Disambiguating prepositional phrase attachment sites with sense information captured in contextualized distributional data

Clayton Greenberg
Department of Computational Linguistics and Phonetics
Universität des Saarlandes
cgreenbe@alumni.princeton.edu

Abstract

This work presents a supervised prepositional phrase (PP) attachment disambiguation system that uses contextualized distributional information as the distance metric for a nearest-neighbor classifier. Contextualized word vectors constructed from the GigaWord Corpus provide a method for implicit Word Sense Disambiguation (WSD), whose reliability helps this system outperform baselines and achieve comparable results to those of systems with full WSD modules. This suggests that targeted WSD methods are preferable to ignoring sense information and also to implementing WSD as an independent module in a pipeline.

1 Introduction

Arriving at meaning from a linguistic expression is hardly a trivial process, but a "simple" four-word expression shows some of the kinds of knowledge and interactions involved:

(1) a. *eat* [*seeds* [*in plants*]]
b. [*eat seeds*] [*in plants*]

(a) and (b) illustrate two possible interpretations for the expression. In (a), the seeds are part of larger organic units, and in (b), the eating takes place in refineries. Choosing (a) or (b) helps the system construct accurate relationships between the events and participants mentioned, which is essential for many natural language processing tasks including machine translation, information extraction, and textual inference.

These two groupings represent an example of the widely-studied phenomenon of prepositional phrase (PP) attachment ambiguity. We define the governor of a PP as the word or phrase that the PP modifies. Ambiguity arises from multiple candidates for the governor. Strings such as in (1) can be represented by quadruples of the form (V, N_1, P, N_2), where V is a transitive verb, N_1 is the head noun of an object of V, P is a preposition, and N_2 is the head noun of the object of P. Then, (a) and (b) reflect the two possible choices of governor for the PP: V (adverbial PP) and N_1 (adjectival PP). Therefore, disambiguation for such quadruples is a binary classification of the PP as adjectival or adverbial, or equivalently, noun-attach or verb-attach.

In our example, classifying the sense of the word *plant* as either `organic_unit` or `refinery` is key to choosing the correct structure. These senses have significantly different respective relationships to *eat* and *seeds*. In particular, we often eat most except, or only, the seeds from an organic unit, but we have no such intuitions about refineries. The training data must be analyzed carefully in order to prevent unwanted mixing of senses, since that causes noise in predictions about word relationships.

Given that $V - N_2$ and $N_1 - N_2$ relationships are very important for PP-attachment disambiguation, it is not surprising that leading PP-attachment disambiguation systems include a Word Sense Disambiguation (WSD) module. The challenging aspect of this is that it introduces a subtask that in the general case has lower accuracy levels than the entire system. Hence, its place and form within the system deserves to be examined closely. Since a representation of the predicted sense is not part of the attachment decision, it does not need to be explicitly present within the procedure. In this paper, we investigate the importance of proper word sense decisions for PP-attachment disambiguation, and describe a highly-accurate system that encodes sense information in contextualized distributional data. Its high performance shows the benefit of representing and handling sense information in a targeted fashion for the task.

Proceedings of the ACL 2014 Student Research Workshop, pages 71–77,
Baltimore, Maryland USA, June 22-27 2014.

2 Background and related work

Sense information provides an illuminating through line for many previous PP-attachment disambiguation systems. We begin by describing a very popular dataset for the problem and its subsequent development, and then trace through the two main approaches to sense information representation and the results obtained using this dataset.

2.1 The corpus

A standard corpus for the binary classification problem described above was developed by Ratnaparkhi, Reynar and Roukos (1994). They systematically extracted (V, N_1, P, N_2) quadruples from the Penn Treebank Wall Street Journal (WSJ) corpus and used the manually-generated constituency parses to obtain attachment decisions for each of the extracted PPs. The final dataset contained 27,937 quadruples. These were divided into 20,801 training quadruples, 4,039 development quadruples, and 3,097 test quadruples. Their maximum entropy model achieved 81.6% accuracy on this dataset and their decision tree achieved 77.7%. Accuracy on this corpus is defined to be the number of quadruples for which the classifier assigned the same attachment site as the site indicated in that sentence's parse tree, divided by the total number of quadruples. Although some parse trees in the corpus are known to have errors, the accuracy figures do not take this into account.

Also, Ratnaparkhi et al. (1994) conducted human experiments with a subset of their corpus. They found that humans, when given just the quadruple, were accurate 88.2% of the time. When given the entire sentence for context, accuracy improved to 93.2%. The perhaps underwhelming human performance is partially due to misclassifications by the Treebank assemblers who made these determinations by hand, and also unclear cases, which we discuss in the next section.

Collins and Brooks (1995) introduced modifications to the Ratnaparkhi et al. (1994) dataset meant to combat data sparsity and used the modified version to train their backed-off model. They replaced four digit numbers with `YEAR`, other numbers with `NUM`. Verbs and prepositions were converted to all lowercase. In nouns, all words that started with an uppercase letter followed by a lowercase letter were replaced with `NAME`. Then, all strings `NAME-NAME` were replaced with `NAME`. Finally all verbs were automatically lemmatized. They did not release statistics on how these modifications affected performance, so it is unclear how to allocate the performance increase between the backed-off model and the modifications to the dataset. The paper also provided some baselines: they achieve 59.0% accuracy on the Ratnaparkhi et al. (1994) corpus by assigning noun-attach to every quadruple, and 72.2% accuracy by assigning a default classification determined for each preposition. They show, and many subsequent papers confirm, that the preposition is the most predictive dimension in the quadruple.

Abney, Schapire, and Singer (1999) used the dataset from Collins and Brooks (1995) with a boosting algorithm and achieved 85.4% accuracy. Their algorithm also was able to order the specific data points by how much weight they were assigned by the learning algorithm. The highest data points tended to be those that contained errors. Thus, they were able to improve the quality of the dataset in a systematic way.

2.2 The WordNet approach

WordNet (Fellbaum, 1998) can be quite a powerful aid to PP-attachment disambiguation because it provides a way to systematically quantify semantic relatedness. The drawback is, though, that since WordNet semantic relations are between explicit word senses (SynSets), the words in the quadruples must be associated with these explicit word senses. The systems described below outline the different ways to make those associations.

Brill and Resnik (1994) trained a transformation-based learning algorithm on 12,766 quadruples from WSJ, with modifications similar to those by Collins and Brooks (1995). As a particularly human-interpretable feature, the rules used word sense hierarchies. Namely, a WordNet rule applied to the named node and all of its hyponyms. For example, a rule involving *boat* would apply to instances of *kayak*. Importantly, each noun in the corpus inherited hypernyms from all of its senses. Therefore, they did not perform explicit WSD. Their accuracy was 81.8%.

The neural network by Nadh and Huyck (2012) also used WordNet word sense hierarchies. Only the first (intended to be the most frequent) sense of the word was used in computations. Hence, they explicitly perform WSD using a baseline method.

On a training corpus of 4,810 quadruples and a test corpus of 3,000 quadruples from WSJ, they achieve 84.6% accuracy. This shows the success of performing baseline WSD as part of a PP-attachment disambiguation system, although the different dataset makes comparison less direct.

At the other extreme, Stetina and Nagao (1997) developed a customized, explicit WSD algorithm as part of their decision tree system. For each ambiguous word in each quadruple, this algorithm selected a most semantically similar quadruple in the training data using unambiguous or previously disambiguated terms. Then, the word was assigned the WordNet sense that was semantically closest to the sense of the corresponding word in the other quadruple. Their distance metric was $L_1/D_1 + L_2/D_2$, where L_i is the distance from word sense i to the common ancestor, and D_i is the depth of the tree (distance to root) at word sense i. Such a metric captures the notion that more fine grained distinctions exist deeper in the WordNet graph, so the same absolute distance between nodes matters less at greater depths. Stetina and Nagao (1997) trained on a version of the Ratnaparkhi et al. (1994) dataset that contained modifications similar to those by Collins and Brooks (1995) and excluded forms not present in WordNet. The system achieved 88.1% accuracy on the entire test set and 90.8% accuracy on the subset of the test set in which all four of the words in the quadruple were present in WordNet.

Finally, Greenberg (2013) implemented a decision tree that reimplemented the WSD module from Stetina and Nagao (1997), and also used WordNet morphosemantic (teleological) links, WordNet evocations, and a list of phrasal verbs as features. The morphosemantic links and evocations brought more semantic relatedness information after the cost of explicit WSD had already been incurred. The system achieved 89.0% on a similarly modified Ratnaparkhi et al. (1994) dataset.

2.3 The distributional approach

As an alternative to the WordNet approach, the distributional tradition allows for implicit sense handling given that contexts from all senses of the word are represented together in the vector. Without modification, the senses are represented according to their relative frequencies in the data. Pantel and Lin (2000) created a collocation database that, for a given word, tracked the words that appeared in specific syntactic relations to it, such as subject (for verbs), adjective-modifier (for nouns), etc. Then, they used the collocation database to construct a corpus-based thesaurus that evaluated semantic relatedness between quadruples. With a mix of unsupervised learning algorithms, they achieved 84.3% accuracy. They also argued that rules involving both V and N_1 should be excluded because they cause over-fitting.

Zhao and Lin (2004) implemented a nearest neighbor system that used various vector similarity metrics to calculate distances between quadruples. The vectors were generated from the ACQUAINT corpus with both syntactic relation and proximity-based (bag of words) models. They found that the cosine of pointwise mutual information metric on a syntactic model performed with the greatest accuracy (86.5%, $k = 33$). They used a version of the Ratnaparkhi et al. (1994) dataset that had all words lemmatized and all digits replaced by `@`.

Using the Web as a large unsupervised corpus, Nakov and Hearst (2005) created a PP-attachment disambiguation system that exploits n-grams, derived surface features, and paraphrases to predict classifications. The system searched for six specific disambiguating paraphrases such as *opened the door (with a key)*, which suggests verb-attach, and *eat: spaghetti with sauce*, which suggests noun-attach. Paraphrases and n-gram models represent the aim to gather context beyond the quadruple as a disambiguation method. Their final system had 85.0% precision and 91.8% recall on the Ratnaparkhi et al. (1994) dataset. When assigning unassigned quadruples to verb-attach, it had 83.6% accuracy and 100% recall. Their system continued the trend that the most common error is classifying a noun-attach quadruple as verb-attach. This is because the majority of difficult cases are verb-attach, so all of the difficult cases get assigned verb-attach as a default.

3 Linguistic analysis

In this section, we will discuss some difficulties with and observations about the task of PP-attachment disambiguation. The analyses and conclusions drawn here set the linguistic foundation for the structure of the system described in the next section.

3.1 Lexically-specified prepositions

Hindle and Rooth (1993) provided many linguistic insights for the PP-attachment disambiguation problem, including the tendency to be verb-attach if N_1 is a pronoun, and that idiomatic expressions (e.g. *give way to mending*) and light verb constructions (e.g. *make cuts to Social Security*) are particularly troublesome for humans to classify. The defining feature of such constructions is a semantically-vacuous preposition. For example, in (2), we have semantically similar verbs appearing with different prepositions and yet the meanings of these sentences are still similar.

(2) a. *She was blamed for the crime.*
b. *She was accused of the crime.*
c. *She was charged with the crime.*

Further, when we nominalize *charged* we can get *charges of murder*, but *charged of murder* is usually unacceptable. Also, (3) gives an analogous three-way preposition variation following nouns.

(3) a. *They proposed a ban on tea.*
b. *They proposed a request for tea.*
c. *They proposed an alternative to tea.*

We argue that in these cases, a preceding word completely determines the preposition selected and that no further meaning is conveyed. In fact, we might say that the prepositions in this case serve analogously to morphological case marking in languages more heavily inflected than English. Freidin (1992) makes a proposal along these lines. The prescriptive rules that dictate "correct" and "incorrect" prepositions associated with certain verbs, nouns, and adjectives, as well as our robust ability to understand these sentences with the prepositions omitted, strongly suggest that this selection is idiosyncratic and cannot be derived from deeper principles.

The extreme case is phrasal verbs, for which it is problematic to posit the existence of a PP because the object can occur before or after the "preposition." As shown in (4d), this is not acceptable for standard prepositions.

(4) a. *He ran up the bill.*
b. *He ran the bill up.*
c. *He ran up the hill.*
d. * *He ran the hill up.*

For these, we say that there is one lexical entry for the transitive verb plus the particle (preposition without an object), as in *to run up*, and an optional operation reverses the order of the object of the phrasal verb and its particle.

Usual paraphrase tests, such as those described in Nakov and Hearst (2005), often do not lead to consistent conclusions about the proper attachment site for these lexically-specified prepositions. Further, two separate governors do not appear to be plausible. Therefore, these constructions probably do not belong as data points in the PP-attachment task. However, if they must conform to the task, the most reasonable attachment decision is likely to be the word that determined the preposition. Therefore, the PPs in (2) are verb-attach and those in (3) are noun-attach. This treatment of lexically-specified prepositions accounts for light verb constructions because the N_1 in those constructions dictates the preposition.

3.2 The special case of *of*

PPs with the preposition *of* attach to nouns with very few exceptions. In fact, 99.1% of the quadruples with *of* in our training set are noun-attach. The other 0.9% were misclassifications and quadruples with verbs that lexically specify *of*, such as *accuse*. The behavior of *of*-PPs has been widely studied. We take the acceptability of (5a) and not (5b) as evidence that *of*-PPs introduce argument-like descriptions of their governors.

(5) a. *a game of cards with incalculable odds*
b. * *a game with incalculable odds of cards*

The extremely high proportion of noun-attachments within *of*-PPs leads some to exclude *of*-PPs altogether from attachment disambiguation corpora. In our data, excluding this most commonly used English preposition shifts the most frequent attachment decision from noun-attach to verb-attach. This is unfortunate for systems aiming to mimic human processing, since Late Closure (Frazier, 1979) suggests a preference for noun-attach as the default or elsewhere case.

4 Methods

Our PP attachment disambiguation system is most closely related to Zhao and Lin (2004). We experimented with several similarity measures on a slightly preprocessed version of the Ratnaparkhi et al. (1994) dataset.

4.1 Training data

Because humans only perform 0.1% better than Stetina and Nagao's (1997) system when given the quadruples but not the full sentences (although technically on different datasets), we found it important to locate the full sentences in the Penn Treebank. So, we carefully searched for the quadruples in the raw version of the corpus. We ensured that the corpus would be searched sequentially, i.e. search for the current quadruple would begin on the previous matched sentence and then proceed forward. By inspection, we could tell that the sentences were roughly in order, so this choice increased performance and accuracy. However, we had to adapt the program to be flexible so that some truncated tokens in the quadruples, such as incorrectly segmented contractions, would be matched to their counterparts.

Next, we created some modified versions of the training corpus. We explored the effect of excluding quadruples with lexically-specified prepositions (usually tagged `PP-CLR` in WSJ), removing sentences in which there was no actual V, N_1, P, N_2 string found, manually removing encountered misclassifications, and reimplementing data sparsity modifications from Collins and Brooks (1995) and Stetina and Nagao (1997). In particular, we used the WordNet lemmatizer in NLTK to lemmatize the verbs in the corpus (Bird, Loper, and Klein 2009). However, for direct comparison with Zhao and Lin (2004), we decided to use in our final experiment a version of the corpus with all words lemmatized and all numbers replaced by `@`, but no other modifications.

4.2 Knowledge base

In order to compute quadruple similarity measures that take context information into account, we adopted the vector space model implemented by Dinu and Thater (2012). This model constructs distributional word vectors from the GigaWord corpus. We used a "filtered" model, meaning that the context for each occurrence is composed of words that are linked to that occurrence in a dependency parse. Therefore, the model is similar to a bag of words model, but does contain some syntactic weighting. To contextualize a vector, the model weights the components of the uncontextualized vector with the components of the context vector, using the formula

$$v(w, c) = \sum_{w\prime \in W} \alpha(c, w\prime) f(w, w\prime) \vec{e}_{w\prime}$$

where w is the target word, c is the context, W is the set of words, α is the cosine similarity of c and $w\prime$, f is a co-occurrence function, and $\vec{e}_{w\prime}$ is a basis vector. Positive pmi-weighting was also applied to the vectors.

4.3 Implementation

We adopted the four-step classification procedure from Zhao and Lin (2004). At each step for each test quadruple, the training examples are sorted by a different vector composition method, a set of best examples is considered, and if these examples cast equal votes for noun-attach and verb-attach, the algorithm moves to the next step. Otherwise, the class with the greatest number of votes is assigned to the test quadruple.

1. Consider only the training examples for which all four words are equal to those in the test quadruple.

2. Consider the k highest (k experimentally determined) scoring examples, with the same preposition as the test quadruple, using the composition function

   $$sim(q_1, q_2) = vn_1 + vn_2 + n_1n_2$$

   where v, n_1, and n_2 are the vector similarities of the V, N_1, and N_2 pairs.

3. Same as (2), except using the function

   $$sim(q_1, q_2) = v + n_1 + n_2$$

4. Assign default class for the preposition (last resort), or noun-attach if there is no default class.

4.4 Similarity measures

We implemented four similarity measures. (1) abs: absolute word similarity, which gives 1 if the tokens are identical, 0 otherwise. (2) $noctxt$: cosine similarity using uncontextualized word vectors. (3) $ctxt_{quad}$: cosine similarity using word vectors contextualized by the quadruple words. (4) $ctxt_{sent}$: cosine similarity using word vectors contextualized by words from the full sentence.

5 Experimentation

We set the k values by using five-fold cross-validation on the training quadruples. Then, for intermediate numerical checks, we tested the systems on the development quadruples. The figures in the next section are the result of a single run of the final trained systems on the test quadruples.

6 Results

Table 1 presents results from our binary classifier using the different similarity measures. Table 2 compares our best binary classifier accuracy (using $ctxt_{quad}$) to other systems. Table 3 shows the number, percentage, and accuracy of decisions by step in the classification procedure for the $ctxt_{quad}$ run.

Similarity measure	**k value**	**Accuracy**
abs	3	80.2%
$noctxt$	11	86.6%
$ctxt_{quad}$	10	**88.4%**
$ctxt_{sent}$	8	81.9%

Table 1: Similarity measure performance comparison.

Method	**Sense handling**	**Accuracy**
BR1994	All senses equal	81.8%
PL2000	Global frequency	84.3%
ZL2004	Global frequency	86.5%
SN1997	Full WSD	88.1%
Our system	Context weighting	**88.4%**
G2013	Full WSD	89.0%

Table 2: Leading PP-attachment disambiguation systems.

Step	**Coverage**	**Coverage %**	**Accuracy**
1	244	7.88%	91.8%
2	2849	91.99%	88.1%
3	0	0.00%	N/A
4	4	0.13%	100.0%

Table 3: Coverage and accuracy for classification procedure steps, using $ctxt_{quad}$.

7 Discussion

The results above show that contextualizing the word vectors, which is meant to implicitly represent sense information, can statistically-significantly boost performance on PP-attachment disambiguation by 1.8% ($\chi^2 = 4.31, p < 0.04$) on an already quite accurate system. We can see that using the full sentence as context, while helpful for human judgment, is not effective in this system because there are not enough examples in the knowledge base for reliable statistics. It seems as though too much context obscures generalizations otherwise captured by the system.

Nominal increases in accuracy aside, this system uses only a knowledge base that is not specific to the task of PP-attachment disambiguation. We obtained highly accurate results without utilizing task-specific resources, such as sense inventories, or performing labor-intensive modifications to training data. Since systems with full WSD modules would likely require both of these, this implicit handling of sense information seems more elegant.

8 Conclusion

This paper describes a PP-attachment disambiguation system that owes its high performance to capturing sense information in contextualized distributional data. We see that this implicit handling is preferable to having no sense handling and also to having a full WSD module as part of a pipeline.

In future work, we would like to investigate how to systematically extract contexts beyond the quadruple, such as sentences or full documents, while maintaining the information captured in less contextualized vectors. Perhaps there are certain particularly informative positions whose words would positively affect the vectors. Given that words tend to maintain the same sense within a document, it is a particularly well-suited context to consider. However, care must be taken to minimize unwanted sense mixing, combat data sparsity, and restrict the number of similarity comparisons for efficiency.

Acknowledgments

We owe sincerest thanks to Prof. Dr. Manfred Pinkal and Dr. Stefan Thater for initial direction and providing the vector space model used in our system. Also, we thank Google for travel and conference support.

References

Steven Abney, Robert E. Schapire and Yoram Singer. 1999. Boosting Applied to Tagging and PP-attachment. In *Proceedings of the Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora, EMNLP-VLC*, College Park, MD. pp. 38–45.

Steven Bird, Edward Loper and Ewan Klein. 2009. *Natural Language Processing with Python*. O'Reilly Media, Inc.

Eric Brill and Philip Resnik. 1994. A rule-based approach to prepositional phrase attachment disambiguation. In *5th International Conference on Computational Linguistics (COLING94)*, Kyoto, Japan.

Michael Collins and James Brooks. 1995. Prepositional Attachment through a Backed-off Model. In David Yarovsky and Kenneth Church (ed.), *Proceedings of the Third Workshop on Very Large Corpora*, Somerset, New Jersey, Association for Computational Linguistics. pp. 27–38.

Georgiana Dinu and Stefan Thater. 2012. Saarland: vector-based models of semantic textual similarity. In *First Joint Conference on Lexical and Computational Semantics (*SEM)*, Montréal. pp. 603–607.

Christiane Fellbaum (ed.) 1998. *WordNet: An Electronic Lexical Database*. MIT Press.

Lyn Frazier. 1979. *On Comprehending Sentences: Syntactic Parsing Techniques*. Unpublished doctoral dissertation, University of Connecticut.

Robert Freidin. 1992. *Foundations of generative syntax*. MIT Press.

Clayton Greenberg. 2013. *Disambiguating prepositional phrase attachment sites with graded semantic data or, how to rule out elephants in pajamas*. Unpublished undergraduate thesis, Princeton University.

Donald Hindle and Mats Rooth. 1993. Structural Ambiguity and Lexical Relations. In *Meeting of the Association for Computational Linguistics*. pp. 229–236.

Kailash Nadh and Christian Huyck. 2012. A neurocomputational approach to prepositional phrase attachment ambiguity resolution. *Neural Computation*, 24(7): pp. 1906–1925.

Preslav Nakov and Marti Hearst. 2005. Using the Web as an Implicit Training Set: Application to Structural Ambiguity Resolution. In *Proceedings of HLT-NAACL*.

Patrick Pantel and Dekang Lin. 2000. An unsupervised approach to prepositional phrase attachment using contextually similar words. In *Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics*. pp. 101–108.

Adwait Ratnaparkhi, Jeff Reynar and Salim Roukos. 1994. A Maximum Entropy Model for Prepositional Phrase Attachment. In *Proceedings of the ARPA Human Language Technology Workshop*, Plainsboro, NJ. pp. 250–255.

Jiri Stetina and Makoto Nagao. 1997. Corpus Based PP Attachment Ambiguity Resolution with a Semantic Dictionary. In *Proceedings of the Fifth Workshop on Very Large Corpora*, Beijing and Hong Kong. pp. 66–80.

Shaojun Zhao Dekang Lin. 2004. Corpus Based PP Attachment Ambiguity Resolution with a Semantic Dictionary. In *Proceedings of the First International Joint Conference on Natural Language Processing*, Sanya, China.