Pattern Dictionary of English Prepositions

Ken Litkowski

CL Research
9208 Gue Road
Damascus, Maryland 20872 USA
301-482-0237
ken@clres.com

Abstract We present a lexical resource for the study of preposition behavior, the Pattern Dictionary of English Prepositions (PDEP). This dictionary, which follows principles laid out in Hanks' theory of norms and exploitations, is linked to 81,509 sentences for 304 prepositions, which have been made available under The Preposition Project (TPP). Notably, 47,285 sentences, initially untagged, provide a representative sample of preposition use, unlike the tagged sentences used in previous studies. Each sentence has been parsed with a dependency parser and our system has near-instantaneous access to features developed with this parser to explore and annotate properties of individual senses. With completion of tagging and sense classification of these sentences, a more complete picture of preposition disambiguation emerges. We present the results of baseline support-vector machine modeling and consider several avenues of further investigation to improve results. While preposition disambiguation remains a challenge, we believe the results described here provide a stepping point for further investigations, particularly for application areas such as semantic role labeling, temporal evaluation, spatial evaluation, and ontology population. We suggest that further efforts will benefit from sense-based and class-based analyses of the preposition data.

1 Introduction

Preposition use in English is notoriously idiosyncratic, posing many difficulties for learning the language. The more common prepositions are highly polysemous, making it even more difficult to treat these words in computational processing. Because of their high frequency, and their relatively low semantic content, they are often treated as stop words, with little analysis. However, by their very definition, they are bearers of semantic relations between discourse elements, comprehensively so. As a result, it is important to have a comprehensive picture of their use.

Section 2 describes The Preposition Project (TPP) and its use in large-scale studies of preposition disambiguation, noting early successes, but then revealing shortcomings. In section 3, we describe the Pattern Dictionary of English Prepositions (PDEP), intended to investigate the difficulties with a more representative sample of preposition use. This section also describes the PDEP interface and the expansion of the database to allow for interactive examination of the data. Section 4 describes support vector machine modeling of the PDEP data, for internal robustness and as applied to two test sets. In section 5, we describe our examination of features generated in modeling, noting the considerable redundancy using recursive feature elimination and feature ablation. We also describe our efforts to analyze features to build preposition word sketches. Section 6 describes our extension of the lexical resources used in feature generation, beyond WordNet to include FrameNet, VerbNet, and the Pattern Dictionary of English Verbs. Section 7 describes our analysis of preposition classes, including their utility in SemEval evaluations, their use in coarse-grained disambiguation, and feature analysis across prepositions.

2 The Preposition Project

Litkowski and Hargraves (2005) and Litkowski and Hargraves (2006) describe The Preposition Project (TPP) as an attempt to describe preposition behavior using a sense inventory made available for public use from the *Oxford Dictionary of English* (Stevenson and Soanes, 2003) by tagging sentences drawn from FrameNet. The TPP data includes, for each sense, the definition, a qualitative description of the object (complement) and attachment point (governor), a characterization of the semantic relation expressed by the sense, a list of similar prepositions, its syntactic behavior (noun postmodifier, adjunct, and verb or adjective complement), and other comments. Where FrameNet instances were not available, other lexical resources, including the Sketch Engine (SE, Kolgarriff et al., 2004), were used to characterize the prepositions, particularly multiword phrases.

The FrameNet sentences were used as the basis for a preposition disambiguation task in SemEval 2007 (Litkowski and Hargraves, 2007). Initial results in SemEval achieved a best accuracy of 69.3 percent (Ye and Baldwin, 2007). The data from SemEval was subsequently been used in several further investigations of preposition disambiguation. Most notably, Tratz (2011) achieved a result of 88.4 percent accuracy and Srikumar and Roth (2013) achieved a similar result.

With these positive results, we wanted to pursue our investigations, beginning with the development of the TPP Corpora (Litkowski, 2013a), consisting of three sets. The first was all FrameNet sentences (57 prepositions, 26739 instances), not just those used in SemEval (24 prepositions, divided into training and test sets). The second was a set of 20 sentences drawn from the Oxford English Corpus (OEC) to exemplify each sense in ODE, notably providing instances for multiword prepositional phrases (7485 sentences). The third was a set of sentences from the written portion of the British National Corpus, drawn with methodology used in the Corpus Pattern Analysis (CPA) project, typically with 250 instances for each preposition (47258 sentences). Each corpus was put into SemEval format. In addition, each sentence was lemmatized, part-of-speech tagged, and parsed with Tratz's dependency parser, in the CoNLL-X format (Tratz and Hovy, 2011).

After implementing the Tratz-Hovy system locally, we re-implemented the preposition disambiguation portion, including modules to extract features from the parses, train support-vector machine (SVM) models from these features, and apply the models to the SemEval test set. We thus replicated Tratz's results. We next applied these models to the OEC corpus we had created. We found that the accuracy of the models applied to this corpus were significantly lower, with a reduction to 39.4 percent. Litkowski (2013b) describes these results in detail. In addition to the main result, we examined other ways of looking at the data, including an examination of Tratz's refined sense inventory and use of the OEC corpus as a training set applied to the SemEval corpus. We concluded that the FrameNet-based corpus may not have been representative. This led to the development of PDEP.

3 The Pattern Dictionary of English Prepositions (PDEP)

PDEP was designed primarily to facilitate the tagging of the 47285 CPA instances with TPP senses. It consists of an online interface to an underlying database containing information about each preposition sense, with links to each of the sentences in the TPP corpora.[1] Litkowski (2104) provides a detailed description of PDEP; here, we provide an overview of components salient to the discussion below.

PDEP is intended to identify the prototypical syntagmatic patterns with which prepositions in use are associated, identifying linguistic units used sequentially to make well-formed structures and characterizing the relationship between these units. These principles are described more fully in Hanks (2013). PDEP is modeled on the principles of Corpus Pattern Analysis (CPA), developed to characterize syntagmatic patterns for verbs.[2] In PDEP, the core pattern for each sense is presented as an instance of the template **[[Governor]] prep [[Complement]]**, followed by its primary implicature, where the current definition is substituted for the preposition. The objective is to specify the details of the complement and the governor.

PDEP follows the principles laid down in the CPA project, reusing much of its underlying code. The PDEP CPA corpus was drawn from the same BNC database as used in the CPA project, using the SE to select a sample of each preposition. The sample size was generally 250, with all instances when the total number of instances

[1] 1 http://www.clres.com/db/TPPEditor.html

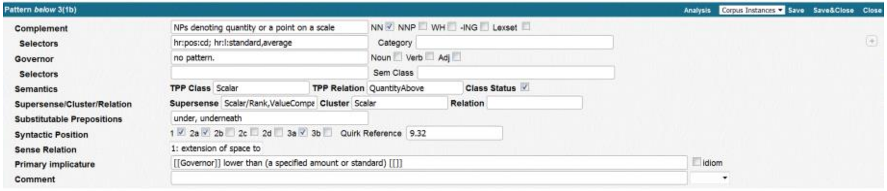
[2] See http://nlp.fi.muni.cz/projects/cpa/.

was fewer than 250 and with 500 or 750 instances for 13 common and highly polysemous prepositions such as *of* or *with*. When these samples were drawn, we noted the total BNC frequency. The BNC frequency provides a basis for extrapolating results from PDEP to the totality of prepositions. In total, the number of instances in the BNC is 5,391,042, which can be used as the denominator when examining the relative frequency of any preposition (e.g., *between* has a frequency of 0.0109, 58,865/5,391,042).

3.1 The PDEP Interface and Database

The PDEP interface consists of four main components. The entry screen shows a list of all the prepositions with the number of senses for each and the number of instances in each of the TPP corpora. When selecting a preposition, the user is presented with a list of the senses for that preposition written in the form of the general syntagmatic pattern. The user can select any sense (pattern) to display the properties of that sense, as present in the underlying database, as shown in Figure 1 for sense 3(1b) of *below*. From either of these two components, the user can examine the instances from any of the TPP corpora, either *in toto* or as tagged with the particular sense. Each instance shows the full sentence, with the preposition, the complement, and the governor highlighted in different colors.

Fig.. 1 Preposition Pattern Box for *below* (3(1b))



Pattern *below* 3(1b)

Complement	NPs denoting quantity or a point on a scale	NN ☑ NNP □ WH □ -ING □ Lexset □				
Selectors	hr:pos:cd; hr:l:standard,average	Category				
Governor	no pattern.	Noun □ Verb □ Adj □				
Selectors		Sem Class				
Semantics	**TPP Class**	Scalar	**TPP Relation**	QuantityAbove	**Class Status**	☑
Supersense/Cluster/Relation	**Supersense**	Scalar/Rank,ValueCompe	**Cluster**	Scalar	**Relation**	
Substitutable Prepositions	under, underneath					
Syntactic Position	1 ☑ 2a ☑ 2b □ 2c □ 2d □ 3a ☑ 3b □	Quirk Reference	9.32			
Sense Relation	1: extension of space to					
Primary implicature	[[Governor]] lower than (a specified amount or standard) [[]]	□ idiom				
Comment		▾				

Almost all TPP data have been imported into the PDEP database, including the fields for the **Complement**, the **Governor**, the **TPP Class**, the **TPP Relation**, the **Substitutable Prepositions**, the **Syntactic Position**, the **Quirk Reference**, the **Sense Relation**, and the **Comment**. We have added the checkboxes for complement type (common nouns, proper nouns, WH-phrases, and -ing phrases), as well as a field to identify a particular lexical item (lexset) if the sense is an idiomatic usage. We have added the **Selector** fields for the complement and the governor. For the complement, we have a field **Category** to hold an ontological category for the complement. We also provided a field for the **Semantic Class** of the governor; this field has not yet been implemented.

We have added three **Supersense/Cluster/Relation** fields. The **Cluster** field is based on data available from Tratz (2011), where senses in the SemEval 2007 data have been put into 34 clusters. The **Relation** field is based on data available from

Srikumar and Roth (2013), where senses in the SemEval 2007 data have been put into 32 classes. A key element of Srikumar and Roth was the use of these classes to model semantic relations across prepositions (e.g., grouping all the Temporal senses of the SemEval prepositions). The **Supersense** filed identifies a category from a hierarchy of English prepositions, described in Schneider et al. (2015).[3] Finally, there is a field **Class Status**, used during PDEP tagging to indicate when a sense had been placed in the PDEP class analysis.[4]

3.2 Tagging the PDEP CPA Corpus

In general, sense-tagging followed standard lexicographic principles, where an attempt is made to group instances that appear to represent distinct senses. PDEP provides an enhanced environment for this process. Firstly, we made use of the current TPP sense inventory to tag sentences. Since the pattern sets (definitions) are based on the *Oxford Dictionary of English*, the likelihood that the coverage and accuracy of the sense distinctions is quite high. However, since prepositions have not generally received the close attention of words in other parts of speech, PDEP provided an opportunity to ensure the coverage and accuracy; along with the tagging done for SemEval, PDEP has about 10 percent more senses than the original ODE sense inventory. Secondly, we were able to make use of the existing sense-tagged instances in the FrameNet and OEC TPP corpora.

More significantly, we were able to take advantage of the parses and the features generated for each sentence. To train and develop the SVM models, the Tratz-Hovy system generates features from the dependency parses for each instance. Each sentence may have as many as 1500 features describing the context of the target preposition. The feature files for these sentences are available for exploration in PDEP. Each feature consists of three parts: a word-finding rule, a feature extraction rule, and the value of the feature. In our implementation, we currently use seven word-finding rules and 14 feature extraction rules.

The word-finding rules fall into two groups: words pertaining to the governor and words pertaining to the complement. The five governor word-finding rules are (1) verb or head to the left (**l**), (2) head to the left (**hl**), (3) verb to the left (**vl**), (4) word to the left (**wl**), and (5) governor (**h**). The two complement word-finding rules are (1) syntactic preposition complement (**c**) and (2) heuristic preposition complement (**hr**). The feature extraction rules are (1) word class (**wc**), (2) part of speech (**pos**), (3) lemma (**l**), (4) word (**w**), (5) WordNet lexical name (**ln**), (6) WordNet immediate hypernym (**h**), (7) WordNet synonyms (**s**), (8) all WordNet hypernyms (**ah**), (9) whether the word is capitalized (**c**), (10) affixes (**af**), (11) VerbNet (**vn**),

[3] http://demo.ark.cs.cmu.edu/PrepWiki/index.php/Main_Page

[4] http://www.clres.com/db/classes/ClassAnalysis.php

(12) FrameNet (**fn**), (13) CPA verb (**cpa**), and (14) ODE noun hierarchy immediate hypernym (**nh**).

In PDEP, we are able to examine any of the 114 **wfr:fer** combinations for whatever set of corpus instances happens to be open. When a particular **wfr:fer** combinations is selected, PDEP displays the values for that feature and the count of each value. For most features (e.g., lemma or part of speech), the number of possible values is relatively small, limited by the number of instances in the corpus set. For features such as the WordNet lexical name, synonyms and hypernyms, the number of values may be much larger (e.g., 7500 hypernym values for 250 preposition complements). In addition to the statistics, PDEP also provides the ability to identify and highlight the instances that have a particular feature value. Finally, when a set of instances has been highlighted, either automatically or manually, they can be tagged with one of the senses from the sense inventory for the preposition. All instances in the CPA subcorpus of the TPP corpora have now been tagged.

Using the methods described in this section, many of the fields in the underlying database (as shown in Figure 1) were filled manually. However, at this point, this task is not yet complete.[5] Below, we consider methods for automatic completion of the fields.

3.3 Assignment of Senses to PDEP Classes

As mentioned above, the PDEP pattern box has a checkbox to indicate **Class Status**. As senses were tagged and characterized, each sense was placed into the PDEP class hierarchy. We carefully placed each sense into a preposition class or subclass, grouping senses together and making annotations that attempt to capture any nuance of meaning that distinguishes the sense from other members of the class. To build a description of the class and its subclasses, we made use of the Quirk references in Figure 1, i.e., the relevant discussions in Quirk et al. (1985).

The original TPP classes were used as the starting point for the PDEP class system. Its evolution was driven by the corpus evidence. As we attempted to place senses, we found it necessary to make refinements to the classes, mostly merging minor classes into other classes. In addition, the evidence occasionally required the addition of senses for a preposition or suggested the need to split a sense (also adding a sense). The need for additional senses had also occurred during the tagging of the FrameNet corpus during SemEval. Overall, the original ODE sense inventory has increased by about 10 percent.

[5] All current data are available (http://www.clres.com/elec_dictionaries.html#pdep) as a set of three MySQL files suitable for upload into a MySQL database. These include (1) definitions for all 1040 senses (patterns) of 304 prepositions, (2) properties for each sense in 26 fields, and (3) tagged instances for all sentences in the TPP corpora.

Each class is described on its own web page. The description provides an overview of the class, making use of the TPP data and the Quirk discussion, and Breaks down the class into subclasses, where each sense was placed. The class page has a table that lists all the preposition senses placed into that class, along with a count of the number of CPA instances tagged with the sense and with the normed frequency per million prepositions in the BNC.

PDEP has classified 1015 senses into 12 preposition classes, as shown in Table 1. Two of the classes, Agent and Exception, have been indented in the table, and arguably are subclasses of Cause and Membership, respectively. We suggest that the BNC-based distribution is representative of prepositions use and thus may be used for comparison purposes.

Table 1 Frequency of Preposition Classes.

Class	Senses	Frequency
Activity	35	0.018
Cause	89	0.106
Agent	58	0.117
Backdrop	63	0.034
MeansMedium	94	0.058
Membership	49	0.190
Exception	30	0.004
Scalar	127	0.043
Spatial	251	0.134
Tandem	72	0.081
Temporal	93	0.105
Topic	54	0.110
Total	**1015**	**1.000**

4 Support Vector Machine (SVM) Modeling

With completion of tagging and sense classification in PDEP, a more complete picture of preposition disambiguation emerges. This section describes our use of the tagged data in developing SVM models and applying them to instances for 120 polysemous prepositions. We began our investigation by using the Tratz-Hovy system to generate and extract features for each of the 47285 sentences in the CPA corpus. We then used this system to train SVM models for each of the 120 polysemous PDEP prepositions using the CPA corpus. We examined two parameters for the modeling, the minimum frequency of features and the C parameter for the SVM models, finding stable results with a minimum of 5 features and C = 0.01.[6]

In the first analysis, we tested the SVM models using 10-fold cross-validation on the CPA instances. We achieved 80 percent accuracy on these instances, suggesting that these instances are internally consistent and that the models will generalize to unseen data that is also representative. However, this leaves open the question of how well the models would apply to data that might not be representative.

We applied the models to the OEC and FrameNet corpora as test sets. The accuracy of the models in predicting the senses in the test sets is shown in Table 2, as given in the highlighted row. The other rows of the table show how the SVM results compare to other metrics, based on the number of instances in the two test sets. The baseline accuracy shows the performance when using the most frequent sense in the training data. OneR is the accuracy obtained using the method described in Holte (1993), as implemented in WEKA, i.e., using a very simple classification rule that identifies the feature that has the highest predictive value in the training set.

Table 2. Baseline Test Accuracy of SVM Models

Model	OEC	SemEval
Instances	6111	27069
Baseline	0.249	0.343
OneR	0.316	0.383
SVM Models	**0.492**	**0.460**
OEC using TPP	0.325	
OEC Train		0.386

The last two rows show the results given in Litkowski (2013b) for comparison. The row "OEC using TPP" shows the performance in predicting the OEC instances when the SemEval instances were used for training. For 24 of 31 prepositions, the accuracy of the new SVM models was an improvement. The row "OEC Train" shows the performance when the OEC data were used as the training set in predicting the SemEval instances. For 23 of 33 prepositions, the accuracy of the new SVM models was an improvement.

As can be seen, the SVM models based on the CPA training set yields a definite improvement over the results given in Litkowski (2013b) and above baseline models. However, the results show that the challenge of preposition disambiguation still exists. Also, while in general, results for highly polysemous prepositions have improved, there is still a considerable gap. In machine learning, a general objective is that the training set represent the general population. However, the difficulty is selecting the appropriate component to handle cases like sense 3(1b)-1 of *of*, which represents 0.4 percent of the CPA training corpus, but 6.2 percent of the SemEval corpus.

[6] We did not perform an exhaustive grid search to optimize these values.

5 Feature Analysis

Notwithstanding the drop-off in performance on the test sets, the CPA data provides voluminous sets of features for use in describing the behavior of each preposition. For most prepositions, the 250 instances lead to 75,000 distinct features. For the highly polysemous prepositions, where 500 or 750 instances are available, the number of features reaches 160,000. These numbers raise several questions. How many of these features are important? Are there classes of features that are not really relevant? Are there particular features that are strongly associated with individual senses? Are there particular features that are associated with preposition classes (e.g., across prepositions, as in Srikumar and Roth (2103))? We describe several research threads intended to build a framework for preposition word sketches.

5.1 Recursive Feature Elimination

The application of SVM models to classification essentially entails applying weights to the presence of features. Thus, for a preposition with four senses and 75,000 features, a total score is computed for each sense, and the sense with the highest score is selected. However, the literature on SVMs suggests that no interpretation can be given to the weights. Instead, it has been suggested that examination of the weights might lead to identification of valuable features.

Guyon et al. (2002) suggest that recursive feature elimination (RFE) can be used to identify important features. They use a criterion (squaring and summing the coefficients) to rank the features, removing the lowest ranked features in stages. In their method, they construct the SVM with the full feature set, eliminate the number of features that will reduce the set to the closest power of 2, and then eliminate half of the remaining set in each subsequent iteration, as shown in Algorithm 1.

5.1.1 Steps is Removing Features

The features for each preposition have been generated previously using the Tratz-Hovy parser and feature generation and extraction methods. Step 1 just reads these features. Steps 4 to 6 are also part of the Tratz-Hovy system, using the chi-square metric to create an initial ranking of the features, which are put into a feature dictionary, translated into SVM format and then used to train and test the SVM model. The function **RemoveFeatures** contains a metric applied to the weights of the SVM model. We evaluated four metrics, described in detail below.

Guyon et al. began with 7129 features, iterating 12 times, whereas we usually had 17 iterations (up to 19 for *of*), with the first few involving significantly larger sets. In generating Table 2, we used the core components of steps 4 to 6 of the algorithm, taking roughly 30 minutes for completion of the entire set. Running Algorithm 1 takes approximately 8 to 30 hours for each metric that is used. The output provides a list of the features as they were eliminated as well as the size of the feature set (*F*) and the metric when they were eliminated. The scores on the test sets identifies the accuracy at each size for the OEC and the SemEval instances

```
Algorithm 1 Recursive Feature Elimination
Input: Preposition feature file
Output: Eliminated features and scores of SVM
  models applied to test sets
$F$ = initialize set of features from training set
$n$ = number of iterations, where $n$ is max($n$)
    such that $2^n < |F|$
while $n > 0$ do
  calculate and sort chi-square values for $F$
  create SVM file and train SVM model
  apply SVM model to test sets
  $numToElim = |F| - 2^{n-1}$
  $F = F$ - RemoveFeatures ($numToElim$)
  $n = n - 1$
end while
```

5.1.2 Benefits of Feature Removal

As indicated above, we had modified the minimum feature frequency when investigating the overall accuracy of the initial SVM models to a level where the results seemed to be stable. When examining the results from the Algorithm 1, we found that, not only were we able to achieve comparable results when significantly fewer features were used, but also that significantly higher scores were achieved with significantly fewer features.

We tested four metrics in **RemoveFeatures**: (1) the squares metric as suggested in Guyon et al., (2) the chi-square metric used in the Tratz-Hovy system to rank features before training the SVM models, (3) the sum of the absolute values of the coefficients across the senses, and (4) the “impact”, the sum of the absolute value of the coefficients weighted by their frequency in the training data (as suggested by Yano et al. (2012)). The metrics were evaluated on the proportion of features needed to achieve optimum accuracy on the two test sets. In all cases, the number of features at the optimum was significantly reduced and the accuracy was improved significantly over the level shown in Table 2. The squares metric was the best, with 54.8 percent accuracy using 3.7 percent of the features for the OEC test set and with 49.7 percent accuracy using 9.8 percent of the features for the SemEval test set. These are significantly better than the results shown in Table 2, up 5 points for the OEC test set and up 4 points for the SemEval test set.

5.2 Feature Ablation

RFE identifies the most important features at an optimum level of performance for each preposition, in some cases using only two features, but in other cases still requiring hundreds or thousands of features. We have modified the **RemoveFeatures** component of Algorithm 1 to examine the relative importance of feature sets. As described above, each feature is a **wfr:fer** combination. We remove each **wfr**, each **fer**, and each **wfro:fer** combination in turn and evaluate the effect on performance on the test sets.

We follow the procedures described in Fraser et al. (2014) and Bethard (2007). We identify the dimension to be investigated, e.g., the word-finding rules (**wfr**s). Starting with the full set of features, we remove one set at a time and re-compute and apply the SVM models. We identify the set that shows the smallest decrease in accuracy (since such a set is most redundant), and remove this set permanently. We continue in this fashion until only one set remains. This procedure thus orders the feature sets in terms of their importance. This procedure is run for each of the polysemous prepositions, since each is likely to have its own rankings. This procedure is also time-consuming, particularly when evaluating the 114 **wfr:fer** combinations. We have not yet completed these analyses, but results are likely to be similar to the analyses in Tratz (2011).

5.3 Characterizing Preposition Behavior

Beyond identifying the features that may be most important in classification tasks, the rich set of features generated for both monosemous and polysemous prepositions can be analyzed in detail to complete the fields in the pattern box shown in Figure 1. We have implemented a simple extension in the Tratz-Hovy system that takes a **wfr** and an **fer** as arguments and prints out the distribution of the feature values by sense for the combination, along with the number of instances for each sense.

Examination of these results is not immediate, requiring further analysis, for several reasons. First, the system of feature generation is not completely accurate. The parser may produce spurious results. The word-finding rules may not identify the correct word (e.g., the head of the preposition complement may not be correct). Second, the number of features for a given combination may not correspond to the number of instances for a given sense. In some cases, the number may be fewer, because a feature was not extracted for every instance (e.g., a part of speech or a lemma might be missing). In other cases, such as WordNet synonyms, hypernyms, or lexical names, the number of features may be much larger than the number of instances. For example, the number of hypernyms for 250 preposition complements of *above* is 6835. This suggests the need to develop criteria for inferring the relative prominence of any features for a sense. In addition, based on the discussion below, it appears as if several features may be desirable to characterize a sense, suggesting the need for a structured representation to provide preposition word sketches.

5.3.1 Syntactic Characteristics of the Complement and the Governor

The most basic feature is the part of speech of the preposition complement and the governor (or point of attachment). For the complement, PDEP has five alternatives: common noun, proper noun, WH-clause, gerundial, and lexical set (i.e., a small number of words). In general, we might assume that it will be a common noun. For the other possibilities, we might use a percentage cutoff as the criterion for checking this property in the PDEP interface. In addition, we might use a minimum cutoff to indicate that a particular sense does not involve a common noun.

For the governor, PDEP has three alternatives: noun, verb, and adjective. As a first step, the analysis of the governor's part of speech can proceed as for the complement. PDEP also has seven checkboxes for identifying the syntactic position of the prepositional phrase: noun postmodifier, adverbial adjunct, adverbial subjunct, adverbial disjunct, adverbial conjunct, verb complement, and adjective complement. Again, further analysis seems warranted to study the criteria useful for characterizing syntactic position. This analysis can include examination of other word-finding rules. For example, features are generated for the word to the left of the preposition. When no such word is found, the prepositional phrase begins the sentence (characterized as fronting in Hovy et al. (2010)). When the part of speech is a comma, this suggests that the prepositional phrase is an adverbial adjunct, not essential to the governor.

5.3.2 Semantic Characteristics of the Complement and the Governor

The next set of features to be examined are those concerned with semantic characterization of the complement and the governor (i.e., the **Selector** fields). A basic feature in the Tratz-Hovy system is the WordNet lexicographer file name (**ln**), i.e., 25 noun classes and 15 verb classes. As described in McCarthy et al. (2015), these classes can be viewed as supersense tags that provide semantic word sketches. In the Tratz-Hovy system, all classes for each lemma are captured, without any attempt at disambiguation of the class. The result is that multiple **ln** tags may be generated for the complements and governors. On average, about 30 of the 40 classes were identified for each preposition. However, despite this apparent ambiguity, the pattern of classes can vary substantially for each sense and can be used as an initial basis for a semantic characterization of the sense. For example, sense 14(5) of *over* has the selector **noun.time**, which applies to 99 of the 137 instances tagged with this sense.

We have extended this notion based the two hypernym features (**h**, the WordNet immediate hypernym, and **nh**, the ODE noun hierarchy immediate hypernym). When viewing a frequency distribution for these features, we notice that many items seem to be subclasses of others. In Litkowski (2016), we developed an algorithm that constructs the path to the root of each item and that then collapses these paths to identify the least common subsumer ontological class. This algorithm extends the notion of measuring semantic similarity of two items by identifying their least common subsumer in a hierarchy. In the example for *over*, this algorithm identifies **time period** as the dominant class in the ODE noun hierarchy, covering 86 percent of the items, with the next highest class at 38 percent (**measure**).[7] This analysis can provide an important method of characterizing the semantic properties of the complement and the governor.

6 The Benefit of Additional Lexical Resources

The Tratz-Hovy system makes extensive use of WordNet in feature generation. The main reason, of course, is that WordNet provides wide coverage, is freely available, and contains considerable structural information about lexical items. Use of WordNet facilitates the generation of features for word class, parts of speech, lexicographer groupings of words, synonyms, and hypernyms. These features are of considerable importance in the SVM models.

Many additional lexical resources are under development. The question arises whether these resources provide information that would also be of importance in preposition disambiguation. Our investigations have shown that this is a fruitful area of research. We have investigated several such resources by building dictionaries that have been integrated into the Tratz-Hovy system to generate additional features used in the modeling.

We have examined FrameNet, VerbNet, and the Pattern Dictionary of English Verbs. Each of these resources is recognized as not having the same coverage as WordNet. In addition, except for FrameNet, each generally focuses on verbs, so in our investigation, we used them to generate features only where they were applicable to the governor of the prepositional phrase and where the entry for the governor in the lexical resource identified the preposition as a dependent element.

For the CPA corpus, 1833 FrameNet features, with the frame as the feature value, were generated in 1524 sentences under 61 prepositions. After recursive feature

[7] It should be noted that this class identification is not a direct feature subject to the SVM modeling, but is examined only after features have been generated for all instances that have been tagged with a particular sense.

elimination, 218 of these features were retained for 23 polysemous prepositions. We generated 424 VerbNet features in 350 sentences under 21 prepositions Recursive feature elimination retained 72 features under 8 prepositions. We generated 225 PDEV features in 225 sentences under 34 prepositions. Of these, 25 were retained under 8 prepositions after recursive feature elimination.

As can be seen, inclusion of these other resources seems to have a relatively small, though possibly valuable, contribution to the feature set. Our dictionaries have not fully explored their potential. For example, both FrameNet and VerbNet have hierarchical aspects to them which can be examined. For FrameNet, we can also examine the contribution of frame elements in attempting to characterize the preposition complements.

7 Class Analyses

The preposition senses included in the classes shown in Table 1 provide an opportunity for more targeted investigations, (1) directly as inventories for the individual classes, (2) in studying the granularity of the senses, and (3) in feature analysis across prepositions in the same class.

7.1 Class Sense Inventories

The set of senses in each class may be directly useful in semantic evaluations, such as temporal evaluations, spatial evaluations, and ontology population. These are areas of particular interest in the SemEval[8] semantic evaluation exercises. The TempEval series is designed to extract timelines for events; these tasks can use the Temporal class. The SpaceEval series is designed to identify particular constructions for talking about spatial information; these tasks can use the Spatial class (as e.g., in Dittrich et al. (2015)). The taxonomy series seeks to improve semantic network coverage, taxonomy enrichment, and ontology population; these tasks can use the Membership and Exception classes.

El Maarouf et al. (2015) describe techniques for ontology population. They build on techniques initially developed by Hearst (1992) and Snow et al. (2004). We suggest that the Membership and Exception classes may provide a comprehensive identification of prepositional phrases that can be exploited in taxonomy investigations.

[8] http://alt.qcri.org/semeval2016/

7.2 Coarse-Grained Disambiguation

In developing his preposition disambiguation modules, Tratz (2011) refined the sense inventory used in SemEval 2007. In many cases, the refinements combined two or more senses he deemed to be similar. We begin a more principled approach to this question by using preposition classes as the basis for coarse-grained disambiguation. We examine the use of the preposition classes in Table 1 and the use of preposition supersense tags.

In the PDEP database, each sense has a TPP class, as identified in Table 1. We considered the question of whether use of this class could be viewed as a coarse-grained sense. For example, there are 10 senses for *above*; of these, four are in the Spatial class and six are in the Scalar class. In the fine-grained analysis (Table 2), the accuracy for OEC was 0.295 and for TPP was 0.225. By essentially collapsing the two classes, i.e., viewing *above* as having only two senses, the OEC accuracy was 0.771 and the TPP accuracy was 0.746.

In evaluating the SVM models against the OEC and TPP test sets, the Tratz-Hovy system generates a prediction for each instance, and includes an identification of the correct sense. We used these results to compute overall coarse-grained accuracies. We took the correct sense and identified its class. We then examined the predicted sense and, if it was in the same class, we counted it as correct.

When we used the 12 classes identified in Table 1, we obtained an overall accuracy of 66.85 for OEC (compared to 49.46 for the fine-grained predictions) and 64.82 for TPP (compared to 46.19). By folding Agent and Exception into Cause and Membership, the OEC results improved to 67.35 and the TPP results improved to 65.21. Thus, we obtained a net gain of about 18 percent in the coarse-grained disambiguation. This suggests that using the preposition classes as the senses may provide much improved results for a task using preposition disambiguation.

The PDEP database also contains a field to record preposition supersenses, taken from the preposition hierarchy described in Schneider et al. (2015), who used them to tag 4100 instances in a corpus of reviews. These instances provide another opportunity for coarse-grained disambiguation. We processed 3183 of these instances, covering 53 single-word prepositions and using 62 distinct supersenses. Each instance is tagged with one supersense; however, the instance is not linked to a single PDEP sense, but only identifies the tagged preposition. We formed a list of all instances for each preposition and then processed these sentences to create an XML file in the Senseval lexical sample format for each preposition. This format includes a

HEAD tag to identify the preposition in the sentence. The format also includes an ANSWER element that contains the correct tag(s) for the instance.

To identify the set of tags for an instance, we first formed a list of the supersenses for all senses, with a list of the senses for each supersense. For example, *on* has eight senses that have a **Location** supersense; one of these has an additional **Locus** supersense and another has two additional supersenses, **Instrument** and **Manner**. **Locus** is also a supersense of another sense and **Manner** is also a supersense of yet another sense of *on*. As a result, when tagging instances of *on*, a given supersense can give rise to multiple correct senses. Thus, this process of assigning correct tags to instances constitutes a coarsening of the senses.

After creating the XML files in a form suitable for the Tratz-Hovy system, we were then able to part-of-speech tag, parse the sentences, and generate features for the instances. We then scored the disambiguation, achieving an accuracy of 55.2 percent. This compares to a baseline accuracy of 42 percent when using a classifier that predicts the most frequent label for a preposition.[9]

The use of supersense tags as described above is clearly much finer-grained than using the PDEP classes. The results suggest again that coarsening the sense inventory can be beneficial. We have not yet examined the effect of further coarsening, using the supersense hierarchy described in Schneider et al.

7.3 Feature Analysis Across Prepositions

Srikumar & Roth (2013) investigated the benefit of examining properties across preposition classes. With the further development of classes in PDEP, we extend this analysis and look more deeply into how individual senses behave. The basic procedure for this analysis is to identify all senses belonging to a class, using the TPP Class field in PDEP senses, and then to perform feature analyses of PDEP instances that have been tagged with one of these senses. Thus, for example, in examining the Scalar class, we would ignore instances of *above* that have been tagged with the Spatial class. We examine the features for these classes through the use of distributional methods, specifically, Kullback-Leibler divergence (KLD) and Jensen-Shannor divergence (JSD).

To perform this analysis, the class is first specified to obtain the list of senses. Next, we specify a word-finding rule (e.g., the complement or the

[9] Personal communication.

governor) and a feature extraction rule (e.g., word class, part of speech, the lexicographer file name, synonyms, or hypernyms), thus creating a **wfr:fer** string that is used to match features. The numbers of each feature value for each sense are then counted and used to determine the relative frequency of each value, for each sense and for the entire set. The first output is the distribution of the feature values for the entire set.

The next step is to compute the KLD for each sense, using the overall distribution as the "true" distribution for the class. The KLDs are sorted and printed out in increasing order. Thus, the senses with the least divergence are printed first, giving some idea of which senses are perhaps prototypical of the class. The senses with the greatest divergence are printed last and perhaps identify outliers which might be incorrectly included in the class.

The next step is to compute the JSD of each sense with each other sense. This is done with the feature-value distributions that have been constructed for each sense, as used in the KLD calculations. The JSD is a true metric, with values in the range 0.0 to 1.0. Calculating this value gives rise to a symmetric matrix. After calculating the matrix, we then examine each column, sorting the JSD values to identify and print out the five closest senses to each sense, along with the JSD divergence between the senses. In general, use of the JSDs provides a principled way of identifying substitutable prepositions (i.e., another field in PDEP as shown in Figure 1).

8 Summary

In this chapter, we have examined many factors affecting preposition disambiguation. Our primary purpose has been to present the latest developments in this area, made possible with the data available in the Pattern Dictionary of English Prepositions. With this update, we hope that we have provided a starting point for further research on preposition behavior. We believe that an important part of this work should be an ongoing refinement of the preposition sense inventories and the development of a framework for preposition word sketches, going beyond the simple use of prepositions as bearers of semantic relations.

Acknowledgments

For over 25 years, Adam Kilgarriff was a constant friend with whom I discussed many of the lexicographic issues underlying preposition behavior. This included the use of the sketch engine, but also touched many other issues of computational lexicography. Orin Hargraves played an important role in the development of TPP and in many discussions surrounding the use of his efforts during the construction of PDEP. Stephen Tratz provided considerable assistance in using his system.

References

Steven Bethard. 2008. *Finding Event, Temporal and Causal Structure in Text: A Machine Learning Approach*. PhD Thesis, University of Colorado.

Andre Dittrich, Maria Varsadani, Stephan Winter, Timothy Baldwin, and Fei Liu. 2015. A Classification Schema for Fast Disambiguation of Spatial Prepositions. In *Proceedings of the 6th ACM SIGSPATIAL International Workshop on Geo-Streaming (IWGS)*. Bellevue, Washington USA.

Ismail El Maarouf, Georgiana Marsci, and Constantin Orasan. 2015. Barbecued Opakapaka: Using Semantic Preferences for Ontology Population. In *RANNLP Proceedings*. Hissar, Bulgaria.

Kathleen Fraser, Graeme Hirst, Naida Graham, Jed Meltzer, Sandra Black, and Elizabeth Roahon. 2014. Comparison of different feature sets for identification of variants in progressive aphasia. In *Proceedings of the Workshop on Computational Linguistics and Clinical Pyschology: From Linguistic Signal to Clinical Reality*. Baltimore, Maryland, USA, 17-26.

Isabelle Guyon, Jason Weston, Stephen Barnhill, and Valdimir Vapnik. 2002. Gene Selection for Cancer Classification using Support Vector Machines. *Machine Learning*, 46: 389-422.

Patrick Hanks. 2013. *Lexical Analysis: Norms and Exploitations*. MIT Press.

Marti A. Hearst. 2000. Automatic Acquisition of Hyponyms from Large Text Corpora. In *COLING 92 Proceedings of the 14th Conference on Computational Linguistics*, Volume 2, 539-45.

Robert. C. Holte. 1999. Very simple classification rules perform well on most commonly used datasets. *Machine Learning*, 11:63-91.

Dirk Hovy, Stephen Tratz, and Eduard Hovy. 2010. What's in a Preposition? Dimensions of Sense Disambiguation for an Interesting Word Class. In *COLING '10 Poster Volume*. Beijing, China, 454-62.

Adam Kilgarriff, Pavel Rychly, Pavel Smrz, and David Tugwell. 2004. The Sketch Engine. In *Proceedings of Euralex*. Pp. 105-116, Lorient, France.

Ken Litkowski. 2013a. *The Preposition Project Corpora*. Technical Report 13-01. Damascus, MD: CL Research.

Ken Litkowski. 2013b. *Preposition Disambiguation: Still a Problem*. Technical Report 13-02. Damascus, MD: CL Research.

Ken Litkowski. 2014. Pattern Dictionary of English Prepositions. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*. Baltimore, Maryland, ACL, 1274-83.

Ken Litkowski. 2016. *Identifying the Least Common Ontological Class*. Technical Report 16-01. Damascus, MD: CL Research.

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. 2006. Coverage and Inheritance in The Preposition Project. In: *Proceedings of the Third ACL-SIGSEM Workshop on Prepositions*. Trento, Italy ACL. 89-94.

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.

Diana McCarthy, Adam Kilgarriff, Milos Jakubicek, and Siva Reddy.2015. Semantic Word Sketches. *Corpus Linguistics 2015*. Lancaster University.

Randolph Quirk, Sidney Greenbaum, Geoffrey Leech, and Jan Svartvik. 1985. *A Comprehensive Grammar of the English Language*. New York: Longman Inc.

Nathan Schneider, Vivek Srikumar, Jena D. Hwang, and Martha Palmer. 2015. A Hierarchy with, of, and for Prepositions. In *Proceedings of the 9th Linguistic Annotation Workshop,* pages 112-123 Denver, Colorado.

Ryan Snow, Daniel Jurafsky, and Andrew K. Ng. 2004. Learning Syntactic Patterns for Automatic Hypernym Discovery. *Advances in Neural Information Processing Systems*.

Vivek Srikumar and Dan Roth. 2013. Modeling Semantic Relations Expressed by Prepositions. *Transactions of the Association for Computational Linguistics*, 1.

Angus Stevenson and Catherine Soanes (Eds.). 2003. *The Oxford Dictionary of English*. Oxford: Clarendon Press.

Stephen Tratz. 2011. *Semantically-Enriched Parsing for Natural Language Understanding*. PhD Thesis, University of Southern California.

Stephen Tratz and Eduard Hovy. 2011. A Fast, Accurate, Non-Projective, Semantically-Enriched Parser. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*. Edinburgh, Scotland, UK.

Patrick Ye and Timothy Baldwin. 2007. MELB-YB: Preposition sense disambiguation using rich semantic features. In *Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval-2007)*, pages 241-4. Prague, Czech Republic.