MINIMAL SUPERVISION FOR LANGUAGE LEARNING: BOOTSTRAPPING GLOBAL PATTERNS FROM LOCAL KNOWLEDGE

BY

MICHAEL JAMES CONNOR

DISSERTATION

Submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy in Computer Science
in the Graduate College of the
University of Illinois at Urbana-Champaign, 2011

Urbana, Illinois

Doctoral Committee:

Professor Dan Roth, Chair & Director of Research
Professor Cynthia Fisher
Professor Suzanne Stevenson, University of Toronto
Professor Gerald DeJong
Assistant Professor Julia Hockenmaier

Abstract

A fundamental step in sentence comprehension involves assigning semantic roles to sentence constituents. To accomplish this, the listener must parse the sentence, find constituents that are candidate arguments, and assign semantic roles to those constituents. Each step depends on prior lexical and syntactic knowledge. Where do children begin in solving this problem when learning their first languages? To experiment with different representations that children may use to begin understanding language, we have built a computational model for this early point in language acquisition. This system, BabySRL, learns from transcriptions of natural child-directed speech and makes use of psycholinguistically plausible background knowledge and realistically noisy semantic feedback to begin to classify sentences at the level of "who does what to whom."

Starting with simple, psycholinguistically-motivated representations of sentence structure, the BabySRL is able to learn from full semantic feedback, as well as a supervision signal derived from partial semantic background knowledge. In addition we combine the BabySRL with an unsupervised Hidden Markov Model part-of-speech tagger, linking clusters with syntactic categories using background noun knowledge so that they can be used to parse input for the SRL system. The results show that proposed shallow representations of sentence structure are robust to reductions in parsing accuracy, and that the contribution of alternative representations of sentence structure to successful semantic role labeling varies with the integrity of the parsing and argument-identification stages. Finally, we enable the BabySRL to improve both an intermediate syntactic representation and its final semantic role classification. Using this system we show that it is possible for a simple learner in a plausible (noisy) setup to begin comprehending simple semantics when initialized with a small amount of concrete noun knowledge and some simple syntax-semantics mapping biases, before acquiring any specific verb knowledge.

To My Parents.

Acknowledgments

The thesis is an odd document to write because in many ways it serves to mark an accomplishment for myself, yet throughout I refer to the ‘we’ that actually deserves the credit. Hopefully I can shed at least a little light onto who is behind this ‘we.’

First and foremost I want to thank my adviser Dan Roth. From the very beginning of graduate school I worked with Dan, and I owe probably my entire understanding of academic research to him (and I am sure I missed a good deal). He sets what I consider to be a near ideal as research director, even as our group swelled in membership, balancing the high level goals, mentoring and salesmanship necessary for that position.

I would also like to thank Cynthia Fisher who I was privileged to collaborate with throughout this project. The frequent discussions between Cindy, Dan, Yael Gertner and myself were always interesting, if often filled with digressions to various related aspects of psycholinguistics that caused at least Dan and myself to view computational linguistics in new ways.

For these many years I have been part of a great and diverse group of researches, the cognitive computation group. For fear of forgetting a name, I won’t list them all here, but this group certainly helped make my time in graduate school possibly too enjoyable, and for the most part did not seem to mind my office juggling and yo-yoing.

Also of course I must thank my family for continued, if sometimes baffled, support. I probably didn’t really understand what I was getting into with graduate school, and my parents certainly did not, but they followed right along. On the other hand my sister was able to start on and thrive in her profession as a lawyer during my time in graduate school, so at least our parents had that.

And to my wife Georgina, who had her own ordeals of thesis followed by medical school to contend with, I can’t really comprehend where I would be without her. As she said in her own acknowledgments, one day we (together with Mxy and Rocky) will rule the world.

Table of Contents

List of Abbreviations . . . . **vi**

Chapter 1 Introduction **1**
1.1 Language Acquisition: Syntactic Bootstrapping 4
1.2 Thesis Statement 5
1.3 Thesis Organization 6

Chapter 2 Background **9**
2.1 Supervised Learning 9
2.2 Semantic Role Labeling 18
2.3 Language Acquisition 22

Chapter 3 Baby SRL Model **28**
3.1 CHILDES Training Data 28
3.2 SRL Learning Model 30
3.3 Experimental Setup 33
3.4 Experimental Results 36
3.5 Conclusion 44

Chapter 4 Baby SRL Noisy Training **45**
4.1 Minimally Supervised Argument Identification 46
4.2 Animacy Feedback 62
4.3 Minimally Supervised BabySRL 73
4.4 Conclusion 76

Chapter 5 Latent Baby SRL **77**
5.1 Related Work 78
5.2 Model 79
5.3 Latent Training 80
5.4 Experimental Evaluation 87
5.5 Conclusion 93

Chapter 6 Conclusions **95**
6.1 Future Work 96

Appendix A Word Lists **99**

References **105**

List of Abbreviations

CDS	Child Directed Speech.
EM	Expectation Maximization.
HMM	Hidden Markov Model.
NLP	Natural Language Processing.
POS	Part of Speech.
SRL	Semantic Role Labeling.
VB	Variational Bayes.
WSD	Word Sense Disambiguation.

Chapter 1

Introduction

Children's ability to rapidly learn language continues to inspire and confound. Faced with noisy, complex speech in complex, ambiguous environments, children are able to acquire language with little explicit instruction. No one provides corrections for every erroneous utterance, or valid interpretations and alternative phrasing for all misunderstood speech. Even with this lack of supervision children seem to follow a regular schedule, starting with knowledge of a few words or structures and quickly generalizing to global language patterns.

Previous computational models of this stage of early language acquisition have demonstrated the abilities of various knowledge and representation schemes using existing learning protocols and potentially unrealistic levels of supervision to acquire the link between syntax and semantics. In this thesis we develop a machine learning model that supports psycholinguistic theories of acquisition, including the ability to deal with noisy input and ambiguous feedback. To date, machine learning algorithms have been successful for a variety of natural language tasks, largely by relying on fine grained supervision over large training sets. We demonstrate a powerful alternative means of supervision motivated by psycholinguistic accounts of child language acquisition.

Statistical machine learning approaches have provided many advances for the Natural Language Processing (NLP) community, allowing some global task representation to be inferred from copious amounts of data. The dominant paradigm in this field is that of supervised learning that relies on the existence of a correct "answer" or label for not only every example, but every decision faced in the data. This labeling defines what the machine will learn; instead of relying on a hand coded model, NLP engineers and experts just need to define what they want the answer to be and create a model to learn from this data.

But what happens when the full "correct answer" is not or cannot be known? When learning language, a child is not provided sentence/interpretation pairs from which to learn. Instead the child must rely on innate assumptions, background knowledge, and feedback from a complex and ambiguous world to drive its learning. In NLP this case abounds, either where a task is too ill defined to provide consistent labeling or else input is available without human labels (e.g. new domains or languages). Most supervised machine learning algorithms expect supervision at the level of individual decisions or structures in the language of the machine representation. In this thesis we develop an architecture that instead combines background knowledge and ambiguous feedback to replace this supervised data.

To train a word sense classifier, a human expert must first partition a word's meaning into a set of senses, label individual occurrences of that word with its sense, define a relevant representation to capture necessary semantics of a context (which may depend on external parsers, dictionaries or other further information investment), and feed the labeled data into a supervised machine learning algorithm. While a machine learning algorithm may be able to learn from this labeling, its usefulness is entirely dependent on how well the human defined the sense repository, or how much vocabulary was covered in the training data. To train a child to distinguish word senses, it is only necessary for the child to understand that words can have different meanings in different contexts. When a word comes up, if the specific meaning is not clear, the child is able to accomodate, using other possible sources of information such as a guess at the possible intended meaning of the entirety of the sentence/dialogue/scene. This feedback is ambiguous and noisy, but task oriented; the child learner understands a high level goal while learning and fits both internal representation and feedback to this goal.

Child language acquisition presents a case where a learner is able to learn a full model of language without being constantly given complete supervision. By assuming that language carries some meaning, the child is able to go a long way towards learning how to extract that meaning. If one's goal is to have computers learn human language, then it seems natural to explore how humans learn language, with a focus on both what knowledge and representations they may use to offset the lack of supervision, and how language allows itself to be learned with this information.

Human language is an unquestionably human phenomena, so far humans are the only known instance of a successful learner of it. Whether one believes this facility is due to an innate Universal Grammar or that language adapts itself to human's general brain capabilities and needs, child language acquisition provides not only an interesting case study, but provides definite evidence that language is learnable. Despite its seeming complexity, elements of language itself allows early generalization and learning from entities with not necessarily complete knowledge and processing capabilities. Here we show that paired with general assumptions about abstract communicative meaning (sentences mean something, and this meaning has structure), specific early knowledge can be used to begin understanding and potentially bootstrap full learning.

The purpose of language is to communicate some meaning, which we can model as a logical predicate with some set of arguments, where each argument serves a role in that predicate. We thus model language acquisition as a learner attempting to learn to predict such a predicate-argument semantic structure given a sentence. This forms a semantic role labeling (SRL) task [Màrquez et al., 2008] that attempts to assign abstract semantic roles to noun arguments of verb predicates (on the level of "who did what to whom"). This represents a useful high level semantic task (used for information extraction [Surdeanu et al., 2003], question answering [Shen and Lapata, 2007], machine translation [Wu and Fung, 2009], etc.) that needs to abstract over various sentence structures and lexical objects to be able to predict semantics for novel sentences with novel verbs.

To complete the model of language acquisition, we build a "BabySRL" system that is trained on semantically tagged transcripts of child directed speech (CDS), and tested using both held out CDS and constructed sentences of fixed structure with known nouns and novel verbs. The BabySRL allows us to experiment with different simple representations of sentences based on knowledge available to young children and varying levels of both input information and semantic feedback. Children are certainly not given semantic feedback along the lines of correct interpretation of the role of every argument in a sentence, nor are they expected to form perfect syntactic parses for every sentence, so we incorporate these restrictions into our training.

We are motivated by the "structure-mapping" account of syntactic bootstrapping. This model proposes that children are able to begin processing sentences by assuming innate abstract semantic roles, early noun knowledge, and a mapping from nouns to semantic arguments, all assumptions that can naturally be encoded in our BabySRL system. The success of the system developed in this thesis when faced with real data provides evidence not only for the effectiveness of this theory for real children, but points the way for more psycholinguistically motivated computer learners to prime the pump when learning language.

1.1 Language Acquisition: Syntactic Bootstrapping

An important stage in early language acquisition is at the beginning of sentence understanding, when semantics are first being applied and generalized to novel multiword units. For example a child encounters the sentence “The girl tickles the boy” with the relevant scene of a girl and a boy playing together. Semantic Bootstrapping [Pinker, 1984] claims that children are able to use their understanding of the meaning of the sentence to constrain the structure or syntax of the sentence, and thus begin learning. In this case the event provides interpretations such as the correct girl tickling the boy, as well as girl and the boy playing, the boy squirming and giggling, the girl laughing, etc. To learn from this sentence the child apparently has to already know the meaning of “tickle”; semantic bootstrapping assumes that word learning must precede syntactic learning. Syntactic bootstrapping [Landau and Gleitman, 1985; Naigles, 1990] counters that children are able to use the structure of the sentence to drive the selection of the meaning of the sentence. Together with semantic bootstrapping the two theories provide a cyclical bootstrapping process for language acquisition, but the question is: where does this cycle start; how can children begin to bootstrap when the language input is mostly ambiguous regarding both structure and meaning. In this thesis we demonstrate a full scale computer model using ‘structure-mapping’ [Fisher et al., 2010]: simple representations depending only on the minimal knowledge available (namely knowing a small number of concrete nouns) can prevail, allowing a language learner to begin to identify verbs and determine who does what to whom.

We instantiate the structure-mapping account through our model’s basic assumptions, or “Universal Grammar”: 1) Sentences contain at least one predicate and arguments for that predicate, 2) Arguments fill unique abstract roles in predicates (Agent, Patient, etc.), 3) Structural/Combinatorial properties of words are encoded and *used* independent of knowing the meaning of the word, and 4) Nouns are treated as arguments for verb predicates. These simple innate assumptions allow the learner to immediately begin to identify verbs and map meaning to a sentence once a set of nouns have been identified.

Applying these assumptions to the “girl tickles boy” example, the child is able to recognize "girl" and "boy" as nouns, and by identifying them as arguments can treat the sentence as conveying an event containing two participant-roles, eliminating such interpretations as "the boy is squirming" or "the girl is laughing". Even without correct identification of exact meaning (deciding between "girl and boy are playing" and "girl tickles boy" interpretations), the occurrence of the word "tickles" in a two argument sentence provides evidence for it as a two argument predicate. If this word commonly appears in similar structures then the child can conclude it is the predicate and knows something about the relative position and order of its arguments.

In this thesis we develop a machine learning model that is able to combine simple, partial syntactic constraints along ambiguous semantic feedback to begin to learn sentence structure and meaning. The idea of syntactic bootstrapping does not throw out the contribution of the semantic scene, it merely acknowledges that faced with the enormous ambiguity of even the simplest real world setting, additional structural cues from the sentence itself are necessary to guide interpretation and learning.

1.2 Thesis Statement

Even with its complexity, language is learned, and thus learnable. We demonstrate with real child directed speech (CDS) that simple abstract representations are able to begin extracting sentence-level semantics in the presence of noise and ambiguity. More specifically we show how small sets of high precision noun knowledge combined with abstract patterns are able to begin both identifying structure and semantics in sentences. This early noun knowledge is defensible in young children, and our process represents a mechanism for them to begin learning structure and syntax early, without relying on complete (and more difficult) verb knowledge. Furthermore we show that a combination of bottom-up noun knowledge and abstract representations allows a learner to gain a foothold when faced with an ambiguous world.

The overall contributions of this thesis is twofold: 1) We develop a machine learning model that learns on a realistic scale and is able to support psycholinguistic theories of syntactic bootstrapping for language acquisition, demonstrating the theory's effectiveness and applicability on real child directed speech. 2) Through this model we demonstrate a useful alternative means of supervision that does not rely on fine grained knowledge injection in the form of fully tagged and labeled data, but instead uses weaker, ambiguous higher level feedback to infer both intermediate representations and final task predictions.

As more specific psycholinguistic contributions, in developing our BabySRL model we show that:

1. Simple, psycholinguistically plausible representations can allow a start to sentence understanding without needing to know every word in the sentence.
2. Simple representations are robust to bootstrapped, partial feedback and noisy input
3. Early noun knowledge can bootstrap both noun *and* verb category knowledge, useful for simple representations
4. A learner with simple representations can recover both syntax and semantics from real sentences with ambiguous semantic feedback, but only when they incorporate some noun knowledge.

Putting these claims together we demonstrate a learning protocol that uses well developed supervised learning techniques in an unsupervised setting, combining simple assumptions and small amounts of background knowledge to generate a supervision signal from real data. This learning can form the basis for further self-improvement when faced with sentences of growing complexity, just as a real child does.

1.3 Thesis Organization

The main focus of this thesis is the development of our BabySRL model from basic representations through processing its input and producing its training signal. This model represents key assumptions of the structure-mapping account, integrated both into the structure of the semantic labeling task and the syntactic representation used. Figure 1.1 traces an example sentences through the complete BabySRL pipeline. Individual aspects of the pipeline will be described in future chapters.

The overall outline of the remainder of this thesis is:

- Chapter 2 gives general background information about some concepts of supervised and unsupervised learning that are used in this thesis, as well as about the specific task of Semantic

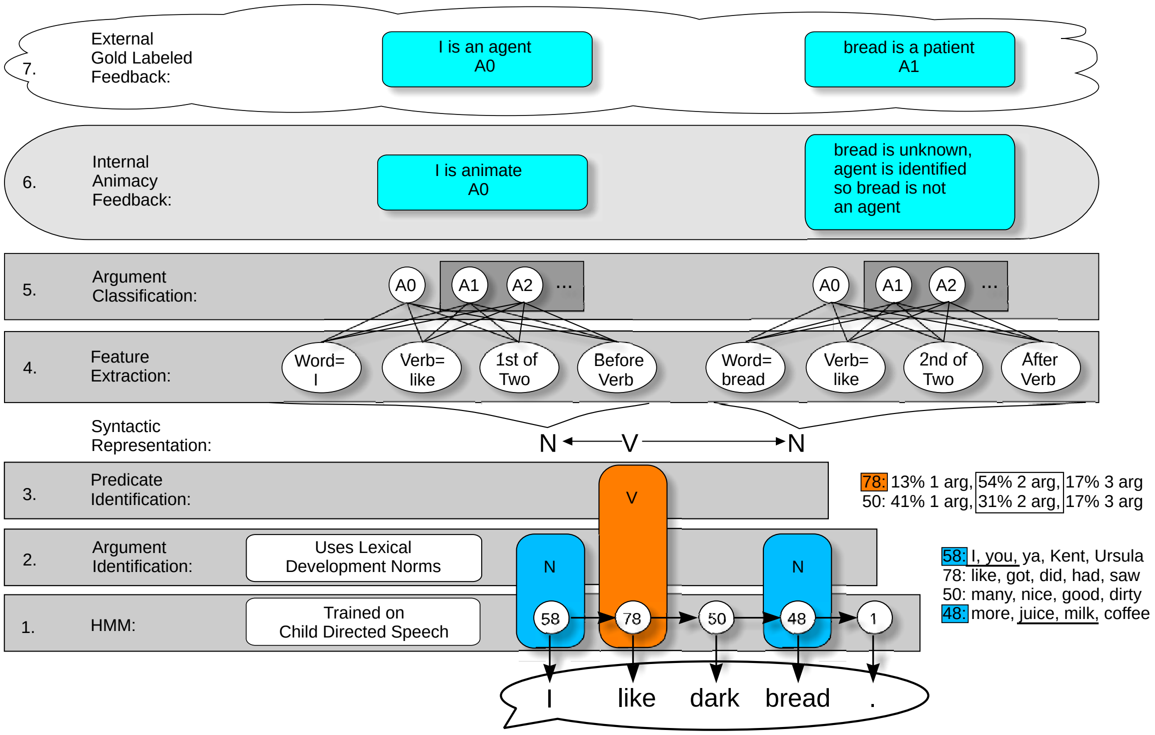


Figure 1.1: Complete BabySRL Pipeline Example for sentence "I like dark bread ." Base semantic classifier and feature representation based on simple linear syntactic representation (number and order of nouns, relative location of verb), box 4 and 5, described in **chapter 3**. This simple syntactic representation is induced with minimal supervision using a Hidden Markov Model (HMM) trained on Child Directed Speech (CDS) (box 1), and a seed set of concrete nouns to identify arguments (box 2) and predicates (box 3), described in **chapter 4**. The full BabySRL can be trained with either true semantically tagged CDS (box 7), or can use internally generated semantic feedback based on world knowledge (box 6; animacy of arguments indicates agent, **chapter 4**). The full pipeline model with minimally supervised arguments, simple representation and animacy feedback demonstrates the effectiveness of a small amount of early noun knowledge in bootstrapping simple representations for whole sentence semantic understanding. **Chapter 5** revises the pipeline approach, viewing the syntactic representation as a latent structure that is inferred to help predict ambiguous semantics.

Role Labeling and Unsupervised Part-Of-Speech tagging. A brief review of relevant child language acquisition experimental and computational modeling work will also be covered.

- Chapter 3 will introduce the basic learning model and simple representations that make up the core of our BabySRL model (boxes 4 and 5 in figure 1.1). Experiments with perfect syntactic input and semantic feedback will demonstrate and provide an upper bound on the abilities of these simple representations.

- Chapter 4 strips the perfect syntax and semantics of the BabySRL model from chapter 3 and replaces them with minimally supervised argument and predicate identification (box 1-3 in figure 1.1) and internally generated semantic feedback based on animacy background knowledge (box 6 in figure 1.1). This entire minimally supervised pipeline shows the robustness of simple representations and demonstrates that this entire model can be trained when starting with the knowledge of a small number of concrete nouns.

- Chapter 5 re-evaluates the pipeline approach of BabySRL from the previous two chapters, and instead views the syntactic structure as a hidden variable that is inferred jointly with semantic interpretation of the sentence. This model allows experimentation with both ambiguous semantic feedback and syntactic constraints, and we show that learning is possible with ambiguous semantics, but it does require interaction with some early syntactic knowledge (as provided by minimally-supervised argument identification).

- And Chapter 6 concludes and describes potential future directions of how this model can be used in the first step of self improvement and language acquisition.

Chapter 2

Background

This chapter covers some of the background material that the rest of the thesis builds upon. All research directions discussed cover vast swaths of literature, so only pertinent or illustrative material will be discussed along with pointers to other related work when appropriate. We will start in section 2.1 with a brief description of our supervised learning framework, focusing on online learning algorithms, then look at how lack of supervision is handled in unsupervised or semi-supervised methods (with a special focus on one specific application, unsupervised part-of-speech). Section 2.2 introduces the NLP Semantic Role Labeling task, and also briefly reviews related work that incorporates alternative training for this task in the form of Unsupervised Semantic Role Labeling. Finally section 2.3 gives some relevant background in language acquisition and other psycholinguistacly relevant computational models.

2.1 Supervised Learning

With supervised learning we are interested in learning a function $h : X \rightarrow Y$ that maps examples from some input space X to an output space Y, when we are given a finite number of samples from the true target distribution $D_{X \times Y}$. The goal of a learning algorithm is to find some function h out of a given hypothesis space H that best matches the true distribution $D_{X \times Y}$, where we can formally define matching as minimizing the expected loss in terms of some loss function $L : Y \times Y \rightarrow \mathbb{R}^+$. In this case the learning algorithm $A^* : H \times D \times L$ would return $h = \arg\min_{h' \in H} \mathbf{E}_{(x,y) \sim D_{X \times Y}} [L(h'(x), y)]$. Of course in the real learning case we do not know the true target distribution $D_{X \times Y}$, so we have to rely on empirically estimating it given some sample $S = \{(x_i, y_i)\}^m$ drawn i.i.d. from $D_{X \times Y}$. This suggests a learning algorithm $A : H \times S \times L$ that returns $\hat{h} = \arg\min_{h' \in H} \frac{1}{m} \sum_{i=1}^{m} L(h'(x_i), y_i)$.

Different classes of learning algorithms can thus be compared by their hypothesis space, what loss functions they use, and how they use the training samples. In this thesis we are interested in online learning algorithms that process the training samples one example at a time. The training sample S is viewed as a sequence of examples that the algorithm may run over, perhaps multiple times[1]. This class of algorithms are both biologically relevant due to their limited memory and processing requirements, and for the same reasons scale to handle truly tremendous amounts of data.

The simplest, and most widely studied classification setting is that of a binary classifier where there are only two labels, $Y = -1, 1$. Now examples can be seen as coming from one of two sets, negative and positive, and often the examples are referred to as such. This setting is both simple and easy to analyze, as well as allowing for generalizations and reductions from many more complicated classification settings. As we will see, binary classifiers form an effective base for which to build classifiers with multiple labels or more complicated output structures.

Additionally we focus on the hypothesis space of linear functions, which can be defined the binary setting as:

$$h(x|\Phi, w) = \begin{cases} 1 & \text{if } w \cdot \Phi(x) > 0 \\ -1 & \text{otherwise} \end{cases}$$

where $\Phi : X \rightarrow \mathbb{R}^n$ is a feature function that maps an input example x into a vector of n real valued features, and $w \in \mathbb{R}^n$ is a vector of weights in the feature space, and here acts as a separating hyperplane between negative and positive examples. Learning a linear function requires setting the weight vector w to minimize loss on the training set. Linear classifiers can be efficiently learned with a polynomial number of training examples [Kearns and Schapire, 1994], provide a natural geometric interpretation of the classification space[2], and perhaps more importantly, they focus the actual engineering work of setting up the learning algorithm into how one defines the feature function Φ.

Since we have already specified that we are interested in an online learning algorithm for linear classifiers, a natural algorithm to use is the single layer perceptron [Rosenblatt, 1958]. This is an

[1] Alternatively, we can imagine online algorithms are provided with some method of stochastically sampling one example at a time from the target distribution, instead of being given a fixed size set of examples. Obviously in a finite amount of time such an algorithm would only see an algorithm could only see a finite number of examples.

[2] Although such intuitions can often lead one astray in high dimensional feature spaces

easy to implement learning algorithm that processes training examples one at a time, making a prediction using the current weight vector and updating the weights with a fixed additive amount whenever a mistake is made. The algorithm is presented in algorithm 1.

Algorithm 1 Binary Perceptron Algorithm

```
Input: $S = \{(x_i, y_i)\}_{i=1}^{m}$ where $x_i \in \mathbb{R}^n, y_i \in \{-1, 1\}$, number of training epochs $T$ and learning rate $\alpha$
Output: $w \in \mathbb{R}^n$
Initialize $w = \vec{0}$
for $t = 1 \rightarrow T$ do
    for all Examples $(x_i, y_i) \in S$ do
        $y^* \leftarrow \text{sign}(w \cdot x_i)$
        if $y^* \neq y_i$ then
            $w \leftarrow w + \alpha y_i x_i$
        end if
    end for
end for
```

When the data is linearly separable (a $u \in \mathbb{R}^n$ exists s.t. $y_i = sign(u \cdot x_i) \forall (x_i, y_i) \in S$), the perceptron algorithm will find such a separating hyperplane after making a bounded number of mistakes during training [Novikoff, 1963]. When the data is not linearly separable (such as the case with the famous binary parity or XOR example [Minsky and Papert, 1969]), the simple response is to add more features, increasing the dimensionality into a region where the data is linearly separable. Hence the focus on feature engineering when working with linear classification.

Over the years (decades even) there have been numerous modifications and improvements to the basic fixed increment perceptron algorithm. Winnow [Littlestone, 1988] modifies the basic additive weight update on misclassification to a multiplicative update, leading to faster training, especially in cases where only a small number of features are important and the vast majority are irrelevant. Average or voted perceptron [Freund and Schapire, 1998] remembers the weight vector at each iteration and averages them all together for a final predictor, which has the effect of decreasing the learning rate over time. Additionally it is possible to set the learning rate per example, depending on size of the error (such as with the family of Passive-Aggressive online algorithms [Crammer et al., 2006]), which may lead to faster learning. Of course perceptrons may be viewed in the same general framework of neural networks, so many of the tricks of the trade used there can also be applied (e.g. [Haykin, 1999; LeCun et al., 1998]). In my own experience additive perceptron is easier to tune than winnow, and average perceptron helps in almost all cases although may take more rounds to converge.

So far we have only described binary classification, the case where there are two possible labels in the Y output space. What if there are multiple possible discrete labels, such as Parts of Speech, or Semantic Roles? Here we assume there are N classes, and that the set of Y outputs are the integers $\{1, \cdots, N\}$. In this work we use the standard "one vs. all" classification technique of reducing the learning and classification of N classes into N binary classification tasks. In the context of linear classifiers we train N weight vectors, $\{w_i\}_{i=1}^{N}$, and final prediction is $h(x|\Phi, \{w_i\}^N) = \arg\max_j = 1^N w_i \cdot \Phi(x)$. During training the "one vs. all" means that for each class i we train w_i where all the training examples labeled with class i form the set of positive examples, and all examples from other classes form the negative training examples. This way, it is hoped that when presented with a new example, the true class's classifier will give a higher score than any other class.

There are of course many other schemes for multiclass classification beyond the standard and direct "one vs. all." For example it is also possible to form $\binom{n}{2}$ classifiers in an "all vs. all" approach, training a classifier to distinguish between every pair and then doing either voting or some sort of tournament for predicting the target class. Alternatively it is possible to do more complicated optimization methods, training N classifiers together as one (e.g. [Vapnik, 1998; Crammer and Singer, 2001]), where they trade off a global error or slack per example. Another major class of multiclass classification approaches is error-correcting codes (e.g. [Dietterich and Bakiri, 1995; Crammer and Singer, 2002]), where some number of binary classifiers are trained (this number not necessarily dependent on the number of classes) where each classifier is trained on some "metatask" that combines multiple classes together as positive and negative sets with the goal of creating separable metatasks allowing good individual binary classifiers. Despite (or perhaps because of) the complexity of the alternative approaches, "one vs. all" is still the dominant approach and empirical results suggest that it should fare well if the underlying binary classifiers are strong [Rifkin and Klautau, 2004].

Later in this thesis we will expand the label output space to structured predictions, where for each input x we are predicting some structure y. This case abounds in NLP because often the object of interest are not just categorical (as is the case with POS), but may have some internal structure, such as predicting the full parse tree for a given sentence. This case can be reduced to multiclass classification where Y is the set of all possible structures, but this set may be exponential size (or greater), so it is not possible to efficiently do "one vs. all" or the like. Various methods of learning with structured prediction have been introduced (e.g. CRF [Lafferty et al., 2001], Structured SVM[Tsochantaridis et al., 2004]), LASO[Daumé and Marcu, 2005], etc.), but in this paper we stick with the online, perceptron approach and build on the ideas of Collin's Structured Perceptron [Collins, 2002].

In the structured prediction case we use a feature function $\Phi : X \times Y \to \mathbb{R}^n$ that extracts features from the entire input + label structure. The goal of a linear structured classifier is to find a weight vector $w \in \mathbb{R}^n$ such that $y_i = \arg\max_y w \cdot \Phi(x_i, y)$, i.e. it gives highest weight to the features of true structures over false. Collin's perceptron finds such a weight vector through online training similar to regular binary perceptron: for each example predict the top structure according to current weight vector, if the prediction does not match the true structure, we increment the weights for those features in true structure, and decrement those in the falsely predicted one. This algorithm is illustrated in algorithm 2:

Algorithm 2 Structured Perceptron Algorithm

```
Input: $S = \{(x_i, y_i)\}_{i=1}^m$, Feature function $\Phi : X \times Y \to \mathbb{R}^n$, number of training epochs $T$ and learning rate $\alpha$
Output: $w \in \mathbb{R}^n$
Initialize $w = \vec{0}$
for $t = 1 \to T$ do
    for all Examples $(x_i, y_i) \in S$ do
        $y^* \leftarrow \arg\max_y w \cdot \Phi(x_i, y)$
        if $y^* \neq y_i$ then
            $w \leftarrow w + \alpha(\Phi(x_i, y_i) - \Phi(x_i, y^*))$
        end if
    end for
end for
```

Again, it can be shown that this algorithm will converge to a separating hypothesis if the training data is linearly separable, and acts "reasonably" if data is close to separable [Collins, 2002]. This algorithm is easy to implement and allows the engineer to focus on the important issue of feature extraction.

2.1.1 Unsupervised Learning

While the previous section laid the foundation for the supervised learning techniques used in this thesis, what about unsupervised methods? The goal of unsupervised learning is for human experts to exert their knowledge in the form of models, constraints, or other knowledge sources, and then hope that untagged data will fill in the gaps. This practice manifests itself in two ways: 1) Experts devise a statistical model of how labels relate to data, and then fit this model to the data, treating labels as hidden values, or 2) Background knowledge is used to inject a supervised signal into otherwise untagged text, allowing the use of standard supervised classifiers. The first approach is essentially clustering, a widely studied, used, and rarely understood methodology, and we will discuss one specific manifestation of this approach for unsupervised POS.

One relevant and commonly cited example of the second class of unsupervised methods is Yarowsky's Word Sense Disambiguation (WSD) bootstrapping algorithm [Yarowsky, 1995]. This algorithm attempts to solve the WSD task (determine the sense, or meaning, of a word in context, selecting from a fixed number of "dictionary definitions" for each word) starting with two basic assumptions about language, a handful of seed examples, and larger set of untagged examples. We'll later return to this important idea that it is possible to (begin to) learn various language phenomena with just such a setting: small seed of high precision knowledge, general assumptions/constraints, and exposure to language. For this task the assumptions are that words tend to have the same sense if they appear multiple times in a discourse (one sense per discourse), and that the sense of the word can be determined from context. These two assumptions allow a supervised classifier (decision list in this case) with features based on context to spread, or generalize sense knowledge from the seed set to those words that share context or discourse.

There are many different such algorithms for unsupervised (really minimally or weakly supervised) learning that are often task specific in terms of assumptions and knowledge brought to bear. In section 2.2.1, we will go through some related unsupervised Semantic Role Labeling models as further examples. It is also possible to define a more general framework for minimal supervision that attempts to abstract away some notion of knowledge injection. Co-training [Blum and Mitchell, 1998] is a prominent framework where the expert imports knowledge through a division of the features into multiple independent subsets or "views." The co-training algorithm uses two weak classifiers (trained on small amount of seed examples) with independent views to bootstrap more labeled data by independently selecting and labeling examples they are confident about. The intuition here is that if both views are compatible with the target function (able to learn the function) and conditionally independent given the label, then confident examples from one classifier will appear randomly distributed to the other classifier. Thus by having independent views (as determined by the human expert) share information through cross-training, label confidence, accuracy, and generalization ability is increased.

Unsupervised Part of Speech

As an example of the model based, or clustering, approach to unsupervised learning we here discuss Unsupervised Part of Speech tagging. In this task each word in a text is assigned to some tag or cluster, and then this clustering is compared to hand annotated syntactic category, or POS. We develop and use such a system in our BabySRL model to identify arguments and predicates in section 4.1. One notable aspect of viewing POS as a clustering task is that it is context sensitive clustering, a single word can appear with multiple different POS tags in real language (”I went for a walk” vs. ”I like to walk”) depending on context, so accurate unsupervised tagging requires putting words in multiple clusters[3].

There are various approaches to unsupervised POS, with various levels of knowledge available to the system. Good results can be obtained if a tagging dictionary is supplied (for every word we know the possible POS for that word, e.g. [Merialdo, 1994; Smith and Eisner, 2005; Toutanova and Johnson, 2007]), or as with previously mentioned unsupervised methods a small number of seed labeled examples, or prototypes, are given to a supervised learner [Haghighi and Klein, 2006]. Here we focus on the induction task where little to no POS categorical information is given, beyond possibly desired number of classes. More specifically we focus on methods that use Hidden Markov Models (HMM) as their base statistical model, and incorporate extra information through priors and training constraints. For a recent empirical review and comparison of POS Induction systems, see [Christodoulopoulos et al., 2010]. Interestingly, the authors of this comparison find that older, purely clustering methods (such as Brown Clusters [Brown et al., 1992]) work almost as well as

[3] Whether this complication is necessary is a separate issue, since a majority of words often appear as a single POS in a given discourse, a simplification that we will see exploited.

more modern and more complicated HMM based methods, and may work better as input for further processing (a finding that matches empirical results in other NLP tasks [Koo et al., 2008; Ratinov and Roth, 2009; Turian et al., 2010]).

HMMs have long been used as a model for POS [Garside et al., 1987], naturally matching the simplified view of the syntax of a sentence being a sequence of syntactic categories and the actual observed words are just a manifestation of that category. Unsupervised POS is implemented as a first-order HMM[4] by specifying a sequence of hidden states (interpreted as syntactic category), where the probability of each state is conditioned only on the previous state (transition probability: $p(t_i|t_{i-1})$), and each state emits an observed word (emission probability: $p(w_i|t_i)$). Given these probabilities (along with an initial distribution for first state in a sequence: $\pi(t_0)$), it is possible to efficiently predict the most likely sequence of hidden states for a new sentence using Viterbi decoding, or to predict the most likely state for a given word in the sentence given the rest of the sentence (marginal likelihood, using Forward-Backward algorithm). See Rabiner [1989] for a thorough review of HMM use, training, and good practices.

Early works using HMM models for POS would estimate such probability tables using hand labeled POS training data, but we are interested in methods that estimate these parameters from untagged text directly. Elworthy [1994] and Merialdo [1994] both first explored the use of Baum-Welch [Baum, 1972] Expectation Maximization (EM) unsupervised training for HMM and POS, both as re-estimation from already trained parameters and from uninitialized probabilities. EM for HMM alternates between finding the likely tagging of the training text using current parameters, and then re-estimating those parameters based on the current likely tagging. This approach can be shown to find a local maximum of the model likelihood. [Johnson, 2007] asked why this method does not seem to work very well for linguistic phenomena, including POS, and concludes that the true distribution of categories, with a few tags covering a large number of words (open class) and most tags covering very few words (closed), differs from that produced by EM, which favors a more uniform distribution of words to hidden state.

It is now the job of the unsupervised POS approach to somehow impart this knowledge of state distribution into the HMM. One method is by specifying Bayesian priors on the transition and

[4] It is possible to use higher order HMM for POS, although for unsupervised training the increase in complexity of training and inference is generally not offset by a comparable increase in accuracy.

emission probabilities which should bias the model towards different distributions. With multinomial transition and emission probabilities in the HMM, it is natural to use Dirichlet priors. The Dirichlet distribution has a single parameter, a vector of positive real values that is commonly all set to the same value (known as the concentration parameter, here referred to as α). Intuitively, as α approaches 0, then the Dirichlet prior will prefer distributions that allocate more weight to fewer entities, thus forming sparser models. These Bayesian HMM can be estimated using either Markov Chain Monte Carlo (MCMC) methods such as Gibb's Sampling, or Variational Bayes (VB). Gibb's Sampling (see [Goldwater and Griffiths, 2007] for demonstration on HMM POS induction) generates many samples from the posterior distribution and converges towards likely samples very slowly. VB inference only requires a simple modification to the standard EM algorithm for HMM to incorporate priors (see [MacKay, 1997; Beal, 2003] for more details regarding VB use in HMM). There is a surprising amount of literature that compares these two inference approaches and the various parameter settings that both require, see [Johnson, 2007; Gao and Johnson, 2008; Christodoulopoulos et al., 2010]. In this thesis we use VB when moving to a Bayesian model due to its ease of implementation when one already has EM working.

To further improve unsupervised POS and incorporate more knowledge into the model systems have to move beyond a single symmetric concentration parameter in the Bayesian model. Following from the intuition that closed and open class words show different word distributions, and thus should have different priors Moon et al. [2010] specify different sparseness for different groups of states by altering the prior. As a means of both simplifying inference and incorporating the reasonable constraint that words tend to appear predominantly with a single tag, it is possible to do inference over all appearances of a word at once Lee et al. [2010], which leads to faster convergence but a slightly weaker model. Instead of relying on purely Bayesian methods, it is also possible to directly regularize the posterior to control for sparseness [Graca et al., 2009], or use integer linear programming to minimize model size [Ravi and Knight, 2009]. Berg-Kirkpatrick et al. [2010] replace the multinomial representation of the transition and emission probabilities with a logistic function, enabling them to incorporate additional features such as whether word contains a digit, is capitalized, or various prefixes and suffix features. In this thesis we introduce an alternative mechanism of injecting information using a list of closed class words, an approach which appears to have been simultaneously developed by [Graca et al., 2009].

2.2 Semantic Role Labeling

Semantic Role Labeling is an NLP task to identify and classify the verbal predicate-argument structures in a sentence, assigning semantic roles to arguments of verbs. Combined with the development of robust syntactic parsers, this level of semantic information should aid other tasks in handling and understanding natural language sentences, such as for information extraction or language understanding. While there was early work with a comparable level of semantic classification, the SRL task itself really caught on with the NLP and ML community with the development of the PropBank semantic annotated corpora [Kingsbury and Palmer, 2002; Palmer et al., 2005] and the introduction of the CoNLL (Annual Conference on Computational Natural Language Learning) shared task SRL competitions [Carreras and Màrquez, 2004, 2005]. For a good review of the SRL task along with summary of the state of the art, see [Màrquez et al., 2008].

As an example, here is a sentence from PropBank:

Mr. Monsky *sees* much bigger changes ahead.

And the task is to identify the arguments of the verb “sees” and classify their role in this structure, producing this labeling:

[$_{A0}$ Mr. Monsky] *sees* [$_{A1}$ much bigger changes] [$_{AM-LOC}$ ahead] .

Where A0 (sometimes written as Arg0) represents the agent, in this case the seer, A1 (also Arg1) represents the patient, or that which is being seen, and AM-LOC is an adjunct argument that specifies where (the location) the thing is being seen. Note that given a sentence and a verb in that sentence (“see” above), the task is to both identify the arguments (square brackets above) and the roles of the arguments (here A0, A1 and AM-LOC). The combination of predicate and arguments is known as a proposition, hence the term Proposition Bank or PropBank.

PropBank defines two types of argument roles: core roles A0 through A5, and adjunct like roles such as the AM-LOC above. The core role labels are general across verbs, so A0 and not “seer” above, but the interpretation is specific to that verb usage. These different verb usages or senses are collected in frame files for each verb, provided as part of PropBank. Each frame file has a different frame set for each sense of a verb that both specifies and defines the possible roles and the allowable syntactic frames for this usage. One consistency across rolesets is the interpretation of A0 as a prototypical agent [Dowty, 1991] and A1 as a prototypical patient or theme. In general some attempt was made such that the rolesets for semantically related words should correspond in some meaningful way.

With these framesets, PropBank annotators set about tagging each verb occurrence in the Penn Treebank Wall Street Journal corpus [Marcus et al., 1993] by identifying the frame it belongs to, identifying the arguments of that verb in the sentence, and classifying the roles of those arguments from its frame. As can be seen in the example above arguments are full phrases, and in fact are derived by labeling sub-branches of the syntactic tree as provided in the treebank.

State of the art SRL approaches (as measured in CoNLL competitions) commonly involve a pipeline approach to this complex and multifaceted classification task (e.g. [Punyakanok et al., 2005a]). Given the sentence it is necessary to 1) parse the sentence (perhaps with multiple different parsers), 2) Identify possible arguments based on the parse, 3) Classify the likely roles of each possible argument, and 4) Combine separate argument predictions into a global (sentence level) classification incorporating any global constraints (arguments cannot overlap and must have unique roles, verb frame role constraints, etc). This naturally structured prediction task is reduced to a series of multiclass predictions through this final global inference step, which can makes use of some general purpose solver (such as integer linear programming [Punyakanok et al., 2005b]).

The performance of such SRL systems is intimately tied to the accuracy of the syntactic representation. The argument identification stage usually involves some sort of pruning step [Xue and Palmer, 2004] that considers siblings and cousins of the verb node in the parse tree as possible arguments, and here the accuracy of the tree is especially important Punyakanok et al. [2008]. As seen in the jump in performance between CoNLL 2004 [Carreras and Màrquez, 2004] when training data contained only syntactic chunking information to 2005 [Carreras and Màrquez, 2005] when full tree information was provided as part of the training data, better syntactic trees were helpful, as was considering many trees at once [P. Koomen and Yih, 2005].

If the syntax and semantics are so intricately linked in these learning models, then it would seem that a model that jointly learns both syntax and semantics together should be able to improve on the strict pipeline model. Recent shared tasks [Surdeanu et al., 2008; Hajič et al., 2009] have sought to explore exactly this point, giving a task where systems have to predict both syntactic and semantic structures for given text. Disappointingly, in both cases mainly pipeline approaches achieved the best results, either reranking a small number of joint candidates [Johansson and Nugues, 2008], or relying on well engineered rich features in a straightforward pipeline [Zhao et al., 2009]. This thesis will revisit this idea of joint syntax and semantics, exploring the idea of simple syntax as a latent structure necessary for prediction of semantics.

2.2.1 Unsupervised Semantic Role Labeling

Since this thesis largely concerns itself with learning SRL without full supervision, it may be relevant to look how this problem has been approached previously in the NLP community. We will pay special attention to the knowledge assumptions that previous models have made use of, and how those may either relate to our assumptions about a child learner, or the question of acquisition in general. We separately address more psycholinguistically motivated computational models of language acquisition in section 2.3.1.

One approach to making an unsupervised semantic distinction between verb-argument structures is to cluster verbs based on some distributional cues such as syntactic frames, or types of arguments the verbs appear with. This line of work is inspired by Levin's work on verb classes [Levin, 1993] and the idea that different semantic classes can appear with different fixed sets of syntactic structures (diathesis alternation). [Stevenson and Merlo, 1999; Merlo and Stevenson, 2001; Stevenson and Joanis, 2003] classify verbs into classes based on various linguistic features in both a supervised and unsupervised setting, evaluating how well specific features are able to capture important aspects of verb-argument structure. [Schulte im Walde, 2003] also experiments with explicitly clustering German verbs into semantic classes based on syntactic frame features, relying on an (unsupervised) syntactic parse. These works demonstrate the importance of linking simple to extract structural cues with verb semantics, but their methodology is less relevant to the current work. By performing a hard clustering of verb types, they ignore both verb polysemy and the task of assigning semantics to a specific instance of a verb. Such unsupervised clustering methods may serve to create data or dictionaries for further unsupervised individual role assignments.

For full unsupervised role labelers, the task is often presented as clustering the arguments of a verb (across many sentences) into semantic classes that should hopefully correspond to the desired roles. Much like the supervised case, these systems often follow the pipeline approach of first identifying arguments and then classifying them, with some models focusing on only one of these steps. Abend et al. [2009] focuses on argument identification, starting from an unsupervised POS parse, combining hand crafted rules and counts over data to refine decisions about possible arguments of a verb. Lang and Lapata [2010] focus on role induction, so given arguments and a syntactic parse with labeled dependency links, they build a classifier to separate roles by essentially smoothing over syntactic roles.

Since SRL represents a fairly high level NLP task, to make any headway with unsupervised approaches often requires a great deal of additional knowledge to supplant the missing supervision. Such knowledge can take the form of verb or noun dictionaries, supervised and unsupervised syntactic parses, or hand coded rules. [Grenager and Manning, 2006] assume a syntactic parse and incorporate hand coded rules for how the model may link syntax and semantics, relying on EM to fill in some blanks. [Swier and Stevenson, 2004, 2005] use the VerbNet [Kipper et al., 2000] verb lexicon to constrain possible frames for verbs, and then build a probabilistic model over class based features where the classes come from either VerbNet and WordNet [Fellbaum, 1998]. As their base syntactic representation they rely on chunking to isolate potential frames for a verb [Swier and Stevenson, 2004] or a full supervised syntactic parser [Swier and Stevenson, 2005]. Abend and Rappoport [2010] make their fully unsupervised SRL model by relying on an unsupervised POS induction system [Clark, 2003] and an unsupervised syntactic parser [Seginer, 2007], then building a classifier to discriminate between core and adjunct roles of prepositional arguments. Using their knowledge of the distribution of the labels in the data, they set by hand thresholds for the resulting classification.

All of these models represent a serious knowledge investment, although still less than that required by a fully annotated corpus. One goal of our model is to rely on a smaller amount of injected knowledge to simulate an earlier, less sophisticated learner. Although it must be said that by focusing on CDS, we do potentially make the classification task easier than when faced with full newswire sentences, but perhaps this is how children are able to learn, by starting small.

2.3 Language Acquisition

Children follow a regular pattern of increasing complexity when learning language. Just to give one rough timeline of the the stages of language development, children typically produce babbling at 6-8 months, single words at 9-18 months, two-word mini-sentences at 18-24, 3 word ”telegraphic” sentences 24-30, and multiword sentences at 3 years of age. By 24 months production vocabulary is estimated at 200-300 words [Nelson, 1973; Dale and Fenson, 1996]. The first 100 words of this vocabulary are dominated by common nouns and by 18 months children are considered to have acquired an internal noun class category [Tomasello, 1992]. Verbs are more difficult to acquire, but necessary for full multiword sentences, so how, between roughly 2 and 3 years of age, are children able to make this jump?

Nouns are easier to both recognize in the scene and in the communicative event (eye gaze, pointing, holding, etc.) even without linguistic cues Gillette et al. [1999]). Verbs on the other hand are much more difficult, depending on both an understanding of speaker intent from a scene that contains many different visible and not visible relations and actions, and understanding the relation between linguistic objects that may span an entire sentence (after first somehow concluding that there are multiple objects in the sentence and that they are somehow related). In this learning situation children hear a sentence while immersed in some scene, and we are interested in how they eventually acquire the knowledge of verb structure and semantics: how the semantic arguments of a verb map to its syntactic frame, what possible frames a verb may appear in, how different semantic categories demonstrate different syntactic behaviour, etc. How children are able to learn and generalize about verbs meaning and behaviour is a key question in the study of language acquisition.

To learn about verbs’ meaning, behaviour and syntax, Semantic Bootstrapping [Pinker, 1984] proposes that children use their understanding of the semantics of the sentence and scene to determine the structure of the sentence. When first encountering the sentence “The dog chases the cat” and the accompanying scene, semantic bootstrapping says that children use their understanding that a dog is chasing a cat, map the word “dog” to the entity dog, and “cat” to the cat, and then they can start to accumulate rules and patterns such as the agent, or the “chaser”, appears before the verb and the patient or “chasee” appears after the verb. Over a large number of such sentence/meaning pairs children can begin to acquire general rules for how syntax and semantics combine in both general and verb specific ways.

The issue with the above scenario is how are the children supposed to interpret the sentence “The dog chases the cat” corresponds to the dog chasing the cat if they do not already know the meaning of the verb chase. Why can the semantics of the sentence not mean that the cat is fleeing the dog? Or that the dog and the cat are running? Without already understanding the meaning of the verb, or how verbs are used in sentences, how can a child identify that there are even two arguments in the sentence, and map those arguments to entities in the scene. The semantics of the world is far more ambiguous than providing a single consistent interpretation to every scene, so the learner must use some additional guidance in making sense of the feedback.

Usage based theories[Tomasello, 2000, 2003] posit that children learn verbs individually, through exposure and repetition, before finally generalizing that there exists some verb class (or classes). Child production experiments show that children do not (or are less willing to) generalize argument structure for novel verbs, thus do not consider novel verbs in the same category, or to share properties with other verbs. Of course to begin learning about an individual verb and to later recognize similarities between verbs implies some sort of abstract representation, and as we will discuss comprehension studies (mentioned below) have shown that children do seem to generalize understanding of situations involving novel verbs from an early age.

Syntactic Bootstrapping [Landau and Gleitman, 1985; Naigles, 1990] complements semantic bootstrapping by proposing that children use early or innate knowledge of syntax to constrain possible meanings given ambiguous semantics. The “structure-mapping” account of syntactic bootstrapping [Fisher et al., 2010] says that children use abstract representations and their early noun class knowledge as a form of structural constraint to quickly generalize from their provided level of semantic feedback. By relying on abstract representations early on, structure-mapping predicts generalization to novel verbs, and thus allows bootstrapping learning.

The structure-mapping view of early verb and syntax acquisition proposes that children start with a shallow structural analysis of sentences: children treat the number of nouns in the sentence as a cue to its semantic predicate-argument structure [Fisher, 1996; Gillette et al., 1999; Lidz et al., 2003]. This view is based on three key assumptions: First, sentence comprehension is grounded by the acquisition of an initial set of concrete nouns. Nouns are arguably less dependent on prior linguistic knowledge for their acquisition than are verbs; thus children are assumed to be able to identify the referents of some nouns via cross-situational observation [Gillette et al., 1999]. Second, these nouns, once identified, yield a skeletal sentence structure. Children treat each noun as a candidate argument, and thus interpret the number of nouns in the sentence as a cue to its semantic predicate-argument structure [Fisher, 1996]. Third, children represent sentences in an abstract format (i.e. a structural representation that goes beyond representing individual specific word forms) that permits generalization to new verbs [Gertner et al., 2006].

The structure-mapping account makes strong predictions. First, as soon as children can identify some nouns, they should interpret transitive and intransitive sentences differently, simply by assigning a distinct semantic role to each noun in the sentence. Second, language-specific syntactic learning should transfer rapidly to new verbs. Third, some striking errors of interpretation can occur. In “Fred and Ginger danced”, an intransitive verb is presented with two nouns. If children interpret any two-noun sentence as if it were transitive, they should be fooled into interpreting the order of two nouns in such conjoined-subject intransitive sentences as conveying agent-patient role information. Experiments with young children support these predictions. First, 21-month-olds use the number of nouns to understand sentences containing new verbs [Yuan et al., 2007]. Second, 21-month-olds generalize what they have learned about English transitive word-order to sentences containing new verbs: Children who heard ”The girl is gorping the boy” interpreted the girl as an agent and the boy as a patient [Gertner et al., 2006]. Third, 21-month-olds make the predicted error, treating intransitive sentences containing two nouns as if they were transitive: they interpret the first noun in “The girl and the boy are gorping” as an agent and the second as a patient [Gertner and Fisher, 2006]. This error is short-lived. By 25 months, children add new features to their representations of sentences, and interpret conjoined-subject intransitives differently from transitives [Naigles, 1990].

2.3.1 Computational Models

In summarizing computational models that explicitly model elements of language acquisition, we focus on those models that investigate the learning of both semantics and syntax. This means leaving out generative syntax models (e.g. [Waterfall et al., 2010; Bannard et al., 2009]) that attempt to model aspects of early syntax acquisition, possibly learning from real child directed speech. As in the discussion of unsupervised SRL models above, our interest here is in what knowledge and feedback is assumed by each model. Because these are computational models of cognitive processes, we are also interested in the possibly cognitively plausible representations used by the models.

Connectionist models have proved to be a vital mainstay for building computational models of various cognitive process, and language acquisition is no different. Desai [2002, 2007] builds a recurrent network [Elman, 1991] that predicts simple semantics of a sentence: whether the verb is causal or non-casual. This network is trained on a generated grammar of phrases and sentences of one or two nouns, similar to test sentences we will later introduce. One interesting aspect of recurrent network training is that the sentence is presented one word at a time to the network, while the semantic feedback for the whole sentence is provided as feedback, thus the network has to figure out a linking between the order it sees and full sentence semantic prediction. Chang et al. [2006] also make use of a recurrent network, although their task is to predict the next word given previous word and meaning. Their model incorporates an internal representation of both lexical semantics (it learns the connection between the word "dog" and symbol DOG) and abstract argument role semantics to help predict the word order for a sentence given its meaning. Both of these models are trained with sentences generated by an artificial grammar paired with true semantics.

Allen [1997] presents a connectionist model whose architecture looks at a representation of an entire sentence when making a prediction about semantics, conceptually closer to some of the representations we use in this paper (representations that need to see the entire sentence to define features). They represent sentences in terms verb, preposition, and semantic features for nominal arguments of the verb (from WordNet), and they predict the semantic role for each argument (abstract roles such as cause, patient, motion, etc.) along with more specific subroles. Trained over a fully labeled sample of CHILDES, their model was able to generalize semantics to sentences with novel verbs, based on the argument semantics and sequence in the sentence, reflecting similar results we will show with our base system in a fully supervised case.

Tourigny [2010] build a recurrent network for predicting the noun argument semantics (at the level of Agent/Undergoer) of a miniature language with six different verb classes. Interestingly this model is not trained with fully semantically labeled sentences. To model syntactic bootstrapping, some sentences without given semantic interpretations were included in the training to complement a set of fully labeled sentences. During training, the model needed to make its own prediction about the scene of unlabeled sentences, and then train on that prediction. While their model was able to learn from the labeled sentences, they report less than successful results for novel words that were only present in unlabeled sentences during training. In this thesis we develop machine learning techniques that focus on some level of supervision between these two extremes where we know a little something about both the input and the semantics, and use that to learn over real sentences and broader semantics.

[Alishahi and Stevenson, 2008] develop a Bayesian clustering approach to learning and generalizing argument structure and semantics. Instead of hard clustering verbs, their system clusters specific verb usage frames, thus allowing the system to predict semantic and syntactic frames for novel sentences and verbs. The input frames that the system clusters contains information about the sentence/meaning pair including head verb, semantics of verb, number of arguments, argument roles and lexical categories, and syntactic pattern (such as NVN, NV, etc.). For verb semantics they use nine semantic primitives (*act, cause, move, become, possess, change-of-state, perceive, contact, and manner*), and for the arguments use 10 abstract thematic roles (*agent, theme, destination, source, beneficiary, stimulus, state, experiencer, instrument, co-agent*) [Jackendoff, 1990]. While they assume that the child is able to extract this level of information from a scene/utterance pair, they do acknowledge that some noise is present in the environment and incorporates this via dropping features from some terms during training.

[Alishahi and Stevenson, 2010] expand on their previous system by replacing the abstract role information with word specific semantic information and allowing the clustering to discover appropriate level of semantics for a given frame cluster. Instead of identifying an abstract role, the input argument structure frame uses relatively deep semantics provided by WordNet hierarchy and some hand coded event-based properties for each argument. Additionally the level of noise during training was augmented with both missing features and some bootstrapped examples where a missing feature was filled in with current most likely prediction of the system. When tested with sentences with a novel verb, the system was able to predict general semantics for argument positions based largely on syntactic pattern.

Previous models have shown that it is possible for simple learners to link semantics to syntax using a variety of representation schemes based on various levels of semantic and syntactic sophistication. One key aspect of these models is that they use a level of semantic feedback, or understanding of the scene, that is unrealistically high in the case of an actual child attempting to learn language. In cases where noise is added to the supervision, this noise either assumes that the interpretation is largely valid, or otherwise there is absolutely no supervision. What we wish to demonstrate with this thesis is an alternative machine learning method that makes use of intermediate levels of supervision, relying on combinations of background knowledge and partial interpretation of scene and sentence to guide language learning.

Chapter 3

Baby SRL Model

This chapter introduces the BabySRL learning model, the training data used and the simple representations that will be the focus of much of the rest of this thesis. We test these representations in a fully supervised setting to validate that they can capture the desired semantic patterns. To test the generalization abilities of this system we introduce a new testing paradigm using constructed test sentences in this chapter, experimenting with various parameters in this initial study. To show off the capabilities of our BabySRL, we explore the ramifications of one possible means of developing knowledge during development, which will also be reflected in more principled results in the next section. Much of this chapter was published in [Connor et al., 2008].

From an NLP perspective this feature study provides evidence for the efficacy of alternative, simpler syntactic representations in gaining an initial foothold on sentence interpretation. It is clear that human learners do not begin interpreting sentences in possession of full part-of-speech tagging, or full parse trees. By building a model that uses shallow representations of sentences and mimics features of language development in children, we can explore the nature of initial representations of syntactic structure and build more complex features from there, further mimicking child development.

3.1 CHILDES Training Data

One goal of the BabySRL project is to attempt to learn language the way a child does, which means use as input the same input available to an actual child. To accomplish this we used samples of parental speech to three children (Adam, Eve, and Sarah; [Brown, 1973]), available via CHILDES [MacWhinney, 2000]. The SRL annotated corpus consists of parental utterances from sections Adam 01-23 (child age 2;3 - 3;2), Eve 01-20 (1;6 - 2;3), and Sarah 01-90 (2;3 - 4;1).

All verb-containing utterances without symbols indicating disfluencies were automatically parsed with the Charniak parser [Charniak, 1997], annotated using an existing SRL system [Punyakanok et al., 2008] and then errors were hand-corrected. The final annotated sample contains about 15,148 sentences, 16,730 propositions, with 32,205 arguments: 3951 propositions and 8107 arguments in Adam, 4209 propositions and 8499 arguments in Eve, and 8570 propositions and 15599 arguments in Sarah.

3.1.1 Preprocessing and Annotation

During preprocessing of the CDS transcripts, only utterances from the Mother and Father were used. Other adults were typically present, including the researchers who collected the data, but we focused on parental speech because we considered it most likely to be typical CDS. Because our goal was to create a corpus for studying input for language learning, we made no attempt to annotate the children's speech.

Removing sentences that contained symbols indicating unintelligible/unidentifiable speech or did not contain a verb represents a relatively strict filter. After an initial experience of annotation of one child (Eve), additional guidelines were set, especially in regard to what constituted a main or auxiliary verb, and it was decided that 'be' verbs would not be annotated even if acting as a main verb in the sentence. Of the 45,166 parental utterances in the sections annotated, only 15,148 were parsed and annotated, less than 34% of total utterances. Many of the ignored utterances were short ("yes .", "what ?", "alright .", etc.), marked as ambiguous by the original transcribers so we decided to ignore them ("we'll get xxx a pencil ."), or did not contain an explicit main verb ("no graham crackers today .", "macaroni for supper ?").

In general annotators were instructed to follow Propbank guidelines [Palmer et al., 2005] in their semantic annotations, matching decisions with Propbank's previously identified verb frames. If no frame exists for a specific verb (such as "tickle", which can be found in CDS but not the newswire that Propbank was developed on), or a frame had to be modified to account for uses specific to casual speech, then the annotators were free to make a new decision and note this addition[1].

To assess the reliability of the SRL annotation, 15 of the 133 files (5 sections from each child)

[1] Corpus, decision files and additional annotation information available at `http://cogcomp.cs.illinois.edu/~connor2/babySRL/`

were annotated by 2 separate annotators and then compared: Eve sections 12, 14, 16, 18, Adam 15, 16, 18, 20, 22 and Sarah 32, 37, 46, 47, 83. Across all files annotators agreed on an average of 96.57% of the annotated arguments (matching both argument boundaries and label). This resulted in a very good level of agreement at the full proposition level as well: The annotators agreed on the boundaries and labels of all arguments in 88.50% of propositions.

The full Propbank SRL annotation labels multiword argument phrases relative to a target predicate, but our focus in these experiments is on individual noun identification. This matches the structure mapping assumption of treating each noun as a potential argument, and classifying its semantic role. To reconcile and simplify the labeled multiword arguments we converted them to labeled single nouns (as identified by POS). A simple heuristic collapsed compound or sequential nouns to their final noun: an approximation of the head noun of the noun phrase. For example, 'Mr. Smith' was treated as the single noun 'Smith'. Other complex noun phrases were not simplified in this way. Thus, a phrase such as 'the toy on the floor' would be treated as two separate nouns, 'toy' and 'floor'. This represents the assumption that young children know 'Mr. Smith' is a single name, but they do not know all the predicating terms that may link multiple nouns into a single noun phrase. Note that this labeled noun data is only used for training; when testing on heldout labeled CDS we compare to the full arguments, although our system only identifies individual nouns.

Using the true labels during training represents a clear upper bound in terms of knowledge available to the child. The world is full of ambiguities, and for a child to understand the intended meaning of each word in the sentence while first learning language is a feat on the order of mind reading. Training with this data reflects both what the representation is capable of and what patterns may exist in the data children receive. In future sections we explore what happens when we limit the amount of supervision the system receives to a more plausible, or even lower bound setting.

As this chapter represents an initial validation of the learning model, we report results only for a single child's data (Eve). In all future chapters we look at results across all three children.

3.2 SRL Learning Model

Our learning task is similar to the full SRL task [Carreras and Màrquez, 2004], except that we classify the roles of individual words rather than full phrases. A full automatic SRL system (e.g.

[Punyakanok et al., 2005a]) typically involves multiple stages to 1) parse the input, 2) identify arguments, 3) classify those arguments, and then 4) run inference to make sure the final labeling for the full sentence does not violate any linguistic constraints. Our simplified SRL architecture (Baby SRL) essentially replaces the first two steps with heuristics. Rather than identifying arguments via a learned classifier with access to a full syntactic parse, the Baby SRL treats each noun in the sentence as a candidate argument and assigns a semantic role to it.

The simplified learning task of the Baby SRL implements a key assumption of the structure-mapping account: that at the start of multiword sentence comprehension children can tell which words in a sentence are nouns [Waxman and Booth, 2001], and treat each noun as a candidate argument. Feedback is provided based on annotation in Propbank style: in training, each noun receives the role label of the phrase that noun is part of. Feedback is given at the level of the macro-role (agent, patient, etc., labeled A0-A4 for core arguments, and AM-* adjuncts). We also introduced a NO label for nouns that are not part of any argument.

For argument classification we use a linear classifier trained with a regularized perceptron update rule [Grove and Roth, 2001]. This learning algorithm provides a simple and general linear classifier that has been demonstrated to work well in other text classification tasks, and allows us to inspect the weights of key features to determine their importance for classification. The Baby SRL does not use inference for the final classification. Instead it classifies every argument independently; thus multiple nouns can have the same role.

3.2.1 Simple Representation

The basic representational assumption of the Baby SRL is that nouns are classified into semantic roles relative to some predicate. For this to happen, nouns must be identified, and a verb is assumed to exist in the sentence (and potentially be identified as well). Because we focus on role classification in this chapter, we assume that the identity of the nouns in the sentence is known to us, and can be used both to identify arguments and act as a structural representation. Figure 3.1 illustrates the classification of an example sentence “I like dark bread” where ‘I’ and ‘bread’ are considered the arguments of ‘like.’ The features extracted from these arguments will be explained below.

Without using any structure from the sentence, one piece of information we do have for a target

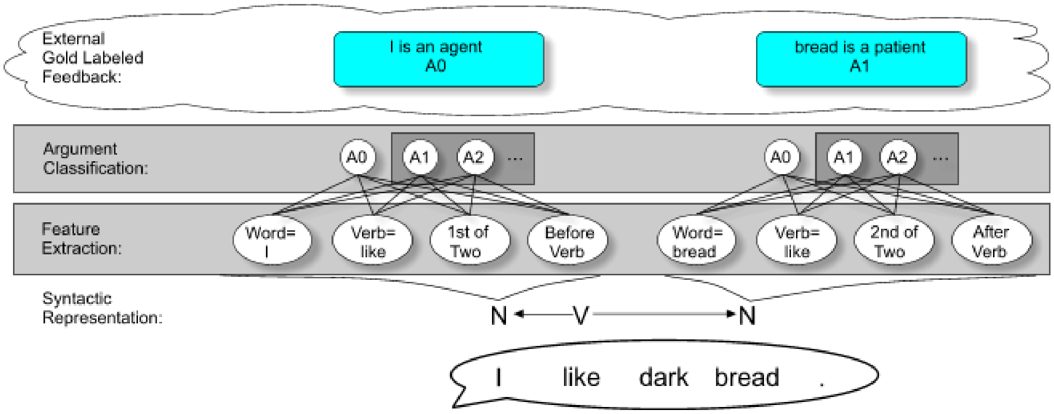


Figure 3.1: BabySRL basic architecture processing sentence "I like dark bread ." Given the sentence, first a syntactic representation is formed using knowledge of the nouns and verb in the sentence (here we assume this is available to the learner). This linear representation recognizes two arguments (nouns), and classifies each one relative to the target verb 'like'. The features for each argument are the noun and verb itself, the noun pattern ('I' is first of two nouns, 'bread' is second of two) and relative position to the verb (before or after). These features are fed into a linear classifier which predicts a semantic role for that argument. Feedback is given to correct the classifier's prediction; here we assume veridical feedback is being given from some exterior source (hand annotation).

noun is the noun itself. We use as a lexical baseline features indicating the word form of the target noun and the predicate it is being classified relative to. These features should indicate per word preferences for roles that a word is often classified as, as well as for predicate the roles that often appear with that verb. What these features do not allow is generalizing to new sentences and novel words. All representations that follow are meant to improve on this memorizing per word in generalizing to new sentences based on simple structures in the sentence.

The basic feature we propose is the noun pattern feature. We hypothesize that children use the number and order of nouns to represent argument structure. To encode this we created a feature (NPattern) that indicates how many nouns there are in the sentence and which noun the target is. For example, in our two-noun test sentences noun A has the feature '_N' active indicating that it is the first noun of two. Likewise for B the feature 'N_' is active, indicating that it is the second of two nouns. This feature is easy to compute once nouns are identified, and does not require fine-grained distinctions between types of nouns or any other part of speech.

At some point the learner must develop more sophisticated syntactic representations. These could include many aspects of the sentence, including noun-phrase and verb-phrase morphological features, and word-order features. If, in addition to nouns we can also identify verbs, then we can incorporate this information into our structural representation. We did this by adding a verb position feature (VPosition) that specifies whether the target noun is before or after the verb. Now simple transitive sentences in training should support the generalization that pre-verbal nouns tend to be agents, and post-verbal nouns tend to be patients.

The representation described above provides four active features for each argument example (as illustrated in figure 3.1) when they are all combined. It is also possible to include more specific combinations of features for each example, such as lexicalizing either of the structural feature with the verb[2]. This new feature, such as lexicalized noun pattern, will be active when the argument is the first of two nouns *and* the verb has some value. These features can now acquire specific structural patterns for each predicate, but will not be applicable when encountering novel verbs.

The rest of this chapter explores the impact of learning with each of these features over the semantically tagged CDS data, especially as they relate to simple patterns of one and two argument sentences. Future chapters build on these experiments, using these simple representations as their basic learning building block.

3.3 Experimental Setup

Throughout this thesis we will use two main types of experimental setups to evaluate SRL performance: full SRL performance on held out set of CHILDES role tagged data and evaluation using constructed test sentences. The constructed test sentences allow us to focus on specific phenomena, especially as they relate to generalization to sentences with known nouns and a novel verb. These sentences were designed to mimic test sentences used in experiments with children described previously.

All constructed test sentences contained a novel verb (‘gorp’) in one of three templates: with one noun ‘A gorps’ or two nouns ‘A gorps B’ and ‘A and B gorp’, where A and B were replaced with nouns that appeared more than twice in training.

We structured our tests of the BabySRL to test the predictions of the structure-mapping account.

[2] It is of course also possible to lexicalize with the noun, but since structure of the sentence is so dominated by verb, this lexicalization was explored first and more fully

(1) NounPat features will improve the SRL's ability to interpret simple transitive test sentences containing two nouns and a novel verb, relative to a lexical baseline. Like 21-month-old children [Gertner et al., 2006], the SRL should interpret the first noun as an agent and the second as a patient. (2) Because NounPat features represent word order solely in terms of a sequence of nouns, an SRL equipped with these features will make the errors predicted by the structure-mapping account and documented in children [Gertner and Fisher, 2006]. (3) NounPat features permit the SRL to assign different roles to the subjects of transitive and intransitive sentences that differ in their number of nouns. This effect follows from the nature of the NounPat features: These features partition the training data based on the number of nouns, and therefore learn separately the likely roles of the '1st of 1 noun' and the '1st of 2 nouns'.

These patterns contrast with the behavior of the VerbPos features: When the BabySRL was trained with perfect parsing, VerbPos promoted agent-patient interpretations of transitive test sentences, and did so even more successfully than NounPat features did, reflecting the usefulness of position relative to the verb in understanding English sentences. In addition, VerbPos features eliminated the errors with two-noun intransitive sentences. Given test sentences such as 'You and Mommy krad', VerbPos features represented both nouns as pre-verbal, and therefore identified both as likely agents. However, VerbPos features did not help the SRL assign different roles to the subjects of simple transitive and intransitive sentences: 'Mommy' in 'Mommy krads you' and 'Mommy krads' are both represented simply as pre-verbal.

We filled the A and B slots by sampling nouns that occurred roughly equally as the first and second of two nouns in the training data. For various experiments we used different selections of nouns to bias the system differently. The three predominant sampling methods we used were for Unbiased nouns, Biased nouns, and Animate nouns. In the unbiased case we uniformly selected A and B from the list of nouns, so the test sentences do not reflect any possible order statistics from the real sentences. For biased case we sample A nouns based on the distribution of first of two nouns, and select B based on the word's distribution of second of two nouns in the training data. This way we can see the lexical ordering effect where words that are more likely to appear first in a sentence will do so in the test sentences. For the animate noun test set (used in future chapters), we identify a set of animate nouns that appear in each child's data and create test sentences where A and B are filled with all pairs of these nouns. Appendix A lists these nouns as used to fill the test sentences, along with statistics about how they appear in the data.

For each of the Unbiased and Biased noun sets used in this chapter we generated a test set of 100 sentences by randomly sampling nouns to fill the templates (depending on the biased or unbiased sampling method). The focus of the experiments in this chapter are how the simple representations compete with lexical baseline, and how both relate to different sampling of nouns; a biased sampling will benefit the lexical baseline since the words themselves carry important information relating to how they are used in training data.

The test sentences with novel verbs ask whether the classifier transfers its learning about argument role assignment to unseen verbs. Does it assume the first of two nouns in a simple transitive sentence (‘A gorps B’) is the agent (A0) and the second is the patient (A1)? Does it over-generalize this rule to two-noun intransitives (‘A and B gorp’), mimicking children’s behavior? We used two measures of success, one to assess classification accuracy, and the other to assess the predicted error. We used a per argument F1 for classification accuracy, with F1 based on correct identification of individual nouns rather than full phrases. The desired labeling for ’A gorps B’ is A0 for the first argument and A1 for the second; for ’A and B gorp’ both arguments should be A0.

Because in general we are comparing the classification of individual nouns to labeled phrases (in the held out test set), we defined precision as as the proportion of nouns that were given the correct label based on the argument they belong to. For example if in the test set the argument “The large man” was tagged as A0, and the classifier predicted ‘man’ to be A0, then this would be correct. Likewise recall was defined as the proportion of complete arguments for which some noun in that argument was correctly labeled. In this case, given the annotated phrase “the man and woman” labeled as A0, if the classifier correctly notes either individual noun ‘man’ or ‘woman’ as A0, then this phrase will be considered correct in terms of argument recall.

To measure predicted errors we also report the proportion of test sentences classified with A0 first and A1 second (%A0A1). This labeling is a correct generalization for the novel ’A gorps B’ sentences, but is an overgeneralization for ’A and B gorp.’

For these experiments the implementation of additive update perceptron in the SNoW learning architecture A. Carlson and Roth [1999] was used as the learning algorithm. Training was run for 5 rounds, with a learning weight of 0.1, prediction threshold of 3.0 for multiclass one vs all training, initial weight of 1.0 and a thick separator of 1.5 (predictions within the thick separator around the prediction threshold are counted as mistakes during training). These parameters were tuned on cross-validation of Eve training data and initial experiments with subsets of Wall Street Journal corpus.

3.4 Experimental Results

	CHILDES				WSJ	
	Unbiased Noun Choice		Biased Noun Choice		Biased Noun Choice	
	A gorps B	A and B gorp	A gorps B	A and B gorp	A gorps B	A and B gorp
Features	%A0A1	%A0A1	%A0A1	%A0A1	%A0A1	%A0A1
1. Words	0.38	0.38	0.65	0.65	0.31	0.31
2. NPattern&V	0.28	0.28	0.67	0.67	0.31	0.31
3. NPattern	0.65	0.65	0.92	0.92	0.44	0.44
4. + NPattern&V	0.65	0.65	0.90	0.90	0.53	0.53
5. + VPosition	0.96	0.00	1.00	0.01	0.88	0.39

Table 3.1: Experiments showing the efficacy of Noun Pattern features for determining agent/patient roles in simple two-noun sentences. The novel verb test sets assess whether the Baby SRL generalizes transitive argument prediction to unseen verbs in the case of ‘A gorps B’ (increasing %A0A1), and overgeneralizes in the case of ‘A and B gorp’ (increasing %A0A1, which is an error). By varying the sampling method for creating the test sentences we can start with a biased or unbiased lexical baseline, demonstrating that the noun pattern features still improve over knowledge that can be contained in typical noun usage. The simple noun pattern features are still effective at learning this pattern when trained with Wall Street Journal training data.

3.4.1 Noun Pattern

Table 3.1 shows the initial feature progression that involves this feature. The baseline system (feature set 1) uses lexical features only: the target noun and the root form of the predicate.

We first tested the hypothesis that children use the NPattern features to distinguish different noun arguments, but only for specific verbs. The NPattern&V features are conjunctions of the target verb and the noun pattern, and these are added to the word features to form feature set 2. Now every example has three features active: target noun, target predicate, and a NPattern&V feature indicating ’the target is the first of two nouns and the verb is X.’ This feature does not improve results on the novel ’A gorps B’ test set, or generate the predicted error with the ’A and B gorp’ test set, because the verb-specific NPattern&V features provide no way to generalize to unseen verbs.

We next tested the NPattern feature alone, without making it verb-specific (feature set 3). The noun pattern feature was added to the word features and again each example had three features active: target noun, target predicate, and the target's noun-pattern feature (first of two, second of three, etc.). The abstract NPattern feature allows the Baby SRL to generalize to new verbs: it increases the system's tendency to predict that the first of two nouns is A0 and the second of two nouns is A1 for verbs not seen in training. Feature set 4 includes both the abstract, non-verb-specific NPattern feature and the verb-specific version. This feature set preserves the ability to generalize to unseen verbs; thus the availability of the verb-specific NPattern features during training did not prevent the abstract NPattern features from gathering useful information.

3.4.2 Lexical Cues for Role-Labeling

Thus far, the target nouns' lexical features provided little help in role labeling, allowing us to clearly see the contribution of the proposed simple structural features. Would our structural features produce any improvement above a more realistic lexical baseline? We created a new set of test sentences, sampling the A nouns based on the distribution of nouns seen as the first of two nouns in training, and the B nouns based on the distribution of nouns seen as the second of two nouns. Given this revised sampling of nouns, the words-only baseline is strongly biased toward A0A1 (biased results for feature set 1 in table 3.1). This high baseline reflects a general property of conversation: Lexical choices provide considerable information about semantic roles. For example, the 6 most common nouns in the Eve corpus are pronouns that are strongly biased in their positions and in their semantic roles (e.g., 'you', 'it'). Despite this high baseline, however, we see the same pattern in the unbiased and biased experiments in table 3.1. The addition of the NPattern features (feature set 3) substantially improves performance on 'A gorps B' test sentences, and promotes over-generalization errors on 'A and B gorp' sentences.

3.4.3 More Complex Training Data

For comparison purposes we also trained the Baby SRL on a subset of the Propbank training data of Wall Street Journal (WSJ) text [Kingsbury and Palmer, 2002]. To approximate the simpler sentences of child-directed speech we selected only those sentences with 8 or fewer words. This provided a training set of about 2500 sentences, most with a single verb and two nouns to be labeled. The CDS and WSJ data pose similar problems for learning abstract and verb-specific knowledge. However, newspaper text differs from casual speech to children in many ways, including vocabulary and sentence complexity. One could argue that the WSJ corpus presents a worst-case scenario for learning based on shallow representations of sentence structure: Full passive sentences are more common in written corpora such as the WSJ than in samples of conversational speech, for example [Roland et al., 2007]. As a result of such differences, two-noun sequences are less likely to display an A0-A1 sequence in the WSJ (0.42 A0-A1 in 2-noun sentences) than in the CDS training data (0.67 A0-A1). The WSJ data provides a more demanding test of the Baby SRL.

We trained the Baby SRL on the WSJ data, and tested it using the biased lexical choices as described above, sampling A and B nouns for novel-verb test sentences based on the distribution of nouns seen as the first of two nouns in training, and as the second of two nouns, respectively. The WSJ training produced performance strikingly similar to the performance resulting from CDS training (last 4 columns of Table 3.1). Even in this more complex training set, the addition of the NPattern features (feature set 3) increases the expected prediction of Agent-Patient on ‘A gorps B’ test sentences, and promotes over-generalization errors on ‘A and B gorp’ sentences. Even on sentences targeted at adults, the number and order of nouns provides a basic structure that can generalize meaning to novel sentences.

3.4.4 Tests with Familiar Verbs

	Total			A0			A1			A2		
Features	Pre	Rec	F1	Pre	Rec	F1	Pre	Rec	F1	Pre	Rec	F1
1. Words	0.73	0.58	0.64	0.79	0.87	0.83	0.69	0.81	0.74	0.57	0.23	0.33
2. NPattern&V	0.77	0.60	0.67	0.82	0.90	0.86	0.75	0.80	0.77	0.57	0.37	0.45
3. NPattern	0.75	0.59	0.66	0.85	0.89	0.87	0.74	0.79	0.76	0.53	0.29	0.37
4. NPattern + NPattern&V	0.78	0.60	0.68	0.86	0.88	0.87	0.78	0.81	0.80	0.56	0.40	0.47
5. + VPosition	0.81	0.62	0.70	0.82	0.94	0.88	0.84	0.81	0.83	0.76	0.37	0.50

Table 3.2: Testing NPattern features on full SRL task of heldout section 8 of Eve when trained on sections 9 through 20. Each result column reflects a per argument precision, recall and F1. Only A0, A1 and A2 individual role predictions are presented, these three labels combine to cover 73.42% of the arguments in the heldout section, the rest of the arguments are spread amongst 7 AM-* labels (AM-MOD, AM-TMP, AM-DIS, AM-LOC, AM-NEG, AM-ADV, AM-MNR).

Learning to interpret sentences depends on balancing abstract and verb-specific structural knowledge. Natural linguistic corpora, including our CDS training data, have few verbs of very high frequency and a long tail of rare verbs. Frequent verbs occur with differing argument patterns. For example, 'have' and 'put' are frequent in the CDS data. 'Have' nearly always occurs in simple transitive sentences that display the canonical word order of English (e.g., 'I have cookies'). 'Put', in contrast, tends to appear in non-canonical sentences that do not display an agent-patient ordering, including imperatives ('Put it on the floor'). To probe the Baby SRL's ability to learn the argument-structure preferences of familiar verbs, we tested it on a held-out sample of CDS from the same source (Eve sample 8, approximately 234 labeled sentences). Table 3.2 shows the same feature progression shown previously, with the full SRL test set. The words-only baseline (feature set 1 in Table 3.2) yields fairly accurate performance, showing that considerable success in role assignment in these simple sentences can be achieved based on the argument-role biases of the target nouns and the familiar verbs. Despite this high baseline, however, we still see the benefit of simple structural features. Adding verb-specific NPattern (feature set 2) leads to small increases overall, and these gains are most noticeable in the more verb specific A2 role. The abstract NPattern features (feature set 3) leads to similar overall classification performance, and the combination of both verb-specific and abstract NPattern features (feature set 4) yields higher performance than either alone. The combination of abstract NPattern features with the verb-specific versions allows the Baby SRL both to generalize to unseen verbs, as seen in earlier sections, and to learn the idiosyncrasies of known verbs.

3.4.5 Verb Position

The noun pattern feature results show that the Baby SRL can learn helpful rules for argument-role assignment using only information about the number and order of nouns. It also makes the error predicted by the structure-mapping account, and documented in children, because it has no way to represent the difference between the ‘A gorps B’ and ‘A and B gorp’ test sentences. When we add verb position information (feature set 5 in table 3.1 and 3.2), performance improves still further for transitive sentences, both with biased and unbiased test sentences. Also, for the first time, the A0A1 pattern is predicted less often for ‘A and B gorp’ sentences. This error diminished because the classifier was able to use the verb position features to distinguish these from ‘A gorps B’ sentences.

Verb position alone provides another simple abstract representation of sentence structure, so it

	Unbiased Lexical			
	A gorps B		A and B gorp	
Features	F1	%A0A1	F1	%A0A1
1. Words	0.59	0.38	0.46	0.38
3. NPattern	0.83	0.65	0.33	0.65
6. VPosition	0.99	0.95	0.97	0.00

Table 3.3: Verb Position vs. Noun Pattern features alone. Verb position features yield better overall performance, but do not replicate the error on ‘A and B gorp’ sentences seen with children.

might be proposed as an equally natural initial representation for human learners, rather than the noun pattern features we proposed. The VPosition features should also support learning and generalization of word-order rules for interpreting transitive sentences, thus reproducing some of the data from children that we reviewed above. In table 3.3 we compared the words-only baseline (set 1), words and NPattern features (set 3), and a new feature set, words and VPosition (set 6). In terms of correct performance on novel transitive verbs (’A gorps B’), the VPosition features out-perform the NPattern features. This may be partly because the same VPosition features are used in all sentences during training, while the NPattern features partition sentences by number of nouns, but is also due to the fact that the verb position features provide a more sophisticated representation of English sentence structure. Verb position features can distinguish transitive sentences from imperatives containing multiple post-verbal nouns, for example. Although verb position is ultimately a more powerful representation of word order for English sentences, it does not accurately reproduce a 21-month-old’s performance on all aspects of this task. In particular, the VPosition feature does not support the overgeneralization of the A0A1 pattern to the ‘A and B gorp’ test sentences with novel verbs. This suggests that children’s very early sentence comprehension is dominated by less sophisticated representations of word order, akin to the NPattern features we proposed, perhaps especially when faced with unfamiliar verbs.

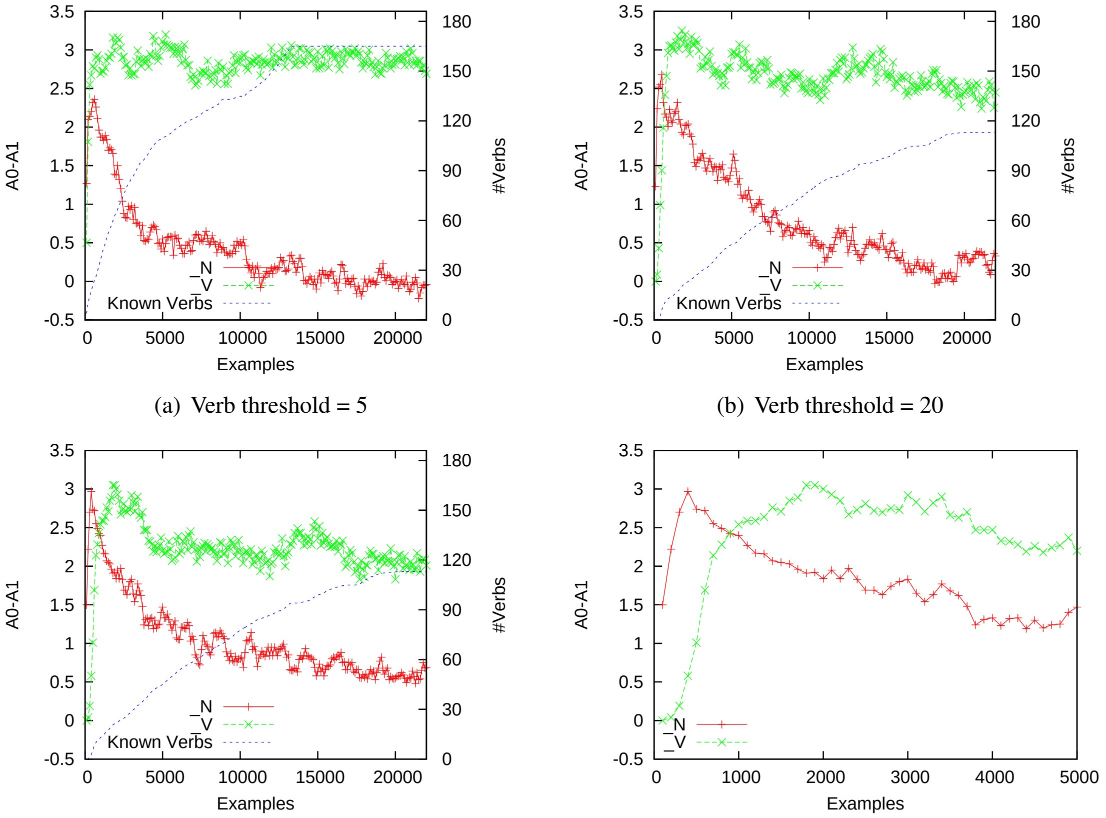
3.4.6 Informativeness vs. Availability

In the preceding sections, we modeled increases in syntactic knowledge by building in more sophisticated features. The Baby SRL escaped the predicted error on two-noun intransitive sentences when given access to features reflecting the position of the target noun relative to the verb. This imposed sequence of features is useful as a starting point, but a more satisfying approach would be to use the Baby SRL to explore possible reasons why NPattern features might dominate early in acquisition, even though VPosition features are ultimately more useful for English.

In theory, a feature might be unavailable early in acquisition because of its computational complexity. For example, lexical features are presumably less complex than relative position features such as NPattern and VPosition. In practice, features can also be unavailable at first because of an informational lack. Here we suggest that NPattern features might dominate VPosition features early in acquisition because the early lexicon is dominated by nouns, and it is easier to compute position relative to a known word than to an unknown word. Many studies have shown that children's early vocabulary is dominated by names for objects and people [Gentner and Boroditsky, 2001].

To test the consequences of this proposed informational bottleneck on the relative weighting of NPattern and VPosition features during training, we modified the Baby SRL's training procedure such that NPattern features were always active, but VPosition features were active during training only when the verb in the current example had been encountered a critical number of times. This represents the assumption that the child can recognize which words in the sentence are nouns, based on lexical familiarity or morphological context [Waxman and Booth, 2001], but is less likely to be able to represent position relative to the verb without knowing the verb well. In future chapters we will explore alternative means of verb identification further.

Figure 3.2 shows the tendency of the NPattern feature ’_N’ (first of two nouns) and the VPosition feature ’_V’ (pre-verbal noun) to predict the role A0 as opposed to A1 as the difference between the weights of these connections in the learned network. Figure 3.2(a) shows the results when VPosition features were active whenever the target verb had occurred at least 5 times; in Figure 3.2(b) the threshold for verb familiarity was 20. In both figures we see that the VPosition features win out over the NPattern features as the verb vocabulary grows. Varying the degree of verb familiarity required to accurately represent VPosition features affects how quickly the VPosition features win out (compare Figures 3.2(a) and 3.2(b)). Figure 3.2(c) shows the same analysis with a threshold of 20, but with verb-specific as well as abstract versions of the NPattern and the VPosition features. In this procedure, every example started with three features: target noun, target predicate, NPattern, and if the verb was known, added NPattern&V, VPosition, and VPosition&V. Comparing Figures 3.2(b) and 3.2(c), we see that the addition of verb-specific versions of the structural features also affects the rate at which the VPosition features come to dominate the NPattern features.



(a) Verb threshold = 5

(b) Verb threshold = 20

(c) Verb threshold = 20, +verb-specific features

(d) Threshold = 20, +verb-specific, Zoomed

Figure 3.2: Testing the consequences of the assumption that Verb Position features are only active for familiar verbs. The figure plots the bias of the features 'first of two nouns' ('_N') and 'before the verb' ('_V') to predict A0 over A1, as the difference between the weights of these connections in the learned network. Verb position features win out over noun pattern features as the verb vocabulary grows. Varying the verb familiarity threshold ((a) vs. (b)) and the presence versus absence of verb-specific versions of the structural features ((b) vs. (c)) affects how quickly the verb position features become dominant. When verbs are learned slowly (threshold set at 20, so must see a verb 20 times before recognizing it as a verb), and verb specific features are included during learning (subfigure (c) and (d)), the noun pattern feature initially provides a stronger bias for predicting agent first sentences until enough verbs have been recognized to prime the verb position feature, after about 1000 examples.

Thus, in training the VPosition features become dominant as the SRL learns to recognize more verbs. However, the VPosition features are inactive when the Baby SRL encounters the novel-verb test sentences. Since the NPattern features are active in test, the system generates the predicted error until the bias of the NPattern features reaches 0. Note in figure 3.2(c) that when verb-specific structural features were added, the Baby SRL never learned to entirely discount the NPattern features within the range of training provided. This result is reminiscent of suggestions in the psycholin-

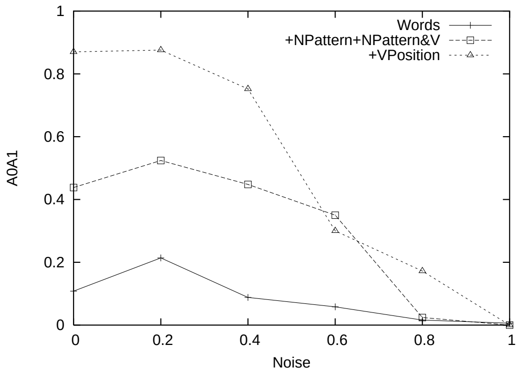


Figure 3.3: Testing the ability of simple features to cope with varying amounts of noisy feedback. Even with noisy feedback, the noun pattern features support learning and generalization to new verbs of a simple agent-patient template for understanding transitive sentences. These results are lower than those found in table 3.1 due to slightly different training assumptions.

guistics literature that shallow representations of syntax persist in the adult parser, alongside more sophisticated representations (e.g., [Ferreira, 2003]).

3.4.7 Noisy Training

So far, the Baby SRL has only been trained with perfect feedback. Theories of human language acquisition assume that learning to understand sentences is naturally a partially-supervised task: the child uses existing knowledge of words and syntax to assign a meaning to a sentence; the appropriateness of this meaning for the referential context provides the feedback (e.g., [Pinker, 1989]). But this feedback must be noisy. Referential scenes provide useful but often ambiguous information about the semantic roles of sentence participants. For example, a participant could be construed as an agent of fleeing or as a patient being chased. In a final set of experiments, we examined the generalization abilities of the Baby SRL as a function of the integrity of semantic feedback.

We provided noisy semantic-role feedback during training by giving a randomly-selected argument label on 0 to 100% of examples. Following this training, we tested with the 'A gorps B' test sentences, using the unbiased noun choices.

As shown in Figure 3.3, feature sets including NPattern or VPosition features yield reasonable performance on the novel verb test sentences up to 50% noise, and promote an A0-A1 sequence over the words-only baseline even at higher noise levels. Thus the proposed simple structural features are robust to noisy feedback.

3.5 Conclusion

The simplified SRL classifier mimicked experimental results with toddlers. We structured the learning task to ask whether shallow representations of sentence structure provided a useful initial representation for learning to interpret sentences. Given representations of the number and order of nouns in the sentence (noun pattern features), the Baby SRL learned to classify the first of two nouns as an agent and the second as a patient. When provided with both verb-general and verb-specific noun pattern features, the Baby SRL learned to balance verb-specific and abstract syntactic knowledge. By treating each noun as an argument, it also reproduced the errors children make. Crucially, verb-position features improved performance when added to the noun-pattern feature, but when presented alone failed to produce the error found with toddlers.

In this chapter we have introduced the basic Baby SRL training and testing methodologies in the simplest fully supervised learning setting. When both true structure (nouns and verbs) and correct semantic feedback is provided to the learner the simple representations of noun pattern and verb position provide an effective starting structure that can accurately generalize beyond lexical patterns. Furthermore these representations were able to handle the introduction of noise both in terms of missing or random semantic feedback and missing verb identification. In the next chapter we will explore both of these issues further with more plausible methods of identifying nouns and verbs, and of providing minimal semantic feedback.

Chapter 4

Baby SRL Noisy Training

The previous chapter's experiments with BabySRL showed that it is possible to learn to assign basic semantic roles based on a shallow sentence representations proposed by the structure-mapping view. Once nouns have been identified a learner can immediatly begin using a representation of sentence structure as simple as 'first of two nouns' to begin learning general patterns in sentence semantics. While this result is useful in teasing out statistics from the corpus of child directed speech, it may not be relevant to real world language acquisition because it relies on perfect knowledge of both noun and verb identification and accurate semantic feedback. Our current BabySRL is modeling not a child hearing unknown sentences in an ambiguous environment, but a child who is given a (shallow) parsed sentence while also being able to read the mind of the speaker.

In this chapter we explore both of these avenues of information that feed into the child learner, removing the dependence on explicit external supervision to form an understanding of structure and semantics while retaining the same core BabySRL role classifier. In section 4.1 we create a minimally supervised argument and predicate identifier that uses a small set of concrete nouns (that children plausibly already recognize) to seed identification of noun clusters, and then uses statistics of noun cooccurence to recognize argument taking (and thus likely verb) clusters. Even with this potentially noisy structure, the simple representation is still able to learn and generalize. In section 4.2 we replace the full semantic feedback (imagined to come from complete understanding of external scene) with internally generated feedback based on background knowledge. Using both expectations based on animacy of identified arguments as well as structural constraints, the animacy trained BabySRL still extracts useful patterns.

In section 4.3 we combine these two approaches to create a complete minimally supervised SRL pipeline that begins with a small set of nouns, unlabeled text, and knowledge of animacy for some set of nouns to 1) identify noun category, 2) use noun category to identify verbs, 3) use nouns and verbs to form structure of sentence and classify semantic roles of the nouns 4) using semantic feedback from background knowledge only. Where the learner from the previous chapter represents an upper bound of information available to the child, the system introduced here represents a lower bound, yet is still able to begin classifying novel multiword sentences given these assumptions.

4.1 Minimally Supervised Argument Identification

Perfect built-in parsing finesses two problems facing the human learner. The first problem involves classifying words by part-of-speech. Proposed solutions to this problem in the NLP and human language acquisition literatures focus on distributional learning as a key data source (e.g., [Mintz, 2003; Johnson, 2007]). Importantly, infants are good at learning distributional patterns [Gomez and Gerken, 1999; Saffran et al., 1996]. Here we use a fairly standard Hidden Markov Model (HMM) to generate clusters of words that occur in similar distributional contexts in a corpus of input sentences.

The second problem facing the learner is more contentious: Having identified clusters of distributionally similar words, how do children figure out what role these clusters of words should play in a sentence interpretation system? Some clusters contain nouns, which are candidate arguments; others contain verbs, which take arguments. How is the child to know which are which? In order to use the output of the HMM tagger to process sentences for input to an SRL model, we must find a way to automatically label the clusters.

Our strategies for automatic argument and predicate identification, spelled out below, reflect core claims of the structure-mapping theory: (1) The meanings of some concrete nouns can be learned without prior linguistic knowledge; these concrete nouns are assumed based on their meanings to be possible arguments; (2) verbs are identified, not primarily by learning their meanings via observation, but rather by learning about their syntactic argument-taking behavior in sentences.

By using the HMM part-of-speech tagger in this way, we can ask how the simple structural features that we propose children start with stand up to reductions in parsing accuracy. In doing so, we move to a parser derived from a particular theoretical account of how the human learner might classify words, and link them into a system for sentence comprehension.

Much of this section originally appeared in [Connor et al., 2010].

4.1.1 Model

As in the previous chapter, we model language learning as a Semantic Role Labeling (SRL) task [Carreras and Màrquez, 2004]. This allows us to ask whether a learner, equipped with particular theoretically-motivated representations of the input, can learn to understand sentences at the level of who did what to whom. We refer to this simplified SRL system as BabySRL.

BabySRL follows a conventional multi-stage pipeline where the stages are: (1) Parsing the sentence, (2) Identifying potential predicates and arguments based on the parse, (3) Classifying role labels for each potential argument relative to a predicate, (4) Applying constraints to find the best labeling of arguments for a sentence. In this section we focus on the first and second stages, attempting to limit the knowledge available to what we argue is available to children in the early stages of language learning: knowledge of a small number of concrete nouns and the exposure to a large amount of language.

We develop here a Minimally Supervised Argument Identification version of BabySRL, where the parsing stage uses an unsupervised parser based on Hidden Markov Models (HMM), modeling a simple 'predict the next word' parser. Next the argument identification stage identifies HMM states that correspond to possible arguments based on a seed set of concrete nouns (the minimal supervision), and these argument states are used to identify verb states. The candidate arguments and predicates identified in each input sentence are passed to an SRL classifier that uses simple abstract features based on the number and order of arguments to learn to assign semantic roles.

As input to our learner we use samples of child directed speech (CDS) from the CHILDES corpora [MacWhinney, 2000]. During initial unsupervised parsing we experiment with incorporating knowledge through a combination of statistical priors favoring a skewed distribution of words into classes, and an initial hard clustering of the vocabulary into function and content words. The argument identifier uses a small set of frequent nouns to seed argument states, relying on the assumptions that some concrete nouns can be learned as a prerequisite to sentence interpretation, and are interpreted as candidate arguments.

The SRL classifier starts with noisy largely unsupervised argument identification, and receives feedback based on annotation in the PropBank style; in training, each word identified as an argument receives the true role label of the phrase that word is part of. This represents the assumption that learning to interpret sentences is naturally supervised by the fit of the learner's predicted meaning with the referential context. The provision of perfect 'gold-standard' feedback over-estimates the real child's access to this supervision, but allows us to investigate the consequences of noisy argument identification for SRL performance. We show that even with imperfect parsing, a learner can identify useful abstract patterns for sentence interpretation. In section 4.2 we experiment with a different method of providing semantic feedback, and in section 4.3 we combine this feedback with the minimally supervised arguments to form a complete end to end BabySRL systme.

4.1.2 Unsupervised Parsing

As a first step of processing, we feed the learner large amounts of unlabeled text and expect it to learn some structure over this data that will facilitate future processing. This stage represents the assumption that the child is naturally exposed to and surrounded by large amounts of language, and even without understanding every utterance they will begin to determine statistics over their input. One caveat to our method is that we use transcripts of child directed speech, so we are assuming that the learner is able to correctly segment speech into words. The source of this text is child directed speech collected from various projects in the CHILDES repository[1]. As an attempt to use only complete sentences from parents to children, we removed utterances with fewer than three words or markers of disfluency. In the end we used 320 thousand sentences from this set, totaling over 2 million tokens and 17 thousand unique words. Note that this set does cover the semantically tagged training data (Adam, Eve and Sarah corpus we use for semantic training and testing).

The goal of the parsing stage is to give the learner a representation permitting it to generalize over word forms. The exact parse we are after is a distributional and context-sensitive clustering of words based on sequential processing. We chose an HMM based parser for this since, in essence the HMM yields an unsupervised POS classifier, but without names for states. An HMM trained with expectation maximization (EM) is analogous to a simple process of predicting the next word in a stream and correcting connections accordingly for each sentence.

We train the HMM in an offline, batch process. While the rest of our training procedure uses

[1] We used parts of the Bloom [Bloom, 1970, 1973], Brent [Brent and Siskind, 2001], Brown [Brown, 1973], Clark [Clark, 1978], Cornell, MacWhinney [MacWhinney, 2000], Post [Demetras et al., 1986] and Providence [Demuth et al., 2006] collections.

online training, we envision this unsupervised HMM process to be reminiscent of an earlier stage of processing whereby a child learns the statistics of a language merely through exposure to large quantities of that language (without requiring further interpretation). In general this can be done alongside our higher semantic training, and it may make sense to use feedback from higher semantics to partially drive and be incorporated into the low level lexical (and lower) statisical model, but for these initial experiments we treat this stage as having already happened in the learner.

With HMM we can also easily incorporate additional knowledge during parameter estimation. The first (and simplest) parser we used was an HMM trained using EM with 80 hidden states. The number of hidden states was made relatively large to increase the likelihood of clusters corresponding to a single part of speech, while preserving some degree of generalization. Other researchers [Huang and Yates, 2009] have also found 80 states to be an effective point for creating a representation to generalize features (using the states to represent out of vocab words, or otherwise help domain adaptation), trading off complexity of training with specifity.

Johnson [Johnson, 2007] observed that EM tends to create word clusters of uniform size, which does not reflect the way words cluster into parts of speech in natural languages. The addition of priors biasing the system toward a skewed allocation of words to classes can help. The second parser was an 80 state HMM trained with Variational Bayes EM (VB) incorporating Dirichlet priors [Beal, 2003].[2]

In the third and fourth parsers we experiment with enriching the HMM POS-tagger with other psycholinguistically plausible knowledge. Words of different grammatical categories differ in their phonological as well as in their distributional properties (e.g., [Kelly, 1992; Monaghan et al., 2005; Shi et al., 1998]); combining phonological and distributional information improves the clustering of words into grammatical categories. The phonological difference between content and function words is particularly striking [Shi et al., 1998]. Even newborns can categorically distinguish content and function words, based on the phonological difference between the two classes [Shi et al., 1999], and infants can use both phonology and frequency to recognize novel content words [Hochmann et al., 2010]. Human learners may treat content and function words as distinct classes from the start.

[2] We tuned the prior using the same set of 8 value pairs suggested by Gao and Johnson [Gao and Johnson, 2008], using a held out set of POS-tagged CDS to evaluate final performance. Our final values are an emission prior of 0.1 and a transitions prior of 0.0001; as dirichlet prior approaches 0 the resulting multinomial becomes peakier with most of the probability mass concentrated in a few points.

To implement this division into function and content words[3], we start with a list of function word POS tags[4] and then find words that appear predominantly with these POS tags, using tagged WSJ data [Marcus et al., 1993]. We allocated a fixed number of states for these function words, and left the rest of the states for the content of the words. This amounts to initializing the emission matrix for the HMM with a block structure; words from one class cannot be emitted by states allocated to other classes. We selected the exact allocation of states through tuning the argument and predicate identification on a held out set of CDS, settling on 5 states for punctuation, 30 states for function words, and 45 content word states. This trick has been used before in speech recognition work [Rabiner, 1989], and requires far fewer resources than the full tagging dictionary that is often used to intelligently initialize an unsupervised POS classifier (e.g. [Brill, 1997; Toutanova and Johnson, 2007; Ravi and Knight, 2009]).

Because the function and content word preclustering preceded parameter estimation, it can be combined with either EM or VB learning. Although this initial split forces sparsity on the emission matrix and allows more uniform sized clusters, Dirichlet priors may still help, if word clusters within the function or content word subsets vary in size and frequency. The third parser was an 80 state HMM trained with EM estimation, with 30 states pre-allocated to function words; the fourth parser was the same except that it was trained with VB EM.

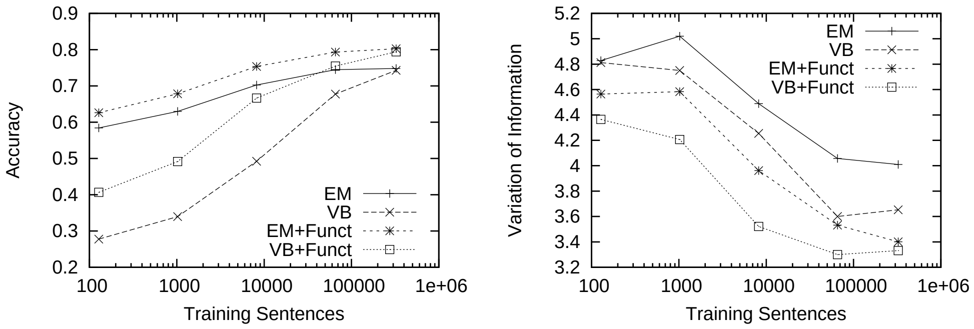
Parser Evaluation

We first evaluate these parsers (the first stage of our SRL system) on unsupervised POS tagging. Figure 4.1 shows the performance of the four systems using both many to one accuracy and variation of information to measure match between fine grained POS and unsupervised parsers as we vary the amount of text they train on. Each point on the graph represents the average result over 10 runs of the HMM with different samples of the unlabeled CDS.

Many to one accuracy is used when there are more states than POS tags, and accuracy is measured by greedily mapping each state to the POS tag it most frequently occurs with in the test data; all other occurences of that state are then considered incorrect. It is known that EM gives a better many to one score than VB trained HMM [Johnson, 2007], and likewise we see that here: with all

[3] We also include a small third class for punctuation, which is discarded.

[4] TO,IN,EX,POS,WDT,PDT,WRB,MD,CC,DT,RP,UH



(a) Many to 1 Accuracy

(b) Variation of Information

Figure 4.1: Unsupervised Part of Speech results, matching states to gold POS labels. All systems use 80 states, and comparison is to POS labeled CDS text (a combined set of all three children's training sections: adam01-20, eve01-18, sarah001-083), which makes up a subset of the HMM training data. The gold POS are according to WSJ Treebank POS, with 44 unique POS appearing in this data, including punctuation. Many to 1 matching accuracy greedily matches states to their most freqent part of speech (figure 4.1(a), higher is better). Variation of Information is an information-theoretic measure summing mutual information between tags and states, proposed by [Meilă, 2002], and first used for Unsupervised Part of Speech in [Goldwater and Griffiths, 2007]. Smaller numbers are better, indicating less information lost in moving from the HMM states to the gold POS tags. Note that incorporating function word preclustering allows both EM and VB algorithms to achieve the same performance with an order of magnitude fewer sentences.

data EM gives 0.75 matching, VB gives 0.74, while both EM+Funct and VB+Funct reach 0.80.

Variation of information is an distance metric between two clusters (true POS labels and HMM states) which measures the loss and gain of information when moving from one clustering to the other. It is defined as $VI(C_1, C_2) = H(C_1|C_2) + H(C_2|C_1) = H(C_1) + H(C_2) - 2 * I(C_1, C_2)$, where $H(C)$ is the entropy of the clustering assignment C and $I(C_1, C_2)$ is the mutual information between the clustering C_1 and C_2. VI is a valid metric, and thus if two clusterings are identical, their VI will be 0. With this measure we see that adding the function word split always improves, for both EM and VB training, indicating that it is adding helpful information regarding the true POS distribution.

Adding the function/content word split to the HMM structure improves both EM and VB estimation in terms of both tag matching accuracy and information. However, these measures look at the parser only in isolation. What is more important to us is how useful the provided word clusters are for future semantic processing. In the next sections we use the outputs of our four parsers to identify arguments and predicates.

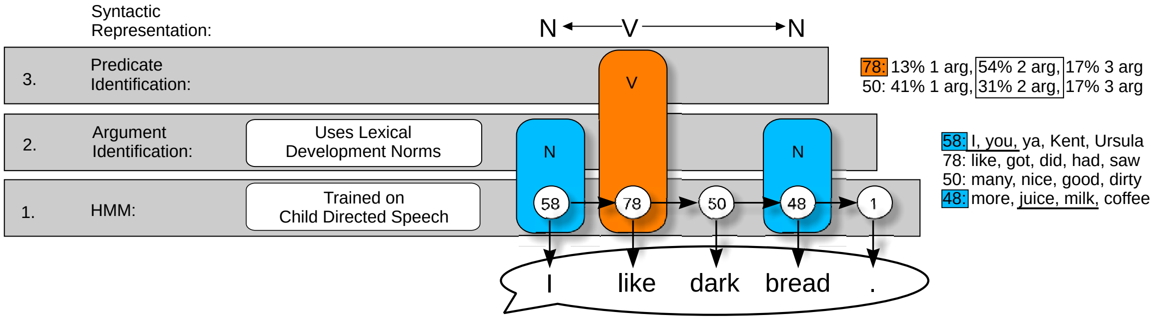

Syntactic Representation:
N
V
N
3.
Predicate Identification:
V
78: 13% 1 arg, 54% 2 arg, 17% 3 arg
50: 41% 1 arg, 31% 2 arg, 17% 3 arg
2.
Argument Identification:
Uses Lexical Development Norms
N
N
58: I, you, ya, Kent, Ursula
78: like, got, did, had, saw
50: many, nice, good, dirty
48: more, juice, milk, coffee
1.
HMM:
Trained on Child Directed Speech
58
78
50
48
1
I
like
dark
bread
.

Figure 4.2: Minimally supervised argument and predicate identification for example sentence "I like dark bread ." (1) Hidden states are found for each word according to a Hidden Markov Model (HMM), trained on a large sample of child directed speech. (2) Nouns are identified based on those states that tend to appear with a small set of known concrete nouns (from lexical development norms). In this example state 58 tends to appear with known person nouns such as pronouns 'I' and 'you', and state 48 appears with known nouns 'juice', and 'milk', so 'bread' (which is not in the lexical development norm) is also assigned as an argument. (3) With the two arguments identified, the predicate is identified as whichever remaining content word is more likely to appear with that number of arguments. The words in state 78, such as 'like' seen here, frequently appear in two argument sentences, indicating this may be a set of two-argument predicates.

4.1.3 Argument Identification

The unsupervised parser provides a state label for each word in each sentence; the goal of the argument identification stage is to use these states to label words as potential arguments, predicates or neither. As described in the introduction, core premises of the structure-mapping account offer routes whereby we could label some HMM states as argument or predicate states.

The structure-mapping account holds that sentence comprehension is grounded in the learning of an initial set of nouns. Children are assumed to identify the referents of some concrete nouns via cross-situational learning [Gillette et al., 1999; Smith and Yu, 2008]. Children then assume, by virtue of the meanings of these nouns, that they are candidate arguments. This is a simple form of semantic bootstrapping, requiring the use of built-in links between semantics and syntax to identify the grammatical type of known words [Pinker, 1984]. We use a small set of known nouns to transform unlabeled word clusters into candidate arguments for the SRL: HMM states that are dominated by known names for animate or inanimate objects are assumed to be argument states.

Given text parsed by the HMM parser and a list of known nouns, the argument identifier proceeds in multiple steps as illustrated in figure 4.3. The first stage identifies as argument states those states that appear at least half the time in the training data with known nouns. This use of a seed list

```
Algorithm ARGUMENT STATE IDENTIFICATION
    INPUT: Parsed Text $T$ = list of (word, state) pairs
        Set of concrete nouns $N$
    OUTPUT: Set of argument states $A$
        Argument count likelihood $ArgLike(s, c)$

    Identify Argument States
    Let $freq(s) = |\{(*, s) \in T\}|$
    Let $freq_N(s) = |\{(w, s) \in T | w \in N\}|$

    For each $s$:
      If $freq_N(s) \geq 4$
        Add $s$ to $A$

    Collect Per Sentence Argument Count statistics
    For each Sentence $S \in T$:
      Let $Arg(S) = |\{(w, s) \in S | s \in A\}|$
      For $(w, s) \in S$ s.t. $s \notin A$
        Increment $ArgCount(s, Arg(S))$

    For each $s \notin A$, and argument count $c$:
      $ArgLike(s, c) = ArgCount(s, c)/freq(s)$
```

(a) Argument Identification

```
Algorithm PREDICATE STATE IDENTIFICATION
    INPUT: Parsed Sentence $S$ = list of (word, state) pairs
        Set of argument states $A$
        Sentence Argument Count $ArgLike(s, c)$
    OUTPUT: Most likely predicate $(v, s_v)$

    Find Number of arguments in sentence
    Let $Arg(S) = |\{(w, s) \in S | s \in A\}|$

    Find Non-argument state in sentence most likely
     to appear with this number of arguments
    $(v, s_v) = argmax_{(w,s) \in S} ArgLike(s, Arg(S))$
```

(b) Predicate Identification

Figure 4.3: Argument identification algorithm. This is a two stage process: argument state identification based on statistics collected over entire text and per sentence predicate identification.

and distributional clustering is similar to Prototype Driven Learning [Haghighi and Klein, 2006], except we are only providing information on one specific class.

As our seed set of plausible concrete nouns we wanted a set of nouns that young children know, should recognize, and that appear in our training data. We used lexical development norms [Dale and Fenson, 1996], selecting all words for things or people that were commonly produced by 20-month-olds (over 50% reported), and that appeared at least 5 times in our training data. Because this list is of words that children produce, it obviously represents a lower bound on the set of words that such a child should comprehend or recognize. This yielded 71 words, including words for common animals (‘pig’, ‘kitty’, ‘puppy’), objects (‘truck’, ‘banana’, ‘telephone’), people (‘mommy’, ‘daddy’), and some pronouns (‘me’ and ‘mine’). To this set we also added pronouns ‘you’ and ‘I’, as well as given names ‘adam’, ’eve’ and ‘sarah’. Pronouns refer to people or objects, but are abstract in that they can refer to any person or object. The inclusion of pronouns in our list of known nouns represents the assumption that toddlers have already identified pronouns as referential terms. Even 19-month-olds assign appropriately different interpretations to novel verbs presented in simple transitive versus intransitive sentences with pronoun arguments (“He’s kradding him!” vs. “He’s kradding!”; [Yuan et al., 2007]). See appendix A for the full list of nouns used as seed set.

The lexical development norm words represent a high precision set of argument nouns, they are not highly frequent in the data (outside of added pronouns), but they nearly always appear as nouns and arguments in the data (over 99% of occurences of words in this list are considered nouns or pronouns in the training data, over 97% are part of arguments). As such we set a very permissive condition that identifies argument states as those HMM states that appear above some frequency with known nouns. Thus the high precision of the seed set is shared with all other words that appear in states that word appears with. In our experiments we set the threshold of known nouns to 4 through tuning argument identifier.

Figure 4.2 acts as a companion to figure 3.1, illustrating how the minimal supervised argument identification can find the syntactic structure used in the previous example. Starting with the sentence “I like dark bread”, first the HMM is used to find the best state for each word in the sentence[5]. Then argument states are identified using the lexical development norm seed set. In this example, the set of known concrete nouns does not contain the word ‘bread’, but does contain words such as ‘juice’ and ‘milk’, which appear with the same HMM state as ‘bread’ in our example sentence, so ‘bread’ is also considered a noun.

What about verbs? A typical SRL model identifies candidate arguments and tries to assign roles to them relative to each verb in the sentence. In principle one might suppose that children learn the meanings of verbs via cross-situational observation just as they learn the meanings of concrete nouns. But identifying the meanings of verbs is much more troublesome. Verbs’ meanings are

[5] We use the state with the highest marginal probability for each word given the sentence, instead of the Viterbi state estimate which finds the sequence of states that have the highest probability for an entire sentence. In previous experiments this has proven to give slightly better results for unsupervised POS.

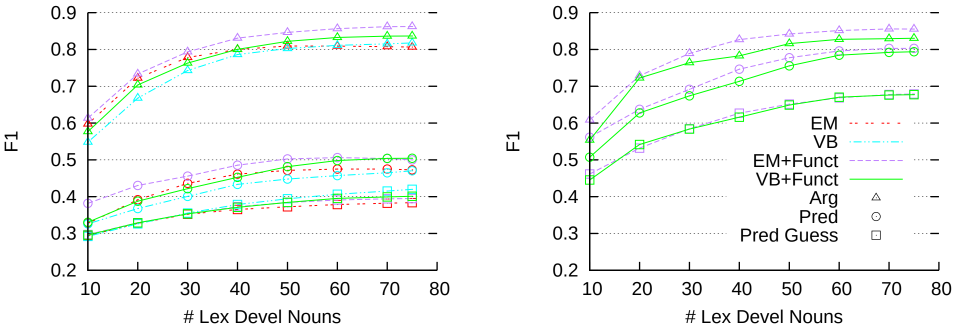
abstract, therefore harder to identify based on scene information alone [Gillette et al., 1999]. As a result, early vocabularies are dominated by nouns [Gentner, 2006]. On the structure-mapping account, learners identify verbs, and begin to determine their meanings, based on sentence structure cues. Verbs take noun arguments; thus, learners could learn which words are verbs by detecting each verb's syntactic argument-taking behavior. Experimental evidence provides some support for this procedure: 2-year-olds keep track of the syntactic structures in which a new verb appears, even without a concurrent scene that provides cues to the verb's semantic content [Yuan and Fisher, 2009].

We implement this behavior by identifying as predicate states the HMM states that appear commonly with a particular number of previously identified arguments. First, we collect statistics over the entire HMM training corpus regarding how many arguments are identified per sentence, and which states that are not identified as argument states appear with each number of arguments. Next, for each parsed sentence that serves as SRL input, the algorithm chooses as the most likely predicate the word whose state is most likely to appear with the number of arguments found in the current input sentence. Note that this algorithm assumes exactly one predicate per sentence. Implicitly, the argument count likelihood divides predicate states up into transitive and intransitive predicates based on appearances in the simple sentences of CDS.

Two arguments are identified in figure 4.2, so predicate identification proceeds by selecting determing which of the non-argument HMM states that appear in the sentence is most likely to appear with two arguments. Deciding between 'like' and 'dark', we see that state 78, which 'like' is assigned to, frequently appears in two argument sentences, indicating that these words may be two argument predicates, or transitive verbs. Looking at the most frequent words that appear with this state, 'like', 'got', 'had', 'saw', these do appear to correctly be verbs that take two arguments, which we were able to identify through distributional similarity and argument count statistics.

Argument Identification Evaluation

Figure 4.4 shows argument and predicate identification accuracy for each of the four parsers when provided with different numbers of known nouns. For each HMM we train 10 models over the training data with different random initializations and take the one with the highest perplexity for



(a) Consider Function and Content Words (b) Distinguish Function Words

Figure 4.4: Effect of number of concrete nouns for seeding argument identification with various unsupervised parsers. Each line represents the mean of 100 runs with a single HMM (which acheived lowest perplexity on training data over 10 runs) and random selection of seed nouns. For each parser (different color and line style) there are three lines indicating: Argument identification (marked with triangle), Predicate identification (circle) and Predicate guessing baseline (square). Argument identification accuracy is computed against true argument boundaries from hand labeled data. The left graph (subfigure (a)) does not distinguish between function and content word states when identifying argument and predicate states while the right graph does include this distinction with the two parsers that support such a seperation. Eliminating function words from consideration when identifying arguments and predicates does not impact argument identification, but greatly helps predicate identification.

experiments. In these graphs, for each number of seed nouns we take the mean over 100 runs of argument and predicate identification with a random selection of seed nouns each time. Because for each of the four unsupervised parsers we use the same HMM over each run, comparison may not be complete reflection of different HMM training approaches.

Three groups of curves appear in figure 4.4: the upper group (marked with triangles) shows the primary argument identification accuracy, the middle group (circle) shows the predicate identification accuracy, and bottom group (squares) shows the lower bound random predicate baseline. We evaluate compared to gold tagged data with true argument and predicate boundaries. The primary argument (A0-4) identification accuracy is the F1 (harmonic mean of precision and recall), with precision calculated as the proportion of identified arguments that appear as part of a true argument, and recall as the proportion of true arguments that have some state identified as an argument. This is a rather lenient measure of accuracy since we are comparing identified individual words to full phrase boundaries.

F1 is calculated similarly for predicate identification, as one state per sentence is identified as the predicate. The predicate guess baseline reflects the expected accuracy if we randomly select a non-argument word to be predicate. As argument identification improves and more arguments are correctly identified, more non-predicates are eliminated from consideration in each sentence and thus predicate guess icreases.

As shown in figure 4.4, argument identification F1 is higher than predicate identification (which is to be expected, given that predicate identification depends on accurate arguments), and as we add more seed nouns the argument identification improves. Surprisingly, despite the clear differences in unsupervised POS performance seen in figure 4.1, the different parsers do not yield very different argument and predicate identification. As we will see in the next section, however, when the arguments identified in this step are used to train SRL classifier, distinctions between parsers reappear, suggesting that argument identification F1 masks systematic patterns in the errors.

An important factor for predicate identification is the contribution of the content/function word split. In the cases of where the HMM contains this division (EM+Funct and VB+Funct) we can make use of this information to only consider content word states for both argument and predicate identification. The two subfigures in figure 4.4 compare the argument and predicate identification performance when we do not eliminate function words (because either the HMM does not contain this division, or else we ignore it) and when we consider only content words. Predicate identification improves dramatically between the two, both because there are naturally fewer options (as can be seen by increase of the guessing baseline), and because the function words are no longer confused with verbs based on their appearing frequently with verbs in the same argument number sentences (such as ‘to’ or ‘and’).

In our previous minimally supervised argument-identification experiments [Connor et al., 2010], we did not find such an improvement of predicate identification over a guessing baseline even as argument identification improved. After this result we did error analysis (on training data) and acheived the current result through three improvements: 1) Evaluate predicate identification on a sentence by sentence basis, instead of per proposition as done for SRL evaluation. Because the same number of arguments are identified per proposition or per sentence, the same predicate would be identified in all cases, meaning we force some entries to be incorrect in sentences with multiple predicates (15% of sentences). 2) Include pronouns and names in lexical development set, improving argument identification and eliminating these words from confusion with predicate, and

3) Distinguish between function and content words when making argument and predicate decisions.

With this minimally supervised argument and predicate identification system we can successfully identify arguments *and* predicates starting with a handful of concrete nouns. With just 10 nouns argument identification is around 0.6 F1, and already this argument information is enough such that predicate identification is doing better than random guessing. With 75 nouns (which is still small relative to the number of nouns children should have acquired by this stage), argument identification improves to over 0.8 F1, and predicate identification continues to improve (especially so when only content words are considered). Partial knowledge of arguments (which can come from knowing a handful of nouns) is enough to identify predicates in a majority of sentences, without knowing anything about the individual verbs other than how often they appear with some number of arguments.

4.1.4 Testing SRL Performance

Note that the results in this section use the original argument identifier that uses a larger set of seed nouns and the poorer predicate identification.

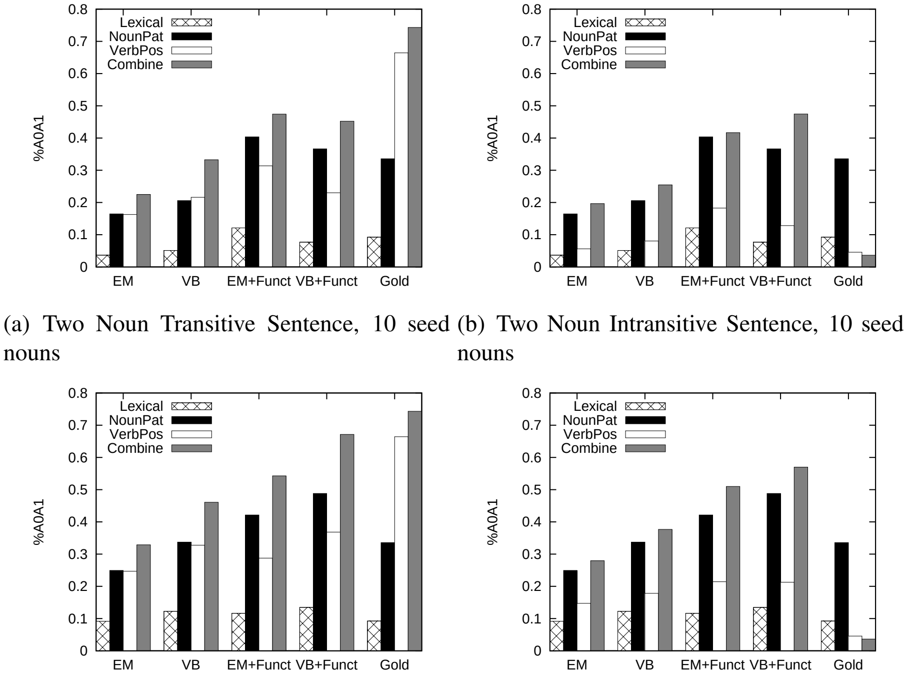
We used the results of the previous parsing and argument-identification stages in training our simplified BabySRL classifier equipped with sets of features derived from the structure-mapping account. In what follows, we compare the performance of the BabySRL across the four parsers. We evaluated SRL performance by testing the BabySRL with constructed sentences like those described in section 3.3. All test sentences contained a novel verb, to test the model's ability to generalize. The goal of these experiments is to see how the simple features that form the base of BabySRL (noun pattern and verb position) deal with the noisy minimally supervised argument and predicate identification. Are the features robust in an environment where a learner only has access to minimal information in making structure predictions?

To test the system's predictions on transitive and intransitive two noun sentences, we constructed two test sentence templates: 'A krads B' and 'A and B krad', where A and B were replaced with familiar animate nouns. The animate nouns were selected from all three children's data in the training set and paired together in the templates such that all pairs are represented.

Figure 4.5 shows SRL performance on test sentences containing a novel verb and two animate

	Two Noun Transitive, % Agent First				One Noun Intransitive, % Agent Prediction			
	Lexical	NPat	VPos	Combine	Lexical	NPat	VPos	Combine
VB+Funct 10 seed	0.48	0.61	0.55	0.71	0.48	0.57	0.56	0.59
VB+Funct 365 seed	0.22	0.64	0.41	0.74	0.23	0.33	0.43	0.41
Gold Arguments	0.16	0.41	0.69	0.77	0.17	0.18	0.70	0.58

Table 4.1: SRL result comparison when trained with best unsupervised argument identifier versus trained with gold arguments. Comparison is between agent first prediction of two noun transitive sentences vs. one noun intransitive sentences. The unsupervised arguments lead the classifier to rely more on noun pattern features; when the true arguments and predicate are known the verb position feature leads the classifier to strongly indicate agent first in both settings.



(a) Two Noun Transitive Sentence, 10 seed nouns

(b) Two Noun Intransitive Sentence, 10 seed nouns

(c) Two Noun Transitive Sentence, 365 seed nouns

(d) Two Noun Intransitive Sentence, 365 seed nouns

Figure 4.5: SRL classification performance on transitive and intransitive test sentences containing two nouns and a novel verb. Performance with gold-standard argument identification is included for comparison. Across parses, noun pattern features promote agent-patient (A0A1) interpretations of both transitive ("You krad Mommy") and two-noun intransitive sentences ("You and Mommy krad"); the latter is an error found in young children. Unsupervised parsing is less accurate in identifying the verb, so verb position features fail to eliminate errors with two-noun intransitive sentences.

nouns. Each plot shows the proportion of test sentences that were assigned an agent-patient (A0-A1) role sequence; this sequence is correct for transitive sentences but is an error for two-noun intransitive sentences. Each group of bars shows the performance of the BabySRL trained using one of the four parsers, equipped with each of our four feature sets. The top and bottom panels in Figure 4.5 differ in the number of nouns provided to seed the argument identification stage. The top row shows performance with 10 seed nouns (the 10 most frequent nouns, mostly animate pronouns), and the bottom row shows performance with 365 concrete (animate or inanimate) nouns treated as known. Relative to the lexical baseline, NPat features fared well: they promoted the assignment of A0-A1 interpretations to transitive sentences, across all parser versions and both sets of known nouns. Both VB estimation and the content-function word split increased the ability of NPat features to learn that the first of two nouns was an agent, and the second a patient. The NPat features also promote the predicted error with two-noun intransitive sentences (Figures 4.5(b), 4.5(d)). Despite the relatively low accuracy of predicate identification noted in section 4.1.3, the VPos features did succeed in promoting an A0A1 interpretation for transitive sentences containing novel verbs relative to the lexical baseline. In every case the performance of the Combined model that includes both NPat and VPos features exceeds the performance of either NPat or VPos alone, suggesting both contribute to correct predictions for transitive sentences. However, the performance of VPos features did not improve with parsing accuracy as did the performance of the NPat features. Most strikingly, the VPos features did not eliminate the predicted error with two-noun intransitive sentences when the NPat feature is also present. As shown in panels 4.5(b) and 4.5(d), the Combined model predicted an A0A1 sequence for these sentences, showing no reduction in this error due to the participation of VPos features.

Table 4.1 shows SRL performance on the same transitive test sentences (‘A krads B’), compared to simple one-noun intransitive sentences (‘A krads’). To permit a direct comparison, the table reports the proportion of transitive test sentences for which the first noun was assigned an agent (A0) interpretation, and the proportion of intransitive test sentences with the agent (A0) role assigned to the single noun in the sentence. Here we report only the results from the best-performing parser (trained with VB EM, and content/function word pre-clustering), compared to the same classifiers trained with gold standard argument identification. When trained on arguments identified via the unsupervised POS tagger, noun pattern features promoted agent interpretations of transitive subjects (relative to the lexical baseline), but not for intransitive subjects. This differentiation between transitive and intransitive sentences was clearer when more known nouns were provided. Verb position features, in contrast, promote agent interpretations of subjects weakly with unsupervised argument identification, but equally for transitive and intransitive.

Noun pattern features were robust to increases in parsing noise. The behavior of verb position features suggests that variations in the identifiability of different parts of speech can affect the usefulness of alternative representations of sentence structure. Representations that reflect the position of the verb may be powerful guides for understanding simple English sentences, but representations reflecting only the number and order of nouns can dominate early in acquisition, depending on the integrity of parsing decisions.

4.1.5 Conclusion and Future Work

The key innovation in this section is the combination of unsupervised part-of-speech tagging and argument identification to permit learning in a simplified SRL system. Children do not have the luxury of treating part-of-speech tagging and semantic role labeling as separable tasks. Instead, they must learn to understand sentences starting from scratch, learning the meanings of some words, and using those words and their patterns of arrangement into sentences to bootstrap their way into more mature knowledge.

We have created a first step toward modeling this incremental process. We combined unsupervised parsing with minimal supervision to begin to identify arguments and predicates. An SRL classifier used simple representations built from these identified arguments to extract useful abstract patterns for classifying semantic roles. Our results suggest that multiple simple representations of sentence structure could co-exist in the child's system for sentence comprehension; representations that will ultimately turn out to be powerful guides to role identification may be less powerful early in acquisition because of the noise introduced by the unsupervised parsing.

Notably we have also shown that a learner can begin to identify verb predicates based on their argument taking behaviour based on the knowledge of a handful of nouns only. This leads the way to a development path where total word learning does not have to precede syntax learning; the two certainly interact and feed each other.

In the next section we explore when the semantic feedback is similarily decreased to rely on only a subset of semantic background knowledge that children have access to. Instead of assuming that the learner can understand the full semantics of the scene, and then uses this information to drive sentence understanding and learning, we seek to use some minimal amount of background knowledge to drive a feedback signal that still allows plausible interpretations and patterns to be learned from real text, in the absence of true semantic feedback. This minimal feedback can be combined with the HMM based argument and predicate identification to form a full minimally supervised BabySRL in section 4.3.

4.2 Animacy Feedback

Theories of human language acquisition assume that learning to understand sentences is naturally a partially-supervised task: the fit of the learner's predicted meaning with the referential context and background knowledge provides corrective feedback (e.g., Pinker [Pinker, 1989]). But this feedback must be noisy; referential scenes provide ambiguous information about the semantic roles of sentence participants. For example, the same participant could be construed as an agent who 'fled' or as a patient who is 'chased'.

In this section, we address this problem by designing a Semantic Role Labeling system (SRL), equipped with shallow representations of sentence structure motivated by the structure-mapping account, that learns with no gold-standard feedback at all. Instead, the SRL provides its own internally-generated feedback based on a combination of world knowledge and linguistic constraints. As a simple stand-in for world knowledge, we assume that the learner has animacy information for some set of nouns, and uses this knowledge to determine their likely roles. In terms of linguistic constraints, the learner uses simple knowledge about the possible arguments verbs can appear with.

This approach has two goals. The first is to inform theories of language learning by investigating the utility of the proposed internally-generated feedback as one component of the human learner's tools. Second, from an NLP and Machine Learning perspective we propose to inject information into a supervised learning algorithm through a channel other than labeled training data. From both perspectives, our key question is whether the algorithm can use these internally labeled examples to extract general patterns that can be applied to new cases.

By building a model that uses shallow representations of sentences and minimal feedback, but that mimics features of language development in children, we can explore the nature of initial representations of syntactic structure.

Our previous computational experiments with BabySRL in Chapter 3 suggest that it is possible to learn to assign basic semantic roles based on the simple representations proposed by the structure-mapping view. The classifier's features were limited to lexical information (nouns and verbs only) and the number and order of nouns in the sentence, and trained on a sample of child-directed speech annotated in PropBank [Kingsbury and Palmer, 2002] style. Given this training, our classifier learned to label the first of two nouns as an agent and the second as a patient. Even amid the variability of casual speech, simply representing the target word as the first or the second of two nouns significantly boosts SRL performance (relative to a lexical baseline) on transitive sentences containing novel verbs. This result depends on key assumptions of the structure-mapping view, including abstract representations of semantic roles, and abstract but simple representations of sentence structure.

However, our previous experimental design has a serious drawback that limits its relevance to the study of how children learn their first language. In training, our SRL received gold standard feedback consisting of correctly labeled sentences. Thus when the SRL made a mistake in identifying the semantic role of any noun in a sentence, it received feedback about the 'true' semantic role of this noun. As noted above, this is an unrealistic assumption for the input to human learners.

Here we ask whether an SRL could learn to interpret simple sentences even without gold-standard feedback by relying on world knowledge to generate its own feedback. This internally-generated feedback was based on the following assumptions. First, nouns referring to animate entities are likely to be agents, and nouns referring to inanimate entities are not. Second, each predicate takes at most one agent. Such role uniqueness constraints are typically included in linguistic discussions of thematic roles [Bresnan, 1982; Carlson, 1998]. The animacy heuristic is not always correct, of course. For example, in "The door hit you", an inanimate object is the agent of action, and an animate being is the patient. Nevertheless, it is useful for two reasons. First, there is a strong cross-linguistic association between agency and animacy [Aissen, 1999; Dowty, 1991]. Second, from the first year of life, children have strong expectations about the capacities of animate and inanimate entities [Baillargeon et al., (in press]. Two-year-olds more readily comprehend sentences with animate than inanimate subjects, suggesting early sensitivity to the tendency for animates to be agents [Corrigan, 1988; Lempert, 1989]. Given the universal tendency for speakers to talk about animate action on less animate objects, many sentences will present useful training data to the SRL: In ordinary sentences such as ”You broke it,” feedback generated based on animacy will resemble gold-standard feedback.

Much of this section was originally published in [Connor et al., 2009].

4.2.1 Learning Model

As a starting point, we will use the BabySRL model introduced in chapter 3, for now assuming that we are given knowledge of true nouns and verbs. We further simplify the SRL task such that classification is between two macro-roles: A0 (agent) and A1 (non-agent; all non-A0 arguments). We did so because we reason that this simplified feedback scheme can be primarily informative for a first stage of learning in which learners identify how their language identifies agents vs. non-agents in sentences. In addition, this level of role granularity is more consistent across verbs [Palmer et al., 2005].

For the final predictions, the classifier uses predicate-level inference to ensure coherent argument assignments. In our task the only active constraints are that all nouns require a label (a NO label is possible for non role assigned arguments), and that they have unique labels, which for this restricted case of A0 vs. not A0 means there will be only one agent.

Training and Feedback

The key feature of this experiment lies in the way feedback is provided. Ordinarily, during training, SRL classifiers predict a semantic label for an argument and receive gold-standard feedback about its correct semantic role. Such accurate feedback is not available for the child learner. Children must rely on their own error-prone interpretation of events to supply feedback. This internally-generated feedback signal is presumably derived from multiple information sources, including the plausibility of particular combinations of argument-roles given the current situation [Chapman and Kohn, 1978]. Here we model this process by combining background knowledge with linguistic constraints to generate a training signal. The ‘unsupervised’ feedback is based on: 1) nouns referring to animate entities are assumed to be agents, while nouns referring to inanimate entities are non-agents and 2) each predicate can have at most one agent.

This internally-generated feedback bears some similarities to Inference Based Training [Punyakanok et al., 2005b]. In both cases the feedback to local supervised classifiers depends on global constraints. With IBT, feedback for mistakes is only considered after global inference, but for BabySRL the global inference is applied to the feedback itself. Figure 4.6 gives an overview of the training and testing procedure, making clear the distinction between training and testing inference.

The results in this section are based on experiments with training data from one child in the semantic role labeled CDS corpus described in section 3.1 ('Sarah'; utterances in samples 1 through 80, recorded at child age 2;3-3;10 years) This child-directed speech training set consists of about 8300 tagged arguments over 4700 sentences, of which a majority had a single verb and two labeled nouns. The annotator agreement on this data set ranged between 95-97% at the level of arguments. In the current section these role-tagged examples provide a comparison point for the utility of animacy-based feedback during training.

Our BabySRL did not receive these hand-corrected semantic roles during training. Instead, for each training example it generated its own feedback based in part on an animacy table. To obtain the animacy table we coded the 100 most frequent nouns in our corpus (which constituted less than 15% of the total number of nouns, but 65% of noun occurrences). We considered 84 of these nouns to be unambiguous in animacy: Personal pronouns and nouns referring to people were coded as animate (30). Nouns referring to objects, body parts, locations, and times, were coded as inanimate (54). The remaining 16 nouns were excluded because they were ambiguous in animacy (e.g., dolls, actions). See Appendix A for a full list of animate and inanimate words used for generating our feedback signal.

We test 3 levels of feedback representing increasing amounts of linguistic knowledge used to generate internal interpretations of the sentences. Using the animacy table, Animacy feedback (**Feedback 1**) was generated as follows: for each noun in training, if it was coded as animate it was labeled A0, if it was coded as inanimate it was labeled A1, otherwise no feedback was given. Because of the frequency of animate nouns this gives a skewed distribution of 4091 animate agents and 1337 inanimate non-agents.

(**Feedback 2**) builds on Feedback 1 by adding another linguistic constraint: if a noun was not found in the animacy-table and there is another noun in the sentence that is labeled A0, then the unknown noun is an A1. In the training set this adds non-agent training examples, yielding 4091 A0 and 2627 A1 examples.

Feedback 1 and Feedback 2 allow two nouns in a sentence to be labeled with A0. **Feedback 3** prevents this; it implements a unique agent constraint that incorporates bootstrapping to make an 'intelligent guess' about which noun is the correct agent. This decision is made based on the current predictions of the classifier. Given a sentence with multiple animate nouns, the classifier predicts a label for each, and the one with the highest score for A0 is declared the true agent and the rest are classified as non-agent. Note that we cannot apply role uniqueness to the A1 (not A0) role, given that this label encompasses multiple non-agent roles. This feedback scheme, allowing at most one agent per sentence, reduces the number of A0 examples and increases the number of A1 examples to 3019 A0 and 3699 A1.

Feature Sets

The basic features of the BabySRL depend on a simple syntactic representation based on number and order of nouns and relative location of verb (Noun pattern and verb position, see section 3.2). We compare the noun pattern (NPat) feature to a baseline lexical feature set (Words): the target noun and the root form of the predicate. The NPat feature set includes lexical features as well as features indicating the number and order of the noun (first of two, second of three, etc.). With gold-standard role feedback, chapter 3 demonstrates that the NPat feature allowed the BabySRL to generalize to new verbs: it increased the system's tendency to predict that the first of two nouns was A0 and the second of two nouns A1 for verbs not seen in training.

To the extent that in child-directed speech the first of two nouns tends to be an agent, and agents tend to be animate, we anticipate that with the NPat feature the BabySRL will learn the same thing, even when provided with internally-generated feedback based on animacy. In the previous chapter we showed that, because this NPat feature set represents only the number and order of nouns, with this feature set the BabySRL reproduced the errors children make as noted in the Introduction, mistakenly assigning agent- and non-agent roles to the first and second nouns in intransitive test sentences containing two nouns. In the present chapter, the linguistic constraints provide an addi-

```
Algorithm BABYSRL TRAINING
    INPUT: Unlabeled Training Sentences
    OUTPUT: Trained Argument Classifier

    For each training sentence
      Generate Internal Feedback: Find interpreted meaning
        Feedback 1: Apply Animacy Heuristic
        For each argument in the sentence (noun)
          If noun is animate → mark as agent
          If noun is inanimate → mark as non-agent
          else leave unknown
        end

        Feedback 2: Known agent constraint
        Beginning with Feedback 1
        If an agent was found
          Mark all unknown arguments as non-agent

        Feedback 3: Unique agent constraint
        Beginning with Feedback 2
        If multiple agents found
          Find argument with highest agent prediction
          Leave this argument an agent, mark rest as non-agent

      Train Supervised Classifier
        Present each argument to classifier
         Update if interpreted meaning does not match classifier prediction
    end
```

(a) Training

```
Algorithm BABYSRL TESTING
    INPUT: Unlabeled Testing Sentences
    OUTPUT: Role labels for each argument

    For each test sentence
      Predict roles for each argument
      Test Inference:
        Find assignment to whole sentence with highest sum of predictions that doesn't violate uniqueness constraint
    end
```

(b) Testing

Figure 4.6: BabySRL training and testing procedures. Internal feedback is generated using animacy plus optional constraints. This feedback is fed to a supervised learning algorithm to create an agent-identification classifier.

tional cause for this error. In addition, as a first step in examining recovery from the predicted error, we added a verb position feature (VPos) specifying whether the target noun is before or after the verb. Given these features, the BabySRL's classification of transitive and two-noun intransitive test sentences diverged, because the gold-standard training supported the generalization that pre-verbal nouns tend to be agents, and post-verbal nouns tend to be patients.

Testing

To evaluate the BabySRL we tested it with both a held-out sample of child-directed speech, and with constructed sentences containing novel verbs as detailed in section 3.3. These sentences provide a more stringent test of generalization than the customary test on a held-out section of the data. Although the held-out section of data contains unseen sentences, it may contain few unseen verbs. In a held out section of our data, 650 out of 696 test examples contain a verb that was encountered in training. Therefore, the customary test cannot tell us whether the system generalizes what it learned to novel verbs.

In this section we report on results with test sentence templates that were filled with random sampling of nouns from two distributions:

Full distribution: The first nouns in the test sentences (A) are chosen from the set of all first nouns in our corpus, taking their frequency into account when sampling. The second nouns in the sentences (B) are chosen from the set of nouns appearing as second nouns in the sentence of our corpus. This way of sampling the nouns will maximize the SRL's test performance based on the baseline feature set of lexical information alone (Words). This is so because in our data many sentences have an animate first noun and an inanimate second noun. Based on these words alone the SRL could learn to predict an A0-A1 role sequence for our test sentences. Nevertheless, we expect that when the BabySRL is also given the NPat feature it should be able to perform better than this high lexical baseline.

Two animate nouns: In these test sentences the A and B nouns are chosen from our list of animate nouns. We chose nouns from this list that were fairly frequent (ranging from 8 to 240 uses in the corpus), and that occurred roughly equally as the first and second noun. This mimics the sentences used in the experiments with children (e.g., "The girl is kradding the boy!"). The lexical baseline system's tendency to assign an A0-A1 sequence to these nouns should be much lower for these test sentences. We therefore expect the contribution of the NPat feature to be more apparent in these test sentences.

The test sentences with novel verbs ask whether the classifier transfers its learning about argument role assignment to unseen verbs. Does it assume the first of two nouns in a simple transitive sentence ('A gorps B') is the agent (A0) and the second is not an agent (A1)? In section 3.4 we

Feedback	Words	+NPat
1. Just Animacy	0.72	0.73
2. + non A0 Inference	0.74	0.75
3. + unique A0 bootstrap	0.70	0.74
10 Gold	0.43	0.47
100 Gold	0.61	0.65
1000 Gold	0.75	0.76

Table 4.2: Agent identification results (A0 F1) on held-out sections of the Sarah Childes corpus. We compare a classifier trained with various amounts of gold labeled data (averaging over 10 different samples at each level of data). For noun pattern features the internally generated bootstrap feedback provides comparable accuracy to training with between 100-1000 fully labeled examples.

showed that a system with the same feature and representations also over-generalized this rule to two-noun intransitives ('A and B gorp'), mimicking children's behavior. In the present experiments this error is over-determined, because the classifier learns only an agent/non-agent contrast, and the linguistic constraints forbid duplicate agents in a sentence. However, for comparison to the earlier results we test our system on the 'A and B gorp' sentences as well.

4.2.2 Experimental Results

Our experiments use internally-generated feedback to train simple, abstract structural features: the NPat features that proved useful with gold-standard training in chapter 3. Section 4.2.2 tests the system on agent-identification in held-out sentences from the corpus, and demonstrates that the animacy-based feedback is useful, yielding SRL performance comparable to that of a system trained with 1000 sentences of gold-standard feedback. Section 4.2.2 presents the critical novel-verb test data, demonstrating that this system replicates key findings of the previous chapter with no gold standard feedback. Using only noisy internally-generated feedback, the BabySRL learned that the first of two nouns is an agent, and generalized this knowledge to sentences with novel verbs.

Comparing Self Generated Feedback with Gold Standard Feedback

Table 4.2 reports for the varying feedback schemes, the A0 F1 performance for a system with either lexical baseline feature (Words) or structural features (+NPat) when tested on a held-out section of the Sarah CHILDES corpus section 84-90, recorded at child ages 3;11-4;1 years. Agent identification based on lexical features is quite accurate given animacy feedback alone (Feedback

	Full Distribution Nouns			Animate Nouns		
Feedback	Words	NPat	VPos	Words	NPat	VPos
‘A gorps B’						
1. Animacy	0.86	0.86	0.87	0.76	0.79	0.70
2. + non A0 Inference	0.87	0.92	0.90	0.63	0.86	0.85
3. + unique A0 bootstrap	0.87	0.95	0.89	0.63	0.82	0.66
‘A and B gorp’						
1. Animacy	0.86	0.86	0.84	0.76	0.79	0.68
2. + non A0 Inference	0.87	0.92	0.85	0.63	0.86	0.66
3. + unique A0 bootstrap	0.87	0.95	0.86	0.63	0.82	0.63

Table 4.3: Percentage of sentences interpreted as agent first (%A0-A1) by the BabySRL when trained on unlabeled data with the 3 internally-generated feedback schemes described in the text. Two different two-noun sentence structures were used (‘A gorps B’, ‘A and B gorp’), along with two different methods of sampling the nouns (Full Distribution, Animate Nouns) to create test sets with 100 sentences each.

1). As expected, because many agents are animate, the animacy tagging heuristic itself is useful. As linguistic constraints are added via non-A0 inference (Feedback 2), performance increases for both the lexical baseline and NPat feature-set, because the system experiences more non-A0 training examples.

When the unique A0 constraint is added (Feedback 3), the lexical baseline performance decreases, because for the first time animate nouns are being tagged as non-agents. With this feedback the NPat feature set yields a larger improvement over lexical baseline, showing that it extracts more general patterns. We discuss the source of these feedback differences in the novel-verb test section below.

We compared the usefulness of the internally-generated feedback to gold-standard feedback by training a classifier equipped with the same features on labeled sentences. We reduced the SRL labeling for the training sentences to the binary agent/non-agent set, and trained the classifier with 10, 100, or 1000 labeled examples. Surprisingly, the simple feedback derived from 84 nouns labeled with animacy information yields performance equivalent to between 100 and 1000 hand-labeled examples.

Comparing Structural Features with Lexical Features

The previous section shows that the BabySRL equipped with simple structural features can use internally generated feedback to learn a simple agent/non-agent classification, and apply it to unseen sentences. In this section we probe what the SRL has learned by testing generalization to new verbs in constructed sentences. Table 4.3 summarizes these experiments. The results are broken down both by what sentence structure is used in test (‘A gorps B’, ‘A and B gorp’) and how the nouns ‘A’ and ‘B’ are sampled (Full Distribution, Animate Nouns). The results are presented in terms of %A0A1: the percentage of test sentences that are assigned an Agent role for ‘A’ and a non-Agent role for ‘B’.

For the transitive ‘A gorps B’ sentences, A0A1 is the correct interpretation; A should be the agent. As predicted, when A and B are sampled from the full distribution of nouns, simply basing classification on the Words feature-set already strongly predicts this A0A1 ordering for the majority of cases. This is because the data (language in general, child directed speech in particular here) are naturally distributed such that particular nouns that refer to animates tend to be agents, and tend to appear as first nouns, and those that refer to inanimates tend to be non-agents and second nouns. Thus, a learner representing sentence information in terms of words only succeeds with full-distribution ’A gorps B’ test sentences even with the simplest animacy feedback (Feedback 1); the A and B nouns in these test sentences reproduce the learned distribution. Also as predicted, given this simple feedback, the additional higher-level features (NPat, VPos) do not improve much upon the lexical baseline. This is due to the strictly lexical nature of the animacy feedback: each lexical item (e.g., ’you’ or ’it’) will always either be animate or inanimate and therefore either A0 or A1. Therefore, in this case lexical features are the best predictors.

Also as expected, higher-level features (NPat, and VPos) improve performance with a more sophisticated self-generated feedback scheme. Adding inferred feedback to label unknown nouns as A1 when the sentence contains a known animate noun (Feedback 2) decreases the ratio of A0 to non-A0 arguments. This feedback is less lexically determined: for nouns whose animacy is unknown, feedback will be provided only if there is another animate noun in the sentence. This leaves room for the abstract structural features to play a role.

Next we test a form of the unique-A0 constraint. In (Feedback 3), in addition to the non-A0 inference added in (Feedback 2), the BabySRL intelligently selects one noun as A0 in sentences with multiple animate nouns. With this feedback we see a striking increase in test performance based on the noun pattern features over the lexical baseline. In principle, this feedback mechanism might permit the classifier to start to learn that animate nouns are not always agents. Early in training, the noun pattern feature learns that first nouns tend to be animate (and therefore interpreted as agents), and it feeds this information back into subsequent training examples, generating new feedback that continues to interpret as agents those animate nouns that appear first in sentences containing two animates.

For the nouns sampled from the full distribution we see that structural features improve over the lexical baseline despite the high performance of the lexical baseline. This finding tells us that simple representations of sentence structure can be useful in learning to interpret sentences even with no gold-standard training. Provided only with simple internally-generated feedback based on animacy knowledge and linguistic constraints, the BabySRL learned that the first of two nouns tends to be an agent, and the second of two does not.

The results for the 'A B gorp' test sentences demonstrate an important way in which predictions based on different simple structure representations can diverge. As expected, the NPat feature makes the same overgeneralization error seen by children and the gold trained BabySRL from chapter 3. However, when the VPos feature is added, different results are obtained for the 'A gorp B' and 'A and B gorp' sentences. The SRL predicts fewer A0A1 for 'A and B gorp' (it cannot predict the expected A0A0 because of the uniqueness constraint used in test inference).

Next, we replicate our findings by performing the same experiments with test sentences in which both 'A' and 'B' are animate. Because lexical features alone cannot determine if 'A' or 'B' should be the agent, it is a more sensitive test of generalization.

When we look at the lexical baseline for animate sentences, the agent-first percentage is lower compared to the full distribution results, because the word features indicate nearly evenly that both nouns should be agents, so the Words baseline model must rely on small, chance differences in its experience with particular words. This percentage is still well above chance due to the method used to apply inference during testing. Recall that the classifier uses predicate-level inference at test to ensure that only one argument is labeled A0. This inference is implemented using a beam search that looks at arguments in a fixed order and roles from A0 up. Thus in the case of ties there is a preference for first seen solutions, meaning A0A1 in this case. This bias has a large effect on the SRL's baseline performance with the test sentences containing two animate nouns. Despite this high baseline, however, because lexical features alone cannot determine if ‘A’ or ‘B’ should be the agent, we are able to see more clearly the improvement gained by including structural features.

Regardless of our testing scheme, we see that as the feedback incorporates more information, both added linguistic constraints and the SRL’s own prior learning, the noun pattern structural feature is better used to identify agents beyond the lexical baseline. The largest improvement over this lexical baseline is obtained by combining knowledge of animacy with a single-agent constraint and bootstrapping predictions based on prior learning.

Conclusion

Conventional approaches to supervised learning require creating large amounts of hand-labeled data. This is labor-intensive, and limits the relevance of the work to the study of how children learn languages. Children do not receive perfect feedback about sentence interpretation. Here we found that our simple SRL classifier can, to a surprising degree, attain performance comparable to training with 1000 sentences of labeled data. This suggests that fully labeled training data can be supplemented by a combination of simple world knowledge (animates make good agents) and linguistic constraints (each verb has only one agent). The combination of these sources provides an informative training signal that allows our BabySRL to learn a high-level semantic task and generalize beyond the training data we provided to it. The SRL learned, based on the distribution of animates in sentences of child-directed speech, that the first of two nouns tends to be an agent. It did so based on representations of sentence structure as simple as the ordered set of nouns in the sentence. This demonstrates that it is possible to learn how to correctly assign semantic roles based on these very simple cues. This together with experimental work (e.g. [Fisher, 1996] suggests that such representations might play a role in children’s early sentence comprehension.

4.3 Minimally Supervised BabySRL

Putting both minimally supervised argument identification and animacy based training together into a single largely unsupervised system allows us to see what simple representations can learn given a clear lower bound on knowledge children have available. This minimally supervised BabySRL model represents an end-to-end SRL system (first pictured in figure 1.1) that uses knowledge of the identity and animacy of a handful of nouns plus exposure to language to identify a class of arguments, use the number of arguments to identify likely predicate, and then infer semantics based on the structure of arguments and predicate.

To normalize the experimental setups of previous results reported here, which may have been trained over data from different children with slightly different parameters, we have trained our entire minimal supervised BabySRL on each of the three children independently and report the average plus individual performance. For HMM argument identification we used the VB trained HMM with function/content word split, discarding function states during identification, seeded with all 75 lexical development norm words. For animacy feedback, the most frequent 365 known animate and inanimate words were used. Finally, inference on the predicted sentences during testing (ensuring role uniqueness) was not used in either the fully or minimally supervised cases, so multiple role predictions are allowed.

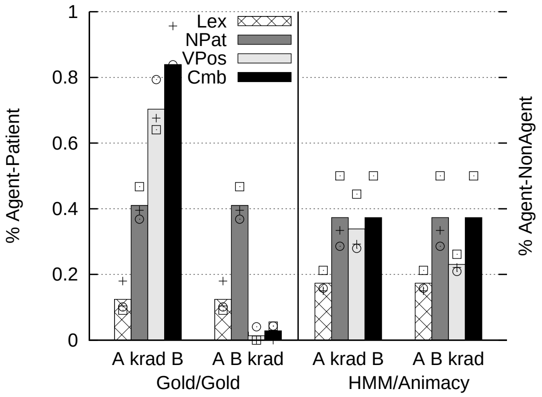


Figure 4.7: Comparison of fully and minimally supervised BabySRL on two noun sentences. Experiments over all three children, with bars representing average performance and points representing each child. A noun pattern based representation alone may cause specific errors in sentence interpretation. With two noun sentences, noun pattern does not distinguish between “A krads B” and “A and B are kradding”, while features that depend on verb position do. On the left, when trained with perfect arguments and complete feedback, the verb position feature strongly predicts agent/patient for two noun transitive, and weakly predicts the same for two noun intransitive. This pattern dominates even when verb position is combined with noun pattern. On the right, with the minimally supervised SRL trained with HMM arguments and animacy feedback, the noun pattern and verb position alone show the same general patterns in terms of increase over lexical baseline as when trained with perfect feedback, but verb position predictions no longer dominates the combined feature representation performance. The number of arguments is a more consistent cue due partly to the weaker ability to identify verbs. The combination of noise in argument identification and low amount of supervision through animacy feedback causes the decline in overall performance.

Figure 4.7 graphs these results for two noun transitive sentences, and the patterns have not changed much from previous results; for both fully and minimally supervised BabySRL all simple structural features succesfully increase the agent-patient interpretation of two-noun transitive sentences over a lexical baseline. Note that due to feedback differences the predictions for the minimally supervised system are between agent and not agent, and not across all different possible roles. The noun pattern feature handles noise from both argument identification and feedback better than verb position so it is able to extract a more reliable pattern regarding the expected order of agents.

While the simple abstract noun pattern feature has been shown to allow early learning of semantic patterns while being robust to noise from both input and feedback, it is clearly not a perfect representation. Namely without verb information a noun only representation will confuse two noun sentences of different structures: transitive “A krads B” and intransitive “A and B are kradding”. Figure 4.7 shows the predictions of the four feature sets on these two sentence structures with both perfect pipeline of gold arguments and feedback, and the minimal supervised case of HMM argument identification and animacy based feedback. With complete information the NPat feature makes the same prediction for both transitive and intransitive sentences, while the verb position feature is able to distinguish these two structures and predict agent/patient for transitive and not for intransitive. The combined feature set is buoyed by both NPat and VPos in the transitive case, but its predictions agree mostly with VPos in the intransitive case. With complete information the verb position feature dominates in the combined feature set.

With the minimally supervised HMM arguments and animacy feedback (right side of figure 4.7), the NPat feature still makes the same predictions for transitive and intransitive, but in this noisy case the combined feature set agrees with it in both cases. Although the verb position alone still learns to treat the two structures differently, it does not handle the noisier input and feedback as well (especially the noisier predicate identification), and so the NPat feature is able to dominate it when the two features are present together in the combined feature set. Although the verb position feature may be a more powerful indicator in the english language, in early language acquisition when not all information in the sentence is available or understood by the learner, noun structure information may serve as a more reliable cue.

4.4 Conclusion

The next step is to ‘close the loop’, using higher level semantic feedback to improve the earlier argument identification and parsing stages. Perhaps with the help of semantic feedback the system can automatically improve predicate identification, which in turn allows it to correct the observed intransitive sentence error. This approach will move us closer to the goal of using initial simple structural patterns and natural observation of the world (semantic feedback) to bootstrap more and more sophisticated representations of linguistic structure.

Chapter 5

Latent Baby SRL

When first learning language, children must cope with enormous ambiguity both in terms of meaning and structure. They have to pick out candidate meanings from the world and align them with the sentence forms presented, without already knowing which parts of the sentence refers to which parts of the scene. Despite this, children do learn to interpret sentences of various structures, and do so without detailed feedback about whether their interpretations were correct.

Computational language learning systems often rely on exactly this level of implausible fine grained feedback to solve this problem, divining structure from a sentence and fitting the true meaning to it. Often this is done in a pipeline where first a fixed structure for each sentence (commonly full parse trees) is learned, and then this structure is used to learn a predefined meaning representation (in our case Semantic Role Labels). The structure learned is not tailored for the final semantic task, and the learning depends on the provision of an exact interpretation of the sentence as feedback for learning. We started with this exact setup to experiment with our BabySRL simple representations in chapter 3.

In this chapter we experiment with a computational system that models early stages of language acquisition, attempting to learn to predict semantic roles from a corpora of child directed speech. The system treats a highly simplified form of sentence structure as a latent structure that must be learned jointly with the role classification based solely on high level semantic feedback in an online, sentence by sentence setting.

With this system we aim to show:

- With just semantic role feedback we can identify latent argument and predicate identifiers.
- We can use the latent structure information to train argument and predicate classifiers, incorporating additional features and prior knowledge.

- Improved hidden structure allows generalization of role feedback to a more realistic, ambiguous level.

- To recover from loss of feedback information, we need to incorporate a small amount plausible bottom-up noun-based background knowledge.

Most of this chapter has previously appeared in [Connor et al., 2011].

5.1 Related Work

In our previous computational experiments with BabySRL we showed that it is possible to learn to assign basic semantic roles based on the shallow sentence representations in chapter 3. Furthermore, these simple structural features were robust to drastic reductions in the integrity of the semantic-role feedback or being used with a minimally supervised parser in chapter 4. These experiments showed that representations of sentence structure as simple as ‘first of two nouns’ are useful as a starting point for sentence understanding, even given the bare minimum of supervised training, and lead to systematic errors.

Other models of early language acquisition such as in [Alishahi and Stevenson, 2010] provide a lexically motivated model of acquisition that is capable of production and comprehension, including argument role understanding. These models assume as input a simple syntactic structure for the sentence, including identifying arguments and predicates. One of the focuses of the current work is how can we identify these structures without being given this information.

A similar task which happens at an earlier stage of language acquisition is the problem of word segmentation. [Johnson et al., 2010] presents a computational model that jointly learns word segmentation along with word referents, and demonstrates synergistic benefits from learning these together. Here we try to use this insight to learn both the structure of the sentence in terms of identifying arguments and predicates along with the higher level semantics.

For the general natural language problem of semantic role labeling, it is well known that the parsing step which gives structure to the sentence is pivotal to final role labeling performance [Gildea and Palmer, 2002; Punyakanok et al., 2008]. There is much interest in trying to learn both syntax and semantics jointly, with two recent CoNLL shared tasks devoted to this problem [Surdeanu et al., 2008; Hajič et al., 2009]. In both cases the best systems learned syntax and semantics separately, then applied together, so at this level the promise of joint synergies have yet to be realized.

5.2 Model

We model language learning with a Semantic Role Labeling (SRL) task [Carreras and Màrquez, 2004]. This allows us to ask whether a learner, equipped with particular theoretically-motivated representations of the input, can learn to understand sentences at the level of who did what to whom with controlled amounts of supervision. As a baseline architecture for our BabySRL system introduced in chapter 3, which is itself based on a standard pipeline architecture of a full SRL system (e.g. [Punyakanok et al., 2008]). The stages are: (1) Unsupervised parsing of the sentence, (2) Identifying potential arguments and predicates based on the parse, (3) Classifying role labels for each potential argument, trained using role-labeled child directed speech.

For the lowest level of representation, after the words themselves, we use an unsupervised Hidden Markov Model (HMM) tagger to provide a context sensitive clustering of the words (essentially an unsupervised POS parse). The HMM states are preclustered into a function/content word division which is both beneficial for unsupervised POS performance (see section 4.1.2), and also psycholinguistically defensible [Shi et al., 1998, 1999]. An alternative approach is to differentiate the prior distribution for different sets of states, which unsurprisingly provides nearly the same division of function and content word states [Moon et al., 2010]. Our HMM model is trained with two million words of child directed speech, in a process that represents the year or so of listening to speech and clustering based on distributional similarity before the child firmly learns any specific words or attempts multi-word sentence interpretation.

Given the sentence and unsupervised tagging, the next step in the system is to determine which words in the sentence are predicates, and which words are potential arguments. We use a structured approach to this, considering the entire predicate/argument identification of the sentence at once, with the constraints that (1) only content words are considered (identified by preclustering of HMM states), (2) there is exactly one predicate, and (3) at most four arguments. These constraints are true of over 98% of the sentences in our training data. The next section describes how we identify these structures.

Once a predicate and arguments have been identified, a role classifier must decide the role for each argument relative to the predicate. We use the abstract roles of Propbank [Kingsbury and Palmer, 2002], with A0 roughly indicating agent, and A1 patient. The role classifier can only rely on features that can be computed with information available at this stage of processing, which means the words themselves, and number and order of arguments and predicates as predicted by the previous step.

As input to our learner we use samples of child directed speech (CDS) from the CHILDES corpora, described in section 3.1. In this chapter we used samples of parental speech to one child (Adam; [Brown, 1973]) as training and test data, sections 01-20 (child age 2;3 - 3;1) for training, and sections 21-23 for test. To simplify evaluation, we restricted training and testing to the subset of sentences with a single predicate (over 85% of the sentences). Additionally we focus on noun arguments in terms of identification, although this may miss some other semantic roles. The final annotated sample contains about 2800 sentences, with 4778 noun arguments.

We want to be able to train this model in an online fashion where we present each sentence along with some semantic constraints (feedback), and the classifier updates itself accordingly. In the next section we will describe how we can train this model without direct supervision, and the representations that are used.

5.3 Latent Training

We can phrase our problem of Semantic Role Labeling as learning a structured prediction task, which depends on some latent structure (argument and predicate identification). As input we have the sequence of words and HMM states for a given sentence, and the output is a role-labeled predicate-argument structure. The goal in our structured prediction task is to learn a linear function $f_w : X \to Y$ that maps from the input space X (sentences) to output space Y (role labeled argument structure):

$$f_w(x) = \arg\max_{y \in Y} \max_{h \in H} w \cdot \Phi(x, h, y) \tag{5.1}$$

Here H is a space of hidden latent structure that describes some connection between X and Y, Φ

is a feature encoding for the complete X, H, Y example structure, w is the learned weight vector that scores structures based on their feature encoding, and both $w, \Phi \in \mathbb{R}^n$. Conventionally this weight vector w would be learned from a set of labeled training examples $(x_i, y_i) \in X \times Y$, attempting to maximize the difference between the score for true structures y_i and all other structures for every training example. In this chapter we present a learning algorithm that does not rely on fully supervised examples (x_i, y_i) to train w, instead as feedback the learner recieves only constraints on possible true output structures $Y_i \subseteq Y$, along with constraints on possible hidden structures $H_i \subseteq H$.

Because of the max over H in the definition of f_w, the general optimization problem for finding the best w (given a training set of $\{x_i, y_i\}_{i=1}^M$ labeled examples) is non-convex[1]. Previously this has been solved using some variant of latent structure optimization such as in [Chang et al., 2010; Yu and Joachims, 2009]. Here we used an online approach and a modification of Collin's Structured Perceptron [Collins, 2002] with margin [Kazama and Torisawa, 2007]. This basic, purely latent algorithm (Algorithm 3) uses an approximation employed in [Felzenszwalb et al., 2008; Cherry and Quirk, 2008] where for each example the best h^* is found (according to the current model and true output structure) and then the classifier is updated using that fixed structure. In this algorithm α_w represents the learning rate and C is the margin.

Algorithm 3 Purely Latent Structure Perceptron

```
Initialize $w_0, t = 0$
repeat
    for all Sentences $(x_i, y_i)$ do
        $h_i^* \leftarrow \arg\max_{h \in H_i} w_t \cdot \Phi_w(x_i, h, y_i)$
        $y' \leftarrow \arg\max_y w_t \cdot \Phi_w(x_i, h_i^*, y) + C * \mathbf{1}[y \neq y_i^*]$
        $w_{t+1} \leftarrow w_t + \alpha_w(\Phi_w(x_i, h_i^*, y_i) - \Phi_w(x_i, h_i^*, y'))$
        $t \leftarrow t + 1$
    end for
until Convergence
```

The intuition behind algorithm 3 is that for every sentence the learner knows the true meaning of that sentence (has the true y), so it is able to find the arrangement of arguments and predicate (h^*) that best fit that meaning according to what it has already learned (current weight vector w_t). Once we identify the latent arguments and predicate, we use this identification to update the weight

[1] The problem arises from attempting to minimize the maximum score on non-true structures.

vector so the true role prediction y_i will be more likely in the future.

An issue here is that h^* is found and then forgotten for each x. If we are interested in h beyond its application to learning w to predict y, say for generalizing between related x, or for use in other examples/prediction problems, then we need a method to not lose this information. For example, in the case of our Semantic Role Labeling system, we may want to use the identified predicates to label verb states from the unsupervised parser, or predict arguments and predicates on new sentences without doing full role labeling.

Instead, we can train a latent predicate and argument classifier along with the role classifier, such that during the latent prediction for each sentence we find the structure that maximizes the score of both role classification and structure prediction. In addition, the exact meaning y_i may not be available for every sentence, so we instead incorporate a looser notion of feedback in terms of constraints on possible labels (Y_i) into the latent prediction step. This algorithm is summarized in algorithm 4. The end result is two classifiers, f_u to predict hidden structure and f_w to use hidden structure for top level task, that have been trained to work together to minimize training error.

Algorithm 4 Online Latent Classifier Training

```
 1: Initialize $w_0, u_0, t = 0$
 2: **repeat**
 3:   **for all** Sentences $(x_i, Y_i)$ **do**
 4:     $(h_i^*, y_i^*) \leftarrow \arg\max_{h \in H_i, y \in Y_i} w_t \cdot \Phi_w(x_i, h, y) + u_t \cdot \Phi_u(x_i, h)$
        // Update $u$ to predict $h^*$
 5:     $h' \leftarrow \arg\max_h u_t \cdot \Phi_u(x_i, h) + C * \mathbf{1}[h \neq h_i^*]$
 6:     $u_{t+1} \leftarrow u_t + \alpha_u(\Phi_u(x_i, h_i^*) - \Phi_u(x_i, h'))$
        // Update $w$ based on $h^*$ to predict $y^*$
 7:     $y' \leftarrow \arg\max_y w_t \cdot \Phi_w(x_i, h_i^*, y) + C * \mathbf{1}[y \neq y_i^*]$
 8:     $w_{t+1} \leftarrow w_t + \alpha_w(\Phi_w(x_i, h_i^*, y_i^*) - \Phi_w(x_i, h_i^*, y'))$
 9:     $t \leftarrow t + 1$
10:   **end for**
11: **until** Convergence
```

The intuition for the online process in algorithm 4 is that for each sentence the learner finds the best joint meaning and structure based on the current classifiers and semantic constraints (line 4), then seperately updates the latent structure f_u and output structure f_w classifiers given this selection. In the case where we have perfect high level semantic feedback $Y_i = y_i$, the role classifier will search for the argument structure that is most useful in predicting the correct labels (as in algorithm 3). More generally, partial feedback, which constrains the set of possible interpretations but does not indicate the one true meaning, may be provided and used for both labeling Y_i and hidden structure H_i.

This learning model allows us to experiment with the trade-offs among different possible sources of information for language acquisition. Given perfect or highly informative semantic feedback, our constrained learner can fairly directly infer the true argument(s) for each sentence, and use this as feedback to train the latent argument and predicate identification (what we might term semantic bootstrapping). On the other hand, if the semantic role feedback is loosened considerably so as *not* to provide information about the true number or identity of arguments in the sentence, the system cannot learn in the same way. In this case, however, the system may still learn if further constraints on the hidden syntactic structure are provided through another route, via a straight-forward implementation of the structure-mapping mechanism for early syntactic bootstrapping.

5.3.1 Argument, Predicate and Role Classification

For the latent structure training method to work, and for the hidden structure classifier to learn, the high level classifier and feature set ($f_w\ and \Phi_w$ respectively) must make use of the hidden structure information h. In our case, the role classifier makes use of (and thus modifies during training) the hidden argument and predicate identification in two ways. The first of these is quite direct: semantic role predictions are made relative to specific arguments and predicates. Semantic-role feedback therefore provides information about the identity of the nouns in the sentence. The second way in which the role classifier makes use of the hidden argument and predicate structure is less direct: The representations used by the SRL classifier determine which aspects of the predictions of the argument and predicate latent classifier are particularly useful in semantic role labeling, and therefore change via the learning permitted by indirect semantic-role feedback.

In the simplest case where we use the full set of correct role labels as feedback, we implement this by providing correct labels for each word in the input sentence that was selected by the latent classifier as an argument. This feedback is provided only for each noun that is the head of an argument-phrase. Thus the optimal prediction by the argument classifier will come to include at least those words. The argument classifier will therefore learn to identify predicates so as to maximize the accuracy of SRL predictions. This is the case of semantically-driven learning where veridical semantic feedback provides enough information to drive learning of both semantics and syntax.

With more ambiguous semantic feedback, the hidden argument and predicate prediction is not directed by straightforward matching of the full set of noun arguments identified via semantic feedback, but instead the drive is to select a hidden structure that allows the role classifier to provide an interpretation that fits the given semantic constraints. Without further constraints on the hidden structure itself, there may not be enough information to drive hidden structure learning.

Likewise, the hidden structure prediction of arguments and predicate depends on the words and HMM states below it, both in terms of features for prediction and constraints on possible structures. The hidden argument and predicate structure we are interested in labels each word in the sentence as either an argument (noun), a predicate (verb), or neither. We used the function/content word state split in the HMM to limit prediction of arguments and predicates to only those words identified as content words. In generating the range of possible hidden structures over content words, the latent structure classifier considers only those with exactly one predicate and one to four arguments.

As an example take the sentence “She likes yellow flowers.” There are four content words; with the constraint that exactly one is a predicate, and at least one is an argument, there are 28 possible predicate/argument structures, including the correct assignment where ‘She’ and ‘flowers’ are arguments of the predicate ‘likes.’ The full semantic feedback would indicate that ‘She’ is an agent and ‘flowers’ is a patient, so the latent score the SRL classifier predicts (line 4 in algorithm 4) will be the sum of the score of assigning agent to ‘She’ and patient to ‘flowers’, assuming both those words are selected as arguments in h. If a word does not have a semantic role (such as non-argument-nouns ‘likes’ or ‘yellow’ here) then its predictions do not contribute to the score. Through this mechanism the full semantic feedback strongly constrains the latent argument structure to select the true argument nouns. Table 5.1 shows the two possible interpretations for “She likes yellow flowers.” given full semantic feedback that identifies the roles of the correct arguments. Decisions regarding ‘likes’ and ‘yellow’ must then depend on the representation used by both the latent-structure predicate identifier and semantic semantic-role classifier.

(a) Sentence Full Feedback	She likes yellow flowers . A0 A1				
(b) Possible Interpretation 1			Possible Interpretation 2		
Sentence Argument Struct.	she likes yellow flowers N V N		Sentence Argument Struct.	she likes yellow flowers N V N	
(c) Feature Representation			Feature Representation		
Semantic Feat. $\Phi_w(x, h, y)$	she	argument:she predicate:likes NPat: 1 of 2 VPos:Before Verb w+1:likes	Semantic Feat. $\Phi_w(x, h, y)$	she	argument:she predicate:yellow NPat: 1 of 2 VPos:Before Verb w+1:likes
	flowers	argument:flowers predicate:likes NPat: 2 of 2 VPos: After Verb w-1:yellow w+1:.		flowers	argument:flowers predicate:yellow NPat: 2 of 2 VPos: After Verb w-1:yellow w+1:.
Structure Feat. $\Phi_u(x, h)$	she=N	word:she hmm:35 verb:likes w+1:likes hmm+1:42 NPat: 1 of 2	Structure Feat. $\Phi_u(x, h)$	she=N	word:she hmm:35 verb:yellow w+1:likes hmm+1:42 NPat: 1 of 2
	likes=V	verb:likes hmm:42 w-1:she hmm-1:35 w+1:yellow hmm+1:57 v:likes&2 args suffixes: s,es,kes		yellow=V	verb:yellow hmm:57 w-1:likes hmm-1:42 w+1:flowers hmm+1:37 v:flowers&2 args suffixes: w,ow,low

Table 5.1: Example Sentence, showing (a) the full (gold standard) semantic feedback that provides true roles for each argument, but no indication of the predicate, as well as (b) two possible hidden structures given this level of feedback. The next rows show (c) the feature representations for individual words. The Semantic Feature set shows the feature representation of each argument as used in SRL classification; the Structure Feature set shows the feature representation of the first argument and the predicate in each possible hidden structure. See text section section 5.3.1 for further description of the features.

Features

In the SRL classifier we started with the same base BabySRL features developed in Connor et al. [2008], simple structures that can be derived from a linear sequence of candidate nouns and verb. These features include 'noun pattern' features indicating the position of each noun in the ordered set of proposed in the sentence (e.g., first of three, second of two, etc; NPat in Table 5.1), and 'verb position' features indicating the position of each noun relative to the proposed verb (before or after; VPos in Table 5.1). In the above example, given the correct argument assignment these features would specify that 'She' is the first of two nouns and 'flowers' is the second of two. No matter whether 'likes' or 'yellow' is selected as a predicate, 'She' is before the verb and 'flowers' is after it. In addition we use a more complicated feature set that includes NPat and VPos features along with commonly-used features such as the words surrounding each proposed noun argument, and conjunctions of NPat and VPos features with the identified predicate (e.g., the proposed predicate is 'likes' and the target noun is before the verb) so that the role classifier is more dependent on correct predicate identification.

For the argument and predicate structure classifiers the representation $\Phi_u(x, h)$ only depends on words and the other arguments and predicates in the structure. We represent each word by its word form, predicted HMM state, and the word before and after. In addition we specify additional features specific to argument or predicate classification: the argument classifier uses noun pattern (NPat in table 5.1, and the predicate representation uses the conjunction of the verb and number of arguments and all suffixes of length up to three as a simple verb ending feature[2].

It should be noted that both the purely latent (algorithm 3 and latent classifier we have been discussing (algorithm 4) require finding the max over hidden structures and labelings according to some set of constraints. As implemented with the sentences found in our child directed speech sample, it is possible to search over all possible argument and predicate structures. In our set of training sentences there were at most nine content words in any one sentence, which requires searching over 1458 structures of exactly one predicate and at most four arguments. On average there were only 3.5 content words a sentence. Once we move on to more complicated language an alternative approximate search strategy will need to be employed.

In terms of actual implementation and parameter settings, we trained our latent structure perceptron with a learning rate of 0.1 for role classifier, and 0.01 for latent structure classifiers, with structure margin set at 1. For each experiment 10 rounds of training was run, with order of example randomized between rounds, and the best result based on held out development set performance was selected. During feature extraction over all possible structures in each of the test and training sentences, all features seen in fewer than 2 of the training sentences were pruned.

[2] This roughly represents phonological/distribution information that might be useful for clustering verbs together (e.g., Monaghan et al. [2005]), but that is not exploited by our HMM because the HMM takes transcribed words as input.

5.4 Experimental Evaluation

For evaluation, we are interested both in how well the final role classifier performs, and how accurately the predicate and argument classifiers identify correct structure when trained with just semantic feedback. Since there is only one true predicate per sentence we report the predicate accuracy: the percentage of sentences with the correct predicate identified. For arguments where there are multiple possible predictions per sentence we report the F1 of identifying arguments: the harmonic mean of precision and recall in predicting true arguments. Likewise since there are many possible role labels and words to label, we report the overall role F1 over all arguments and label predictions[3].

Note that unlike in previous chapters, in this chapter we only focus on evaluation of the semantic role labeling and predicate/argument identification on the test set of the Adam corpus. Specifically, Adam sections 01-20 were split into 2305 sentences for training and 257 sentences as held out development, and sections 21-23 were used for testing.

Our first experiment tests online latent training with full semantic feedback. As an upper bound comparison we train with perfect argument knowledge, so both classifiers are fully supervised. As a lower bound of predicate-argument classification we also include the expected result of selecting a random predicate/argument structure for each sentence.

Training	Predicate %	Argument F1	Role F1
Gold Arguments	0.9740	0.9238	0.6920
Purely Latent	0.5844	0.6992	0.5588
Latent Classifier	0.9263	0.8619	0.6623
Random Arguments	0.3126	0.4580	-

Table 5.2: Results on test set of SRL with argument/predicate as latent structure. With gold arguments, both structure classifier and role classifier are trained with full knowledge of the correct arguments for each sentence. Purely Latent does not use a latent argument and predicate classifier, it selects a structure for each sentence that maximizes role classification of true labels during training, and tests using the structure and labels that maximize role classification, algorithm 3 . Latent Classifier training trains an argument identifier using the structure that the role classifier considers most likely to give the correct labeling (where we know correct labels for each noun argument), algorithm 4.

Table 5.2 shows the performance of the two algorithms from section 5.3 compared to the previ-

[3] Since we focus on noun arguments, we miss those predicate arguments that do not include any nouns; the maximum SRL role F1 with only noun arguments correct is 0.8255

ously mentioned upper and lower bounds. All classifiers use the full feature sets from section 5.3.1. The purely latent method (algorithm 3) does not use an intermediate latent structure classifier, so the arguments and predicates it selects are only relative to maximizing the role classifier prediction. By incorporating a latent classifier into the training (algorithm 4) we see a large boost in both argument and predicate identification, as well as final role performance. The argument and predicate classifier effectively generalizes the training signal provided by the latent semantic feedback to achieve nearly the performance of being trained on the true arguments explicitly. Of special note is the predicate identification performance; while the semantic feedback implicitly indicates true arguments, it says nothing about the true predicates. The predicate classifier is able to extract this information solely from what latent structures help the role classifier make the correct role predictions.

To investigate the interaction between the two classifier's (hidden structure and SRL) representation choices, we test the latent classifier with the full argument and predicate feature sets when the role classifier incorporates the four feature types: just words, noun pattern, verb position, and a full model containing all these features as well as surrounding words and predicate conjunctions. As we add feature complexity that depends on more accurate latent structure identification, we should see improvement in both final role accuracy and argument and predicate identification.

Role Feat	Predicate %	Argument F1	Role F1
Words	0.6406	0.8108	0.6261
+NounPat	0.7296	0.8108	0.6154
+VPos	0.9328	0.8291	0.6530
+Surrounding words and Predicate conjunctions	0.9263	0.8619	0.6623

Table 5.3: With the full role feedback and latent classifier training, the role classifier features interact with the structure classifier. Better role classification through improved feature representation feedsback to allow for improved argument and predicate identification. The last two feature sets make strong use of the identity of the predicate, which encourages the predicate classifier to accurately identify the predicate.

Table 5.3 shows the increasing performance as the feature complexity increases. Most notable is the large difference in predicate identification performance between the feature sets that heavily depend on accurate predicate information and those that only use the word form of the identified predicate as a feature (+VPos and the full feature set in Table 5.3). In contrast, the argument identification performance varies much less across feature sets in this experiment, because the full semantic feedback always implicitly drives accurate argument identification. The increase in role classification performance across feature sets can be attributed both to a useful increase in representations used for SRL classification, and to the increased argument and predicate structure accuracy during both SRL training and testing. The relatively high level of performance given the lexical features alone in Table 5.3 reflects the repetitive character of the corpus from which our training and test sentences were drawn: Given full semantic feedback, considerable success in role assignment can be achieved based on the argument-role biases of the target nouns and the familiar verbs in our corpus of child-directed speech.

5.4.1 Loosening Feedback

Feedback	Pred	Arg	A0	A1	Role F1
Full Labels	0.94(0.02)	0.89(0.02)	0.85(0.02)	0.75(0.02)	0.64(0.02)
Set of Labels[4]	0.40(0.23)	0.62(0.14)	0.47(0.28)	0.38(0.17)	0.34(0.14)
Superset	0.35(0.20)	0.57(0.11)	0.46(0.27)	0.33(0.13)	0.29(0.11)
Superset + HMM Args	0.87(0.10)	0.88(0.01)	0.68(0.25)	0.54(0.16)	0.48(0.15)
Superset + HMM Args&Pred	0.79(0.02)	0.89(0.00)	0.73(0.23)	0.58(0.13)	0.51(0.14)
Superset + True Args	0.86(0.09)	0.92(0.01)	0.69(0.21)	0.61(0.13)	0.52(0.13)
Superset + True Args&Pred	0.97(0.00)	0.93(0.00)	0.68(0.19)	0.61(0.12)	0.52(0.11)
Random	0.31	0.46			

Table 5.4: Results when the amount of semantic feedback is decreased. Each value represents the mean over twenty training runs with shuffled sentence order, while the numbers in parenthesis are the standard deviations. Full label feedback provides true role feedback for each noun argument, which is unreasonable in the case of actual language learning. Set of Labels feedback only provides the set of true labels as feedback for each sentence, so the learner must pick a structure and label assignment from this set. Superset goes one step further and provides a super set of labels that includes the true labels, so the learner does not even know how many or which roles are mentioned in the sentence. With these ambiguous feedback schemes the classifiers are barely able to begin interpreting correctly, and with superset the argument and predicate accuracy is only slightly better than random. We introduce extra information through constraining the possible argument and predicate structures for each training example using bottom-up knowledge, either from an HMM based minimally supervised identifier or knowledge of true argument and predicate. Once extra information about even just argument identity is introduced, whether true arguments or the HMM identified arguments, the learner is able to make use of the superset feedback, and especially begin to identify agent and patient roles (A0 and A1), and predicate.

The full semantic feedback used in the previous experiments, while less informative than absolute gold knowledge of true arguments and predicates, is still an unreasonable amount of feedback to expect for a child first learning language. Often in the real learning case the learner only has available an understanding of the scene around her which may involve a number of possible objects, relations and semantic roles, and a sentence without any indication of the true argument labels for the sentence or even how many arguments are present.

We are able to mimic this level of feedback by modifying the constraining sets H_i and Y_i used in line 4 of algorithm 4. By loosening these sets we still provide feedback in terms of restricting the search space, but not an exact labeling.

We test two levels of reduced role feedback. The first level uses the true role labels that are present in the sentence, but does not indicate which words correspond to which role. In this case Y_i is just the set of all labellings that use exactly the true labels present, and H_i is constrained to be only those argument structures with the correct number of arguments. This feedback scheme represents a setting where the child knows the semantic relation involved, but either does not know the nouns in the sentence, or doesn't know whether the speaker means chase or flee (so can't fix role order). In our "She likes yellow flowers" example the feedback would be that there is an agent and a patient, but no indication of order.

Even this feedback scheme includes the number of true arguments in the sentence, so we can go a step further with a second level of feedback where for each sentence we supply a superset of the true labels for the learner to select a labeling. In this case Y_i includes the true labels, plus random other labels such that for every sentence there are 4 labels to choose from, no matter the number of true arguments. We are no longer constrained by the number of arguments, so we must search over all argument structures and role labellings that come from some subset of the feedback set Y_i. This case corresponds to the setting that the learner must select a possible interpretation of the sentence from the abundance of information provided by the world around them. For our 'yellow flowers' example the feedback would be a set of possible labels that include the correct agent and patient roles, but also two unrelated roles such as recipient or location, and no indication of how many are actually in the sentence.

As seen in table 5.4, the set and superset feedback schemes definitely degrade performance compared to full labellings. With superset feedback the learner is not able to get a good foothold to begin correctly identifying structure and interpreting sentences, so its argument and predicate

[4] In the original paper, [Connor et al., 2011], the unordered set feedback results were higher than reported here. Those results were most likely due to non-shuffled ordering of sentences during training.

identification accuracy is no better than random. This suggests that information about the number and identity of arguments might be a necessary constraint in learning to understand sentences.

5.4.2 Recovering Argument Knowledge

In a sense, there's something fundamentally unnatural about looking at semantic role labeling before the learner knows the meanings of *any* nouns. Considerable psycholinguistic evidence suggests that children do learn some nouns before they start to interpret multi-word sentences, and that this noun knowledge therefore is available to scaffold the beginnings of sentence interpretation. If we can incorporate this extra source of knowledge with the superset feedback then perhaps there will be enough information on repeated sentence training for the system to improve.

Again taking inspiration from the 'structure-mapping' account of syntactic bootstrapping (described in section 2.3), we incorporate a starting point by using the minimally-supervised argument identification developed in section 4.1. In that case nouns are identified based on a seed set of concrete nouns combined with the clustering provided by an unsupervised HMM. Once some nouns have been identified, the HMM states they are seen with are treated as potential argument states. Predicates are identified by finding non-argument content words that are seen often in sentences with a given number of arguments, and considered to be likely predicates that take that number of arguments.

We can use this bottom-up identification system to constrain the argument search in the our latent classifier training. During training, we restrict the true argument set (H_i in algorithm 4) such that only those structures that agree with the HMM identification are considered, and the best labeling from the superset of labels is selected for this structure. If we use both the argument and predicate identified by HMM to essentially fix the latent structure during training, the learning problem is then similar to the noisy argument identification BabySRL in section 4.1, except instead of true feedback for those roles the learner must select from superset of possible labels. Additionally we can only consider the arguments identified by the HMM, ignoring the predicate identified by noun counting heuristic. In this case the latent inference must select both the identity of the predicate and the argument labeling that is most consistent with previous training.

Table 5.4 shows that once we add the HMM bottom-up argument identification to the Superset feedback scheme the argument and predicate performance increases greatly (due to accuracy of the HMM argument identification). Note in Table 5.4 that bottom-up HMM argument identification is strong (0.88 F1 compared to 0.93 when trained with true arguments), and that this effective argument-identification in turn permits strong performance on verb identification. Thus our procedure for tagging some HMM classes as argument (noun) classes based on a seed set of concrete nouns, combined with ambiguous Superset semantic feedback that does not indicate the number identity of semantic arguments, yields enough information to begin learning to identify predicates (verbs) in input sentences.

Looking at the final role classification performance of the Superset+argument constraint training schemes, we see the Role F1 increases over both straight Superset and unordered Set of Labels feedback schemes. This increase is most dramatic for the more common A0 and A1 roles. This represents one possible implementation of the structure-mapping procedure for early syntactic bootstrapping. If we assume the learner can learn some nouns with no guidance from syntactic knowledge (represented by our seed nouns), that noun knowledge can be combined with distributional learning (represented by our HMM parser) to tag some word-classes as noun classes. Representing each sentence as containing some number of these nouns (HMM argument identification) then permits the latent SRL to begin learning to assign semantic roles to those nouns in sentences given highly ambiguous feedback, and also to use that ambiguous semantic feedback, combined with the constraints provided by the set of identified nouns in the sentence, to improve the latent syntactic representation, beginning to identify verbs in sentences.

This latent training method with ambiguous feedback works because it is seeking consistency in the features of the structures it sees. At the start of training, or when encountering a novel sentence with features not seen before, the latent inference will essentially choose a structure and labeling at random (since all structures will have the same score of 0, and ties are broken randomly). From this random labeling the classifier will increase connection strengths between lexical and structural features in the input sentence, and the (at first randomly) selected semantic role labels. Assuming that some number of random or quasi-random predictions are initially made, the learner can only improve if some set of feature weights increase above the others and begin to dominate predictions, both in the latent structure classifier and in the linked SRL classifier. This dominance can emerge only if there are structural features of sentences that frequently co-occur with frequent semantic roles.

A0 and A1 roles are learnable with this latent SRL learner both because of their frequency in the training data and their consistency with simple representation features that make use of the bottom-up information provided by the HMM argument identification. If "She likes yellow flowers." is encountered early during latent training, the feedback may be the superset {A0, A1, A4, AM-LOC}, where the true labels A0 and A1 are present along with two other random labels. With the accurate argument identification of 'she' and 'flowers' from the HMM, we focus on only those role labelings that use two of the four roles. When we look at a large enough number of different sentences such as "She kicks the ball" (true labels are A0, A1), "She writes in her book" (A0, A2), and "She sleeps" (A0), the most consistent labeling amongst the true and random labelings provided by Superset feedback is that both 'she' and the first of two arguments are more likely to be labeled as an A0. This consistent labeling is then propogated through the weights of the learner and used for future predictions and learning.

Starting with a reasonable identification of the noun arguments in each input sentence, the latent SRL trained with Superset feedback can use the consistency of certain roles' appearance with cues based on the identified arguments to boost their linking with specific words and structures. Without the identified arguments, the chance of randomly assigning the correct arguments and roles decreases dramatically and so the likelihood of encountering the correct interpretation enough for it to dominate disappears. By limiting the search space of structures, the set of features is also focused on those relevant to true structure and that may indicate true meaning.

5.5 Conclusion

In this chapter we showed that it is possible to train a semantic role classifier jointly with a simplified latent syntactic structure based solely on semantic feedback and simple linguistic constraints. Even with highly ambiguous semantic feedback, our system was able to identify arguments and predicates, and begin to interpret roles when primed with knowledge of a small set of nouns.

Of course in this chapter we used the high level semantic feedback to improve only part of the intermediate representation. The current system is still held back by noise from lower unsupervised parse level, so it is natural to attempt to jointly improve all levels of representation. This brings up some combinatorial challenges for implementation and feedback, but structural constraints can be introduced to keep this process manageable.

Chapter 6

Conclusions

In this thesis we demonstrated that it is possible to begin learning rules that map syntactic patterns to verb-argument semantics from real language input starting only with knowledge of a small number of nouns. Nouns are easy to identify in the scene when they are referred to by the speaker, and children's early vocabulary are dominated by them. By assuming that children then use this noun knowledge to form an "argument class", and that sentences convey some semantic relation between arguments, learning is able to proceed. The nature of child directed language allows for both accurate identification of arguments starting with concrete nouns, and the robust learning of semantic patterns from simple linear syntax of noun arguments.

Learning semantics cannot be solely handled by noun knowledge; some amount of feedback from the world is necessary to match sentence structure with meaning. While we showed that it is possible for background knowledge in terms of noun animacy may serve as a potential feedback signal, we also experimented with other mechanisms for the scene to provide ambiguous semantic feedback. The real world may convey the true meaning or interpretation of a sentence, but this meaning may not be clear to a naive listener, without the sentence providing some guidance itself. Our model was able to extract and learn helpful semantics and syntax jointly from ambiguous feedback when it was initialized with some knowledge of noun arguments.

To support these claims we developed a machine learning model that incorporates explicit psycholinguistic assumptions about a child's language abilities. Furthermore, we developed and demonstrated machine learning methods that successfully makes use of both incomplete knowledge of input and ambiguous feedback, mirroring the difficulties that children face when learning language. By exploring language learning the way children learn it, we may find more natural representations and methods for allowing a computational learner to both understand language, and better interact and deal with a noisy, ambiguous world.

6.1 Future Work

The long term vision for this project is to build a system that grows and improves itself as a child quickly is able to learn and handle more and more complex sentences. This work has demonstrated that it is possible for a learner to get a foothold in understanding sentences starting with simple representations and limited noun based knowledge. In the future we want to this system to serve as a starting point for bootstrapping more complicated representations. There are a number of avenues available to reach this goal.

Complex Arguments and Multiple Predicates More complicated sentences mean longer sentences with both multiple predicates and complex arguments (more than just a single noun). Handling multiple predicates represents a shift in basic assumptions, but in terms of machinery much of the learning we have already demonstrated can be brought to bear. Additionally, if we assume a separate argument identifier (for now not specifying where such knowledge would come from), then when we identify more complicated arguments it may be possible to collapse them down to a simpler form of a single head noun. Essentially an accurate latent predicate and argument identifier can begin identifying multiple likely predicates per sentence, and simplify complex sentences to a form that matches what our role classifier with simple representations expects and was trained on. The questions now are how far can this approach take us, how much of the patterns learned on simple, single predicate sentences can be passed on to simplified complex sentences, and are there other ramifications of the shift of assumptions from single noun arguments and single predicate sentences.

Generalize Syntax While the above direction accomplished moving to more complicated sentences without a change in the basic syntactic representation, a separate but related direction would be to generalize the latent syntax of our model. Instead of relying on a linear syntax of just nouns and verbs, we would need to incorporate more structure that covers embedding of clauses and arguments. Of course much of NLP and linguistics in general seek the hierarchical syntactic structures that humans may use, but here we are constrained both by what can be learnable from latent feedback and what structures are actually required by real data. In our current model the role classifier depends on essentially tree path features in our simple representation as is, just the trees are restricted to be strictly linear. This linear tree may serve as a starting point for structure, leading to smooth transitions from our original BabySRL, allowing the current model to truly be a foothold for bootstrapping.

Approximate Inference Of course with almost any change to the internal syntactic representation it will be necessary to revisit the inference procedure, especially in the joint learning setting. With short sentences and simple linear structure it was entirely possible to enumerate and score all possible structures for each sentence in a reasonable time, but this quickly becomes unfeasible with almost any sort of added complexity. In a move for both efficiency and cognitive plausibility, such structural inference can be made online, in a word by word manner, building representation as sentence is processed. This can lead to exploration of such psycholinguistically relevant phenomena as garden path and other sentence processing errors, while also further constraining possible syntactic structures to be considered.

Feature Extraction One question that has not been somewhat brushed aside in this thesis is where do the actual feature specifications come from. While we show that simple representations based on a linear order of nouns and verbs can suffice when beginning to learn, the actual features we extract from this representation, noun pattern and verb position, were manually specified to the classifier as being important. We do not try to argue that these exact feature specifications or encodings are biologically relevant, just that it is possible to use the information of the simple syntactic structure for learning. But if we allow the system to improve its own internal syntactic representation due to feedback and to handle growing complexity, it will also be necessary for it to extract new features from this representation. This may be accomplished by using general tree features such as path based features, of which the specific features we use can be viewed as one instantiation, but again this represents a form of Universal Grammar on our part, instilling this assumption into the model. It may be possible to grow such a feature extractor along with the internal representation, but this is even less clear an objective.

Training Order As we begin to incorporate growing complexity we will need to rely on more language input. As the corpus size grows, and in theory sentence complexity increases as the model "ages", it will become necessary to look at sentence training order. As recently explored with Self-Paced Learning [Kumar et al., 2010] or Curriculum Learning [Bengio et al., 2009], when training large, possibly non-convex models, improvement is seen when ordering the training from easy to hard examples. This makes even more intuitive sense with latent models because the initialization and early direction of what hidden values are learned prove pivotal for the final performance (as demonstrated by large variation in accuracy for the latent BabySRL experiments across different runs), once the ball gets rolling its hard to change its direction so its best to get it rolling the right direction early. In some sense the current training corpus represents almost entirely "easy examples", so it is hard to draw conclusions just yet on how helpful an improved ordering would be, but it may be interesting to see how an optimal sentence ordering from latent training perspective compares to the natural ordering that parents use with children.

We now have more data and computational resources available to us than ever before, yet children's abilities to learn language far outpace any other learning system currently available. By mimicking a child's ability to bootstrap itself, starting from natural assumptions about meaning and structure, we may one day be able to let loose computer systems over fields of Internet text and data, learning from noisy, unstructured and ambiguous text the same way a child handles its noisy, unstructured and ambiguous parents.

Appendix A

Word Lists

Table A.1 lists animate and inanimate nouns used to generate feedback in chapter 4.2. This list was created by identifying the animacy or inanimacy of the 100 most frequent nouns across all three children's data. Only 84 nouns were considered to have clear animacy or inanimacy.

Table A.2 lists Lexical Development Norm nouns used to seed Minimally Supervised HMM Argument Identification in chapter 4.1. We used lexical development norms [Dale and Fenson, 1996], selecting all words for things or people that were commonly produced by 20-month-olds (over 50% reported), and that appeared at least 5 times in our training data. Because this list is of words that children produce, it obviously represents a lower bound on the set of words that such a child should comprehend or recognize. This yielded 71 words, including words for common animals ('pig', 'kitty', 'puppy'), objects ('truck', 'banana', 'telephone'), people ('mommy', 'daddy'), and some pronouns ('me' and 'mine'). To this set we also added pronouns 'you' and 'I', as well as given names 'adam', 'eve' and 'sarah'.

Tables A.3,A.4,A.5 list Animate nouns for each child used to create animate test sets used in chapter 4. Each child's test sentences created independently, creating a sentence for all pairs of nouns. Included in the tables are frequency counts for occurences of the word in each child's training data, including role labels and when the word appears in two noun sentences if it more often appears first or second.

Word	Animate?	Frequency	Word	Animate?	Frequency
you	Yes	7920	it	No	2855
i	Yes	2441	what	No	2155
he	Yes	843	me	Yes	752
ya	Yes	748	she	Yes	657
we	Yes	651	can	No	512
her	Yes	446	him	Yes	298
who	Yes	263	back	No	225
ursula	Yes	162	daddy	Yes	161
fraser	Yes	134	baby	Yes	124
something	No	122	head	No	116
chair	No	114	lunch	No	113
mommy	Yes	110	bed	No	108
kent	Yes	103	book	No	103
today	No	100	way	No	99
time	No	95	cromer	Yes	95
watch	No	93	paper	No	89
sarah	Yes	88	pencil	No	85
papa	Yes	81	mouth	No	81
water	No	79	table	No	78
coffee	No	75	floor	No	74
drink	No	72	yourself	Yes	71
night	No	70	eve	Yes	70
box	No	69	ride	No	69
milk	No	69	minute	No	66
anything	No	63	mummy	Yes	62
cream	No	61	juice	No	60
adam	Yes	59	nose	No	59
house	No	59	yesterday	No	59
story	No	58	nana	Yes	58
school	No	56	hair	No	56
day	No	54	bite	No	52
beach	No	52	fingers	No	51
cheese	No	51	song	No	51
room	No	51	bath	No	50
piece	No	49	shoes	No	49
hand	No	49	cha	Yes	48
finger	No	48	money	No	47
home	No	46	stool	No	43
courtney	Yes	43	cup	No	43
everything	No	43	sandwich	No	42
car	No	42	tape	No	42
while	No	42			
Total Animate	26	16488	Total Inanimate	57	9314

Table A.1: Animacy Training Noun list.

Word	Cumulative Distribution	Word	Cumulative Distribution
you	24.59%	shoe	40.75%
i	32.17%	cookie	40.82%
me	34.51%	kitty	40.89%
daddy	35.01%	boy	40.95%
baby	35.39%	light	41.01%
chair	35.75%	train	41.07%
mommy	36.09%	bunny	41.12%
bed	36.42%	mine	41.17%
book	36.74%	bird	41.22%
sarah	37.02%	tummy	41.26%
mouth	37.27%	bottle	41.31%
eve	37.48%	apple	41.35%
milk	37.70%	boat	41.39%
dog	37.91%	telephone	41.43%
juice	38.09%	toe	41.47%
adam	38.28%	bear	41.50%
nose	38.46%	airplane	41.54%
hair	38.63%	blanket	41.57%
cheese	38.79%	ear	41.60%
hand	38.94%	tooth	41.63%
cup	39.08%	cow	41.66%
car	39.21%	cat	41.69%
door	39.33%	bicycle	41.71%
hat	39.45%	pig	41.73%
toy	39.57%	banana	41.75%
spoon	39.69%	pool	41.76%
truck	39.79%	balloon	41.78%
outside	39.89%	duck	41.79%
diaper	39.99%	clock	41.81%
tree	40.08%	soap	41.82%
eye	40.18%	rain	41.83%
button	40.27%	block	41.84%
ball	40.36%	puppy	41.85%
foot	40.44%	keys	41.85%
cracker	40.52%	sock	41.86%
doll	40.61%	bubbles	41.86%
horse	40.68%	flower	41.87%

Table A.2: Lexical Development Norms Noun List, with cumulative frequency distribution over training data.

		Role Freq.			Two Arg. Freq.	
Word	Frequency	A0	A1	Other	1st	2nd
you	2305	1558	347	400	762	312
ursula	150	20	34	96	13	23
cromer	66	14	18	34	9	7
who	54	34	16	4	21	6
adam	53	28	14	11	21	10
mommy	47	22	16	9	9	5
daddy	31	17	7	7	5	3
cowboy	20	11	7	2	9	3
somebody	12	4	5	3	1	2
baby	11	4	5	2	4	1
paul	10	3	5	2	0	2
doctor	9	1	3	5	1	0
robin	9	3	2	4	1	0
anybody	7	0	4	3	1	2
man	6	2	4	0	0	2
david	6	1	1	4	0	2
boy	6	3	2	1	1	1
people	5	5	0	0	0	0
cowboys	4	3	1	0	0	1
joshua	4	1	1	2	1	0
nobody	4	3	0	1	2	0
fireman	4	1	1	2	1	1
bengy	3	1	0	2	0	0

Table A.3: Adam Animate Nouns

		Role Freq.			Two Arg. Freq.	
Word	Frequency	A0	A1	Other	1st	2nd
you	3479	2227	620	632	1162	458
i	1243	883	69	291	462	32
ya	724	465	100	159	199	148
who	165	99	39	27	84	11
daddy	121	53	33	35	34	23
baby	82	21	44	17	12	26
kent	80	21	14	45	24	12
mummy	60	35	10	15	22	5
nana	56	11	12	33	7	7
sarah	30	11	5	14	2	4
gloria	28	3	8	17	1	5
courtney	27	0	6	21	5	5
girl	25	6	7	12	3	4
babies	22	1	14	7	6	9
mike	22	5	12	5	5	8
mommy	21	14	3	4	3	1
donna	21	8	8	5	4	3
blanche	18	8	6	4	2	5
kids	18	8	7	3	4	3
mother	18	7	6	5	5	3
mumma	16	7	6	3	1	5
boy	12	4	5	3	2	3
man	11	5	5	1	3	1
mama	11	3	5	3	3	1
richard	11	0	6	5	0	1
grampy	10	2	5	3	1	2
father	10	3	3	4	1	4
sandra	9	5	1	3	2	2
momma	8	3	4	1	1	3
jo_ann	8	2	1	5	0	1
brother	8	2	6	0	0	2
arthur	8	3	5	0	3	1
michael	8	2	4	2	0	4
aunt_dot	8	0	3	5	0	6
doctor	7	2	2	3	2	3
people	7	3	2	2	2	3
teppy	7	3	2	2	2	1
ann_marie	7	2	1	4	2	1
girls	6	2	3	1	1	0
everybody	6	5	1	0	3	0
chantilly	6	3	2	1	2	3
auntie	5	1	1	3	0	1

Table A.4: Sarah Animate Nouns

		Role Freq.			Two Arg. Freq.	
Word	Frequency	A0	A1	Other	1st	2nd
you	2077	1425	301	351	738	234
fraser	122	39	28	55	26	22
papa	81	31	16	34	22	10
eve	69	31	19	19	26	7
sarah	51	16	14	21	9	9
mommy	38	21	6	11	17	1
who	28	17	6	5	12	0
becky	28	6	7	15	4	4
cromer	25	10	9	6	8	2
man	21	12	7	2	10	3
mom	16	11	1	4	9	1
yourself	7	0	2	5	0	5
mama	6	2	0	4	3	0
baby	5	0	3	2	0	1
jerry	5	0	4	1	0	0
lady	5	4	1	0	2	0
jack	5	0	4	1	0	0
sheila	5	0	0	5	0	2

Table A.5: Eve Animate Nouns

References

J. Rosen A. Carlson, C. Cumby and D. Roth. The snow learning architecture. Technical Report UIUCDCS-R-99-2101, UIUC Computer Science Department, 5 1999. URL `http://l2r.cs.uiuc.edu/~danr/Papers/CCRR99.pdf`.

O. Abend and A. Rappoport. Fully unsupervised core-adjunct argument classification. In *ACL*, 2010.

O. Abend, R. Reichart, and A. Rappoport. Unsupervised argument identification for semantic role labeling. In *ACL*, 2009.

J. Aissen. Markedness and subject choice in optimality theory. *Natural Language and Linguistic Theory*, 17:673–711, 1999.

Afra Alishahi and Suzanne Stevenson. A Computational Model of Early Argument Structure Acquisition. *Cognitive Science: A Multidisciplinary Journal*, 32(5):789–834, July 2008. ISSN 0364-0213. doi: 10.1080/03640210801929287. URL `http://doi.wiley.com/10.1080/03640210801929287`.

Afra Alishahi and Suzanne Stevenson. A computational model of learning semantic roles from child-directed language. *Language and Cognitive Processes*, 25(1):50–93, January 2010. ISSN 0169-0965. doi: 10.1080/01690960902840279. URL `http://www.informaworld.com/openurl?genre=article\&doi=10.1080/01690960902840279\&magic=crossref||D404A21C5BB053405B1A640AFFD44AE3`.

J Allen. Probabilistic constraints in acquisition. In *Proceedings of the GALA*, 1997.

R. Baillargeon, D. Wu, S. Yuan, J. Li, and Y. Luo. Young infants expectations about self-propelled objects. In B. Hood and L. Santos, editors, *The origins of object knowledge*. Oxford University Press, Oxford, (in press).

Colin Bannard, Elena Lieven, and Michael Tomasello. Modeling children's early grammatical knowledge. *Proceedings of the National Academy of Sciences of the United States of America*, 106(41):17284–9, October 2009. ISSN 1091-6490. doi: 10.1073/pnas.0905638106. URL `http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=2765208\&tool=pmcentrez\&rendertype=abstract`.

L. E. Baum. An inequality and an associated maximization technique in statistical estimation of probabilistic functions of a markov process. *Inequalities*, 3:1-8, 1972.

M.J. Beal. *Variational Algorithms for Approximate Bayesian Inference*. PhD thesis, Gatsby Computational Neuroscience Unit, University College London, 2003.

Yoshua Bengio, Jerome Louradour, Ronan Collobert, and Jason Weston. Curriculum learning. In *ICML*, 2009.

Taylor Berg-Kirkpatrick, Alexandre Bouchard-Cote, John DeNero, and Dan Klein. Painless unsupervised learning with features. In *NAACL*, 2010.

B. H. Bloom. Space/Time trade-offs in Hash Coding with allowable errors. *Communications of the ACM*, 13(7):422–426, 1970.

L. Bloom. *One word at a time: The use of single-word utterances before syntax*. Mouton, The Hague, 1973.

A. Blum and T. Mitchell. Combining labeled and unlabeled data with co-training. In *Proc. of the Annual ACM Workshop on Computational Learning Theory (COLT)*, pages 92–100, 1998.

M.R. Brent and J.M. Siskind. The role of exposure to isolated words in early vocabulary development. *Cognition*, 81:31–44, 2001.

J. Bresnan. *The mental representation of grammatical relations*. MIT Press, Cambridge MA, 1982.

E. Brill. Unsupervised learning of disambiguation rules for part of speech tagging. In *Natural Language Processing Using Very Large Corpora*. Kluwer Academic Press, 1997.

P. Brown, V. Della Pietra, P. deSouza, J. Lai, and R. Mercer. Class-based n-gram models of natural language. *Computational Linguistics*, 18(4):467–479, 1992.

R. Brown. *A First Language*. Harvard University Press, Cambridge, MA, 1973.

G. Carlson. Thematic roles and the individuation of events. In S. D. Rothstein, editor, *Events and Grammar*, pages 35–51. Kluwer, Dordrecht, 1998.

X. Carreras and L. Màrquez. Introduction to the CoNLL-2004 shared tasks: Semantic role labeling. In *Proceedings of CoNLL-2004*, pages 89–97. Boston, MA, USA, 2004.

X. Carreras and L. Màrquez. Introduction to the CoNLL-2005 shared task: Semantic role labeling. In *Proceedings of the Ninth Conference on Computational Natural Language Learning (CoNLL-2005)*, pages 152–164, Ann Arbor, Michigan, June 2005. Association for Computational Linguistics. URL `http://www.aclweb.org/anthology/W/W05/W05-0620`.

Franklin Chang, Gary S Dell, and Kathryn Bock. Becoming syntactic. *Psychological review*, 113 (2):234–72, April 2006. ISSN 0033-295X. doi: 10.1037/0033-295X.113.2.234. URL `http://www.ncbi.nlm.nih.gov/pubmed/16637761`.

M. Chang, D. Goldwasser, D. Roth, and V. Srikumar. Discriminative learning over constrained latent representations. In *Proc. of the Annual Meeting of the North American Association of Computational Linguistics (NAACL)*, 6 2010. URL `http://l2r.cs.uiuc.edu/~danr/Papers/CGRS10.pdf`.

R. S. Chapman and L. L. Kohn. Comprehension strategies in two- and three-year-olds: Animate agents or probable events? *Journal of Speech and Hearing Research*, 21:746–761, 1978.

E. Charniak. Statistical parsing with a context-free grammar and word statistics. In *Proceedings of the National Conference on Artificial Intelligence (AAAI)*, 1997.

C. Cherry and C. Quirk. Discriminative, syntactic language modeling through latent svms. In *Proc. of the Eighth Conference of AMTA*, Honolulu, Hawaii, October 2008.

C. Christodoulopoulos, S. Goldwater, and M. Steedman. Two decades of unsupervised pos induction: How far have we come? In *Proc. of the Conference on Empirical Methods for Natural Language Processing (EMNLP)*, 2010.

A. Clark. Combining distributional and morphological information for part of speech induction. In *EACL*, 2003.

E.V. Clark. Awwareness of language: Some evidence from what children say and do. In R. J. A. Sinclair and W. Levelt, editors, *The child's conception of language*. Springer Verlag, Berlin, 1978.

M. Collins. Discriminative training methods for hidden Markov models: Theory and experiments with perceptron algorithms. In *EMNLP*, 2002.

M. Connor, Y. Gertner, C. Fisher, and D. Roth. Baby srl: Modeling early language acquisition. In *Proc. of the Annual Conference on Computational Natural Language Learning (CoNLL)*, 8 2008. URL `http://l2r.cs.uiuc.edu/~danr/Papers/CGFR08.pdf`.

M. Connor, Y. Gertner, C. Fisher, and D. Roth. Minimally supervised model of early language acquisition. In *Proc. of the Annual Conference on Computational Natural Language Learning (CoNLL)*, 6 2009. URL `http://l2r.cs.uiuc.edu/~danr/Papers/CGFR09.pdf`.

M. Connor, Y. Gertner, C. Fisher, and D. Roth. Starting from scratch in semantic role labeling. In *Proc. of the Annual Meeting of the Association of Computational Linguistics (ACL)*, Uppsala, Sweden, 7 2010. Association for Computational Linguistics. URL `http://l2r.cs.uiuc.edu/~danr/Papers/CGFR10.pdf`.

Michael Connor, Cynthia Fisher, and Dan Roth. Online latent structure training for language acquisition. In *Proc. of the International Joint Conference on Artificial Intelligence (IJCAI)*, 7 2011. URL `http://cogcomp.cs.illinois.edu/papers/ConnorFiRo11.pdf`.

R. Corrigan. Children's identification of actors and patients in prototypical and nonprototypical sentence types. *Cognitive Development*, 3:285–297, 1988.

K. Crammer and Y. Singer. On the algorithmic implementation of multiclass kernel-based vector machines. *Journal of Machine Learning Research*, 3, 2001.

K. Crammer and Y. Singer. On the learnability and design of output codes for multiclass problems. *Machine Learning*, 47, 2002.

K. Crammer, O. Dekel, J. Keshet, S. Shalev-Shwartz, and Y. Singer. Online passive-aggressive algorithms. *Journal of Machine Learning Research*, 2006.

P.S. Dale and L. Fenson. Lexical development norms for young children. *Behavior Research Methods, Instruments, & Computers*, 28:125–127, 1996.

H. Daumé and D. Marcu. Learning as search optimization: approximate large margin methods for structured prediction. In *ICML*, pages 169–176, New York, NY, USA, 2005. ACM. ISBN 1-59593-180-5. doi: http://doi.acm.org/10.1145/1102351.1102373.

M. Demetras, K. Post, and C. Snow. Feedback to first-language learners. *Journal of Child Language*, 13:275–292, 1986.

K. Demuth, J. Culbertson, and J. Alter. Word-minimality, epenthesis, and coda licensing in the acquisition of english. *Language & Speech*, 49:137–174, 2006.

R. Desai. Bootstrapping in miniature language acquisition. *Cognitive Systems Research*, 3, 2002.

R. Desai. A model of frame and verb compliance in language acquisition. *Neurocomputing*, 70, 2007.

T. G. Dietterich and G. Bakiri. Solving multiclass learning problems via error-correcting output codes. *Journal of Artificial Intelligence Research*, 2:263–286, 1995.

D. Dowty. Thematic proto-roles and argument selection. *Language*, 67:547–619, 1991.

Jeffrey L. Elman. Distributed representations, simple recurrent networks, and grammatical structure. *Machine Learning*, 7:195–225, 1991.

David Elworthy. Does Baum-Welch re-estimation improve taggers? In *Proc. of ACL Conference on Applied Natural Language Processing*, pages 53–58, 1994.

C. Fellbaum. *WordNet: An Electronic Lexical Database*. MIT Press, 1998.

P. Felzenszwalb, D. Mcallester, and D. Ramanan. A discriminatively trained, multiscale, deformable part model. In *IEEE International Conference on Computer Vision and Pattern Recognition (CVPR) Anchorage, Alaska, June 2008.*, June 2008. URL `http://www.ics.uci.edu/~dramanan/papers/latent.pdf`.

F. Ferreira. The misinterpretation of noncanonical sentences. *Cognitive Psychology*, 47:164–203, 2003.

C. Fisher. Structural limits on verb mapping: The role of analogy in children's interpretation of sentences. *Cognitive Psychology*, 31:41–81, 1996.

C. Fisher, Y. Gertner, R. Scott, and S. Yuan. Syntactic bootstrapping. *Wiley Interdisciplinary Reviews: Cognitive Science*, 2010.

Y. Freund and R. Schapire. Large margin classification using the Perceptron algorithm. In *Proc. of the Annual ACM Workshop on Computational Learning Theory (COLT)*, pages 209–217, 1998.

Jianfeng Gao and Mark Johnson. A comparison of bayesian estimators for unsupervised hidden markov model pos taggers. In *Proceedings of EMNLP-2008*, pages 344–352, 2008.

Roger Garside, Geoffrey Leech, and Geoffrey Sampson, editors. *The Computational Analysis of English – A Corpus Based Approach*. Longman, 1987.

D. Gentner. Why verbs are hard to learn. In K. Hirsh-Pasek and R. Golinkoff, editors, *Action meets word: How children learn verbs*, pages 544–564. Oxford University Press, 2006.

D. Gentner and L. Boroditsky. Individuation, relativity and early word learning. In M. Bowerman and S. C. Levinson, editors, *Language acquisition and conceptual development*, pages 215–256. Cambridge University Press, New York, 2001.

Y. Gertner and C. Fisher. Predicted errors in early verb learning. In *31st Annual Boston University Conference on Language Development*, 2006.

Y. Gertner, C. Fisher, and J. Eisengart. Learning words and rules: Abstract knowledge of word order in early sentence comprehension. *Psychological Science*, 17, 2006.

D. Gildea and M. Palmer. The necessity of parsing for predicate argument recognition. In *ACL*, pages 239–246, 2002.

J. Gillette, H. Gleitman, L. R. Gleitman, and A. Lederer. Human simulations of vocabulary learning. *Cognition*, 73:135–176, 1999.

Sharon Goldwater and Tom Griffiths. A fully bayesian approach to unsupervised part-of-speech tagging. In *ACL*, pages 744–751, Prague, Czech Republic, June 2007. Association for Computational Linguistics. URL `http://www.aclweb.org/anthology/P/P07/P07-1094`.

R. Gomez and L. Gerken. Artificial grammar learning by 1-year-olds leads to specific and abstract knowledge. *Cognition*, 70:109–135, 1999.

Joao Graca, Kuzman Ganchev, Ben Taskar, and Fernando Pereira. Posterior vs parameter sparsity in latent variable models. In *Advances in Neural Information Processing Systems 22*, pages 664–672, 2009.

T. Grenager and C. Manning. Unsupervised discovery of a statistical verb lexicon. In *Proc. of the Conference on Empirical Methods for Natural Language Processing (EMNLP)*, 2006.

A. Grove and D. Roth. Linear concepts and hidden variables. *Machine Learning*, 42(1/2):123–141, 2001. URL `http://l2r.cs.uiuc.edu/~danr/Papers/hiddenJ.pdf`.

A. Haghighi and D. Klein. Prototype-driven learning for sequence models. In *Proc. of HTL-NAACL*, 2006.

Jan Hajič, Massimiliano Ciaramita, Richard Johansson, Daisuke Kawahara, Maria Antònia Martí, Lluís Màrquez, Adam Meyers, Joakim Nivre, Sebastian Padó, Jan Štěpánek, Pavel Straňák, Mihai Surdeanu, Nianwen Xue, and Yi Zhang. The conll-2009 shared task: Syntactic and semantic dependencies in multiple languages. In *Proceedings of the Thirteenth Conference on Computational Natural Language Learning (CoNLL 2009): Shared Task*, 2009.

S. Haykin. *Neural Networks: A Comprehensive Foundation, 2nd Edition*. Prentice-Hall, 1999.

Jean-Remy Hochmann, Ansgar D. Endress, and Jacques Mehler. Word frequency as a cue for identifying function words in infancy. *Cognition*, 115:444–457, 2010.

Fei Huang and Alexander Yates. Distributional representations for handling sparsity in supervised sequence-labeling. In *ACL*, 2009.

R. Jackendoff. *Semantic Structures*. MIT Press, Cambridge, MA, 1990.

Richard Johansson and Pierre Nugues. Dependency-based syntactic–semantic analysis with propbank and nombank. In *CoNLL 2008: Proceedings of the Twelfth Conference on Computational Natural Language Learning*, pages 183–187, Manchester, England, August 2008. Coling 2008 Organizing Committee. URL `http://www.aclweb.org/anthology/W08-2123`.

M. Johnson, K. Demuth, M. C. Frank, and B. Jones. Synergies in learning words and their meanings. In *Neural Information Processing Systems, 23*, 2010.

Mark Johnson. Why doesn't EM find good HMM POS-taggers? In *Proceedings of the 2007 Joint Conference of EMNLP-CoNLL*, pages 296–305, 2007. URL http://www.aclweb.org/anthology/D/D07/D07-1031.

Jun'ichi Kazama and Kentaro Torisawa. A new perceptron algorithm for sequence labeling with non-local features. In *Proceedings of the 2007 Joint Conference of EMNLP-CoNLL*, pages 315–324, 2007. URL http://www.aclweb.org/anthology/D/D07/D07-1033.

M. Kearns and R. Schapire. Efficient distribution-free learning of probabilistic concepts. *Journal of Computer and System Sciences*, 48:464–497, 1994.

M.H. Kelly. Using sound to solve syntactic problems: The role of phonology in grammatical category assignments. *Psychological Review*, 99:349–364, 1992.

P. Kingsbury and M. Palmer. From Treebank to PropBank. In *Proceedings of LREC-2002*, Spain, 2002.

Karin Kipper, Hoa Trang Dang, and Martha Palmer. Class-based construction of a verb lexicon. In *Proceedings of the National Conference on Artificial Intelligence (AAAI)*, Austin, TX, 2000. AAAI.

T. Koo, X. Carreras, and M. Collins. Simple semi-supervised dependency parsing. In *ACL*, 2008.

P. Kumar, B. Packer, and D. Koller. Self-paced learning for latent variable models. In *Advances in Neural Information Processing Systems (NIPS 2010)*, 2010.

J. Lafferty, A. McCallum, and F. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *ICML*, 2001.

B. Landau and L Gleitman. *Language and experience*. Harvard University Press, Cambridge, MA, 1985.

J. Lang and M. Lapata. Unsupervised induction of semantic roles. In *Human Language Technologies 2010: The Conference of the North American Chapter of the Association for Computational Linguistics; Proceedings of the Main Conference*, 2010.

Y. LeCun, L. Bottou, G. Orr, and K. Muller. Efficient backprop. In G. Orr and K. Muller, editors, *Neural Networks: Tricks of the trade*. Springer, 1998.

Yoong Keok Lee, Aria Haghighi, and Regina Barzilay. Simple type-level unsupervised pos tagging. In *Proc. of the Conference on Empirical Methods for Natural Language Processing (EMNLP)*, 2010.

H. Lempert. Animacy constraints on preschool children's acquisition of syntax. *Child Development*, 60:237–245, 1989.

B. Levin. *English Verb Classes and Alternations: A Preliminary Investigation*. University of Chicago Press, Chicago, IL, 1993.

J. Lidz, H. Gleitman, and L. R. Gleitman. Understanding how input matters: verb learning and the footprint of universal grammar. *Cognition*, 87:151–178, 2003.

N. Littlestone. Learning quickly when irrelevant attributes abound: A new linear-threshold algorithm. *Machine Learning*, 2(4):285–318, 1988.

David MacKay. Ensemble learning for hidden markov models. *Technical report, Cavendish Laboratory, Cambridge*, 1997.

B. MacWhinney. *The CHILDES project: Tools for analyzing talk. Third Edition*. Lawrence Elrbaum Associates, Mahwah, NJ, 2000.

M. P. Marcus, B. Santorini, and M. Marcinkiewicz. Building a large annotated corpus of English: The Penn Treebank. *Computational Linguistics*, 19(2):313–330, June 1993.

L. Màrquez, X. Carreras, K. Litkowski, and S. Stevenson. Semantic role labeling: An introduction to the special issue. *Computational Linguistics*, 34:145–159, 2008.

Marina Meilă. Comparing clusterings. Technical Report 418, University of Washington Statistics Department, 2002.

Bernard Merialdo. Tagging text with a probabilistic model. *Computational Linguistics*, 20(2): 155–172, 1994.

P. Merlo and S. Stevenson. Automatic verb classification based on statistical distributions of argument structure. *Computational Linguistics*, 27:373–408, 2001.

Marvin Minsky and Seymour Papert. *Perceptrons: An Introduction to Computational Geometry*. The MIT Press, 1969.

T. Mintz. Frequent frames as a cue for grammatical categories in child directed speech. *Cognition*, 90:91–117, 2003.

P. Monaghan, N. Chater, and M.H. Christiansen. The differential role of phonological and distributional cues in grammatical categorisation. *Cognition*, 96:143–182, 2005.

Taesun Moon, Katrin Erk, and Jason Baldridge. Crouching dirichlet, hidden markov model: Unsupervised POS tagging with context local tag generation. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, 2010.

L. R. Naigles. Children use syntax to learn verb meanings. *Journal of Child Language*, 17:357–374, 1990.

K. Nelson. Structure and strategy in learning to talk. *Monographs of the Society for Research in Child Development*, 38, 1973.

A. Novikoff. On convergence proofs for perceptrons. In *Proceeding of the Symposium on the Mathematical Theory of Automata*, volume 12, pages 615–622, 1963.

D. Roth P. Koomen, V. Punyakanok and W. Yih. Generalized inference with multiple semantic role labeling systems shared task paper. In Ido Dagan and Dan Gildea, editors, *Proc. of the Annual Conference on Computational Natural Language Learning (CoNLL)*, pages 181–184, 2005. URL `http://l2r.cs.uiuc.edu/~danr/Papers/PunyakanokRoYi05a.pdf`.

M. Palmer, D. Gildea, and P. Kingsbury. The Proposition Bank: An Annotated Corpus of Semantic Roles. *Computational Linguistics*, 31(1):71–106, March 2005. URL `http://dx.doi.org/10.1162/0891201053630264`.

S. Pinker. *Language learnability and language development*. Harvard University Press, Cambridge, MA, 1984.

S. Pinker. *Learnability and Cognition*. Cambridge: MIT Press, 1989.

V. Punyakanok, D. Roth, and W. Yih. The necessity of syntactic parsing for semantic role labeling. In *Proc. of the International Joint Conference on Artificial Intelligence (IJCAI)*, pages 1117–1123, 2005a. URL `http://l2r.cs.uiuc.edu/~danr/Papers/PunyakanokRoYi05.pdf`.

V. Punyakanok, D. Roth, W. Yih, and D. Zimak. Learning and inference over constrained output. In *Proc. of the International Joint Conference on Artificial Intelligence (IJCAI)*, pages 1124–1129, 2005b. URL `http://l2r.cs.uiuc.edu/~danr/Papers/PRYZ05.pdf`.

V. Punyakanok, D. Roth, and W. Yih. The importance of syntactic parsing and inference in semantic role labeling. *Computational Linguistics*, 34(2), 2008. URL `http://l2r.cs.uiuc.edu/~danr/Papers/PunyakanokRoYi07.pdf`.

L. R. Rabiner. A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–285, 1989.

L. Ratinov and D. Roth. Design challenges and misconceptions in named entity recognition. In *Proc. of the Annual Conference on Computational Natural Language Learning (CoNLL)*, 6 2009. URL `http://l2r.cs.uiuc.edu/~danr/Papers/RatinovRo09.pdf`.

Sujith Ravi and Kevin Knight. Minimized models for unsupervised part-of-speech tagging. In *Proceedings of the Joint Conferenceof the 47th Annual Meeting of the Association for Computational Linguistics and the 4th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing (ACL-IJCNLP)*, 2009.

R. Rifkin and A. Klautau. In defense of on-vs-all classification. *Journal of Machine Learning Research*, 5, 2004.

D. Roland, F. Dick, and J. L. Elman. Frequency of basic english grammatical structures: A corpus analysis. *Journal of Memory and Language*, 57:348–379, 2007.

F. Rosenblatt. The perceptron: A probabilistic model for information storage and organization in the brain. *Psych. Rev.*, 65:386–407, 1958. (Reprinted in Neurocomputing (MIT Press, 1988).).

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

S. Schulte im Walde. Experiments on the choice of features for learning verb classes. In *EACL*, 2003.

Yoav Seginer. Fast unsupervised incremental parsing. In *ACL*, pages 384–391, Prague, Czech Republic, June 2007. Association for Computational Linguistics. URL `http://www.aclweb.org/anthology/P/P07/P07-1049`.

Dan Shen and Mirella Lapata. Using semantic roles to improve question answering. In *Proceedings of the 2007 Joint Conference of EMNLP-CoNLL*, pages 12–21, 2007. URL http://www.aclweb.org/anthology/D/D07/D07-1002.

Rushen Shi, James L. Morgan, and Paul Allopenna. Phonological and acoustic bases for earliest grammatical category assignment: a cross-linguistic perspective. *Journal of Child Language*, 25(01):169–201, 1998. doi: 10.1017/S0305000997003395. URL http://journals.cambridge.org/action/displayAbstract?fromPage=online&aid=36873&fulltextType=RA&fileId=S0305000997003395.

Rushen Shi, Janet F. Werker, and James L. Morgan. Newborn infants' sensitivity to perceptual cues to lexical and grammatical words. *Cognition*, 72(2):B11 – B21, 1999. ISSN 0010-0277. doi: DOI:10.1016/S0010-0277(99)00047-5. URL http://www.sciencedirect.com/science/article/B6T24-3XM2T3P-6/2/9ed3deb83dcb7b96c91e110065fc1563.

L.B. Smith and C. Yu. Infants rapidly learn word-referent mappings via cross-situational statistics. *Cognition*, 106:1558–1568, 2008.

N. Smith and J. Eisner. Contrastive estimation: Training log-linear models on unlabeled data. In *ACL*, 2005.

S. Stevenson and E. Joanis. Semi-supervised verb class discovery using noisy features. In *Proc. of the Annual Conference on Computational Natural Language Learning (CoNLL)*, 2003.

S. Stevenson and P. Merlo. Automatic verb classification using distribution of grammatical features. In *EACL*, 1999.

M. Surdeanu, S. Harabagiu, J. Williams, and P. Aarseth. Using predicate-argument structures for information extraction. In *Proceedings of ACL 2003*, pages 8–15, 2003.

Mihai Surdeanu, Richard Johansson, Adam Meyers, Lluís Màrquez, and Joakim Nivre. The conll 2008 shared task on joint parsing of syntactic and semantic dependencies. In *CoNLL 2008: Proceedings of the Twelfth Conference on Computational Natural Language Learning*, 2008.

R. Swier and S. Stevenson. Unsupervised semantic role labeling. In *Proc. of the Conference on Empirical Methods for Natural Language Processing (EMNLP)*, 2004.

R. Swier and S. Stevenson. Exploiting a verb lexicon in automatic semantic role labeling. In *Proc. of the Conference on Empirical Methods for Natural Language Processing (EMNLP)*, 2005.

M. Tomasello. *First verbs: A case study of grammatical development*. Cambridge University Press, Cambridge, UK, 1992.

M. Tomasello. Do young children have adult syntactic competence? *Cognition*, 74:209–253, 2000.

M. Tomasello. *Constructing a language: A Usage-Based theory of language acquisition*. Harvard University Press, 2003.

H. Tourigny. Exploiting systematicity: A connectionist model of bootstrapping in language acquisition. M. S. thesis, Institute for Logic, Language and Computation, Univeriteit van Amsterdam, 2010.

Kiristina Toutanova and Mark Johnson. A bayesian lda-based model for semi-supervised part-of-speech tagging. In *Proceedings of NIPS*, 2007.

I. Tsochantaridis, T. Hofmann, T. Joachims, and Y. Altun. Support vector machine learning for interdependent and structured output spaces. In *ICML*, 2004.

Joseph Turian, Lev Ratinov, and Yoshua Bengio. Word representations: A simple and general method for semi-supervised learning. In *Proc. of the Annual Meeting of the Association of Computational Linguistics (ACL)*, 2010. URL `http://cogcomp.cs.illinois.edu/papers/TurianRaBe2010.pdf`.

V. Vapnik. *Statistical Learning Theory*. Springer, 1998.

H. Waterfall, B. Sandbank, L. Onnis, and S. Edelman. An empirical generative framework for computational modeling of language acquisition. *Journal of Child Language*, 37:671–703, 2010.

S. R. Waxman and A. Booth. Seeing pink elephants: Fourteen-month-olds's interpretations of novel nouns and adjectives. *Cognitive Psychology*, 43:217–242, 2001.

D. Wu and P. Fung. Semantic roles for smt: A hybrid two-pass model. In *Proceedings of North American Annual Meeting of the Association for Computational Linguistics HLT 2009: Short Papers*, 2009.

N. Xue and M. Palmer. Calibrating features for semantic role labeling. In *Proceedings of the EMNLP-2004*, pages 88–94, Barcelona, Spain, 2004.

D. Yarowsky. Unsupervised woed sense disambiguation rivaling supervied methods. In *Proceedings of ACL-95*, 1995.

C. Yu and T. Joachims. Learning structural svms with latent variables. In *ICML*, 2009.

S. Yuan and C. Fisher. "really? she blicked the baby?": Two-year-olds learn combinatorial facts about verbs by listening. *Psychological Science*, 20:619–626, 2009.

S. Yuan, C. Fisher, Y. Gertner, and J. Snedeker. Participants are more than physical bodies: 21-month-olds assign relational meaning to novel transitive verbs. In *Biennial Meeting of the Society for Research in Child Development*, Boston, MA, 2007.

Hai Zhao, Wenliang Chen, Jun'ichi Kazama, Kiyotaka Uchimoto, and Knetaro Torisawa. Multilingual dependency learning: Exploiting rich features for tagging syntactic and semantic dependencies. In *Proceedings of the Thirteenth Conference on Computational Natural Language Learning (CoNLL 2009): Shared Task*, 2009.