Models and Training for Unsupervised Preposition Sense Disambiguation

Dirk Hovy and **Ashish Vaswani** and **Stephen Tratz** and
David Chiang and **Eduard Hovy**
Information Sciences Institute
University of Southern California
4676 Admiralty Way, Marina del Rey, CA 90292
{dirkh,avaswani,stratz,chiang,hovy}@isi.edu

Abstract

We present a preliminary study on unsupervised preposition sense disambiguation (PSD), comparing different models and training techniques (EM, MAP-EM with L_0 norm, Bayesian inference using Gibbs sampling). To our knowledge, this is the first attempt at unsupervised preposition sense disambiguation. Our best accuracy reaches 56%, a significant improvement (at p <.001) of 16% over the most-frequent-sense baseline.

1 Introduction

Reliable disambiguation of words plays an important role in many NLP applications. Prepositions are ubiquitous—they account for more than 10% of the 1.16m words in the Brown corpus—and highly ambiguous. The Preposition Project (Litkowski and Hargraves, 2005) lists an average of 9.76 senses for each of the 34 most frequent English prepositions, while nouns usually have around two (WordNet nouns average about 1.2 senses, 2.7 if monosemous nouns are excluded (Fellbaum, 1998)). Disambiguating prepositions is thus a challenging and interesting task in itself (as exemplified by the SemEval 2007 task, (Litkowski and Hargraves, 2007)), and holds promise for NLP applications such as Information Extraction or Machine Translation.[1] Given a sentence such as the following:

In the morning, he shopped in Rome

we ultimately want to be able to annotate it as

in/TEMPORAL the morning/TIME he/PERSON
shopped/SOCIAL in/LOCATIVE
Rome/LOCATION

Here, the preposition *in* has two distinct meanings, namely a temporal and a locative one. These meanings are context-dependent. Ultimately, we want to disambiguate prepositions not by and for themselves, but in the context of sequential semantic labeling. This should also improve disambiguation of the words linked by the prepositions (here, *morning*, *shopped*, and *Rome*). We propose using unsupervised methods in order to leverage unlabeled data, since, to our knowledge, there are no annotated data sets that include both preposition and argument senses. In this paper, we present our unsupervised framework and show results for preposition disambiguation. We hope to present results for the joint disambiguation of preposition and arguments in a future paper.

The results from this work can be incorporated into a number of NLP problems, such as semantic tagging, which tries to assign not only syntactic, but also semantic categories to unlabeled text. Knowledge about semantic constraints of prepositional constructions would not only provide better label accuracy, but also aid in resolving prepositional attachment problems. Learning by Reading approaches (Mulkar-Mehta et al., 2010) also crucially depend on unsupervised techniques as the ones described here for textual enrichment.

Our contributions are:

- we present the first unsupervised preposition sense disambiguation (PSD) system

[1] See (Chan et al., 2007) for how using WSD can help MT.

Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics:shortpapers, pages 323–328,
Portland, Oregon, June 19-24, 2011.

- we compare the effectiveness of various models and unsupervised training methods
- we present ways to extend this work to prepositional arguments

2 Preliminaries

A preposition p acts as a link between two words, h and o. The head word h (a noun, adjective, or verb) governs the preposition. In our example above, the head word is *shopped*. The object of the prepositional phrase (usually a noun) is denoted o, in our example *morning* and *Rome*. We will refer to h and o collectively as the *prepositional arguments*. The triple h, p, o forms a syntactically and semantically constrained structure. This structure is reflected in dependency parses as a common construction. In our example sentence above, the respective structures would be *shopped in morning* and *shopped in Rome*. The senses of each element are denoted by a barred letter, i.e., $\bar{p}$ denotes the preposition sense, $\bar{h}$ denotes the sense of the head word, and $\bar{o}$ the sense of the object.

3 Data

We use the data set for the SemEval 2007 PSD task, which consists of a training (16k) and a test set (8k) of sentences with sense-annotated prepositions following the sense inventory of *The Preposition Project*, TPP (Litkowski and Hargraves, 2005). It defines senses for each of the 34 most frequent prepositions. There are on average 9.76 senses per preposition. This corpus was chosen as a starting point for our study since it allows a comparison with the original SemEval task. We plan to use larger amounts of additional training data.

We used an in-house dependency parser to extract the prepositional constructions from the data (e.g., "shop/VB in/IN Rome/NNP"). Pronouns and numbers are collapsed into "PRO" and "NUM", respectively.

In order to constrain the argument senses, we construct a dictionary that lists for each word all the possible lexicographer senses according to WordNet. The set of lexicographer senses (45) is a higher level abstraction which is sufficiently coarse to allow for a good generalization. Unknown words are assumed to have all possible senses applicable to their respective word class (i.e. all noun senses for words labeled as nouns, etc).

4 Graphical Model

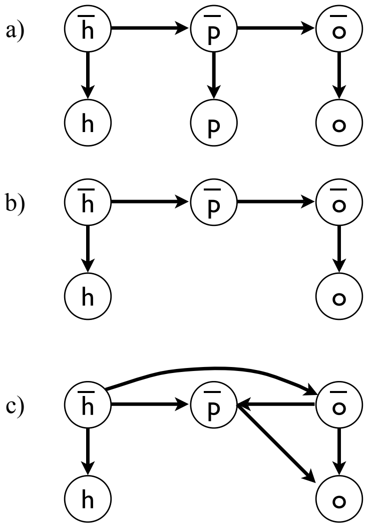

Figure 1: Graphical Models. a) 1^{st} order HMM. b) variant used in experiments (one model/preposition, thus no conditioning on p). c) incorporates further constraints on variables

As shown by Hovy et al. (2010), preposition senses can be accurately disambiguated using only the head word and object of the PP. We exploit this property of prepositional constructions to represent the constraints between h, p, and o in a graphical model. We define a good model as one that reasonably constrains the choices, but is still tractable in terms of the number of parameters being estimated.

As a starting point, we choose the standard first-order Hidden Markov Model as depicted in Figure 1a. Since we train a separate model for each preposition, we can omit all arcs to p. This results in model 1b. The joint distribution over the network can thus be written as

$$P_p(h, o, \bar{h}, \bar{p}, \bar{o}) = P(\bar{h}) \cdot P(h|\bar{h}) \cdot P(\bar{p}|\bar{h}) \cdot P(\bar{o}|\bar{p}) \cdot P(o|\bar{o}) \quad (1)$$

We want to incorporate as much information as possible into the model to constrain the choices. In Figure 1c, we condition $\bar{p}$ on both $\bar{h}$ and $\bar{o}$, to reflect the fact that prepositions act as links and determine their sense mainly through context. In order to constrain the object sense $\bar{o}$, we condition on $\bar{h}$, similar to a second-order HMM. The actual object o is conditioned on both $\bar{p}$ and $\bar{o}$. The joint distribution is equal to

$$P_p(h, o, \bar{h}, \bar{p}, \bar{o}) = P(\bar{h}) \cdot P(h|\bar{h}) \cdot P(\bar{o}|\bar{h}) \cdot P(\bar{p}|\bar{h}, \bar{o}) \cdot P(o|\bar{o}, \bar{p}) \quad (2)$$

Though we would like to also condition the preposition sense $\bar{p}$ on the head word h (i.e., an arc between them in 1c) in order to capture idioms and fixed phrases, this would increase the number of parameters prohibitively.

5 Training

The training method largely determines how well the resulting model explains the data. Ideally, the sense distribution found by the model matches the real one. Since most linguistic distributions are Zipfian, we want a training method that encourages sparsity in the model.

We briefly introduce different unsupervised training methods and discuss their respective advantages and disadvantages. Unless specified otherwise, we initialized all models uniformly, and trained until the perplexity rate stopped increasing or a predefined number of iterations was reached. Note that MAP-EM and Bayesian Inference require tuning of some hyper-parameters on held-out data, and are thus not fully unsupervised.

5.1 EM

We use the EM algorithm (Dempster et al., 1977) as a baseline. It is relatively easy to implement with existing toolkits like Carmel (Graehl, 1997). However, EM has a tendency to assume equal importance for each parameter. It thus prefers "general" solutions, assigning part of the probability mass to unlikely states (Johnson, 2007). We ran EM on each model for 100 iterations, or until the perplexity stopped decreasing below a threshold of 10^{-6}.

5.2 EM with Smoothing and Restarts

In addition to the baseline, we ran 100 restarts with random initialization and smoothed the fractional counts by adding 0.1 before normalizing (Eisner, 2002). Smoothing helps to prevent overfitting. Repeated random restarts help escape unfavorable initializations that lead to local maxima. Carmel provides options for both smoothing and restarts.

5.3 MAP-EM with L_0 Norm

Since we want to encourage sparsity in our models, we use the MDL-inspired technique introduced by Vaswani et al. (2010). Here, the goal is to increase the data likelihood while keeping the number of parameters small. The authors use a smoothed L_0 prior, which encourages probabilities to go down to 0. The prior involves hyper-parameters α, which rewards sparsity, and β, which controls how close the approximation is to the true L_0 norm.[2] We perform a grid search to tune the hyper-parameters of the smoothed L_0 prior for accuracy on the preposition *against*, since it has a medium number of senses and instances. For HMM, we set α_{trans} =100.0, β_{trans} =0.005, α_{emit} =1.0, β_{emit} =0.75. The subscripts $_{trans}$ and $_{emit}$ denote the transition and emission parameters. For our model, we set α_{trans} =70.0, β_{trans} =0.05, α_{emit} =110.0, β_{emit} =0.0025. The latter resulted in the best accuracy we achieved.

5.4 Bayesian Inference

Instead of EM, we can use Bayesian inference with Gibbs sampling and Dirichlet priors (also known as the Chinese Restaurant Process, CRP). We follow the approach of Chiang et al. (2010), running Gibbs sampling for 10,000 iterations, with a burn-in period of 5,000, and carry out automatic run selection over 10 random restarts.[3] Again, we tuned the hyper-parameters of our Dirichlet priors for accuracy via a grid search over the model for the preposition *against*. For both models, we set the concentration parameter α_{trans} to 0.001, and α_{emit} to 0.1. This encourages sparsity in the model and allows for a more nuanced explanation of the data by shifting probability mass to the few prominent classes.

[2] For more details, the reader is referred to Vaswani et al. (2010).

[3] Due to time and space constraints, we did not run the 1000 restarts used in Chiang et al. (2010).

	baseline	**Vanilla EM**	**EM, smoothed, 100 random restarts**	**MAP-EM + smoothed L0 norm**	**CRP, 10 random restarts**
HMM	0.40 (0.40)	0.42 (0.42)	*0.55 (0.55)*	*0.45 (0.45)*	*0.53 (0.53)*
our model		0.41 (0.41)	*0.49 (0.49)*	***0.55 (0.56)***	*0.48 (0.49)*

Table 1: Accuracy over all prepositions w. different models and training. Best accuracy: MAP-EM+smoothed L_0 norm on our model. Italics denote significant improvement over baseline at p <.001. Numbers in brackets include *against* (used to tune MAP-EM and Bayesian Inference hyper-parameters)

6 Results

Given a sequence h, p, o, we want to find the sequence of senses $\bar{h}, \bar{p}, \bar{o}$ that maximizes the joint probability. Since unsupervised methods use the provided labels indiscriminately, we have to map the resulting predictions to the gold labels. The predicted label sequence $\hat{h}, \hat{p}, \hat{o}$ generated by the model via Viterbi decoding can then be compared to the true key. We use many-to-1 mapping as described by Johnson (2007) and used in other unsupervised tasks (Berg-Kirkpatrick et al., 2010), where each predicted sense is mapped to the gold label it most frequently occurs with in the test data. Success is measured by the percentage of accurate predictions. Here, we only evaluate $\hat{p}$.

The results presented in Table 1 were obtained on the SemEval test set. We report results both with and without *against*, since we tuned the hyperparameters of two training methods on this preposition. To test for significance, we use a two-tailed t-test, comparing the number of correctly labeled prepositions. As a baseline, we simply label all word types with the same sense, i.e., each preposition token is labeled with its respective name. When using many-to-1 accuracy, this technique is equivalent to a most-frequent-sense baseline.

Vanilla EM does not improve significantly over the baseline with either model, all other methods do. Adding smoothing and random restarts increases the gain considerably, illustrating how important these techniques are for unsupervised training. We note that EM performs better with the less complex HMM.

CRP is somewhat surprisingly roughly equivalent to EM with smoothing and random restarts. Accuracy might improve with more restarts.

MAP-EM with L_0 normalization produces the best result (56%), significantly outperforming the baseline at $p < .001$. With more parameters (9.7k vs. 3.7k), which allow for a better modeling of the data, L_0 normalization helps by zeroing out infrequent ones. However, the difference between our complex model and the best HMM (EM with smoothing and random restarts, 55%) is not significant.

The best (supervised) system in the SemEval task (Ye and Baldwin, 2007) reached 69% accuracy. The best current supervised system we are aware of (Hovy et al., 2010) reaches 84.8%.

7 Related Work

The semantics of prepositions were topic of a special issue of *Computational Linguistics* (Baldwin et al., 2009). Preposition sense disambiguation was one of the SemEval 2007 tasks (Litkowski and Hargraves, 2007), and was subsequently explored in a number of papers using supervised approaches: O'Hara and Wiebe (2009) present a supervised preposition sense disambiguation approach which explores different settings; Tratz and Hovy (2009), Hovy et al. (2010) make explicit use of the arguments for preposition sense disambiguation, using various features. We differ from these approaches by using unsupervised methods and including argument labeling.

The constraints of prepositional constructions have been explored by Rudzicz and Mokhov (2003) and O'Hara and Wiebe (2003) to annotate the semantic role of complete PPs with FrameNet and Penn Treebank categories. Ye and Baldwin (2006) explore the constraints of prepositional phrases for semantic role labeling. We plan to use the constraints for argument disambiguation.

8 Conclusion and Future Work

We evaluate the influence of two different models (to represent constraints) and three unsupervised training methods (to achieve sparse sense distributions) on PSD. Using MAP-EM with L_0 norm on our model, we achieve an accuracy of 56%. This is a significant improvement (at p <.001) over the baseline and vanilla EM. We hope to shorten the gap to supervised systems with more unlabeled data. We also plan on training our models with EM with features (Berg-Kirkpatrick et al., 2010).

The advantage of our approach is that the models can be used to infer the senses of the prepositional arguments as well as the preposition. We are currently annotating the data to produce a test set with Amazon's Mechanical Turk, in order to measure label accuracy for the preposition arguments.

Acknowledgements

We would like to thank Steve DeNeefe, Jonathan Graehl, Victoria Fossum, and Kevin Knight, as well as the anonymous reviewers for helpful comments on how to improve the paper. We would also like to thank Morgan from Curious Palate for letting us write there. Research supported in part by Air Force Contract FA8750-09-C-0172 under the DARPA Machine Reading Program and by DARPA under contract DOI-NBC N10AP20031.

References

Tim Baldwin, Valia Kordoni, and Aline Villavicencio. 2009. Prepositions in applications: A survey and introduction to the special issue. *Computational Linguistics*, 35(2):119–149.

Taylor Berg-Kirkpatrick, Alexandre Bouchard-Côté, John DeNero, and Dan Klein. 2010. Painless Unsupervised Learning with Features. In *North American Chapter of the Association for Computational Linguistics*.

Yee Seng Chan, Hwee Tou Ng, and David Chiang. 2007. Word sense disambiguation improves statistical machine translation. In *Annual Meeting – Association For Computational Linguistics*, volume 45, pages 33–40.

David Chiang, Jonathan Graehl, Kevin Knight, Adam Pauls, and Sujith Ravi. 2010. Bayesian inference for Finite-State transducers. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 447–455. Association for Computational Linguistics.

Arthur P. Dempster, Nan M. Laird, and Donald B. Rubin. 1977. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, 39(1):1–38.

Jason Eisner. 2002. An interactive spreadsheet for teaching the forward-backward algorithm. In *Proceedings of the ACL-02 Workshop on Effective tools and methodologies for teaching natural language processing and computational linguistics-Volume 1*, pages 10–18. Association for Computational Linguistics.

Christiane Fellbaum. 1998. *WordNet: an electronic lexical database*. MIT Press USA.

Jonathan Graehl. 1997. Carmel Finite-state Toolkit. *ISI/USC*.

Dirk Hovy, Stephen Tratz, and Eduard Hovy. 2010. What's in a Preposition? Dimensions of Sense Disambiguation for an Interesting Word Class. In *Coling 2010: Posters*, pages 454–462, Beijing, China, August. Coling 2010 Organizing Committee.

Mark Johnson. 2007. Why doesn't EM find good HMM POS-taggers. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 296–305.

Ken Litkowski and Orin Hargraves. 2005. The preposition project. *ACL-SIGSEM Workshop on "The Linguistic Dimensions of Prepositions and Their Use in Computational Linguistic Formalisms and Applications"*, pages 171–179.

Ken Litkowski and Orin Hargraves. 2007. SemEval-2007 Task 06: Word-Sense Disambiguation of Prepositions. In *Proceedings of the 4th International Workshop on Semantic Evaluations (SemEval-2007)*, Prague, Czech Republic.

Rutu Mulkar-Mehta, James Allen, Jerry Hobbs, Eduard Hovy, Bernardo Magnini, and Christopher Manning, editors. 2010. *Proceedings of the NAACL HLT 2010 First International Workshop on Formalisms and Methodology for Learning by Reading*. Association for Computational Linguistics, Los Angeles, California, June.

Tom O'Hara and Janyce Wiebe. 2003. Preposition semantic classification via Penn Treebank and FrameNet. In *Proceedings of CoNLL*, pages 79–86.

Tom O'Hara and Janyce Wiebe. 2009. Exploiting semantic role resources for preposition disambiguation. *Computational Linguistics*, 35(2):151–184.

Frank Rudzicz and Serguei A. Mokhov. 2003. Towards a heuristic categorization of prepositional phrases in english with wordnet. Technical report, Cornell University, arxiv1.library.cornell.edu/abs/1002.1095-?context=cs.

Stephen Tratz and Dirk Hovy. 2009. Disambiguation of Preposition Sense Using Linguistically Motivated Features. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics, Companion Volume: Student Research Workshop and Doctoral Consortium*, pages 96–100, Boulder, Colorado, June. Association for Computational Linguistics.

Ashish Vaswani, Adam Pauls, and David Chiang. 2010. Efficient optimization of an MDL-inspired objective function for unsupervised part-of-speech tagging. In *Proceedings of the ACL 2010 Conference Short Papers*, pages 209–214. Association for Computational Linguistics.

Patrick Ye and Tim Baldwin. 2006. Semantic role labeling of prepositional phrases. *ACM Transactions on Asian Language Information Processing (TALIP)*, 5(3):228–244.

Patrick Ye and Timothy Baldwin. 2007. MELB-YB: Preposition Sense Disambiguation Using Rich Semantic Features. In *Proceedings of the 4th International Workshop on Semantic Evaluations (SemEval-2007)*, Prague, Czech Republic.