

Online Latent Structure Training for Language Acquisition

Michael Connor
University of Illinois
connor2@illinois.edu

Cynthia Fisher
University of Illinois
cfisher@cyrus.psych.uiuc.edu

Dan Roth
University of Illinois
danr@illinois.edu

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. Where do children learning their first languages begin in solving this problem? Even assuming children can derive a rough meaning for the sentence from the situation, how do they begin to map this meaning to the structure and the structure to the form of the sentence? In this paper we use feedback from a semantic role labeling (SRL) task to improve the intermediate syntactic representations that feed the SRL. We accomplish this by training an intermediate classifier using signals derived from latent structure optimization techniques. By using a separate classifier to predict internal structure we see benefits due to knowledge embedded in the classifier's feature representation. This extra structure allows the system to begin to learn using weaker, more plausible semantic feedback.

1 Introduction

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 pre-defined 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.

In this work 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.

1.1 Related Work

Previous computational experiments with a system for automatic semantic role labeling (BabySRL: [Connor *et al.*, 2008]) showed that it is possible to learn to assign basic semantic roles based on the shallow sentence representations. Furthermore, these simple structural features were robust to drastic reductions in the integrity of the semantic-role feedback [Connor *et al.*, 2009] or being used with a minimally supervised parser [Connor *et al.*, 2010]. 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.

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 system we use the model of [Connor *et al.*, 2010], 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 [Connor *et al.*, 2010], 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 one 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 rel-

ative 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 natural child directed speech (CDS) from the CHILDES corpora [MacWhinney, 2000]. The training data were samples of parental speech to one child (Adam; [Brown, 1973]). The SRL training corpus consists of parental utterances in Adam sections 01-20 (child age 2;3 - 3;1), with test set from sections 21-23. 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. 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.

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). The goal with a structured prediction task is, given a set of labeled examples $(x_i, y_i) \in X \times Y$, where both X and Y are the space of some structures, to learn a function f_w :

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

Here H is a space of hidden latent structure that describes some connection between X and Y , and Φ is a feature encoding for the complete X, H, Y example structure. For the Semantic Role labeling case X are sentences, Y are argument role labellings, and H ranges over the possible argument and predicate structures in each input sentence x .

Because of the max over H in the definition of f_w , the general optimization problem for finding best w is non-convex. 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 use an online approach and a modification of Collin's Structure Perceptron [Collins, 2002] with margin [Kazama and Torisawa, 2007]. This basic algorithm (Algorithm 1) uses an approximation employed in [Felzenszwalb *et al.*, 2008; Cherry and Quirk, 2008] where for each example the best h^* is found (using current model and true labels) 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 1 Purely Latent Structure Perceptron

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

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 2. 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 2 Online Latent Classifier Training

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

The intuition for this online process is that for each sentence the learner finds the best joint meaning and structure based on the current classifiers and semantic constraints, then updates the classifiers separately to predict this selection. In the case where we have perfect high level semantic feedback $Y_i = y_i$, so the role classifier will search for the argument structure that is most useful in predicting the correct labels. In future sections we will experiment with loosening these semantic feedback constraints.

3.1 Argument, Predicate and Role Classification

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 and 5 of algorithm 1 and 2 respectively) 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 at least minimally the true argument nouns. Decisions regarding ‘likes’ and ‘yellow’ must then depend on the representation used by the SRL classifier.

We use features inspired by the psycholinguistic account of [Connor *et al.*, 2008]: simple structures that only represent the number and order of candidate arguments, and location of the predicate. For representing overall argument structure we use the noun pattern feature (indicates number and order of proposed arguments: first of three, second of two, etc; NounPat), as well as the position relative to the proposed verb (before or after; VPos). In the above example, with the correct argument assignment we have 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. In addition we also use a more complicated feature set that includes NounPat and VPos along with common features such as surrounding words and HMM tags, and conjunctions of NounPat and VPos features with the identified predicate (e.g. the proposed predicate is ‘likes’ and the target word is before it) so that the role classifier is more dependent on a correct predicate prediction.

For the argument and predicate structure classifiers the representation $\Phi_u(x, h)$ only depends on words, HMM tags and the other arguments and predicates in the structure. We represent each word by its word form, HMM tag, word before and after. The argument classifier also uses noun pattern, and the predicate representation uses the number of arguments and all suffixes of length up to three as a simple verb ending feature.

It should be noted that both algorithms require finding the max over hidden structures and labellings 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. Once we move on to more complicated language an alternative search strategy will need to be employed.

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 predictions¹.

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 1: 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 1. 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 2.

Table 1 shows the performance of the two algorithms from section 3 compared to the previously mentioned upper and lower bounds. All classifiers use the full feature sets from section 3.1. The purely latent method (algorithm 1) 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 2) 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 2: 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 feeds-back 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 2 shows the increasing performance as the feature complexity increases. Most notable is the large drop 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. Comparatively the argument performance drops less because the full semantic feedback will always implicitly drive an accurate argument identification. The role classification performance drop can be attributed to both a natural decrease in representation, as well as decreased argument and predicate structure accuracy during training and testing.

4.1 Loosening Feedback

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 5 of algorithm 2. 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.

¹Since 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

Feedback	Pred	Arg	A0	A1	Role F1
Full Labels	0.93(0.03)	0.86(0.04)	0.90(0.01)	0.78(0.02)	0.66(0.01)
Set of Labels	0.57(0.25)	0.72(0.15)	0.60(0.28)	0.48(0.18)	0.42(0.16)
Superset	0.23(0.14)	0.50(0.08)	0.31(0.28)	0.36(0.13)	0.24(0.11)
Superset + HMM Args	0.79(0.02)	0.89(0.00)	0.67(0.24)	0.55(0.13)	0.47(0.14)
Random	0.31	0.46			

Table 3: Results on test set when the amount of semantic feedback is decreased. Each value represents the mean over ten 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 no better than random. Once extra information driven by bottom-up, minimally supervised noun knowledge is introduced (HMM Args) the learner is able to make use of the superset feedback, and especially begin to identify agent and patient roles (A0 and A1).

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

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.

This starting point of knowledge is inspired by the syntactic bootstrapping theory, and by an account of syntactic bootstrapping known as ‘structure-mapping’ [Fisher, 1996; Gillette *et al.*, 1999; Lidz *et al.*, 2003]. Syntactic bootstrapping theory proposes that young children use their very par-

tial knowledge of syntax to guide sentence comprehension. The structure-mapping account makes three key assumptions: First, sentence comprehension is grounded by the acquisition of an initial set of concrete nouns; children are assumed to be able to identify the referents of some nouns via cross-situational observation [Gillette *et al.*, 1999]. Second, 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 that permits generalization to new verbs [Gertner *et al.*, 2006].

We incorporate this starting point by using the minimally-supervised argument identification of [Connor *et al.*, 2010]. In their 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. During latent classifier training, for each sentence the arguments and predicate identified by this HMM method are considered the true argument set (H_i), and then the best labeling from the superset of labels is selected for them.

Starting with a more accurate identification of arguments and predicate, the superset feedback is able to use the consistency of certain roles’ appearance in sentences to boost their linking with specific words and structures. For instance an agent or patient appears in most sentences, so once common arguments are identified and frequently appear with sentences where the agent role is a possible label, the chance of predicting this labeling increases. Table 3 shows that once we add the HMM argument identification to the superset feedback scheme the argument and predicate performance increases greatly (due to accuracy of the HMM argument identification), and the role classification also begins to increase, especially among the more common agent and patient roles. By linking nouns to arguments, argument number to predicate, and then argument structure to the prediction of role labels through the role classifier, we have enabled the system to begin to extract meaning from ambiguous feedback.

5 Conclusion and Future Work

In this work 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.

An eventual goal of this work is to use this latent training not only to improve intermediate representations, but to naturally grow the complexity as the system bootstraps itself. One natural notion of increasing complexity is dealing with sentences with multiple predicates, and larger numbers of arguments. By training an accurate argument and predicate identifier with this latent feedback on simple sentences, we can then lift the constraint of a single predicate per sentence and start to identify multiple predicates and argument structures per sentence, applying role classification to these structures without necessarily having to retrain on more complicated sentences.

Acknowledgments

This research is supported by NSF grant BCS-0620257 and NIH grant R01-HD054448.

References

- [Alishahi and Stevenson, 2010] 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.
- [Brown, 1973] R. Brown. *A First Language*. Harvard University Press, Cambridge, MA, 1973.
- [Carreras and Màrquez, 2004] X. Carreras and L. Màrquez. Introduction to the CoNLL-2004 shared tasks: Semantic role labeling. In *CoNLL*, 2004.
- [Chang et al., 2010] 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)*, 2010.
- [Charniak, 1997] E. Charniak. Statistical parsing with a context-free grammar and word statistics. In *AAAI*, 1997.
- [Cherry and Quirk, 2008] C. Cherry and C. Quirk. Discriminative, syntactic language modeling through latent svms. In *Proceedings of the Eighth Conference of AMTA*, 2008.
- [Collins, 2002] M. Collins. Discriminative training methods for hidden Markov models: Theory and experiments with perceptron algorithms. In *SIGDAT*, 2002.
- [Connor et al., 2008] 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.
- [Connor et al., 2009] 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.
- [Connor et al., 2010] 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, 2010.
- [Felzenszwalb et al., 2008] P. Felzenszwalb, D. Mcallester, and D. Ramanan. A discriminatively trained, multiscale, deformable part model. In *CVPR*, 2008.
- [Fisher, 1996] C. Fisher. Structural limits on verb mapping: The role of analogy in children’s interpretation of sentences. *Cognitive Psychology*, 31:41–81, 1996.
- [Gertner et al., 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.
- [Gildea and Palmer, 2002] D. Gildea and M. Palmer. The necessity of parsing for predicate argument recognition. In *ACL*, 2002.
- [Gillette et al., 1999] J. Gillette, H. Gleitman, L. R. Gleitman, and A. Lederer. Human simulations of vocabulary learning. *Cognition*, 73:135–176, 1999.
- [Hajič et al., 2009] 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.
- [Johnson et al., 2010] M. Johnson, K. Demuth, M. C. Frank, and B. Jones. Synergies in learning words and their meanings. In *Neural Information Processing Systems*, 23, 2010.
- [Kazama and Torisawa, 2007] Jun’ichi Kazama and Kentaro Torisawa. A new perceptron algorithm for sequence labeling with non-local features. In *EMNLP-CoNLL*, 2007.
- [Kingsbury and Palmer, 2002] P. Kingsbury and M. Palmer. From Treebank to PropBank. In *LREC*, 2002.
- [Lidz et al., 2003] 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.
- [MacWhinney, 2000] B. MacWhinney. *The CHILDES project: Tools for analyzing talk. Third Edition*. Lawrence Erlbaum Associates, Mahwah, NJ, 2000.
- [Moon et al., 2010] 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.
- [Punyakanok et al., 2008] V. Punyakanok, D. Roth, and W. Yih. The importance of syntactic parsing and inference in semantic role labeling. *Computational Linguistics*, 34(2), 2008.
- [Shi et al., 1998] 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.
- [Shi et al., 1999] 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.
- [Surdeanu et al., 2008] 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.
- [Yu and Joachims, 2009] C. Yu and T. Joachims. Learning structural svms with latent variables. In *ICML*, 2009.