word-based grammar. This would yield both high parsing performance with respect to understanding utterances captured by the initial knowledge and also adaptation to novel input. Further, the word-based induced patterns may also be applied to determine novel words. For instance, assume that a word-based pattern is recognized by the ASR with the recognized words having high confidence but the confidence for a recognized lexical item appearing at the position of a slot is low. Then, one might assume that at this position actually a new word has been spoken which is not contained in the ASR's lexicon, at least if a novel semantic referent is observed concurrently. One might then resort to phoneme recognition and extract the sequence corresponding to the lexical unit based on timestamps and hypothesize it to be a new lexical item. Moreover, one might directly fast map this sequence onto a novel referent.

In addition, parsing performance might be improved further by investigating grammars based on different units, in particular phonologically motivated ones. Vale & Mast (2012) found that using phonologically motivated units, in particular the foot and the syllable, speech recognition grammars can yield improved parsing performance compared to applying word-based grammars. They concluded that applying a foot-syllable CFG appears to be a good choice for application in Ambient Assisted Living environments, i.e. an intelligent wheel chair based in an apartment in their case. However, these grammars were created manually. Aiming to reduce manual effort needed for grammar creation and to improve parsing performance of the grammars induced by our system, one of the main directions for future work will be to explore the data-driven induction of foot-syllable grammars by applying and extending learning methods presented in this thesis.

Our results have implications concerning further possible computational investigations of child language acquisition. In particular, while to date several models addressing language acquisition have been proposed, these models typically focus on a subset or certain aspects of language acquisition learning tasks. In doing so, they often assume other learning tasks, e.g. those of lower complexity, as already solved by the learner. For instance, models addressing the acquisition of grammatical constructions and their meaning (e.g. Kwiatkowski et al., 2012; Alishahi & Stevenson, 2008; Chang & Maia, 2001) typically learn from symbolic input. Assuming that the child is already able to segment a speech signal into a stream of words and to extract structured representations from the visual context, such models typically explore learning from sequences of words and symbolic descriptions of the non-linguistic context. By contrast, models addressing the acquisition of word-like units directly from a speech signal (e.g. Räsänen, 2011; Räsänen et al., 2009) have also been explored. Those, however, typically do not address learning of more complex linguistic structures. Taken together, lexical acquisition from speech and syntactic acquisition have been mainly studied independently of each other, often assuming that syntactic knowledge follows from knowledge of words. However, learning processes might actually be interleaved, and top-down learning processes may play an important role in language acquisition. In this chapter we have shown how top-down information of syntactic patterns can be utilized in order to improve boundary detection and language learning. Children may apply similar learning mechanisms. Their potential role along with the possible role of several top-down learning processes and their interaction with bottom-up learning mechanisms may be investigated in the framework of computational models addressing the learning task presented in this chapter since children also learn language by observing *spoken* utterances in some environment. However, – in contrast to the work presented in this chapter – it appears to be important to explicitly take cognitive plausibility into account, in particular constraints on the infant learner regarding memory. Moreover, it appears to be important to address phonetic acquisition instead of applying a phoneme recognizer, and in doing so the potential interaction between bottom-up and top-down learning processes could also be addressed. In fact, with respect to lexical and phonetic acquisition – which have traditionally also been studied independently of each other –, recent work (Martin et al., 2012; Feldman et al., 2009) has already shown how lexical information – even in rather rudimentary form – can support/boost phonetic acquisition. Thus, capturing several learning tasks in a unified model and studying the potential interaction of bottom-up and top-down learning mechanisms rather than focusing on sequential models for language acquisition appears to be important for future work concerning cognitive modeling of child language acquisition.

Addressing this issue, as mentioned previously, one of our main goals for future work is to explore how grounded syntactic patterns can be learned from sub-symbolic input, and we have already collected a dataset for this purpose (Gaspers, Panzner, et al., 2014). In the framework of the developed models we will then investigate the role of several top-down learning processes and their interaction with bottom-up learning processes.

5.6. Summary

In this chapter, we have presented an approach which learns a semantic parser in the form of a lexicon and an inventory containing syntactic constructions from speech. In particular, the parser is learned from spoken utterances transcribed by a phoneme recognizer, i.e. without making use of word transcriptions, coupled with ambiguous context information. Due to the additional segmentation task and noise in the form of recognition errors which must be tackled, this learning scenario is far more challenging compared to the word-based one explored in the previous chapters. We have therefore proposed a different method for generalization which is based on alignments between form and meaning in order to allow faster language learning, and we have taken further learning methods into account. For instance, we have introduced a top-down step in which alignments/segmentations are refined using top-down information of previously induced syntactic patterns. However, the system presented in this chapter is similar to the computational model presented previously in that in both cases we applied a cross-situational learning mechanism at different levels and in both cases lexical knowledge was used to bootstrap generalization.

We have presented empirical results showing that when applied to text, i.e. the RoboCup dataset also used for evaluating the computational model, a parser achieving state-of-the-art performance can be induced straightforwardly. Further, we have shown that when applied to spoken utterances, a parser can be induced which can be successfully applied to parse several unseen spoken utterances. In fact, our results even indicate that the results are comparable to those which can be expected when applying an in-domain word-based ASR, while not making a-priori restrictions concerning the vocabulary the parser can posses. Furthermore, our results indicate that while the system can correctly detect and segment the phoneme sequence appearing most frequently in the data for several referents, parsing performance can be improved by taking different phoneme sequences into account, even if several of them are incorrectly segmented and do not correspond to actual words.

Moreover, we have shown that in cases where training data in the form of text or manual transcriptions are available, the system can be applied successfully to induce semantic speech recognition grammars which enable semantic parsing of speech with a rather low loss in performance compared to parsing of correct transcriptions. Since semantic speech recognition grammars are typically created manually or learned in a supervised setting, making use of weak supervision provides a relaxation of manual effort needed for grammar creation.

We have also shown that top-down knowledge of syntactic patterns can yield useful segmentation cues, improving both boundary detection and language learning. Children may also make use of such cues during language acquisition and thus our results indicate that computational investigations exploring the potential role of syntactic information in segmentation with respect to child language acquisition might be an interesting point for future research.

Chapter 6

Summary

In this thesis, we have explored how language can be learned by observing natural language utterances coupled with concurrent ambiguous context information. Semantic information was represented symbolically using first-order logic formulas. We have considered two different learning settings, one with respect to utterances in the form of sequences of words and one with respect to spoken utterances. In the latter case, we applied a phoneme recognizer and thus – in contrast to the word-based setting – addressed learning without predefined lexical knowledge. Given the input, we attempted to acquire a lexicon and an inventory of rudimentary syntactic patterns; in both cases entries were acquired along with a mapping to their corresponding semantics.

In Chapter 3, we have addressed the word-based learning task by formalizing and modeling ideas from usage-based theories to language acquisition. In particular, we presented a computational and formal model for the gradual emergence of verb-specific slot-and-frame patterns. In the model, linguistic knowledge is represented in the form of an interrelated network comprising constructions at varying levels of complexity and abstraction.

The model is able to learn two types of constructions: (short sequences of) words and their meanings as well as bottom-up induced verb-specific slot-and-frame-patterns. In doing so, our model proposes uniform representational devices and learning mechanisms for all levels of constructions in order to determine an appropriate meaning out of ambiguous contexts. More specifically, all correspondences between form and meaning are modeled by associative networks, and linguistic knowledge captured by the model is measured based on the weights of connections contained in those networks. Within the scope of our language learning algorithm, observed natural language utterances are first incorporated into the network as a whole. Once sufficient knowledge is regarded as learned, the model starts to gradually induce slot-and-frame patterns. That is, the model searches for natural language utterances and already (partially) generalized patterns representing the same pattern. Roughly speaking, this is the case if the utterances under consideration show minimal variation in the surface structure, i.e. varying elements in one position, and these elements represent a set of elements corresponding to an argument slot in an associated predicate. Further, we explicitly built the fast mapping ability observed in children into the model by incorporating a disambiguation bias.

Our proposed model is in line with usage-based psycholinguistic theories stating that in early language acquisition children maintain an inventory of lexically-specific and item-based constructions which are gradually generalized by replacing concrete lexical items by slots which can be filled by (a restricted group of) words or short sequences of words. More specifically, it is represented in the form of an interrelated network of constructions at varying degrees of complexity and abstraction without assuming precoded linguistic knowledge. Knowledge emerges gradually from specific words to partially productive slot-and-frame patterns to fully productive patterns.

We provided empirical results on the RoboCup dataset showing that the employed learning and generalization mechanisms are appropriate in order to i) generalize beyond specific examples seen, while ii) not overgeneralizing, and to iii) assess confidence in the acquired knowledge accurately. In our experiments, the model's performance was highly precise and it achieved a large reduction in the number of stored patterns compared to the number of individual utterances observed in the input data. This in turn yielded understanding of several novel utterances, i.e. utterances not observed in the input. In line with findings from psycholinguistic studies with infants in the framework of usage-based theories, our model learns language gradually. Initially, in our experiments the model's generalization abilities were limited, but increased over the time course and finally converged, suggesting that during further processing of examples the employed mechanisms allow accurate learning without (severe) deterioration of the knowledge already captured by the network. Taken together, our model thus yields a compact and precise model of the input data generalizing well to unseen data. The model provides an interesting framework for future research in language acquisition research since it can be utilized for experiments aiming to shed light on the mechanisms at play during language acquisition.

In Chapter 4, we have presented an extension of the computational model to also capture the emergence of verb-general constructions. The induction of verb-general constructions builds on the induction of verb-specific constructions and similar learning mechanisms as those used for inducing verb-specific constructions are explored. In particular, verb-general constructions are learned in a bottom-up fashion based on verb-specific constructions only once verb-specific knowledge has been derived with sufficient confidence. Further, generalization occurs in an item-based fashion – albeit with respect to more complex structures – by searching for variation at a linguistic layer which has corresponding variation at a meaning layer.

Our model infers form-meaning mappings under referential uncertainty by applying the same cross-situational learning mechanism at different levels, implemented via associative networks. More specifically, in contrast to previous models exploring cross-situational learning, we apply the same cross-situational learning mechanism beyond simple word-referent mappings, i.e. between *NL* patterns/syntactic frames and actions, including thematic relations. Hence, our model can represent verb entries in the framework of these *NL* patterns, and – in line with children – store additional information about possible referents with verb entries. Both, information concerning possible referents and co-occurrences with different semantic frames are updated incrementally over time, enabling the acquisition of verb meanings and verb-general constructions starting from ambiguous contexts.

We have presented empirical results replicating findings from psycholinguistic studies with children with the model, showing how it can establish verb meanings under referential uncertainty. Moreover, we have shown how the model can learn verb-general constructions and how it can use this knowledge to create initial verb entries based on syntactic information alone. Thus, the model suggests possible learning mechanisms at play concerning the emergence of verb-general constructions and the representation of early verb entries by providing one formal explanation for the observed behavior. Future psycholinguistic studies may reveal whether children indeed apply learning mechanisms similar to those implemented in the model by testing its predictions.

In Chapter 5, we have presented an approach which learns a semantic parser in the form of a lexicon and an inventory containing syntactic constructions from speech input. In particular, the parser is learned with non-linguistic ambiguous context information directly from spoken utterances transcribed by a phoneme recognizer, i.e. without word transcriptions. Due to the additional segmentation task and noise in the form of recognition errors which must be tackled, this learning scenario is more challenging than the word-based one explored in the previous chapters. We have hence adapted and extended learning mechanisms explored within the framework of the computational model. More specifically, generalization is based on alignments between form and meaning in order to enable faster language learning. Further, we have introduced a top-down step in which alignments/segmentations are refined based on top-down information of previously induced syntactic patterns. However, the presented system is inspired by the cognitive model presented previously in that in both cases we applied cross-situational learning at different levels and in both cases initial lexical knowledge is used to bootstrap generalization.

We have presented empirical results showing that when applied to text, a parser achieving state-of-the-art performance can be induced straightforwardly. Further, we have shown that when applied to spoken utterances, a parser can be induced which can be successfully applied to parse several unseen spoken utterances. In fact, our results even indicate that the results are comparable to those which can be expected when applying an in-domain word-based speech recognizer, without making a-priori restrictions concerning the vocabulary the parser can process. Furthermore, our results indicate that the system can correctly detect and segment the phoneme sequences appearing most frequently in the data for several referents. Yet, they also indicate that parsing performance can be improved by taking different phoneme sequences into account, even if several of them might be incorrectly segmented and do not correspond to actual words.

Moreover, we have shown that in cases where training data in the form of text or manual transcriptions are available, the system can be successfully applied to induce semantic speech recognition grammars, allowing semantic parsing of speech with a rather low loss in performance compared to parsing of correct transcriptions. Since semantic speech recognition grammars are typically created manually or learned in a supervised setting, making use of weak supervision provides a relaxation of manual effort needed for grammar creation.

Our experiments have also revealed that making use of top-down knowledge of syntactic patterns can yield useful segmentation cues, improving both boundary detection and language learning. Children may also make use of such cues during language acquisition and our results hence indicate that computational investigations exploring the potential role of syntactic information in segmentation with respect to child language acquisition might be an interesting point for future research.

We have further discussed several possible extensions and relevant points for future work, and there are two major points for future research which we will begin with. First, we have addressed language learning with contextual information in symbolic form, i.e. with actions represented by means of predicate logic formulas, and hence our cognitive model and system are not grounded in the sense of Harnad (1990). In future work we will address an extension towards working with perceptually grounded representations of meaning, such as image or cognitive schemas, which may be derived from the visual context and utilized in the model and system instead of predicate logic formulas. Moreover, we will address learning from a speech signal without applying a speech recognizer. To address this issue, we have already collected a multimodel corpus designed with the main goal of allowing the evaluation of computational models that address the acquisition of rather complex grounded linguistic structures, i.e. syntactic patterns, from sub-symbolic input (Gaspers, Panzner, et al., 2014). In on-going work we use this dataset to explore how the model can be grounded. In the framework of the models developed to learn syntactic patterns from sub-symbolic input, we will investigate the role of several top-down processes and their interaction with bottom-up processes. In particular, we will investigate the potential role of knowledge about syntactic patterns on segmentation.

The second main direction for future research concerns prosody and phonologically motivated units. In particular, we will explore how prosodic cues can be utilized to improve segmentation and language learning and how grammars based on phonologically motivated units, in particular foot-syllable grammars which have been shown to yield improved parsing performance over word-based grammars in situated language understanding (Vale & Mast, 2012), can be acquired.

References

Aho, A. V., & Ullman, J. D. (1972). *The Theory of Parsing, Translation, and Compiling.* Prentice Hall.

Alishahi, A., & Chrupala, G. (2012). Concurrent acquisition of word meaning and lexical categories. In *Proceedings of the Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning* (pp. 643–654). Stroudsburg, PA, USA: Association for Computational Linguistics.

Alishahi, A., & Fazly, A. (2010). Integrating Syntactic Knowledge into a Model of Cross-situational Word Learning. In *Proceedings of the 32nd Annual Conference of the Cognitive Science Society.* Boston, Massachusetts, USA: Cognitive Science Society.

Alishahi, A., Fazly, A., & Stevenson, S. (2008). Fast mapping in word learning: What probabilities tell us. In *Proceedings of the 12th Conference on Computational Natural Language Learning* (pp. 57–64). Stroudsburg, PA, USA: Association for Computational Linguistics.

Alishahi, A., & Stevenson, S. (2008). A Computational Model of Early Argument Structure Acquisition. *Cognitive Science*, *32*(5), 789–834.

Arunachalam, S., & Waxman, S. R. (2010). Meaning from syntax: Evidence from 2-year-olds. *Cognition*, *114*, 442–446.

Aslin, R. N., Saffran, J. R., & Newport, E. L. (1999). Statistical Learning in Linguistic and Nonlinguistic Domains. In *Emergence of Language* (pp. 359–380). Hillsdale, NJ: Lawrence Earlbaum Associates.

Bannard, C., Lieven, E., & Tomasello, M. (2009). Modeling children's early grammatical knowledge. *Proceedings of the National Academy of Sciences*, *106*(41), 17284–17289.

Bayer, A. O., & Riccardi, G. (2012). Joint Language Models for Automatic Speech Recognition and Understanding. In *Proceedings of the IEEE/ACL Workshop on Spoken Language Technology.* Washington, DC, USA: IEEE Computer Society.

Beekhuizen, B., Bod, R., Fazly, A., Stevenson, S., & Verhagen, A. (2014). A Usage-Based Model of Early Grammatical Development. In *Proceedings of the ACL Workshop on Cognitive Modeling and Computational Linguistics.*

Behrens, H. (2009). Usage-based and emergentist approaches to language acquisition. *Linguistics*, *47*(2), 383–411.

Bion, R. A., Borovsky, A., & Fernald, A. (2013). Fast mapping, slow learning: Disambiguation of novel word-object mappings in relation to vocabulary learning at 18, 24, and 30 months. *Cognition*, *126*(1), 39–53.

Bloom, P. (2000). *How Children Learn the Meanings of Words.* Cambridge, MA: MIT Press.

Bod, R. (2009). From exemplar to grammar: A probabilistic analogy-based model of language leaning. *Cognitive Science*, *33*(5), 752–793.

Börschinger, B., Jones, B. K., & Johnson, M. (2011). Reducing Grounded Learning Tasks to Grammatical Inference. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing* (pp. 1416–1425). Stroudsburg, PA, USA: Association for Computational Linguistics.

Branavan, S., Chen, H., Zettlemoyer, L. S., & Barzilay, R. (2009). Reinforcement Learning for Mapping Instructions to Actions. In *Proceedings of the Joint conference of the Annual Meeting of the Association for Computational Linguistics and the International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing.* Stroudsburg, PA, USA: Association for Computational Linguistics.

Branavan, S., Zettlemoyer, L. S., & Barzilay, R. (2010). Reading Between the Lines: Leaning to Map High-level Instructions to Commands. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics.* Stroudsburg, PA, USA: Association for Computational Linguistics.

Brandl, H. (2009). *A computational model for unsupervised childlike speech acquisition.* Unpublished doctoral dissertation.

Brent, M. R. (1999). An efficient, probabilistically sound algorithm for segmentation and word discovery. *Machine Learning*, *34*, 71–106.

Brown, R. (1973). *A first language: the early stages.* Harvard University Press, Cambridge MA.

Bybee, J. (1995). Regular morphology and the lexicon. *Language and Cognitive Processes*, *10*(5), 425–455.

Bybee, J., & Scheibman, J. (1999). The effect of usage on degrees of constituency: the reduction of don't in english. *Linguistics*, *37*(4), 575–596.

Carey, S., & Bartlett, E. (1978). Acquiring a single new word. *Papers and reports on Child Language Development*, *15*, 17–29.

Çöltekin, c., & Nerbonne, J. (2014). An explicit statistical model of learning lexical segmentation using multiple cues. In *Proceedings of the 5th Workshop on Cognitive Aspects of Computational Language Learning* (pp. 19–28). Association for Computational Linguistics.

Cerisara, C. (2009). Automatic discovery of topics and acoustic morphemes from speech. *Computer Speech and Language*, *23*(2), 220–239.

Chang, N. C., & Maia, T. V. (2001). Learning grammatical constructions. In *Proceedings of the Annual Conference of the Cognitive Science Society* (pp. 176–181). Boston, Massachusetts, USA: Cognitive Science Society.

Chen, D. L., Kim, J., & Mooney, R. J. (2010). Training a multilingual sportscaster: Using perceptual context to learn language. *Journal of Artificial Intelligence Research*, *37*(1), 397–435.

Chen, D. L., & Mooney, R. J. (2008). Learning to sportscast: A test of grounded language acquisition. In *Proceedings of the International Conference on Machine Learning.*

Chen, S. F., & Goodman, J. (1998). *An empirical study of smoothing techniques for language modeling* (Tech. Rep.). Computer Science Group, Harvard University.

Chomsky, N. (1965). *Aspects of the theory of syntax.* Cambridge, MA: MIT Press.

Clark, E. (1993). *The lexicon in acquisition.* Cambridge, UK: Cambridge University Press.

Cohen, P., & Adams, N. (2001). An algorithm for segmenting categorical time series into meaningful episodes. In *Proceedings of the Fourth Symposium on Intelligent Data Analysis.*

Cohen, P. R., Adams, N., & Heeringa, B. (2006). Voting Experts: An Unsupervised Algorithm for Segmenting Sequences. *Journal of Intelligent Data Analysis*, *11*(6), 607–625.

Cover, T. M., & Thomas, J. A. (1991). *Elements of information theory*. John Wiley & Sons.

De Mori, R. (2011). History of knowledge and processes for spoken language understanding. In G. Tur & R. D. Mori (Eds.), *Spoken language understanding: Systems for extracting semantic information from speech* (pp. 11–40). John Wiley & Sons.

Dempster, A. P., Laird, N. M., & Rubin, D. (1977). Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society: Series B*, *39*(1), 1–38.

Deoras, A., Tur, G., Sarikaya, R., & Hakkani-Tur, D. (2013). Joint Discriminative Decoding of Words and Semantic Tags for Spoken Language Understanding. *IEEE Transactions on Audio, Speech and Language Processing*, *21*(8), 1612-1621.

Dinarelli, M., Moschitti, A., & Riccardi, G. (2012). Discriminative reranking for spoken language understanding. *IEEE Transactions on Audio Speech and Language Processing*, *20*(2), 526–539.

Dollaghan, C. (1985). Child meets word: Fast mapping in preschool children. *Journal of Speech and Hearing Research*, *28*, 449–454.

Dominey, P. F., & Boucher, J.-D. (2005). Learning to talk about events from narrated video in a construction grammar framework. *Artificial Intelligence*, *167*(1-2), 31–61.

Eisenbeiß, S. (2009). Generative approaches to language learning. *Linguistics*, *42*(2), 273–310.

Elman, J. L. (1990). Finding structure in time. *Cognitive Science*, *14*(2), 179–211.

Elsner, M., Goldwater, S., Feldman, N., & Wood, F. (2013). A Joint Learning Model of Word Segmentation, Lexical Acquisition, and Phonetic Variability. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing* (pp. 42–54). Seattle, Washington, USA: Association for Computational Linguistics.

Fazly, A., Alishahi, A., & Stevenson, S. (2010). A probabilistic computational model of cross-situational word learning. *Cognitive Science*, *34*(6), 1017–1063.

Feldman, N., Griffiths, T., & Morgan, J. (2009). Learning phonetic categories by learning a lexicon. In *Proceedings of the Annual Conference of the Cognitive Science Society* (pp. 2208–2213). Boston, Massachusetts USA: Cognitive Science Society.

Fillmore, C. J. (1976). Frame semantics and the nature of language. *Annals of the New York Academy of Sciences: Conference on the Origin and Development of Language and Speech*, *280*(1), 20-32.

Fiscus, J., Garofolo, J., Przybocki, M., Fisher, W., & Pallett, D. (1998). *1997 English Broadcast News Speech (HUB4) LDC98S7.* Linguistic Data Consortium.

Frank, M. C., Goodman, N. D., & Tenenbaum, J. B. (2007). A bayesian framework for cross-situational word-learning. In *Advances in Neural Information Processing Systems* (pp. 1–8). Red Hook, NY, USA: Curran Associates, Inc.

Gaspers, J., & Cimiano, P. (2012). A usage-based model for the online induction of constructions from phoneme sequences. In *Proceedings of the joint IEEE International Conference on Development and Learning and on Epigenetic Robotics* (pp. 1–6). Washington, DC, USA: IEEE Computer Society.

Gaspers, J., & Cimiano, P. (2014a). A computational model for the item-based induction of construction networks. *Cognitive Science*, *38*(3), 439–488.

Gaspers, J., & Cimiano, P. (2014b). Learning a semantic parser from spoken utterances. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing.* Washington, DC, USA: IEEE Computer Society.

Gaspers, J., Cimiano, P., Griffiths, S., & Wrede, B. (2011). An unsupervised algorithm for the induction of constructions. In *Proceedings of the joint IEEE International Conference on Development and Learning and on Epigenetic Robotics* (pp. 1–6). Washington, DC, USA: IEEE Computer Society.

Gaspers, J., Foltz, A., & Cimiano, P. (2014). Towards the emergence of verb-general constructions and early representations for verb entries: Insights from a computational model. In *Proceedings of the Annual Conference of the Cognitive Science Society.* Boston, Massachusetts USA: Cognitive Science Society.

Gaspers, J., Panzner, M., Lemme, A., Cimiano, P., Rohlfing, K. J., & Wrede, S. (2014). A multimodal corpus for the evaluation of computational models for (grounded) language acquisition. In *Proceedings of the Workshop on Cognitive*

Aspects of Computational Language Learning (pp. 30–37). Association for Computational Linguistics.

Gladfelter, A., & Goffman, L. (2013). The influence of prosodic stress patterns and semantic depth on novel word learning in typically developing children. *Language Learning and Development*, *9*(2), 151-174.

Gleitman, L. (1990). The structural sources of verb meaning. *Language Acquisition*, *1*(1), 3–55.

Gleitman, L. R., & Fisher, C. (2005). The Cambridge companion to Chomsky. In J. A. McGilvray (Ed.), (chap. Universal aspects of word learning).

Goldberg, A. (2003). Constructions: a new theoretical approach to language. *Trends in Cognitive Science*, *7*(5), 219–224.

Goldberg, A., & Suttle, L. (2010). Construction grammar. *Wiley Interdisciplinary Reviews: Cognitive Science*, *1*(4), 468–477.

Goldwasser, D., Reichart, R., Clarke, J., & Roth, D. (2011). Confidence Driven Unsupervised Semantic Parsing. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics.* Stroudsburg, PA, USA: Association for Computational Linguistics.

Goldwasser, D., & Roth, D. (2011). Learning from natural instructions. In *Proceedings of the International Joint Conference on Artificial Intelligence.* San Francisco, CA, USA: Morgan Kaufmann Publishers Inc.

Goldwater, S., Griffiths, T. L., & Johnson, M. (2009). A bayesian framework for word segmentation: Exploring the effects of context. *Cognition*, *112*, 21–54.

Golinkoff, R. M., Hirsh-Pasek, K., Bailey, L. M., & Wegner, N. R. (1992). Young children and adults use lexical principles to learn new nouns. *Developmental Psychology*, *28*(2), 99–108.

Gorin, A. L., Petrovska-Delacrétaz, D., Riccardi, G., & Wright, J. (1999). Learning spoken language without transcriptions. In *Proceedings of the IEEE Workshop on Automatic Speech Recognition and Understanding.* Washington, DC, USA: IEEE Computer Society.

Gorniak, P., & Roy, D. (2005). Speaking with your sidekick: Understanding situated speech in computer role playing games. In *Proceedings of the Conference on Artificial Intelligence and Interactive Digital Entertainment.*

Harnad, S. (1990). The symbol grounding problem. *Physica D: Nonlinear Phenomena*, *42*(1-3), 335–346.

He, Y., & Young, S. (2005). Semantic processing using the hidden vector state model. *Computer Speech and Language*, *19*, 85–106.

Hebb, D. O. (1949). *The organization of behavior: A neuropsychological theory.* New York, USA: John Wiley and Sons, Inc.

Hewlett, D., & Cohen, P. (2009). Bootstrap Voting Experts. In *Proceedings of the International Joint Conference on Artificial Intelligence.* San Francisco, CA, USA: Morgan Kaufmann Publishers Inc.

Hewlett, D., & Cohen, P. (2011). Fully Unsupervised Word Segmentation with BVE and MDL. In *Proceedings of The Annual Meeting of the Association for Computational Linguistics: Human Language Technologies.* Stroudsburg, PA, USA: Association for Computational Linguistics.

Heymann, J., Walter, O., Haeb-Umbach, R., & Raj, B. (2014). Iterative Bayesian Word Segmentation for Unspuervised Vocabulary Discovery from Phoneme Lattices. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing.* Washington, DC, USA: IEEE Computer Society.

Hinaut, X., & Dominey, P. F. (2013). Real-time parallel processing of grammatical structure in the fronto-striatal system: a recurrent network simulation study using reservoir computing. *PLoS ONE*, *8*(2), 1–18.

Hinaut, X., Petit, M., Pointeau, G., & Dominey, P. F. (2014). Exploring the Acquisition and Production of Grammatical Constructions Through Human-Robot Interaction with Echo State Networks. *Frontiers in Neurorobotics*, *8*(16), 1–17.

Horst, J. S., McMurray, B., & Samuelson, L. K. (2006). Online processing is essential for leaning: Understanding fast mapping and word learning in a dynamic connectionist architecture. In *Proceedings of the Annual Conference of the Cognitive Science Society* (pp. 339–344). Boston, Massachusetts, USA: Cognitive Science Society.

Horst, J. S., & Samuelson, L. K. (2008). Fast mapping but poor retention by 24-month-old infants. *Infancy*, *13*(2), 128–157.

Hunt, A., & McGlashan, S. (2004). *Speech Recognition Grammar Specification (SRGS) Version 1.0* (Tech. Rep.). Available from `http://www.w3.org/TR/speech-grammar/`

Huttenlocher, J., Waterfall, H., Vasilyeva, M., Vevea, J., & Hedges, L. V. (2010). Sources of variability in children's language growth. *Cognitive Psychcology*, *61*(4), 343–365.

Jansen, A., Dupoux, E., Goldwater, S., Johnson, M., Khudanpur, S., Church, K., et al. (2013). A summary of the 2012 jhu clsp workshop on zero resource speech technologies and models of early language acquisition. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing.* Washington, DC, USA: IEEE Computer Society.

Jarosz, G., & Johnson, J. A. (2013). The richness of distributional cues to word boundaries in speech to young children. *Language Learning and Development*, *9*(2), 175-210.

Johnson, E. K., & Jusczy, P. W. (2001). Word segmentation by 8-month-olds: When speech cues count more than statistics. *Journal of Memory and Language*, *44*, 1–20.

Johnson, M., & Goldwater, S. (2009). Improving nonparameteric Bayesian inference: experiments on unsupervised word segmentation with adaptor grammars. In *Proceedings of the Annual Conference of the North American Chapter of the Association for Computational Linguistics* (pp. 317–325). Stroudsburg, PA, USA: Association for Computational Linguistics.

Johnson, M., Griffiths, T. L., & Goldwater, S. (2007). Adaptor grammars: A framework for specifying compositional nonparametric Bayesian models. In *Advances in Neural Information Processing Systems.*

Jones, B. K., Johnson, M., & Frank, M. C. (2010). Learning words and their meanings from unsegmented child-directed speech. In *Human Language Technologies: The Annual Conference of the North American Chapter of the Association for Computational Linguistics* (pp. 501–509). Stroudsburg, PA, USA: Association for Computational Linguistics.

Jung, S., Lee, C., Kim, K., Jeong, M., & Lee, G. G. (2009). Data-driven user simulation for automated evaluation of spoken dialog systems. *Computer Speech and Language*, *23*, 479–509.

Kachergis, G., Yu, C., & Shiffrin, R. (2012a). An associative model of adaptive inference for learning word-referent mappings. *Psychonomic Bulletin & Review*, *19*(2), 317–324.

Kachergis, G., Yu, C., & Shiffrin, R. M. (2012b). Cross-situational word learning is better modeled by associations than hypotheses. In *Proceedings of the joint IEEE International Conference on Development and Learning and on Epigenetic Robotics.* Washington, DC, USA: IEEE Computer Society.

Kate, R. J., & Mooney, R. J. (2006). Using string-kernels for learning semantic parsers. In *Proceedings of the International Conference on Computational Linguistics and Annual Meeting of the Association for Computational Linguistics* (pp. 913–920). Stroudsburg, PA, USA: Association for Computational Linguistics.

Kate, R. J., & Mooney, R. J. (2007). Learning language semantics from ambiguous supervision. In *Proceedings of the Twenty-Second Conference on Artificial Intelligence* (pp. 895–900).

Kidd, E. (2012). Implicit statistical learning is directly associated with the acquisition of syntax. *Developmental Psychology*, *48*(1), 171-184.

Klasinas, I., Potamianos, A., Iosif, E., Georgiladakis, S., & Mameli, G. (2013). Web Data Harvesting for Speech Understanding Grammar Induction. In *Proceedings of Interspeech.* International Speech Communication Association.

Kriz, S., Anderson, G., & Trafton, J. G. (2010). Robot-directed Speech: Using Language to Assess First-time Users' Conceptualizations of a Robot. In *Proceedings of the ACM/IEEE International Conference on Human-robot Interaction* (pp. 267–274). Piscataway, NJ, USA: IEEE Press.

Kruskal, J. B. (1999). An overview of sequence comparison. In D. S. . J. Kruskal (Ed.), *Time warps, string edits and macromolecules: The theory and practice of sequence comparison* (pp. 1–44). Addison-Wesley, Boston.

Kullback, S. (1959). *Information theory and statistics.* John Wiley & Sons.

Kwiatkowski, T., Goldwater, S., Zettlemoyer, L., & Steedman, M. (2012). A probabilistic model of syntactic and semantic acquisition from child-directed utterances and their meanings. In *Proceedings of the Conference of the European Chapter of the Association for Computational Linguistics* (pp. 234–244). Stroudsburg, PA, USA: Association for Computational Linguistics.

Ladefoged, P. (1993). *A course in phonetics.* Orlando, FL: Harcourt Brac.

Lafferty, J., McCallum, A., & Pereira, F. (2001). Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the International Conference on Machine Learning.*

Lamel, L., & Gauvain, J.-L. (2003). Speech recognition. In R. Mitkov (Ed.), *The oxford handbook of computational linguistics* (pp. 305–322). Oxford University Press.

Levit, M., Nöth, E., & Gorin, A. (2002). Using em-trained string-edit distances for approximate matching of acoustic morphemes. In *Proceedings of Interspeech.* International Speech Communication Association.

Liang, P., Jordan, M. I., & Klein, D. (2009). Learning semantic correspondences with less supervision. In *Proceedings of the Joint conference of the Annual Meeting of the Association for Computational Linguistics and the International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing* (pp. 91–99). Stroudsburg, PA, USA: Association for Computational Linguistics.

Lieven, E., Pine, J. M., & Baldwin, G. (1997). Lexically-based learning and early grammatical development. *Journal of Child Language*, *24*(1), 187–219.

Liu, Y., & Hakkani-Tür, D. (2011). Speech summmarization. In G. Tur & R. D. Mori (Eds.), *Spoken language understanding: Systems for extracting semantic information from speech* (pp. 357–396). John Wiley & Sons.

MacWhinney, B. (2000). *The CHILDES Project: Tools for Analyzing Talk.* Lawrence Erlbaum.

Markman, E. M., & Wachtel, G. F. (1988). Children's use of mutual exclusivity to constrain the meanings of words. *Cognitive Psychology*, *20*(2), 121–157.

Martin, A., Peperkamp, B. S., & A, A. E. D. (2012). Learning phonemes with a proto-lexicon. *Cognitive Science*, *37*(1), 103–124.

Mattys, S. L., Jusczyk, P. W., Luce, P. A., & Morgan, J. L. (1999). Phonotactic and prosodic effects on word segmentation in infants. *Cognitive Psychology*, *38*, 465–494.

Maurits, L., Perfors, A. F., & Navarro, D. J. (2009). Joint acquisition of word order and word reference. In *Proceedings of the Annual Conference of the Cognitive Science Society.* Boston, Massachusetts USA: Cognitive Science Society.

McInnes, F. R., & Goldwater, S. J. (2011). Unsupervised extraction of recurring words from infant-directed speech. In *Proceedings of the Annual Conference of the Cognitive Science Society.* Boston, Massachusetts USA: Cognitive Science Society.

McMurray, B., Samuelson, L. K., & Horst, J. S. (2012). Word Learning Emerges From the Interaction of Online Referent Selection and Slow Associative Learning. *Psychological Review*, *119*(4), 831–877.

Medina, T. N., Snedeker, J., Trueswell, J. C., & Gleitman, L. R. (2011). How words can and cannot be learned by observation. *Proceedings of the National Academy of Sciences*, *108*, 9014–9019.

Meng, H. M., & Siu, K.-C. (2002). Semiautomatic Acquisition of Semantic Structures for Understanding Domain-Specific Natural Language Queries. *IEEE Transactions on Knowledge and Data Engineering*, *14*(1), 172-181.

Merriman, W., & Bowman, L. (1989). The mutual exclusivity bias in children's word learning. *Monographs of the Society for Research in Child Development*, *54*(3–4), 1–129.

Muscariello, A., Gravier, G., & Bimbot, F. (2012). Unsupervised Motif Acquisition in Speech via Seeded Discovery and Template Matching Combination. *IEEE Transactions on Audio, Speech & Language Processing*, *20*(7), 2031-2044.

Neubig, G., Mimupa, M., Mori, S., & Kawahara, T. (2012). Bayesian learning of a language model from continuous speech. *IEICE Transactions on Information and Systems*, *59*(2), 614–625.

Neubig, G., Mimura, M., Mori, S., & Kawahara, T. (2010). Learning a language model from continuous speech. In *Proceedings of Interspeech.* International Speech Communication Association.

Nobel, C. H., Rowland, C. F., & Pine, J. M. (2011). Comprehension of argument structure and semantic roles: Evidence from english-learning children and the forced-choice pointing paradigm. *Cognitive Science*, *35*, 963–982.

Och, F. J., & Ney, H. (2003). A systematic comparison of various statistical alignment models. *Computational Linguistics*, *29*(1), 19-â51.

Olguin, R., & Tomasello, M. (1993). Twenty-five-month-old children do not have a grammatical category of verb. *Cognitive Development*, *8*(3), 245–272.

Orkin, J., & Roy, D. (2007). The restaurant game: Learning social behavior and language from thousands of players online. *Journal of Game Development*, *3*(1), 39–60.

Parisien, C., & Stevenson, S. (2010). Learning verb alternations in a usage-based bayesian model. In *Proceedings of the Annual Conference of the Cognitive Science Society.* Boston, Massachusetts USA: Cognitive Science Society.

Pearl, L., Goldwater, S., & Steyvers, M. (2011). Online learning mechanisms for bayesian models of word segmentation. *Research on Language and Computation*, *8*(2–3), 107–132.

Perfors, A., Tenenbaum, J. B., & Wonnacott, E. (2010). Variability, negative evidence, and the acquisition of verb argument constructions. *Journal of Child Language*, *37*(3), 607–642.

Pickering, M. J., & Garrod, S. (2004). Towards a mechanistic psychology of dialogue. *Behavioral and Brain Science*, *27*(2), 169–225.

Pickering, M. J., & Garrod, S. (2006). Alignment as the basis for successful communication. *Research on Language and Computation*, *4*, 203-228.

Pine, J. M., & Lieven, E. V. M. (1997). Slot and frame patterns and the development of the determiner category. *Applied Psycholinguistics*, *18*(2), 123–138.

Pinker, S. (1989). *Learnability and cognition.* The MIT Press, Cambridge M.A.

Placeway, P., Chen, S., Eskenazi, M., Jain, U., Parikh, V., Raj, B., et al. (1997). The 1996 Hub-4 sphinx-3 system. In *Proceedings of the DARPA Speech recognition workshop.*

Poon, H., & Domingos, P. (2009). Unsupervised semantic parsing. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing.* Stroudsburg, PA, USA: Association for Computational Linguistics.

Qu, S., & Chai, J. Y. (2010). Context-based word acquisition for situated dialogue in a virtual world's. *Journal of Artificial Intelligence Research*, *37*, 247-277.

Quine, W. V. O. (1960). *Word and object.* Cambridge, MA, USA: MIT Press.

Rabiner, L. R. (1989). A tutorial on hidden markov models and selected applications in speech recognition. In *Proceedings of the IEEE* (pp. 257–286). Washington, DC, USA: IEEE Computer Society.

Räsänen, O. (2011). A computational model of word segmentation from continuous speech using transitional probabilities of atomic acoustic events. *Cognition*, *120*, 149–176.

Räsänen, O. (2012). Computational modeling of phonetic and lexical learning in early language acquisition: existing models and future directions. *Speech Communication*, *54*, 975–997.

Räsänen, O., Laine, U. K., & Altosaar, T. (2009). Computational language acquisition by statistical bottom-up processing. In *Proceedings of Interspeech.* International Speech Communication Association.

Reckman, H., Orkin, J., & Roy, D. (2010). Learning meanings of words and constructions, grounded in a virtual game. In *Proceedings of the Conference on Natural Language Processing.*

Rojas, R. (1993). *Theorie der neuronalen Netze.* Berlin, Germany: Springer-Verlag.

Romberg, A. R., & Saffran, J. R. (2010). Statistical learning and language acquisition. *Wiley Interdisciplinary Reviews: Cognitive Science*, *1*(6), 906–914.

Rosenfeld, R. (2000). Two decades of statistical language modeling: Where do we go from here? In *Proceedings of the IEEE* (Vol. 88, p. 1270-1278). Washington, DC, USA: IEEE Computer Society.

Rosset, S., Galibert, O., & Lamel, L. (2011). Spoken question answering. In G. Tur & R. D. Mori (Eds.), *Spoken language understanding: Systems for extracting semantic information from speech* (pp. 147–170). John Wiley & Sons.

Rowland, C. F. (2007). Explaining errors in children's questions. *Cognition*, *104*(1), 106–134.

Roy, D., & Pentland, A. (2002). Learning words from sights and sounds: A computational model. *Cognitive Science*, *26*(1), 113-146.

Saffran, J. R. (2003). Statistical language learning; mechanisms and constraints. *Current Directions in Psychological Science*, *12*, 110–114.

Saffran, J. R., Aslin, R. N., & Newport, E. L. (1996). Statistical learning by 8-month-old infants. *Science*, *274*(5294), 1926–1928.

Saffran, J. R., & Wilson, D. P. (2003). From syllable to syntax: Multilevel statistical learning by 12-month-old infants. *Infancy*, *4*(2), 273–284.

Schatten, R. (2003). Systemic architecture for audio signal processing. In *Proceedings of the European Conference on Artificial Life* (pp. 491–498). Berlin, Germany: Springer-Verlag.

Schröder, M., & Trouvain, J. (2003). The German Text-to-Speech Synthesis System MARY: A Tool for Research, Development and Teaching. *International Journal of Speech Technology*, *6*, 365–377.

Schukat-Talamazzini, E. G. (1995). *Automatische Spracherkennung. Statistische Verfahren der Musteranalyse.* Braunschweig: Vieweg.

Scott, R. M., & Fisher, C. (2012). 2.5-year-olds use cross-situational consistency to learn verbs under referential uncertainty. *Cognition*, *122*(2), 163–180.

Sheline, L., Waxman, S. R., & Arunachalam, S. (2013). Understanding conjoined-subject intransitives: Two-year-olds perform like adults. *Poster presented at the 2013 Biennial Meeting of the Society for Research in Child Development,* Seattle, WA, USA.

Siskind, J. M. (1996). A computational study of cross-situational techniques for learning word-to-meaning mappings. *Cognition*, *61*(1–2), 39-91.

Siu, K.-c., & Meng, H. M. (1999). Semi-Automatic Acquisition of Domain-Specific Semantic Structures. In *Proceedings of Eurospeech.* International Speech Communication Association.

Sloetjes, H., & Wittenburg, P. (2008). Annotation by category: ELAN and ISO DCR. In *Proceedings of the International Conference on Language Resources and Evaluation.* Paris: European Language Resources Association.

Smith, K., Smith, A., & Blythec, R. A. (2011). Cross-situational learning: An experimental study of word-learning mechanisms. *Cognitive Science*, *35*(3), 480–498.

Smith, L., & Yu, C. (2008). Infants rapidly learn word-referent mappings via cross-situational statistics. *Cognition*, *106*(3), 1558–1568.

Solan, Z., Horn, D., Ruppin, E., & Edelman, S. (2005). Unsupervised learning of natural languages. *Proceedings of the National Academy of Sciences*, *102*(33), 11629-11634.

Steedman, M. (2000). *The syntactic process.* Cambridge, MA: MIT Press.

Stolcke, A. (2002). SRILM - An Extensible Language Modeling Toolkit. In *Proceedings of Interspeech* (pp. 901–904). International Speech Communication Association.

Svec, J., Smidle, L., & Ircing, P. (2013). Hierarchical discriminative model for spoken language understanding. In *IEEE International Conference on Acoustics, Speech, and Signal Processing.* Washington, DC, USA: IEEE Computer Society.

Taguchi, R., Iwahashi, N., Nose, T., Funakoshi, K., & Nakano, M. (2009). Learning lexicons from spoken utterances based on statistical model selection. In *Proceedings of Interspeech.* International Speech Communication Association.

Tomasello, M. (1992). *First verbs: A case study of early grammatical development.* Cambridge: Cambridge University Press.

Tomasello, M. (2000a). First steps toward a usage-based theory of language acquisition. *Cognitive Linguistics*, *11*(1–2), 61–82.

Tomasello, M. (2000b). The item-based nature of children's early syntatic development. *Trends in Cognitive Science*, *4*(4), 156–163.

Tomasello, M. (2003). *Constructing a language: A usage-based theory of language acquisition.* Cambridge, MA: Harvard University Press.

Tomasello, M., Akhtar, N., Dodson, K., & Rekau, L. (1997). Differential productivity in young children's use of nouns and verbs. *Journal of Child Language*, *24*(2), 373–387.

Trueswell, J. C., Medina, T. N., Hafri, A., & Gleitman, L. R. (2013). Propose but verify: Fast mapping meets cross-situational word learning. *Cognitive Psychology*, *66*(1), 126–156.

Tur, G., & Mori, R. D. (Eds.). (2011). *Spoken language understanding: Systems for extracting semantic information from speech.* John Wiley & Sons.

Vale, D. C., & Mast, V. (2012). Using foot-syllable grammars to customize speech recognizers for dialogue systems. In P. Sojka, A. Horak, I. Kopecek, & K. Pala (Eds.), *Tsd* (Vol. 7499, p. 591-598). Springer.

Van Tichelen, L., & Burke, D. (2007). *Semantic Interpretation for Speech Recognition (SISR) Version 1.0* (Tech. Rep.). Available from `http://www.w3.org/TR/semantic-interpretation/`

Vlach, H. A., & Sandhofer, C. M. (2012). Fast Mapping Across Time: Memory Processes Support Children's Retention of Learned Words. *Frontiers in Developmental Psychology*, *46*(3), 1–8.

Vogel, A., & Jurafsky, D. (2010). Learning to follow navigational directions. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics.* Stroudsburg, PA, USA: Association for Computational Linguistics.

Vogt, P., & Divina, F. (2007). Social symbol grounding and language evolution. *Interaction Studies*, *8*(1), 31-52.

Walker, W., Lamere, P., Kwok, P., Raj, B., Singh, R., Gouvea, E., et al. (2004). *Sphinx-4: A flexible open source framework for speech recognition* (Tech. Rep.). Sun Microsystems.

Wang, Y.-Y., & Acero, A. (2003). Combination of CFG and N-gram Modeling in Semantic Grammar Learning. In *Proceedings of Eurospeech.* International Speech Communication Association.

Wang, Y.-Y., & Acero, A. (2005). SGStudio: Rapid Semantic Grammar Development for Spoken Language Understanding. In *Proceedings of the European Conference on Speech Communication and Technology.*

Wang, Y.-Y., & Acero, A. (2006a). Discriminative models for spoken language understanding. In *Proceedings of Interspeech.* International Speech Communication Association.

Wang, Y.-Y., & Acero, A. (2006b). Rapid Development of Spoken Language Understanding Grammars. *Speech Communication*, *48*(3–4), 390–416.

Wang, Y.-Y., Acero, A., & Chelba, C. (2003). Is word error rate a good indicator for spoken language understanding accuracy. In *IEEE Workshop on Automatic Speech Recognition and Understanding.* Washington, DC, USA: IEEE Computer Society.

Wang, Y.-Y., Deng, L., & Acero, A. (2011). Semantic Frame-based Spoken Language Understanding. In G. Tur & R. D. Mori (Eds.), *Spoken language understanding: Systems for extracting semantic information from speech* (pp. 41–92). John Wiley & Sons.

Waterfall, H. R., Sandbank, B., Onnis, L., & Edelman, S. (2010). An empirical generative framework for computational modeling of language acquisition. *Journal of Child Language*, *37*(3), 671–703.

Wilkinson, K. M., & Mazzitelli, K. (2003). The effect of 'missing' information on children's retention of fast-mapped labels. *Journal of Child Language*, *30*, 47–73.

Wong, C.-C., & Meng, H. (2001). Improvements on a semi-automatic grammar induction framework. In *Proceedings fo the IEEE Automatic Speech Recognition and Understanding Workshop.* Washington, DC, USA: IEEE Computer Society.

Wong, Y. W., & Mooney, R. J. (2006). Learning for Semantic Parsing with Statistical Machine Translation. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics* (pp. 439–446). Stroudsburg, PA, USA: Association for Computational Linguistics.

Wu, W.-L., Lu, R.-Z., Duan, J.-Y., Liu, H., Gao, F., & Chen, Y.-Q. (2010). Spoken language understanding using weakly supervised learning. *Computer*, *24*, 358–382.

Yu, C. (2005). The emergence of links between lexical acquisition and object categorization: A computational study. *Connection Science*, *17*(3–4), 381–397.

Yu, C. (2006). Learning Syntax-Semantics Mappings to Bootstrap Word Learning. In *Proceedings of the Annual Conference of the Cognitive Science Society.* Boston, Massachusetts USA: Cognitive Science Society.

Yu, C., & Ballard, D. H. (2002). *A computational model of embodied language learning* (Tech. Rep.). Department of Computer Science, University of Rochester.

Yu, C., & Ballard, D. H. (2007, August). A unified model of early word learning: Integrating statistical and social cues. *Neurocomputing*, *70*(13-15), 2149–2165.

Yu, C., Ballard, D. H., & Aslin, R. N. (2005). The role of embodied intention in early lexical acquisition. *Cognitive Science*, *29*, 961–1005.

Yu, C., & Smith, L. B. (2007). Rapid word learning under uncertainty via cross-situational statistics. *Psychological Science*, *18*, 414–420.

Yu, D., Ju, Y. C., Wang, Y.-Y., & Acero, A. (2006). N-gram based filler model for robust grammar authoring. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing.* Washington, DC, USA: IEEE Computer Society.

Zaanen, M. van, & Adriaans, P. (2001a). Alignment-Based Learning versus EMILE: A Comparison. In *Proceedings of the Belgian-Dutch Conference on Artificial Intelligence* (pp. 315–322). Brussels, Belgium: Royal Flemish Academy for Science and Art.

Zaanen, M. van, & Adriaans, P. (2001b). *Comparing Two Unsupervised Grammar Induction Systems: Alignment-Based Learning vs. EMILE* (Tech. Rep.).

Zettlemoyer, L. S., & Collins, M. (2007). Online Learning of Relaxed CCG Grammars for Parsing to Logical Form. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning* (pp. 678–687). Stroudsburg, PA, USA: Association for Computational Linguistics.

Appendix A

Similarity computation between phoneme strings

We compute the phonetic similarity between two phoneme string roughly following Yu et al. (Yu & Ballard, 2002; Yu et al., 2005). In order to compare two phoneme strings, the authors first convert phonemes into vectors of articulatory features, such as e.g. "voicing", called distinctive features (Ladefoged, 1993). The vectors are binary where 0 and 1 denote the absence and presence of a feature, respectively. We utilize the same 12 distinctive features as Yu et al. (2005). They are presented in Table A.1; they are taken from Yu & Ballard (2002) and converted from ARPABET into X-SAMPA.

Yu et al. (Yu & Ballard, 2002; Yu et al., 2005) compute the similarity between two such vectors as the Hamming distance between the two of them. The intuition is that sounds which differ in fewer features are more likely to be similar/related. Subsequently, a similarity matrix is computed which contains scores for the similarity for each pair of phonemes. This score is set to the negative hamming distance in case of comparing different phonemes. Additionally, a positive reward is set for two matching phonemes in two strings. Following Yu & Ballard (2002) we set the reward to be half of the dimension of distinctive features, i.e. 6 in our case.

Yu et al. (Yu & Ballard, 2002; Yu et al., 2005) compute the similarity between two phoneme strings using the similarity scores based on the dynamic programming principle (Kruskal, 1999). In our case we applied the procedure presented in Algorithm 4 which is similar to the one applied by Yu et al. (2005).

The phonetic similarity $sim(s_1, s_2)$ between two phonetic strings s_1 and s_2 is computed as described previously. The algorithm returns a value for each pair of sequences, even if no phoneme appears in both of them. However, we only consider two sequences as potentially phonetically similar if at least some phoneme(s) appear in both sequences. Thus, we set a threshold by multiplying the maximal sequence length $max(|s_1|, |s_2|)$ with a fraction, i.e. $\frac{1}{3}$, of the reward set for matching phonemes, i.e. $max(|s_1|, |s_2|) * 2$. We consider only sequences with a value above this threshold as *similar*.

Algorithm 4 Comparing phoneme strings

Input: Two phoneme strings s_1 and s_2
Output: The similarity score

```
$l_1 = length(s_1)$
$l_2 = length(s_2)$
$M = array[0..l_1, 0..l_2]$

for $i = 1$ to $l_1$ do
    $M[i, 0] = 0$
end for

for $j = 1$ to $l_2$ do
    $M[0, j] = 0$
end for

for $i = 1$ to $l_1$ do
    for $j = 1$ to $l_2$ do
        $M[i, j] = max(M[i-1, j-1] + hamDist(s_{1_i}, s_{2_j}),$
                       $M[i, j-1] + hamDist(s_{1_i}, s_{2_j}),$
                       $M[i-1, j] + hamDist(s_{1_i}, s_{2_j}),$
                       $M[i-1, j] + min(hamDist(s_{1_i}, s_{1_{i-1}}), hamDist(s_{1_i}, s_{1_{i+1}}))$
                       $M[i, j-1] + min(hamDist(s_{1_j}, s_{1_{j-1}}), hamDist(s_{1_j}, s_{1_{j+1}})))$
    end for
end for

return $M[l_1, l_2]$
```

Table A.1.: Overview of distinctive features utilized in this thesis based on Yu & Ballard (2002).

	consonantal	**vocalic**	**continuant**	**nasal**	**anterior**	**coronal**	**high**	**low**	**back**	**voicing**	**strident**	**sonorant**
p	1	0	0	0	1	0	0	0	0	0	0	0
b	1	0	0	0	1	0	0	0	0	1	0	0
t	1	0	0	0	1	1	0	0	0	0	0	0
d	1	0	0	0	1	1	0	0	0	1	0	0
tS	1	0	0	0	0	1	1	0	0	0	1	0
dZ	1	0	0	0	0	1	1	0	0	1	1	0
k	1	0	0	0	0	0	1	0	1	0	0	0
g	1	0	0	0	0	0	1	0	1	1	0	0
f	1	0	1	0	0	0	0	0	0	0	1	0
v	1	0	1	0	1	0	0	0	0	1	1	0
T	1	0	1	0	1	1	0	0	0	0	0	0
D	1	0	1	0	1	1	0	0	0	1	0	0
s	1	0	1	0	1	1	0	0	0	0	1	0
z	1	0	1	0	1	1	0	0	0	1	1	0
S	1	0	1	0	0	1	1	0	0	0	1	0
Z	1	0	1	0	0	1	1	0	0	1	1	0
m	1	0	0	1	1	0	0	0	0	1	0	1
n	1	0	0	1	1	1	0	0	0	1	0	1
N	1	0	0	1	0	0	1	0	1	1	0	1
l	1	1	1	0	1	1	0	0	0	1	0	1
w	0	0	1	0	0	0	1	0	1	1	0	1
j	0	0	1	0	0	0	1	0	0	1	0	1
r	1	1	1	0	0	1	0	0	0	1	0	1
h	0	0	1	0	0	0	0	1	0	0	0	0
E	0	1	1	0	0	0	0	0	0	1	0	1
u	0	1	1	0	0	0	1	0	1	1	0	1
i	0	1	1	0	0	0	1	0	0	1	0	1
I	0	1	1	0	0	0	1	0	0	1	0	1
A	0	1	1	0	0	0	0	1	1	1	0	1
@	0	1	1	0	0	0	0	0	1	1	0	1
V	0	1	1	0	0	0	0	0	1	1	0	1
U	0	1	1	0	0	0	1	0	1	1	0	1
O	0	1	1	0	0	0	0	1	1	1	0	1
{	0	1	1	0	0	0	0	1	0	1	0	1
@U	0	1	1	0	0	0	0	0	1	1	0	1
EI	0	1	1	0	0	0	0	0	0	1	0	1
aU	0	1	1	0	0	0	0	0	0	0	0	0
AI	0	1	1	0	0	0	0	0	0	0	0	0
OI	0	1	1	0	0	0	0	0	0	0	0	0
r=	0	1	1	0	0	0	0	0	0	0	0	0