

# PP-attachment disambiguation using large context

Marian Olteanu and Dan Moldovan

Human Language Technology Research Institute

The University of Texas at Dallas

Richardson, TX 75080

marian@hlt.utdallas.edu

moldovan@utdallas.edu

## Abstract

Prepositional Phrase-attachment is a common source of ambiguity in natural language. The previous approaches use limited information to solve the ambiguity – four lexical heads – although humans disambiguate much better when the full sentence is available. We propose to solve the PP-attachment ambiguity with a Support Vector Machines learning model that uses complex syntactic and semantic features as well as unsupervised information obtained from the World Wide Web. The system was tested on several datasets obtaining an accuracy of 93.62% on a Penn Treebank-II dataset; 91.79% on a FrameNet dataset when no manually-annotated semantic information is provided and 92.85% when semantic information is provided.

## 1 Problem description

### 1.1 PP-attachment ambiguity problem

Prepositional Phrase-attachment is a source of ambiguity in natural language that generates a significant number of errors in syntactic parsing. For example the sentence “I saw yesterday the man in the park with a telescope” has 5 different semantic interpretations based on the way the prepositional phrases “in the park” and “with the telescope” are attached: *I saw yesterday [the man [in the park [with a telescope]]]; I saw yesterday [the man [in the park*

*[with a telescope]]]; I saw yesterday [the man [in the park]] [with a telescope]; I saw yesterday [the man] [in the park [with a telescope]] and I saw yesterday [the man] [in the park] [with a telescope].*

The problem can be viewed as a decision of attaching a prepositional phrase (PP) to one of the preceding head nouns or verbs. The ambiguity expressed by the number of potential parse trees generated by Context-Free Grammars increases exponentially with the number of PPs. For a PP that follows the object of a verb there are 2 parse trees, for a chain of 2, 3, 4 and 5 PPs there are respectively 5, 14, 42 and 132 parse trees. Usually the average number of consecutive PPs in a sentence increases linearly with the length of the sentence.

Lexical and syntactic information alone is not sufficient to resolve the PP-attachment problem; often semantic and/or contextual information is necessary. For example, in “*I ate a pizza with anchovies*”, “*with anchovies*” attaches to the noun “*pizza*”, where as in “*I ate a pizza with friends.*”, “*with friends*” attaches to the verb “*eat*” – example found in (McLauchlan, 2001). There are instances of PP-attachment, like the one in “*I saw the car in the picture*” that can be disambiguated only by using contextual discourse information.

Usually, people don’t have much trouble in finding the right way to attach PPs. But if one limits the information used for disambiguation of the PP-attachment to include only the verb, the noun representing its object, the preposition and the main noun in the PP, the accuracy for human decision degrades from 93.2% to 88.2% (Ratnaparkhi et al., 1994) on a dataset extracted from Penn Treebank (Marcus et

al., 1993).

## 1.2 Motivation

Syntactic parsing is essential for many natural language applications such as Machine Translation, Question Answering, Information Extraction, Information Retrieval, Automatic Speech Recognition. Since parsing occurs early in the chain of NLP processing steps it has a large impact on the overall system performance.

## 2 Approach

Our approach to solve the PP-attachment ambiguity is based on a Support Vector Machines learner (Cortes and Vapnik, 1995). The feature set contains complex information extracted automatically from candidate syntax trees generated by parsing (Charniak, 2000), trees that will be improved by more accurate PP-attachment decisions. Some of these features were proven efficient for semantic information labeling (Gildea and Jurafsky, 2002). The feature set also includes unsupervised information obtained from a very large corpus (World Wide Web). Features containing manually annotated semantic information about the verb and about the objects of the verb have also been used. We adopted the standard approach to distinguish between verb and noun attachment; thus the classifier has to choose between two classes: V when the prepositional phrase is attached to the verb and N when the prepositional phrase is attached to the preceding head noun.

## 3 Data

To be able to extract the required features from a dataset instance, one must identify the verb, the phrase identifying the object of the verb that precedes the prepositional phrase in question (*np1*) which usually is part of the predicate-argument structure of the verb, its head noun, the prepositional phrase (*np2*), its preposition and its head noun (the second most important word in the PP).

We have adopted the notation from (Collins and Brooks, 1995), where *v* is the verb, *n*<sub>1</sub> is the head noun of object phrase, *p* is the preposition and *n*<sub>2</sub> is the head noun of the prepositional phrase.

Compared to our datasets, Ratnaparkhi's dataset (Ratnaparkhi et al., 1994) contains only the lexical heads *v*, *n*<sub>1</sub>, *p* and *n*<sub>2</sub>. Thus, our methodology cannot be applied to Ratnaparkhi's dataset (RRR).

In our experiments we used two datasets:

- **FN** – extracted from FrameNet II 1.1 (Baker et al., 1998)
- **TB2** – extracted from Penn Treebank-II

Table 1 presents the datasets<sup>1</sup>. The creation of the datasets is described in details in (Olteanu, 2004).

## 4 Features

The experiments described in this paper use a set of discrete (alphanumeric) and continuous (numeric) features. All features are fully deterministic, except the features **count-ratio** and **pp-count** that are based on information provided by an external resource - Google search engine (<http://www.google.com>).

In describing the features, we will use the Penn Treebank-II parse tree associated with the sentence “*The Lorillard spokeswoman said asbestos was used in “very modest amounts” in making paper for the filters in the early 1950s and replaced with a different type of filter in 1956*”.

Table 2 describes the features and the origin of each feature. The preposition is the feature with the most discriminative power, because of preferences of particular prepositions to attach to verbs or nouns. Table 3 shows the distribution of top 10 most frequently used prepositions in the FN and TB2 datasets.

The features were carefully designed so that, when they are extracted from gold parse trees, they don't provide more information useful for disambiguation than when they are automatically generated using a parser. This claim is validated by the experimental results that show a strong correlation between the results on the two datasets – one based on automatically generated parse trees (FN) and one based on gold parse trees (TB2).

Next, we describe in further detail the features presented in Table 2.

**v-frame** represents the *frame of the verb* – the frame to which the verb belongs, as it is present in FrameNet (manually annotated). We used this feature because the frame of the verb describes very well the semantic behavior of the verb including the predicate-argument structure of the verb, which entails the affinity of the verb for certain prepositions.

<sup>1</sup>The datasets are available at <http://www.utdallas.edu/~mgo031000/ppa/>

	FN	TB2
Source	FrameNet annotation samples (British National Corpus)	Penn Treebank-II (WSJ articles)
Instance identification	Semantic-centered (related to Frame Elements)	Syntactic-centered (related to the structure of the parse tree)
Parse trees	Automatically generated (Charniak)	Gold standard
Total size	27,421 instances	60,699 instances
Distribution statistics	70.28% ambiguous verb attachments 2.36:1 v-attach:n-attach	35.71% ambiguous verb attachments 1:1.8 v-attach:n-attach
Training / test sets	90% - 10% – homogenously distributed (one in every 10 instances is selected for the test set)	
Location of PP	Both before and after verb	Only after verb
Other properties	<ul style="list-style-type: none"> <li>– Partial identification of ambiguous PP-attachment instances in the corpus, derived from manual annotation of FEs (Olteanu, 2004)</li> <li>– Semantic information readily available</li> </ul>	

Table 1: The datasets and their characteristics

Feature: description [origin]
<b>v-surface</b> : surface form of the verb [Hindle'93, ...]
<b>n1-surface</b> : surface form of $n_1$ . May be morphologically processed [Hindle'93, ...]
<b>p</b> : the preposition, lower-cased [Hindle'93, ...]
<b>n2-surface</b> : surface form of $n_2$ . May be morphologically processed [Ratnaparkhi'94, Collins'95, ...]
<b>n1-mp/n1-mpf</b> : morph. processing of $n_1$ [Collins'95]
<b>n2-mp/n2-mpf</b> : morph. processing of $n_2$ [Collins'95]
<b>v-lemma</b> : lemma of the verb [Collins'95]
<b>path</b> : path in the candidate parse tree between the verb and $np1$ [Gildea'02]
<b>subcategorization</b> : subcategorization of the verb [modified from Pradhan'03]
<b>v-pos</b> : part-of-speech of the verb
<b>v-voice</b> : voice of the verb
<b>n1-pos</b> : part-of-speech of $n_1$
<b>n1-lemma</b> : lemma of $n_1$ . May be morphologically processed
<b>n2-pos</b> : part-of-speech of $n_2$
<b>n2-lemma</b> : lemma of $n_2$ . May be morphologically processed
<b>position</b> : position of $np1$ relative to the verb [new]
<b>v-frame</b> : frame of the verb [new in PPA]
<b>n1-sr</b> : semantic role of $np1$ [new in PPA]
<b>n1-tr</b> : thematic role of $np1$ [new in PPA]
<b>n1-preposition</b> : preposition that heads $np1$ , if $np1$ is a PP [new]
<b>n1-parent</b> : label of the parent of $np1$ in the candidate parse tree [new in PPA]
<b>n1-np-label</b> : label of $np1$ in the candidate parse tree [new in PPA]
<b>n2-det</b> : determination of $np2$ [new]
<b>parser-vote</b> : choice of the automatic parser in attaching PP [new in PPA]
<b>count-ratio</b> : WWW statistics about verb-attachment vs. noun-attachment for that particular instance [new]
<b>pp-count</b> : WWW statistics about co-occurrence of $v$ and $n_2$ [new]
<b>n1-p-distance</b> : the distance between $n_1$ and $p$ [new]

Table 2: Features

Prep.	% of FN	% v-att FN	% of TB2	% v-att TB2
of	13.47%	6.17%	30.14%	2.74%
to	13.27%	80.14%	9.55%	60.49%
in	12.42%	73.64%	16.94%	42.58%
for	6.87%	82.44%	8.95%	39.72%
on	6.21%	75.51%	5.16%	47.73%
with	6.17%	86.30%	3.79%	46.92%
from	5.37%	75.90%	5.76%	52.76%
at	4.09%	76.63%	3.21%	66.02%
as	3.95%	86.51%	2.49%	51.69%
by	3.53%	88.02%	3.27%	68.11%

Table 3: Distribution of the first 10 most-frequent prepositions in the FN and TB2 datasets

**n1-sr** represents the *semantic role of the object phrase  $np1$*  – the label attached to the Frame Element (manual semantic annotation that can be found in FrameNet). This feature was introduced because of the relation between the underlying meaning of  $np1$  and its semantic role.

**n1-tr** represents the *thematic role of the object phrase  $np1$*  – a coarse-grained role based on the label attached to the Frame Element (manual semantic annotation that can be found in FrameNet). It was introduced to reduce data sparseness for the **n1-sr** feature. The conversion from fine-grained semantic role to coarse-grained semantic role is done automatically using a table that maps a pair of a frame-level semantic role (FE label) and a frame to a thematic role.

**subcategorization** contains a semi-lexicalized description of the structure of the verb phrase. A subcategorization frame is closely related to the

predicate argument structure and to the underlying meaning of the verb. It contains an ordered set of all the phrase labels that are siblings of the verb, plus a marker for the verb. If the child phrase of the verb is a PP, then the label will also contain the preposition (the headword of the PP). This feature is a modified form of the sub-categorization feature described in (Pradhan et al., 2003): the differences in various part-of-speeches for the verb were ignored and the preposition that heads a prepositional phrase is also attached to the label. Therefore, for the sentence “*The stock declined in June by 4%*”, the value for this feature is *\*-PPin-PPby*.

In the TB2 dataset the parse trees are gold standard (contain the expected output value for PP-ambiguity resolution). In the case of a verb attachment, if the selected PP is a child of the selected VP, then by applying the algorithm, the value of the feature will contain the PP label plus the preposition. This clearly is a clue for the learner that the instance is a *verb attachment*. To overcome this problem for datasets based on gold-standard parse trees, when computing the value of the **subcategorization** feature the selected PP will not be used. Figure 1 shows the subcategorization for the phrase “*replaced with a different type of filter in 1956*”.

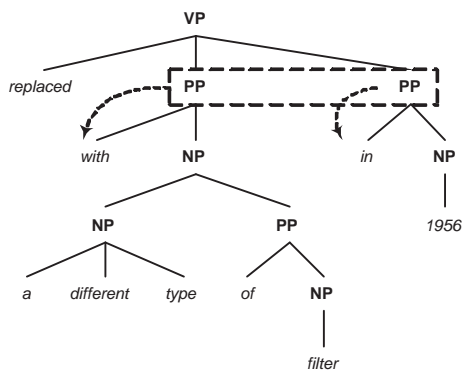


Figure 1: Subcategorization feature: *\*-PPin-PPby*

**path** expresses the syntactic relation between the verb *v* and the object phrase *np1*. Its purpose is to describe the syntactic relation of *np1* to the rest of the clause by the syntactic relation of *np1* with the head of the clause – *v*. We adopted this feature from (Gildea and Jurafsky, 2002). **path** describes the chain of labels in the tree from *v* to *np1*, includ-

ing the label of *v* and *np1*. Ascending movements and descending movements are depicted separately. We used two variants of this feature to determine the optimum version for our problem – one with full POS of the verb and one with POS reduced to “VB”. The experiments proved that the second variant provides a better performance. Figure 2 depicts the path between “replaced” and “a different type of filter”:  $VP \uparrow NP \downarrow PP \downarrow NP$  or  $VB \uparrow VP \downarrow PP \downarrow NP$ .

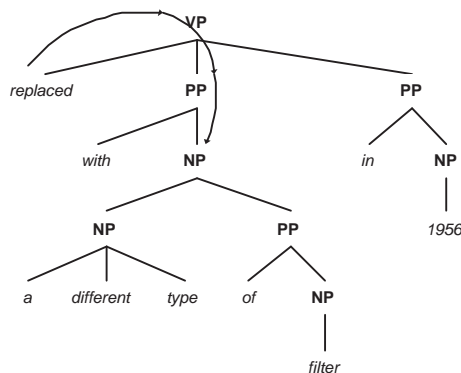


Figure 2: Example of a path feature

**position** indicates the *position of the  $n_1$ - $p$ - $n_2$  construction relative to the verb*, i.e. whether the prepositional phrase in question lies before the verb or after the verb in the sentence. Position is very important in deciding the type of attachment, considering the totally different distribution of PPs constructions preceding the verb and PPs constructions following the verb.

Morphological processing applied to  $n_1$  and  $n_2$  was inspired by the algorithm described in (Collins and Brooks, 1995). We analyzed the impact of different levels of morphological processing by using two types: partial morphological processing (only numbers and years are converted) – identified by adding **-mp** as a suffix to the name of this feature – and full morphological processing (numbers, years and capitalized names) – identified by adding **-mpf** as a suffix to the name of this feature. The purpose of morphological processing is data sparseness reduction by clustering similar values for this feature.

**n1-parent** represents the phrase label of the parent of *np1* and it cannot be used on gold parse trees (TB2 dataset) because it will provide a clue about the correct attachment type.

**n2-det** is called the *determination of the prepositional phrase np2*. This novel feature tells if  $n_2$  is preceded in  $np2$  by a possessive pronoun or by a determiner. This is used to differentiate between “*buy books for children*” (which is probably a noun attachment) and “*buy books for her children*” (which very probably is a verb attachment).

**parser-vote** feature represents the choice of the parser (Charniak’s parser) in the PP-attachment resolution. It cannot be used with gold-standard parse trees because it will provide the right answer.

**count-ratio** represents the estimated ratio between the frequency of an unambiguous verb attachment construction based on  $v$ ,  $p$  and  $n_2$  and the frequency of a probably unambiguous noun attachment construction based on  $n_1$ ,  $p$  and  $n_2$  in a very large corpus. A very large corpus is required to overcome the data sparseness inherent for complex constructions like those described above.

We chose the World Wide Web as a corpus and Google as a query interface (see (Olteanu, 2004) for details).

Let’s consider the estimated frequency of unambiguous verb-attachments and respectively noun-attachments defined as:

$$f_v = \frac{c_{v-p-n_2}}{c_v \cdot c_{p-n_2}}$$

$$f_n = \frac{c_{n_1-p-n_2}}{c_{n_1} \cdot c_{p-n_2}}$$

where:

- $c_{v-p-n_2}$  is the number of occurrences of the phrase “ $v p n_2$ ”, “ $v p * n_2$ ” (where \* symbolizes any word), “ $v$ -lemma  $p n_2$ ” or “ $v$ -lemma  $p * n_2$ ” in World Wide Web, as reported by Google
- $c_v$  is the number of occurrences of the word “ $v$ ” or “ $v$ -lemma” in WWW
- $c_{p-n_2}$  is the number of occurrences of the phrase “ $p n_2$ ” or “ $p * n_2$ ” in WWW
- $c_{n_1-p-n_2}$  is the number of occurrences of the phrase “ $n_1 p n_2$ ” or “ $v p * n_2$ ” in WWW
- $c_{n_1}$  is the number of occurrences of the word “ $n_1$ ” in WWW

The value for this feature is:

$$\text{count - ratio} = \log_{10} \frac{f_v}{f_n} = \log_{10} \frac{c_{v-p-n_2} \cdot c_{n_1}}{c_{n_1-p-n_2} \cdot c_v}$$

We chose logarithmic values for this feature because experiments showed that logarithmic values provide a higher accuracy than linear values. Also, by experimentation we concluded that value bounding is helpful, and the feature was bounded to values between -3 and 3 on the logarithmic scale, unless specified otherwise in the experiment description.

This feature resembles the approach adopted in (Volk, 2001).

**pp-count** depicts the estimated count of occurrences in World Wide Web of the prepositional phrases based on  $p$  and  $n_2$ . The count is estimated by  $c_{p-n_2}$ . Therefore **pp-count** =  $\log_{10}(c_{p-n_2} + c_{p-*n_2})$ .

**n1-p-distance** depicts the distance (in tokens) between  $n_1$  and  $p$ . Let  $d_{n_1-p}$  be the distance between  $n_1$  and  $p$  ( $d = 1$  if there is no other token between  $n_1$  and  $p$ ). Thus **n1-p-distance** =  $\log_{10}(1 + \log_{10} d_{n_1-p})$ .

## 5 Learning model and procedure

We used in our experiments a Support Vector Machines learner with Radial Basis Function kernel as implemented in the LIBSVM toolkit (<http://www.csie.ntu.edu.tw/~cjlin/libsvm/>).

We converted the feature tuples (containing discrete alphanumeric and continuous values) to multi-dimensional vectors using the following procedure:

- Discrete features: assign to each possible value of each feature a dimension in the vector space, and to each feature value in each training or test example put 1 in the dimension corresponding to the feature value and 0 in all other dimensions associated with that feature.
- Continuous features: assign a dimension and put the scaled value in the multi-dimensional vector (all examples in training data will span between 0 and 1 for that particular dimension).

SVM training was preceded by finding the optimal  $\gamma$  and  $C$  parameters required for training using 2-fold cross validation, which was found to be superior in model accuracy and training time over higher folds cross-validations (Olteanu, 2004).

The criterion for selecting the best set of features was the accuracy on the cross-validation. Thus, the development of the models was performed entirely

on the training set, which acted also as a development set. We later computed the accuracy on the test set on some representative models.

## 6 Experiments, results and analysis

For each dataset, we conducted experiments to determine an efficient combination of features and the accuracy on test data for the best combination of features. We also run the experimental procedure on the original Ratnaparkhi’s dataset in order to compare SVM with other machine learning techniques applied to PP-attachment problem. Table 4 summarizes the experiments performed on all datasets.

Experiment	% on dev / x-val	% on test
<i>FN-basic-flw</i>	86.25	86.44
<i>FN-lex-syn-flw</i>	88.55	89.61
<i>FN-best-no-sem</i>	90.93	91.79
<i>FN-best-sem</i>	91.87	92.85
<i>TB2-basic</i>	85.75	87.47
<i>TB2-best-no-www</i>	92.06	92.81
<i>TB2-best</i>	92.92	93.62
<i>RRR-basic</i>	84.32	84.60
<i>RRR-basic-mpf</i>	84.34	85.14

Table 4: Results

*FN-basic-flw* uses **v-surface**, **n1-surface**, **p** and **n2-surface** on examples that follow the verb. *FN-lex-syn-flw* uses **v-surface**, **v-pos**, **v-lemma**, **subcategorization**, **path** (full POS), **position**, **n1-preposition**, **n1-surface**, **n1-pos**, **n1-lemma**, **n1-parent**, **p**, **n2-surface**, **n2-pos**, **n2-lemma**, **n2-det** and **parser-vote** on examples that follow the verb. *FN-best-no-sem* uses **v-surface**, **v-pos**, **v-lemma**, **subcategorization**, **path** (reduced POS), **position**, **n1-preposition**, **n1-surface**, **n1-pos**, **n1-lemma-mpf**, **n1-parent**, **p**, **n2-surface**, **n2-pos**, **n2-lemma-mpf**, **n2-det**, **parser-vote**, **count-ratio** and **pp-count** on all examples. *FN-best-sem* uses the same set of features as *FN-best-no-sem* plus **v-frame** and **n1-sr**.

*TB2-basic* uses **v-surface**, **n1-surface-mpf**, **p** and **n2-surface-mpf**. *TB2-best-no-www* uses **v-surface**, **v-pos**, **v-lemma**, **subcategorization**, **path** (reduced POS), **n1-preposition**, **n1-surface**, **n1-mpf**, **n1-pos**, **n1-lemma**, **n1-np-label**, **p**, **n2-surface**, **n2-mpf** and **n1-p-distance**. *TB2-best* also uses **count-ratio** and **pp-count**.

*RRR-basic* uses **v-surface**, **n1-surface**, **p** and **n2-surface**. *RRR-basic-mpf* uses **v-surface**, **n1-surface-mpf**, **p** and **n2-surface-mpf**.

On the FN dataset, all features except **v-voice** have a positive contribution to the system (**n2-det**, choice between semantic vs. thematic role and how should morphological processing be applied is questionable). The negative impact for the **v-voice** feature may be explained by the fact that the only situation in which it may potentially help is extremely rare: passive voice and the agent headed by “by” appears after another argument of the verb (i.e.: “*The painting was presented to the audience by its author.*”). Moreover the PP-attachment based on the preposition “by” is not highly ambiguous; as seen in Table 3 in the FrameNet dataset, 88% of the “by” ambiguity instances are verb-attachments.

The experiment with the highest cross-validation accuracy has an accuracy of 92.85% on the test data. The equivalent experiment that doesn’t include manually annotated semantic information has an accuracy of 91.79% on the test data.

On TB2 dataset, the results are close to the results obtained on the FrameNet corpus, although the distribution of noun and verb attachment differs considerably between the two data sets (70.28% are verb-attachments in FN2 and 35.71% in TB2). The best accuracy in cross-validation is 92.92%, which leads to an accuracy on test set of 93.62%.

## 7 Comparison with previous work

Because we couldn’t use the standard dataset used in PP-attachment resolution (Ratnaparkhi’s), we implemented back-off algorithm developed by Collins and Brooks (1995) and applied it to our TB2 dataset. Both RRR and TB2 datasets are extracted from Penn Treebank. This algorithm, trained on TB2 training set, obtains an accuracy on TB2 test set of 86.1% (85.8% when no morphological processing is applied). The same algorithm provides an accuracy on RRR dataset of 84.5% (84.1% without morphological processing). The difference in accuracy between the two datasets is 1.6% (1.7% without morphological processing when using Collins and Brooks’s algorithm).

The difference in accuracy between a SVM model applied to RRR dataset (*RRR-basic* experiment) and the same experiment applied to TB2 dataset (*TB2-*

Description	Accuracy	Data	Extra Supervision
Always noun	55.0	RRR	
Most likely for each P	72.19	RRR	
<b>Most likely for each P</b>	<b>72.30</b>	<b>TB2</b>	
<b>Most likely for each P</b>	<b>81.73</b>	<b>FN</b>	
Average human, headwords (Ratnaparkhi et al., 1994)	88.2	RRR	
Average human, whole sentence (Ratnaparkhi et al., 1994)	93.2	RRR	
Maximum Likelihood-based (Hindle and Rooth, 1993)	79.7	AP	
Maximum entropy, words (Ratnaparkhi et al., 1994)	77.7	RRR	
Maximum entropy, words & classes (Ratnaparkhi et al., 1994)	81.6	RRR	
Decision trees (Ratnaparkhi et al., 1994)	77.7	RRR	
Transformation-Based Learning (Brill and Resnik, 1994)	81.8		WordNet
Maximum-Likelihood based (Collins and Brooks, 1995)	84.5	RRR	
<b>Maximum-Likelihood based (Collins and Brooks, 1995)</b>	<b>86.1</b>	<b>TB2</b>	
Decision trees & WSD (Stetina and Nagao, 1997)	88.1	RRR	WordNet
Memory-based Learning (Zavrel et al., 1997)	84.4	RRR	LexSpace
Maximum entropy, unsupervised (Ratnaparkhi, 1998)	81.9		
Maximum entropy, supervised (Ratnaparkhi, 1998)	83.7	RRR	
Neural Nets (Alegre et al., 1999)	86.0	RRR	WordNet
Boosting (Abney et al., 1999)	84.4	RRR	
Semi-probabilistic (Pantel and Lin, 2000)	84.31	RRR	
Maximum entropy, ensemble (McLauchlan, 2001)	85.5	RRR	LSA
SVM (Vanschoenwinkel and Manderick, 2003)	84.8	RRR	
Nearest-neighbor (Zhao and Lin, 2004)	86.5	RRR	DWS
<b>FN dataset, w/o semantic features (FN-best-no-sem)</b>	<b>91.79</b>	<b>FN</b>	<b>PR-WWW</b>
<b>FN dataset, w/ semantic features (FN-best-sem)</b>	<b>92.85</b>	<b>FN</b>	<b>PR-WWW</b>
<b>TB2 dataset, best feature set (TB2-best)</b>	<b>93.62</b>	<b>TB2</b>	<b>PR-WWW</b>

Table 5: Accuracy of PP-attachment ambiguity resolution (our results in bold)

basic experiment) is 2.9%. Also, the baseline – the most probable PP type for each preposition – is approximately the same for the two datasets (72.19% on RRR and 72.30% on TB2).

One may hypothesize that the majority of the algorithms for PP-attachment disambiguation obtain no more than 4% increase in accuracy on the TB2 compared to the results on the RRR dataset. One important difference between the two datasets is the size – 20,801 training examples in RRR vs. 54,629 training examples in TB2. We plan to implement more algorithms described in literature in order to verify this statement.

Table 5 summarizes the results in PP-attachment ambiguity resolution found in literature along with our best results.

Other acronyms used in this table:

- AP – dataset of 13 million word sample of Associated Press news stories from 1999 (Hindle and Rooth, 1993).
- LexSpace - Lexical Space – a method to measure the similarity of the words (Zavrel et al., 1997).

- LSA – Latent Semantic Analysis – measure the lexical preferences between a preposition and a noun or a verb (McLauchlan, 2001)
- DWS – Distributional Word Similarity. Words that tend to appear in the same contexts tend to have similar meanings (Zhao and Lin, 2004)
- PR-WWW – the probability ratio between verb-preposition-noun and noun-preposition-noun constructs measured using World Wide Web searching.

## 8 Conclusions

The Penn Treebank-II results indicate that the new features used for the disambiguation of PP-attachment provide a very substantial improvement in accuracy over the base line (from 87.48% to 93.62%). This represents an absolute improvement of approximately 6.14%, equivalent to a 49% error drop. The performance of the system on Penn Treebank-II exceeds the reported human expert performance on Penn Treebank-I (Ratnaparkhi et al., 1994) by about 0.4%. A significant improvement comes from the unsupervised information collected

from a very large corpus; this method proved to be efficient to overcome the data sparseness problem.

By analyzing the results from the FrameNet dataset, we conclude that the contribution of the gold semantic features (frame and semantic role) is significant (1.05% difference in accuracy; 12.8% reduction in the error). We will further investigate this issue by replacing gold semantic information with automatically detected semantic information. Our additional lexico-syntactic features increase the accuracy of the system from 86.44% to 89.61% for PPs following the verb. This suggests that on the FrameNet dataset the proposed syntactic features have a considerable impact on the accuracy.

The best TB2 feature set is approximately the same as the best FN feature set in spite of the differences between the datasets (Parse trees: TB2 – gold standard; FN – automatically generated. PP-attachment ambiguity identification: TB2 – parse trees; FN – a combination of trees and FE annotation. Data source: TB2 – WSJ articles; FN – BNC). This fact suggests that the selected feature sets do not exploit particularities of the datasets and that the features are relevant to the PP-attachment ambiguity problem.

## References

- Steven Abney, Robert E. Schapire, and Yoram Singer. 1999. Boosting applied to tagging and PP Attachment. In *Proceedings of EMNLP/VLC-99*, pages 38–45.
- Martha A. Alegre, Josep M. Sopena, and Agusti Lloberas. 1999. Pp-attachment: A committee machine approach. In *Proceedings of EMNLP/VLC-99*, pages 231–238.
- Collin F. Baker, Charles J. Fillmore, and John B. Lowe. 1998. The Berkeley FrameNet Project. In *Proceedings of the 17th international conference on Computational Linguistics*, pages 86–90.
- Eric Brill and Philip Resnik. 1994. A rule-based approach to prepositional phrase attachment disambiguation. In *Proceedings of the 15th conference on Computational Linguistics*, pages 1198–1204.
- Eugene Charniak. 2000. A Maximum-Entropy-Inspired Parser. In *Proceedings of NAACL-2000*, pages 132–139.
- Michael Collins and James Brooks. 1995. Prepositional Phrase Attachment through a Backed-Off Model. In *Proceedings of the Thirds Workshop on Very Large Corpora*, pages 27–38.
- Corinna Cortes and Vladimir Vapnik. 1995. Support-Vector Networks. *Machine Learning*, 20(3):273–297.
- Daniel Gildea and Daniel Jurafsky. 2002. Automatic Labeling of Semantic Roles. *Computational Linguistics*, 28(3):245–288.
- Donald Hindle and Mats Rooth. 1993. Structural Ambiguity and Lexical Relations. *Computational Linguistics*, 19(1):103–120.
- Mitchell Marcus, Beatrice Santorini, and Mary Ann Marcinkiewicz. 1993. Building a large annotated corpus of English: the Penn Treebank. *Computational Linguistics*, 19(2):313–330.
- Mark McLauchlan. 2001. Maximum Entropy Models and Prepositional Phrase Ambiguity. Master’s thesis, University of Edinburgh.
- Marian G. Olteanu. 2004. Prepositional Phrase Attachment ambiguity resolution through a rich syntactic, lexical and semantic set of features applied in support vector machines learner. Master’s thesis, University of Texas at Dallas.
- Patrick Pantel and Dekang Lin. 2000. An unsupervised approach to Prepositional Phrase Attachment using contextually similar words. In *Proceedings of the 38th Meeting of the Association for Computational Linguistic*, pages 101–108.
- Sameer Pradhan, Kadri Hacioglu, Wayne Ward, James H. Martin, and Daniel Jurafsky. 2003. Semantic Role Parsing: Adding Semantic Structure to Unstructured Text. In *Proceedings of the International Conference on Data Mining*, pages 629–632.
- Adwait Ratnaparkhi, Jeff Reynar, and Salim Roukos. 1994. A Maximum Entropy Model for Prepositional Phrase Attachment. In *Proceedings of the Human Language Technology Workshop*, pages 250–255.
- Adwait Ratnaparkhi. 1998. Statistical Models for Unsupervised Prepositional Phrase Attachment. In *Proceedings of the 36th conference on Association for Computational Linguistics*, pages 1079–1085.
- 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*, pages 66–80.
- Bram Vanschoenwinkel and Bernard Manderick. 2003. A weighted polynomial information gain kernel for resolving Prepositional Phrase attachment ambiguities with Support Vector Machines. In *Proceedings of the Eighteenth International Joint Conference on Artificial Intelligence*, pages 133–140.
- Martin Volk. 2001. Exploiting the WWW as a corpus to resolve PP attachment ambiguities. In *Proceedings of Corpus Linguistics*, pages 601–606.
- Jakub Zavrel, Walter Daelemans, and Jorn Veenstra. 1997. Resolving PP attachment Ambiguities with Memory-Based Learning. In *Proceedings of CoNLL-97*, pages 136–144.
- Shaojun Zhao and Dekang Lin. 2004. A Nearest-Neighbor Method for Resolving PP-Attachment Ambiguity. In *Proceedings of IJCNLP-04*.