Preposition Analysis Using Correspondence Analysis

Ken Litkowski
CL Research
9208 Gue Road
Damascus, MD 20872 USA
ken@clres.com

April 8, 2021

Abstract

Several proposed characterization of preposition sense groupings have been developed over the years. In general, such groupings have involved discussions of fine- and coarse-grained senses. All of these discussions are based on qualitative judgments, beginning with lexicographers creating dictionaries and continuing with computational linguists using distributional methods. Correspondence analysis (CA) offers a different approach for examining sense similarities, using features developed in parsing preposition instances. CA methods first provide graphical visualizations of the similarities and then provide quantitative distances between senses, analyzing the variances of contingency tables in the expected values. We examine these methods in enhancing characterizations of preposition behavior patterns.

Contents

1 Introduction **3**

2 Motivation and Objectives **5**

3 Cross-Tabulation of PDEP Features **6**

4 Instance Analysis **9**

5 Dictionary Analysis **12**

6 Feature Selection **17**
6.1 Framework for Chi-Square Analysis 18
6.2 Analyzing the Feature Tabulations 20
6.3 Technique for Examining the Feature Cells 21
6.3.1 Main Elements of Feature Cross-Tabulations 21
6.3.2 Detailed Examination of Specific Features 22

7 Substitutable Prepositions **25**
7.1 Examining the Substitutable Prepositions 25
7.2 Correspondence Analysis to Identify Substitution Senses . . . 27
7.2.1 Tabulating the Features for the Anchor Preposition . 27
7.2.2 Creating the Supplementary Row for the Target Sense 28
7.2.3 Substitutable Analysis 29
7.2.4 Lexicographic Benefits for Correspondence Analysis . 30
7.3 Digraph after Substitutable Prepositions 30
7.4 Sense without Substitutes 30

8 Supersenses **31**

9 Multiword Expressions **31**

10 Reviewing the Corpora Tagging **31**

11 Related Work **32**

A Correspondence Analysis Techniques **35**
A.1 Simple Correspondence Analysis 35
A.2 Supplementary Points 36
A.3 Multiple Correspondence Analysis 36

B Word-Finding Rules **36**
B.1 Governor (h) 37
B.2 Verb or Head to the Left (l) 37
B.3 Head to the Left (hl) 37
B.4 Verb to the Left (vl) 37
B.5 Word to the Left (wl) 37
B.6 Syntactic Preposition Complement (c) 37
B.7 Heuristic Preposition Complement (hr) 37

C Feature Extraction Rules **37**
C.1 Immediate WordNet Hypernyms (h) 38
C.2 All WordNet Hypernyms (ah) 38
C.3 Affixes (af) 38
C.4 Capitalized Word (c) 38
C.5 All WordNet Gloss Words (g) 38
C.6 Lemma (l) 38
C.7 Word (w) 38
C.8 Word Class (wc) 39
C.9 WordNet Lexical Name (ln) 39
C.10 Part of Speech (pos) 39
C.11 Rule Itself (ri) 39
C.12 WordNet Immediate Synonym (s) 39
C.13 WordNet All Synonyms (as) 39
C.14 Pattern Dictionary of English Verbs (cpa) 39
C.15 FrameNet Entry (fn) 40
C.16 VerbNet Entry (vn) 40
C.17 Oxford Noun Hierarchy (o) 40

1 Introduction

Litkowski (2019) discussed future plans for the Pattern Dictionary of English Prepositions (PDEP, Litkowski (2014)), subsequent to honing the files installed into Sketch Engine[1](SE). These plans described improvements in PDEP supersenses, reviewing the corpora tagging, completing fields in the preposition patterns, analyzing substitutable prepositions, and extending preposition idioms in multiword expressions. As stated, there was no ordering for these areas. Several aspects of these tasks have been started, but now it seems that concrete steps have suggested that the tasks can be integrated, and in a way that may lead to a novel perspective for analyzing similarities of preposition senses.

Completing further fields in the preposition patterns had generally involved judgment of how each field should be filled[2]. The most basic field involves determining the general part of speech of the preposition's complement and governor. For the complement, PDEP envisioned common nouns, proper nouns, wh-forms, and gerunds, as well as the possibility of indicating that a sense complement could be a small set of lexical items. For the

[1] `https://www.sketchengine.eu/`
[2] `https://www.clres.com/db/TPPEditor.html`

governor, PDEP envisions that a noun, a verb, or an adjective governed the prepositional phrase. The PDEP software included several kinds of interactive analysis that could be used to help fill the various fields, particularly using features that were generated in parsing the corpora. Instead of examining the corpora associated with the tagged senses, one at a time, it was clear that writing some simple scripts could examine all the corpora at one time.

The first script[3] simply created tables of parts of speech for each sense for each preposition, for each of the three corpora. These tables are cross-tabulations, suitable for using correspondence analysis (CA), prompted by McGillivray et al. (2008), as further described in Greenacre (2017), summarized in Appendix A. CA provides spatial visualizations of cross-tabs showing the multidimensional relations for the preposition senses based on a singular-value decomposition of the table. In particular, the CA analysis shows how the senses of a preposition are related to each other. In this paper, we describe how each of the planned improvements can built on the correspondence analyses.

Section 2 describes what is being addressed and why it is relevant to attempting disambiguate among very polysemous prepositions, particularly when needed for semantic role labeling. Section 3 details how tables are generated from the features that were used in support-vector machines used in modeling the preposition sense disambiguation. Section 4 provides a more detailed multiple correspondence analysis that allows examination of individual corpus instances. Section 5 shows how it is possible to compare independent tagging against the dictionary definitions for each of the senses. Section 7 enables an examination of the **substitutable prepositions** field in PDEP, to allow the null hypothesis that the features across these substitutes are essentially highly similar. Section 8 allows similar examination of the PDEP field for **supersenses**, also allowing the examples used in the guidelines for supersenses in Schneider et al. (2017). Section 9 discusses multiword expressions (MWEs) added to PDEP, not included in the original sense inventories for 70 prepositions; these as well as other MWEs need to be analyzed in conjunction with the base sense inventories for these prepo-

[3]The script featanal.py is available at `https://github.com/kenclr/ca4pdep`. The project **ca4pdep** contains details of the processes, code, and data used in this paper. The feature files can be accessed online with simple R functions to create cross-tabulations, described in `https://www.clres.com/ca/pdepca00.html`. The files can be downloaded from `https://www.clres.com/db/feats/` by specifying one of the three corpora and a preposition. There are 580 feature files with almost 96 million features totaling 1.2 GB. The features were generated using code developed by Tratz (2011).

sitions. Section 10 describes techniques for comparing the tagging of each instance in the CPA corpus, based on the distance to the other senses.

2 Motivation and Objectives

Several developments and issues about preposition disambiguation have emerged in the past several years. Litkowski (2002) investigated a digraph of preposition definitions, establishing **an initial hierarchy of the definitions, requiring further analysis**. This led to The Preposition Project (TPP, Litkowski and Hargraves (2005)), providing a lexicographic characterization of preposition definitions from the *Oxford Dictionary of English* (Stevenson and Soanes (2003)). As indicated, TPP included sense tagging for prepositions from FrameNet sentences by a lexicographer. In some cases, **"the sense division found in ODE [did] not quite match the reality of preposition use"**, leading to the addition of senses.

Litkowski and Hargraves (2006) further characterized the hierarchy of definitions, particularly focusing on the importance of prepositions for semantic role analysis. With this and the further completion of TPP, a sufficient corpus (using FrameNet data) enabled SemEval task on word-sense disambiguation of prepositions (Litkowski and Hargraves (2007)). The description of this task emphasized **greater urgency to understanding preposition behavior** in semantic role analysis.

The SemEval task established a disambiguation level corresponding to open class words, achieving 69.3 precision and an F1 level of 0.818. This led to some others working on improving this level. Hovy et al. (2010) obtained accuracies of 91.8% for coarse-grained and 84.8% for fine-grained disambiguation. In the underlying code (Tratz (2011)), one important aspect was that the definitions were examined in detail so that several senses were clustered; this clustering thus also suggested **the need for further analysis of the preposition sense inventories**. Srikumar and Roth (2013a) also achieved similar accuracies and also showed the value of an **"inventory of 32 relations, building on the word sense disambiguation task for prepositions and collapsing related senses across prepositions"**. Srikumar and Roth (2013b) elaborated the inventory by identifying the TPP senses, "collapsing semantically related senses across prepositions."

The clusters and the relations have been added as fields in PDEP. Building on the semantic relations, Schneider et al. (2015) and Schneider et al. (2016) established a corpus of **preposition supersenses**. This corpus has been used for a detailed linguistic description of SNACS (Semantic Network

of Adposition and Case Supersenses), an inventory of 50 semantic labels ("supersenses") that characterize the use of adpositions and case markers, providing guidelines for applying these labels. Notably, each item in the inventory has been described with several corpus sentences that exemplify each supersense. Supersenses have also been added as a field in PDEP. **None of the clusters, relations or supersenses have been entered for all PDEP senses.**

(In development) Event of disambiguation still a problem. When representative corpus (CPA), found the need to add more senses because of idioms that belonged there. Describes what is being addressed and why it is relevant to attempting disambiguate among very polysemous prepositions, particularly when needed for semantic role labeling.

3 Cross-Tabulation of PDEP Features

Simple correspondence analysis (as described in Appendix A.1) begins with the generation of cross-tabulations in contingency tables, with sense lists in the rows and features (such as parts of speech) in the columns.[4]

As described in Litkowski (2016b), PDEP parsed 81509 sentences, using a dependency parser (Tratz and Hovy, 2011), each sentence focused on one preposition. On average, about 1250 features were generated for each sentence; for a typical set of 250 sentences for a preposition, about 70,000 distinct features were generated. Features are comprised of three components, (1) a word-finding rule (**wf**), (2) a feature extraction rule (**fer**), and (3) the value of the feature (**wf:fer:**). The initial cross-tabulation looks at the feature (**hr:pos:**), the part of speech of the heuristic identification of the preposition complement. The Python script above created a table of the parts-of-speech for each preposition sense for each corpus (CPA, OEC, and FN[5]). Table 1 shows the table for *above* in the CPA corpus[6] [7]. In this

[4]See `https://www.clres.com/ca/pdepca01.html` for code and output in this section. See also `https://www.clres.com/ca/pdepca01a.html` for a demonstration of the essentials for CA as described online.

[5]Corpus Pattern Analysis, Oxford English Corpus, and FrameNet

[6]Senses: 1(1) in extended space over and not touching; 2(1a) extending upwards over, 3(1b) higher than and to one side of; overlooking; 4(2) at a higher level or layer than; 5(2a) higher in grade or rank than; 6(2b) considered of higher status or worth than, too good for; 7(2c) in preference to; 8(2d) at a higher volume or pitch than; 9(3) higher than (a specified amount, rate, or norm); 10(n) more so than anything else

[7]Parts of Speech: cd (cardinal number), dt (determiner), jj (adjective), nn (noun (sing. or mass), nnp (proper noun, singular), nnps (proper noun, plural), nns (noun (plu.), pdt (predeterminer), prp (personal pronoun), vbg (verb, gerund or present participle), wp

Table 1: Parts of Speech for Complements of *above* in CPA Corpus

CPA	cd	dt	jj	nn	nnp	nnps	nns	pdt	prp	vbg	wp
1(1)	0	0	0	23	1	0	2	0	2	1	0
2(1a)	0	2	0	8	0	0	2	0	2	0	0
3(1b)	0	0	0	10	1	0	0	0	2	0	0
4(2)	0	1	0	10	0	0	4	0	1	0	0
5(2a)	0	1	0	3	2	0	0	0	1	0	0
6(2b)	0	1	0	5	1	0	1	0	2	1	0
7(2c)	0	0	0	0	0	0	1	0	0	1	0
8(2d)	0	0	0	5	0	0	0	0	0	0	0
9(3)	18	3	1	33	2	1	13	0	2	1	1
10(n)	0	48	2	1	0	0	3	3	0	0	0

case, there were 250 instances in the corpus, but only 229 were prepositions and the remaining 21 were adverbs.

A cursory examination of Table 1 provides some indication of how the pattern for each sense might be marked in the editor, e.g., noting the presence of cardinal numbers for sense 9(3) and determiners for sense 10(n). Since neither of these emphasized parts of speech are currently checkmark options in the pattern manager, describing the behavior requires characterization in the Selectors box.

The first question about this table is whether there is any difference between any of the senses. It is difficult to discern the differences among the other senses by inspection. A chi-square test determines if the distributions of the categorical variables differ from each another, i.e., testing the null hypothesis that there is no difference. This is the beginning step in correspondence analysis. In examining a cross-tabulation, if the null hypothesis is true, the observed and expected frequencies will be close in value. In Table 1, the question is whether the several senses have similar behavior. In this case, the chi-square statistic χ^2 is 286.95, indicating that the patterns are different. The chi-square divided by the sum of the table (229) is known as the (total) inertia (ϕ^2), characterized the variance in the table, in this case equal to 1.253063.

The objective of CA is to determine where the variance lies in the table. The first step creates an independence or correspondence matrix (CM), dividing each cell by the sum of all the cells, so that the sum of the cells of the CM is equal to 1. This is a matrix of standardized residuals, which is used to

(wh-pronoun)

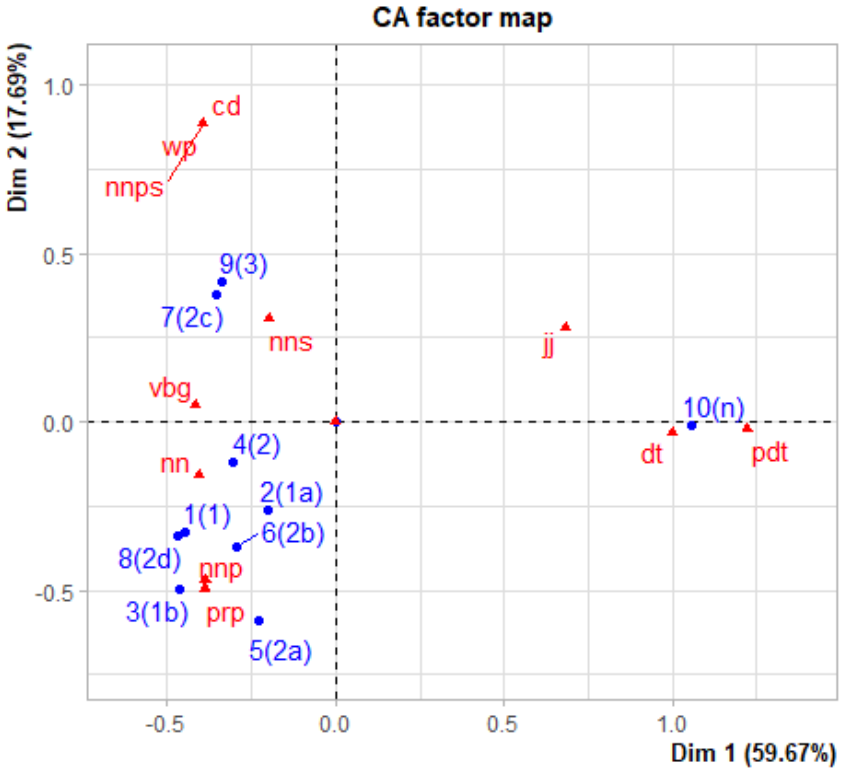

Figure 1: Sense Similarities Based on CPA Corpus

perform a singular value decomposition (SVD) into its factorizations. With the eigenvalues from the SVD, the CA can be used to identify the contribution for each of the rows, columns, and cells into their contributions to the inertia. This permits a plot of the rows and columns of the original table, in this case shown in Figure 1. The figure visualizes how the senses and the parts of speech relate to one another.[8]

Usually, the result summary first identifies the chi-square value. Next, the summary identifies the eigenvalues resulting from the singular value decomposition of the table, particularly showing the variance for each dimension. The total inertia (the sum of the variances) is 1.253063. A summary next provides the details for each row and each column, showing an analysis

[8]There are several packages for correspondence analysis, particularly in R, one in Python, and others in various statistical software. They can also perform the components in CA or can be implemented by developing the computations.

of these details. First, a column identifies the inertia for each row or column; the sum of these individual amounts is equal to the total inertia. Thus, it is possible to see, in this case, which senses and parts of speech have the largest portion of the variance. For Table 1, this indicates that senses 10(n), 9(3), and 7(2c) and the parts of speech "dt", "nn", "cd", and "vbg" account for the greatest variances. The summary results next identify where each sense and part of speech should be placed in factor map. These locations correspond to the first two dimensions for each

From the visualization, several further observations can be made and analyzed further. As indicated above, sense 10(n) seems to be somewhat different from the others. The figure shows that sense 10(n) is very extreme, very different from the others, probably based on the likelihood that the sense is idiomatic ("above all"), with its complement either a determiner or predeterminer. Having made this observation, it is possible to drop this sense, and thus provide a better idea of how the blob can be distinguished. In dropping that sense, it is necessary to drop the "pdt" (predeterminer) column, since dropping 10(n) leaves only 0 values and would result in a degenerate matrix for the SVD. With the smaller table, χ^2 is 96.46, still rejecting the null hypothesis.

The plot (not shown) that removes sense 10(n) still shows a blob of most of the senses, with outliers for sense 7(2c) ("in preference to") and the part of speech "vbg" (gerundial). Examining Table 1, we see that this sense has only 2 instances and the part of speech has only 4 instances. With these small counts, it seems that their significance in the plot appear too dominant, suggesting that these outliers can be dropped. With the smaller table, χ^2 is 68.49, still rejecting the null hypothesis. Figure 2 shows the resulting plot. This figure shows more spread of the senses. The first set of senses (1(1), 2(1a), 3(1b)) are close to one another; the second group (4(2), 5(2a), 6(2b), 8(2d)) are spread out, but also in the negative hemisphere, suggesting similarity; and the third group (9(3)) is alone in the positive hemisphere.

This introduction to the CA provides an overview for the technique. The discussions below will provide more details.

4 Instance Analysis

The contingency table above (Table 1) is actually a summary based on the features for each of the instances in the source corpus.[9] These instances can

[9]See `https://www.clres.com/ca/pdepca02.html` for code and output in this section.

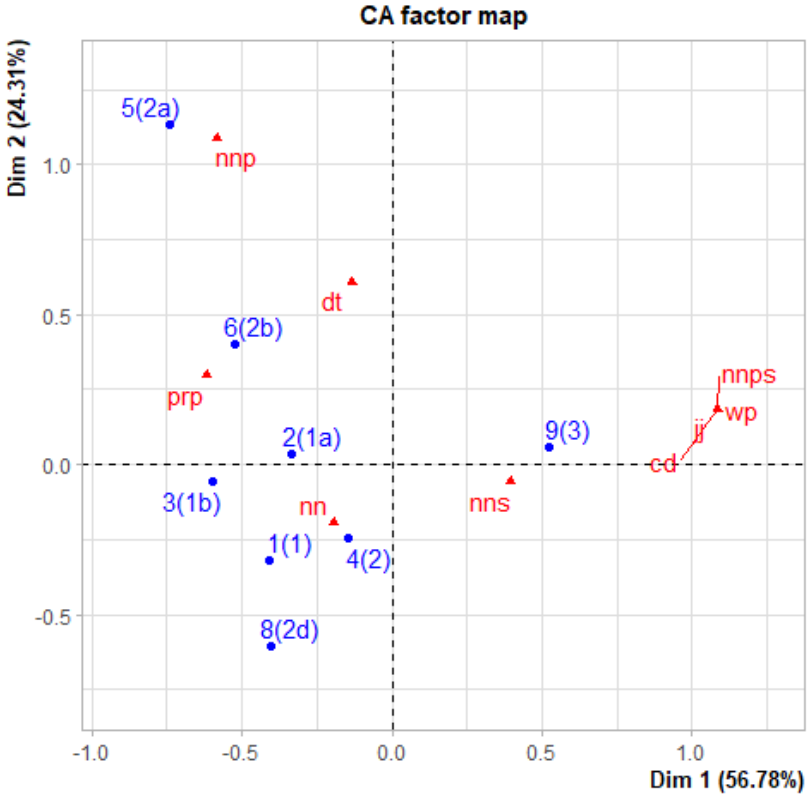

Figure 2: CA Factor Map with Some Removed Instances on CPA Corpus

be entered in another table (Table 2), with the rows corresponding to the corpus instance number (here listing only 10 of the 225 rows). Each row has four columns. The first column is the corpus instance identifier (Inst); the second column is the sense tag (S); the third column is the complement's part of speech (C); the fourth column is the governor's part of speech (G). This table can be analyzed using the techniques of **multiple correspondence analysis** (MCA).

In MCA, there is no specification of what should be the rows and the columns in a contingency table. Rather, the categorical variables are determined by the data that are available in the instances. In the table that is subjected to the computation, the rows are the instances (in this case, the 225 instances) and the columns are the values in the instances (in this case, the 10 senses and the 11 parts of speech). In this table, each row (a corpus instance) has a 0 value for 19 of the columns and a 1 value for 2 of

the columns.

As discussed above (see Section 3), it is possible to use any feature as the type for each instance. Thus, for example, we can use the lemma of the complement. In such a case, the MCA technique would establish a column for each distinct lemma in the instances we are examining. (See further discussion of selecting features in Section 6.)

The senses and the parts of speech are treated as equal. When the MCA is performed, the factor map (Figure 3) emplaces 21 points for the senses and the part-of-speech complements. Note that the map has points with parts of speech, essentially equal to points for the senses. This figure should be compared to the factor map in Figure 1, where the senses use blue. More imporantly, the maps of the two figures are essentially the same, i.e., the relative positions and differences are the same, except that Figure 3 is a reflection of Figure 1. Also, because the parts of speech are treated as the same level as the senses, the axes for the first two dimensions are much smaller than before, 9.8% and 7.7%.

The initial plotting for the MCA analysis (see footnote 9) also show two additional figures. The first shows a factor map for all the rows (instances). This map is full of blobs, where 225 instances are included in the map. The blobs correspond to many instances and have instances that have smaller distances than others. For example, this map contains 63 instances in the negative x-axis, 47 of which are located in the same position (all instances containing "above all") and with the remaining have similarities ("above all others", "above all else"). Thus, the distances between instances may serve as the basis for clustering into similar senses.

The MCA technique can also be performed on the entire table, i.e., also including the governor parts of speech. The first resultant figure (also included in the footnote 9) shows an additional set of points, beginning with a "C" or "G" and an underscore for the complements and the governors, respectively. In this case, with an additional 18 points, it becomes more difficult to get an impression of the visualization of the results, suggesting that a quantitative analysis of the distances is desirable.

The second factor map (showing the locations of the individual sentences) is similar to the one that looked only at the complements, but the raw data is more interesting. The footnote link also shows the coordinates of the 225 instances for each of the corpus instances. It is instructive to examine instances having the same coordinates. For example, there are 11 instances with the location (-1.99, -0.09) and have a similar combinations of complement and governor. This suggests that the MCA makes it possible to home in on similar patterns.

Table 2: Instance Data for *above* in CPA Corpus

Inst	S	C	G
c1	3(1b)	prp	vbd
c2	1(1)	nn	nn
c3	9(3)	nns	vbd
c4	9(3)	prp	vbz
c5	9(3)	nns	vb
c6	1(1)	nn	vb
c7	10(n)	dt	cc
c8	9(3)	nn	vbp
c9	9(3)	cd	nn
c10	10(n)	dt	vbz

5 Dictionary Analysis

The PDEP senses (footnote 6) were provided from the *Oxford Dictionary of English* (ODE, Stevenson and Soanes (2003)).[10] In addition, as described in Litkowski (2013), 7650 example sentences were also made available, from the Oxford English Corpus (OEC). In general, the object was to provide 20 example sentences for each sense; while this goal was not always attained, the examples are generally suitable for analysis. These sentences were also parsed, generated with the same sets of features, characterizing the preposition behavior. One notable aspect of this corpus is that the sentences are generally simple sentences, not compound sentences, thus likely having more accurate parses and features. The quality of this corpus can be examined as an anchor point for assessing the other corpora in PDEP.

The PDEP data include features for the OEC sentences used to exemplify each of the senses. Table 3 shows the parts of speech for the complements of *above* in the OEC corpus. This table shows several variations that may occur, particularly in comparison with the CPA cross-tabulation in Table 1. First, there is no sense for "10(n)" in the OEC corpus. This was added to the PDEP senses since the last sense occurred frequently in the CPA corpus. This sense is also an occurrence of a multiword expression (MWE) that should be considered along with the main senses of *above* (MWEs are further discussed in section 9 below). Second, the parts of speech occurring in the OEC corpus are slightly different from those in the CPA corpus. Table 1 includes "jj" (adjective), "nnps" (plural capital nouns), "pdt" (pre-

[10] See `https://www.clres.com/ca/pdepca03.html` for code and output in this section.

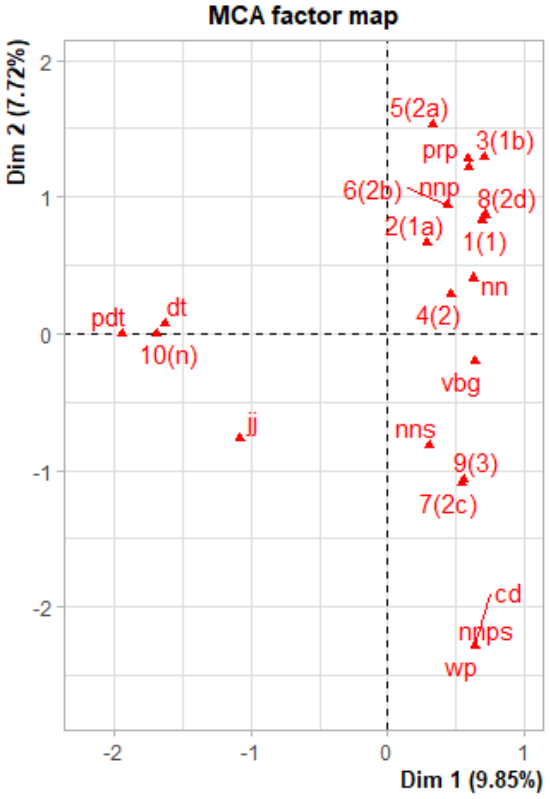

Figure 3: MCA Factor Map with Containing All Variables

determiners), and "wp" (*wh*-pronouns) that do not occur in Table 3. It is worth commenting that (1) these parts of speech are very infrequent, (2) they may not have been correctly parsed, and (3) these instances do not arise to a frequency that suggests they should have been incorporated into the set of dictionary senses. Third, the number of senses (in each row) differ slightly for the object of having 20 examples. Even when 20 examples were obtained, it should be kept in mind that the distribution of the parts of speech (or any other feature) may not be absolutely true.

Figure 4 shows the factor map using the OEC corpus, as based on the parts of speech for Table 3. This figure might best be compared with Figure 2 in which several outliers had been removed from the full CPA corpus for *above*. Such a comparison only provides an impression. In this case, it appears that the two figures are somewhat different from each. This suggests quantifying the distances between two coordinates. To do so, this establishes

Table 3: Parts of Speech for Complements of *above* in OEC Corpus

OEC	cd	dt	nn	nnp	nns	prp	vbg
1(1)	0	0	9	3	3	5	0
2(1a)	0	0	16	0	2	2	0
3(1b)	0	0	8	8	3	1	0
4(2)	0	0	12	0	2	2	0
5(2a)	0	1	8	2	1	8	0
6(2b)	0	0	7	0	4	0	0
7(2c)	0	1	12	0	5	0	1
8(2d)	0	0	20	0	0	0	0
9(3)	3	0	11	0	6	0	1

the OEC figure as an anchor and assesses the distances from the senses in the CPA figure to the anchor.

CA provides a mechanism for the distances between two coordinates, as described in Appendix A.2. Here, Table 3 is the anchor contingency table and the rows in other contingency tables are viewed as supplementary points. The CA of the anchor table establishes coordinates for each sense, constituting the total inertia for the *active* points. With a supplementary point, it is possible to compute what inertia it would have and how it might be related to the anchor table. To do this, we examine each row (sense) in a comparison contingency table, one for the CPA corpus and one for the FN corpus as identified in Litkowski (2013). The details are provided in footnote 9.

As shown above, the senses and parts of speech in the comparison tables may not always match up with the anchor table. As a result, some modifications are necessary. There are two routines that can be applied. The first method determines the set differences between the rows and columns of the comparison table against those in the anchor table. For the CPA table, as described above, this yields one row ("10(n)") and four columns ("jj", "nnps", "pdt", and "wp") not in the anchor table. To synchronize, a modified table drops the specified items. The resultant table can then be used to analyze each row against the anchor table. The second method determines manually by inspection. For the FN table, there are no senses for three of the rows; this absence is not a problem for the analysis. There are two differences in the columns. There is no "nnps" (plural proper nouns) in the FN table, although there is a "nnp" (singular proper noun) in both tables. In the FN table, we add the columns in "nnp" and "nnps", making

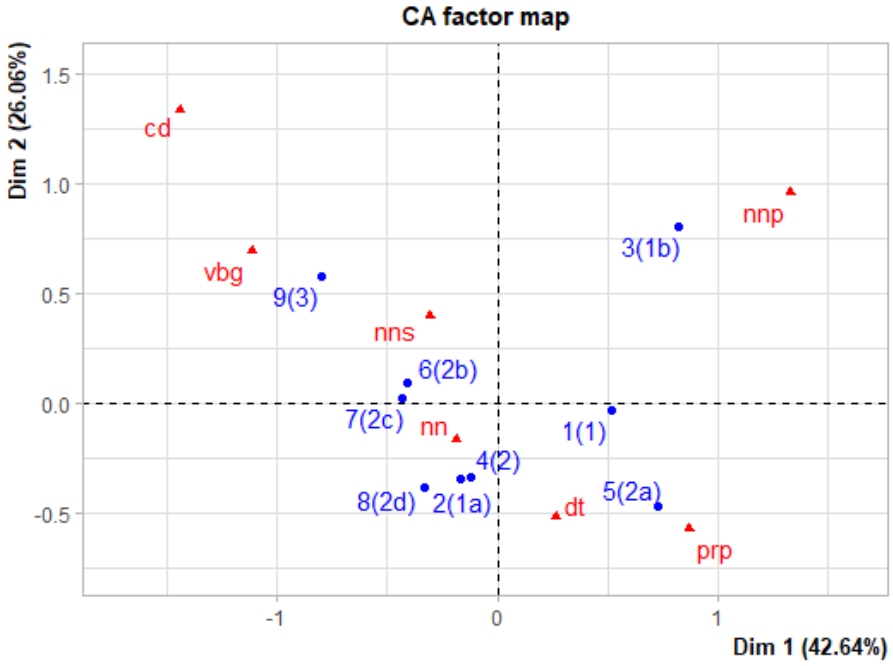

Figure 4: Sense Similarities Based on OEC Corpus

the sum as the new column for "nnp" (since these are similar). The other change in the FN table is that there is no "vbg" (gerundial). To make this consistent with the anchor table, we add a new column "vbg" with a "0" in each cell of the column. The resultant table can then be used to analyze each row against the anchor table.

To analyze the supplementary rows, we append rows to the anchor (OEC) contingency table. We then perform the correspondence analysis with the anchor table, as before, but indicating that we have one or more supplementary rows. Figure 5 is the result of adding two supplementary rows from the comparison tables, one with the row *C9(3)* from the CPA table and one with the row *F9(3)* from the FN table. This figure should be compared with Figure 4. The two figures are essentially identical, except for the addition of points where the supplementary rows are located. In the summary results for the CA, the only information associated with these

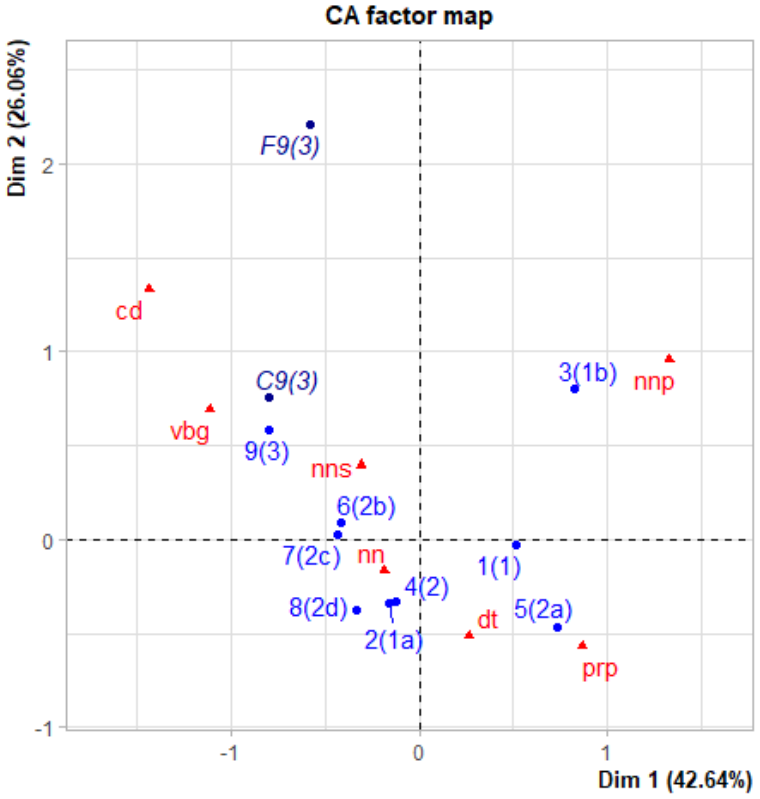

Figure 5: OEC Corpus with Supplementary Rows

rows is their coordinate locations. That is, there is no other information because they are passive points, not the active points that determine the plot.

From simple inspection, the two supplementary points (*C9(3)* and *F9(3)*) appear to be closest to the anchor point *9(3)*. To confirm this, we compare the distances from the supplementary points to all the other points in the anchor map. This process suggests (1) determining the distances for the rows of each comparison table and (2) ranking where each row would occur. Since the anchor table has 9 rows (senses), each comparison row will have a ranking between 1 to 9. For example, *C4(2)* has the rank 2 (i.e., there is one closer point, *2(1a)*) and *F4(2)* has rank 3 (i.e., there are two closer points, *1(1)* and *5(2a)*).

The routine for determining the ranks for all the comparison tables consists of simple steps. For each row in each comparison table, we perform

Table 4: Ranking of Comparison Tables

Senses	CPA	FN
1(1)	6	7
2(1a)	2	2
3(1b)	8	NA
4(2)	2	3
5(2a)	2	NA
6(2b)	5	NA
7(2c)	3	6
8(2d)	1	3
9(3)	1	1
Ave	3.3	3.7

a supplementary analysis function. This function gets the row from the comparison table, appends the row to the anchor table, and performs the correspondence analysis (with the row identified as a supplementary point). The result of the CA is then passed to a function that determines the distances of the supplementary coordinate with the coordinates of the anchor table rows. The distances are then sorted to find the rank of the supplementary point.

The ranking results are shown in the Table 4. That is, each ranking indicates how far its sense in the comparison table from its position in the anchor table. In the "FN" column, the three cells with "NA" indicates that the FN table had no instances for the corresponding sense. As indicated in the table, there are 9 senses in the anchor OEC table (3). The average rank for each sense (3.3 for CPA, 3.7 for FN) is thus out of 9. Ideally, we would like to have an average of 1.0, indicating that the comparison senses are closest to the senses in the anchor table. We consider this in the next section, where we examine the possibility of developing criteria to assess the mappings.

6 Feature Selection

Glynn (2014) indicates that it is difficult to interpret results when there is a "large number of features at play".[11] Krawczak and Glynn (2019) show the detailed complexity appropriate for a comprehensive analysis of a correspondence analysis for one small set of instances, including relevant

[11] See `https://www.clres.com/ca/pdepca04.html` for code and output in this section.

linguistic properties. With the size of PDEP, it will be very difficult to match this amount of detailed analysis. Here, we provide some procedures that may be of some assistance in selecting appropriate features for a preposition.

As mentioned above (Section 3), PDEP used 7 word-finding and 17 feature extraction rules to create features to characterize the behaviors of a focus preposition in sentences. The objective here is to identify which of the potential 119 combinations that best fit that distinguish among the senses for a preposition, "which usage-features co-occur with other usage-features, giving a map of their overall patterning". Glynn suggests that significance tests "can be used to show that the occurrence of certain features is substantially more common than could be expected by chance". Litkowski (2016c) described the overall strength of each combination, using feature ablation procedures. Of the top six combinations, five of the word-finding rules used the "all hypernyms" (**ah**) feature extraction rule. It was only the 7th combination where "part of speech" (**pos**) for the preposition complement was identified as important; two of the next ranks used the "WordNet lexical name" (**ln**) feature extraction rule. While these results gave a general perspective of which combinations were the most importance, they do not provide the kinds of details for characterizing the behaviors of an individual preposition.

There are three corpora that can be used to determine distinctions. In theory, the OEC senses are intended to identify the distinctions; these may be considered the most important for this purpose. The CPA senses are intended to be representative, so we will keep this point in mind. The TPP senses (from the FrameNet selections) may not be the best for obtaining distinctions, but they can be used in assessing their distinctions. Using an estimate of 1171 features for each of the 81,509 sentences in the corpora, 96 million features need to be brought down to a level of describing the behaviors for the 1040 PDEP senses.

6.1 Framework for Chi-Square Analysis

Here, the primary information for making the distinctions uses the features that have the largest chi-square values, i.e., the ones that are statistically significant. To do so, the analysis uses the features for each of the instances in each corpus for each preposition, with each feature comprised of three parts, **wfr:fer:value** (combination of word-finding rule, feature-extraction rule, and the value of the feature).[12] There two primary components for

[12]The word-finding rules and the feature extraction rules are described in detail in Litkowski (2016c) and quickly named at footnote 11.

Table 5: CPA 'above' with 4059 significant features

	h	ah	af	c	g	l	w	wc	ln	pos	ri	s	as	cpa	fn	vn	o
h	50	117	1	0	205	3	1	1	8	1	0	57	166	0	0	0	1
hl	42	105	2	0	191	2	1	0	4	0	0	42	93	0	0	0	0
l	66	131	1	0	218	2	1	0	4	0	0	53	122	0	0	0	0
vl	42	64	0	0	128	1	0	0	0	0	0	38	104	0	0	0	0
wl	16	90	1	0	157	3	3	1	11	3	0	15	23	0	0	0	0
c	80	220	3	1	352	3	2	2	17	5	0	41	75	0	0	0	0
h	79	222	5	1	377	3	2	1	17	4	0	41	75	0	0	0	40

determining chi-square values. The first component is the set of senses tagged for the preposition, as included in the feature files. This set identifies the number of instances tagged with each sense. The second component is the number of each of the 119 features, also tabulating the number of occurrences for each sense. There are two criteria to compute χ^2. The first is that there are at least two senses.[13] The second is that the feature has at least the minimum frequency, here using 5. The χ^2 is computed by determining whether the distribution for the feature corresponds to the distribution of the senses, meeting the significance level for the degrees of freedom (i.e., using the number of senses), here using 0.05.

A chi-square file is generated for each preposition for each corpus that has at least two senses. There are 97 files for the OEC corpus, 125 for the CPA corpus, and 45 for the TPP corpus (out of 304 prepositions in PDEP). The first two lines of each file lists the senses and the number of instances for the senses. Each other line contains the feature, its chi-square value, and the number of instances for the feature. The lines are sorted in decreasing χ^2. The number of features runs from 67 in *outside of* in the OEC corpus to 134209 in *of* in the TPP corpus. The number of significant features runs from 0 (17 files, e.g., none of the features in *outside of* is significant) to 72997 in *of* in the TPP corpus.

In the accompanying code, we use data for the preposition **above**. The chi-square analyses for the three corpora are provided in the accompanying data. Note that the number of senses is different for the three corpus (9 for OEC, 10 for CPA, and 6 for TPP), so that a different significance levels will be appropriate for the three corpora.

In our analysis, we are going to count the number of features that are

[13] Another aspect of these corpora is that many of the prepositions are monosemous, so their senses and their instances cannot be immediately useful for characterizing distinctions. These instances will be discussed below.

considered significant. This will be done in a counting matrix of 7 x 17. The rows are the word-finding rules (wfr); the columns are the feature-extraction rules(fer). There are 247 tables for the prepositions tagged with two or more senses, such as in Table 5 which counts the number of significant features for **above** in the CPA corpus. The script loops over the directories for the three corpora. The next step gets the list of chi-square files and loops over the files. The first step reinitializes the counting matrix, setting 0 to each cell. Next, we load the file for the particular preposition and gets its length (i.e., the number of features). The script then gets the first line in the file, showing the number of senses, based on the length of the line. We set the significance level (p-value) appropriate for the preposition, based on the degrees of freedom (the number of senses). Next, we count the number of significant features. To do this, we loop over the length of the file, examining each line. We set the line to the feature and look at its second element, which is the chi-square value. We break the loop if the value is less than the significance, indicating that we have ended the feature counting. If the feature is significant, we split the feature name to get the word-finding rule name and the feature-extraction rule name to increment the cell in the counting matrix.

6.2 Analyzing the Feature Tabulations

In looking at the tabulations, we remind that a cell does not correspond to the feature occurrences in the feature files for each corpus. Instead, the cells correspond to the number of occurrences that have the greatest differences from the expected proportions for a feature combination. For example, there may be several parts of speech for preposition's complement and governor, but only a few of them are distinctive (i.e., significant), and possibly useful in the correspondence analyses. For example, in the 173 OEC instances for **above**, there are 167 occurrences of hr:pos:; there are 7 different feature values, of which only 2 are significant ("nnp" and "prp").

The first observation for the feature tabulations is that there are different results from the three corpora. The main conclusion is that there is no main conclusion that will encompass the three corpora and all the prepositions. Instead, it will be necessary to examine multiple chi-square files in themselves. These tabulations are provided for all the chi-square files for all the corpora (97 for OEC, 125 for CPA, and 45 for TPP).[14]

[14]They are available at `https://github.com/kenclr/ca4pdep/tree/main/data` in the files **feats*.txt**. In these files, each preposition tabulation is a space-separated lists. The tables are available in raw forms and also in an Excel table FeatsSignif.xlsx. The Excel

The OEC tabulations frequently have the largest number of significant features. As discussed in dictionary analysis in section 5, the instances for the several senses for the corpus were probably selected to represent the differences. The tabulations for this corpus seems to support the differences, resulting in different distributions of the important features. Since the CPA instances are intended to be representative of a preposition's behavior, it is expected that there will be fewer differences from the proportions for each sense. This is likely to reflect in the smaller number of significant features. Another difference for this corpus may be the number of senses. For example, for **above**, there is an added sense (corresponding to an idiom *above all*); we can expect that more detailed analysis will be observed in some specific feature combinations. As indicated above, the TPP data correspond to fewer senses (6 for **above** compared with 9 for OEC and 10 for CPA). In addition, some of the prepositions have a much larger number of instances (e.g., 4496 for of and 2089 for in); as a result, the numbers of significant features are much larger for such prepositions (72997 for of, 32420 for in).

6.3 Technique for Examining the Feature Cells

6.3.1 Main Elements of Feature Cross-Tabulations

An initial perspective of the feature cross-tabulations can be obtained by examining a text that includes all of the tables and examining a spreadsheet that also includes all of the tables. We can search the **feats*.txt** files for the regular expression **[0-9]+ significant**, obtain the matches, and sort the 267 results. We can first see that the number of significant features ranges from 0 to 72997 (for **of** in the TPP corpus). There are 17 tabulations with no significant features and 61 tabulations that have fewer than 100 significant features. There are 20 files that have more than 10,000 significant features; it may be difficult to get information from such plethora. In general, it may be difficult to find important information from these tables.

From the spreadsheets, it is possible to assess the feature-extraction features. We use three sheets, one for each corpus. We then sum the cells for each of the 17 columns for all the prepositions. The WordNet gloss word (**g**-C.5) was the most frequent feature-extraction rule, followed by WordNet immediate hypernym (**h**-C.1), all hypernym (**ah**-C.2), immediate synonym (**s**-C.12), and all synonym (**as**-C.13). The numbers of significant features seem to be so large as to be likely difficult to discern meaningful distinctions.

Feature-extraction rules with an intermediate frequency seem likely to

table facilitates further examination.

be worth more detailed examination. The word class (**wc**-C.8) is relatively infrequent, in part because there are only 4 possible values ("noun", "verb", "adjective", or "adverb"); these attain significant only when the word class is significantly different from the usual for a sense. The part of speech (**pos**-C.10) appears to be significant to a level that is worth examining. The word (**w**-C.7) and lemma (**l**-C.6) features appear to be significant for some prepositions and thus worth examining in detail. The WordNet lexical name (**ln**-C.9) reach significance in the greatest frequency in this group; since there are only 40 possible values for this feature, attaining significance is worth examining in detail. The Oxford noun hierarchy feature (**o**-C.17) occurs frequently; these are essentially hypernyms; they have not been examined in detail, but may be of considerable value, as suggested in Litkowski (2016a). The feature-extraction rule on whether the word is capitalized (**c**-C.4) reaches significant in a relatively number of cases, but such cases may be useful for characterizing a sense's behavior. The number of cases for affixes (**af**-C.3) is relatively small; their importance has not yet been examined.

To examine word-finding rules, we search the feats*.txt files for lines beginning with a particular rule (i.e., regular expressions, e.g., **^hr:** for the heuristic complements or **^h:** for the governor). We can copy such lines and then paste the lines into a spreadsheet sheet. We can head the sheet with the column names (i.e., the feature-extraction rule codes) and copy the prepositions and number of senses into the first two columns. With this result, we can compare and contrast the behaviors with the prepositions. This will help us to identify which features to examine in more detail.

6.3.2 Detailed Examination of Specific Features

This process includes five steps to make a detailed examination of the specified feature analysis.

Corpus Identification: To analyze specific features, we need to identify which corpus is to be examined. A function is designed to obtain all the chi-square files that will be processed. This function is called before the examination in the main function for the feature analysis, where it processes each file.

Specify Preposition, Feature, and Frequency: The main function (**fanal**) specifies the preposition, the feature (the word-finding rule and the feature extraction rule), and the minimum frequency required for the feature. The function then looks in the specified directory for each chi-square file (***.chisq**). The first two lines of the file identify the senses and the number of instances in the corpus. Based on the number of senses, the significance

level (p-value) is identified based on the degrees of freedom. The function counts the number of significant features, lists the lines in chi-square file (at hits), and the number of features (min) that meet the minimum frequency.

The function next loops through the lines of the chi-square file, examining each line which consists of the feature, the chi-square value for the feature, and the number of occurrences of the feature for each sense. The loop breaks when the chi-square for a feature is less than the p-value. Otherwise, the counter for the number of significant features is incremented. Next, the feature is examined to make sure that the name of the feature matches that was specified (beginning with **wfr:fer**). If so, the line number is added to hits. The feature is tested to see if there are at least the minimum frequency for the occurrences of the feature. If it does, we analyze the contribution of each sense to the total chi-square, as described below.

After all lines have been processed, the results of the function are summarized, listing (1) the number of significant features, (2) the line numbers of the features that have been analyzed, and (3) the number of features with the required frequency.

Contributions to Chi-Square: Each feature has a chi-square computed on the difference between the observed and expected values based on the frequency for the occurrences for the feature. The objective here is to determine how much can be allocated to each sense. This function (**contrib**) apportions the total chi-square for a feature for each of senses. The feature counts for each sense constitutes the **observed** for the feature; if this is less than or equal to the minimum, we return with a FALSE return. Otherwise, we are ready to determine how much of the chi-square should be allocated to each sense.

We print the beginning for the output, indicating the feature name and its chi-square value and the number of occurrences for the feature. We sum the number of instances for the full corpus as the total for the preposition. computing the proportion of each sense in the full corpus as percents. Next, we sum the counts for the feature and compute the expected value, i.e., apportioning how the counts for the feature *ought* to occur for each sense. We use the chi-square equation, as above, computing the difference between the observed and the expected, with a vector showing the portion for each sense.

We print a table showing the situation for each feature that meets the criteria. Table 6 shows the results for the corpus name, the preposition, the full feature (**hr:pos:dt** for a determiner), the frequency of the feature, and the chi-square value. The table consists of (1) the senses, (2) the instances for each sense (**insts**), (3) the feature counts (**fcnts**), and (4) the chi-square

Table 6: CPA 'above' hr:pos:dt, freq = 56, chi = 113.60

senses	2(1a)	3(1b)	4(2)	1(1)	5(2a)	10(n)	6(2b)	7(2c)	9(3)	8(2d)
insts	14	13	16	29	7	57	12	2	75	6
fcnts	2	0	1	0	1	48	1	0	3	0
chi	0.6	3.2	2.1	7.0	0.3	84.6	1.3	0.5	12.7	1.5

contributions to each sense (**chi**). We use this table to describe the results of the analysis.

Selecting What to Examine: With the scripts above, it next becomes easy to examine any of the almost 32,000 combinations of the corpus, preposition, word-finding rule, and feature extraction rule. In addition, specifying the minimum frequency can be used to adjust the results to ensure that there is an appropriate set of feature counts. Table 6 is only one of the 8 tables generated from the script in footnote 11. In the first four, we examine two ways of looking the OEC complement's part of speech for the syntactic (**c**-B.6) and the heuristic (**hr**-C.10) features (both of which have significant features for *proper nouns* and *personal pronouns*). We are attempting to examine whether there is some significant difference between the methods. In this situation, we see that the two tables are quite similar, where the parts of speech do not explain the differences from the two methods. In the other examination, we change the corpus from the OEC to the CPA corpus (using the heuristic word-finding rule (**hr**-B.7) and the feature-extraction rule (**pos**-C.10) feature), seeing that with the CPA corpus, the significant features are for determiners ("dt"), common nouns ("nn"), and cardinals ("cd"), as well as proper nouns ("nnp").

Interpreting Feature Results: Each set of results will require the characterizations. Examination of the results is similar to what would be done by a lexicographer. The first thing that we look is whether the number of counts are sufficient. In the examples shown in footnote 11 the CPA corpus, the number of 7 occurrences for **hr:pos:nnp** had 2 proper nouns, more than expected; the question is whether this is behavior for a distinctive difference.

The next thing that we examine is which of the senses have the largest contribution to the total chi-square. We see that sense "3(1b)" (higher than and to one side of) has a larger number of proper nouns, named places. We also see that sense "5(2a)" (higher in grade or rank than) has a large number of **hr:pos:prp**, *personal pronouns*.

We also note in Table 6 that, for the CPA corpus, **hr:pos:dt** (a determiner) is significant for sense ("10(n)"), (*more so than anything else*), which

corresponds to the idiom *above all*, explaining 74 percent of the chi-square. For **hr:pos:cd** (a cardinal number), sense ("9(3)") (*higher than (a specified amount, rate, or norm)*), 68 percent of the chi-square.

In this analysis, we have indicated that the sense is the one that has the most proportion of the chi-square. These results correspond to intuition. However, this is not always the case. We have seen some situations where a zero or one occurrence has the largest contribution to the chi-square, i.e., because the infrequency is distinctive.

7 Substitutable Prepositions

The object of analyzing substitutable prepositions is to collapse senses so that ambiguity or polysemy among prepositions is minimized.[15] In describing TPP, Litkowski and Hargraves (2005) indicates that the lexicographer used several criteria in identifying other prepositions that have substitutes for each sense. To do this, the lexicographer would imagine substituting some of the suggestions in example sentences, examine the definitions in the other preposition's inventory in ODE for similarity, and look for some meaning component of the attachment point (usually a verb) with some meaning component of the preposition. Litkowski and Hargraves (2005) indicated that this issue would await further data when the other prepositions undergo their sense tagging. This section describes procedures for such analysis.

Substitutes are identified for 778 of the 1039 senses in PDEP. When these substitutions were developed, there was no intention to subject them to a rigorous analysis. With the addition of correspondence analysis, several avenues of investigation arise. There are three components to the analysis: (1) assuring the integrity of the substitutable prepositions field (section 7.1), (2) using correspondence analysis to identify which sense to use when a substitutable has multiple senses (section 7.2), and (3) hierarchizing and graphing the senses into a digraph (section 7.3).

7.1 Examining the Substitutable Prepositions

When a sense has a non-null preposition substitutable (*opreps*) field, it is necessary to determine which sense of each preposition is applicable. If a sense has more than one substitution, it is necessary to disambiguate each one. For any substitution, we determine how many senses the substitution

[15] See `https://www.clres.com/ca/pdepca05.html` for code and output in this section.

has. In many cases, a substitution has only one sense, thus simplifying the task. When the calling preposition also has only one sense and each substitution also has only one sense, we can say that the two preposition are complete synonyms. Analyzing the substitutions iterates through the PDEP senses (a file containing the preposition, its sense number, and the value of the substitutable prepositions field), counting six items and noting three lists that will be used in further analysis.

The main loop in the script increments the count of the number of senses (1039). With each iterate, the preposition and the sense are recorded. The first test for a sense is to determine whether it has a non-empty string the Substitutable Prepositions field. The script increments the count of the senses with substitutions (initially 778). When a sense has substitutions, the string in the field is split using a slash ("/") and then iterates over each posited substitution. We need to test whether each substring is a preposition and whether the substrings constitute synonyms, initially assuming as true. We also need to check whether each substring is a preposition that has only one sense (i.e., the preposition is monosemous). After iterating through the substrings, we can conclude something about the preposition substitutions and enter data into one of the lists.

If a substring is not in the preposition list, we set that it is not valid and that it cannot be a synonym. When it is not valid, we increment the *without all good prepositions* counter and add the sense to the list of *bad substitutions*. The substitutions field needs correction so that all correspond to prepositions. For example, *about 3(2)-1* has a substitution "round (Brit)"; the parenthetical needs to be removed so that the further analysis can be performed correctly. There are 156 senses that fall into this category. Some senses have more than one non-preposition substring; 14 senses have multiple non-preposition substrings. This makes a total of 170 non-preposition substrings.

When each substring has only one sense, we make an initial judgment that the substitution(s) are synonym(s) with the sense. For example, *by reason of 1(1)* has the substitutions "because of/owing to/on account of". In this case, every substitution has only one sense in the PDEP entry. This is defeasible, but is initially plausible. There are 58 senses for which this is the case. When this is the case, the counter of synonyms is incremented and the sense is added to the list of synonyms.

When the initial test for the substitutions failed, a sense had no substitutions. The lexicographer in TPP did not identify the sense of the substitutions. This is incremented in the counter for these senses and added to the list of this item. There are 261 senses with no substitutions. This

analysis includes procedures intended to identify the specific sense(s), particularly below in the locating of these senses in the preposition digraph analysis below.

There are 622 senses that have acceptable sets of substitutable prepositions, with 564 having substitutions that will require correspondence analysis to disambiguate the closest sense of the substitution.

7.2 Correspondence Analysis to Identify Substitution Senses

There are 778 senses in PDEP that have a list of substitutable preposition and that need to be subjected to correspondence analysis. The analysis uses the features of the OEC corpus, i.e., the sentences that exemplify the preposition senses. For each sense that has viable substitutable prepositions, for each such preposition, we use the OEC sense inventory as the anchor. The basic procedure for identifying the sense that corresponds to the instant sense follows the steps describe in the dictionary analysis (section 5). For example, *abaft 1(1)* has "*behind*" in the substitutable preposition field. The entry *behind* has 9 senses. In the analysis, we treat *abaft 1(1)* as a supplementary row to the anchor senses in *behind* and determine which anchor sense has the most similar set of features.

In the analysis, we need to specify the feature **wfr:fer:** that will be used in the contingency table. In the feature files, we are obtaining the values of the feature combination. For example, for **hr:pos:**, we are looking for the parts of speech for the preposition complements. We will find **hr:pos:nnp** for proper nouns and **hr:pos:nns** for plural common nouns; in this case, we will use **nnp** and **nns** as column values in the contingency table.

7.2.1 Tabulating the Features for the Anchor Preposition

To create the tabulation for the anchor preposition for a particular feature, we use a function to get the features for the substitution senses from which we will make a selection. We select the directory containing the OEC features for the file corresponding to the preposition. This file will contain the feature analysis for about 20 sentences for each sense for the preposition. Each line will contain on the order of a couple of thousand features. The function is called with the anchor preposition and the selected feature. The function will generate a data frame constituting the anchor contingency table, used below.

This function reads the feature file and then processes each line. We split the line based on a character demarcating features into an array of

the features. The first "feature" is a sentence number (not further used). The second "feature" is the sense used for the sentence, used to specify the row where the feature value(s) will be tabulated. The next step is a regular expression over the features to obtain a list of the indexes containing the specified feature. If it is not present (e.g., the part of speech has not been identified for the instance), we proceed to the next line of the file.

Next, we obtain the feature's value (removing feature name for the feature), which is used to specify the column where the feature value will be tabulated. The cell of the data frame (**sense number**,**feature value**) will next be initialized or incremented. If the cell has not yet been initialized, there are three ways this may occur: (1) the value is not already a column, but the sense row exists in the data frame, (2) the value is already a column, but the sense number has not yet been initialized, (3) the cell has an "NA" (Not Available) value. After all lines have been processed, we make sure that the tabulation is completed. If a cell has not been tallied, it initially has an "NA" value. Each such cell is reset to 0. When this is completed, the function returns the data frame as the anchor contingency table.

7.2.2 Creating the Supplementary Row for the Target Sense

To create a supplementary row that can be analyzed with the tabulation for the anchor preposition, we use a function whose arguments are (1) the target preposition, (2) the sense number, and (3) the feature. This is essentially similar to what had been done above, with a couple of minor adjustments. The feature file for the target preposition is loaded and we iterate over each line of the file. Again, we read the features for each line (feats) and obtain the sense number for the sentence. Here, if the the sense number is not equal to the desired sense, we continue to the next line of the file.

For the lines corresponding to the desired sense, we create a data frame based on these lines. The main difference is that this data frame consists of only one row, generating the supplementary row that will be used. As for the data frame for the tabulation of the anchor, the columns of the initial supplementary row are based on the values of the feature in the lines of the target preposition and sense. The columns here may not correspond to the columns in anchor tabulation. We need to use the columns of the anchor tabulation as obtained in the previous section. To synchronize the columns, we initialize a new data frame and iterate over the column names in anchor table. We will use "T" as the row name. For each column, we establish the cell for each column in the anchor table based on the value in supplementary data frame. If there is a value in that cell, we will include that count; if the

supplementary row does not have a value, it is set to 0. If the target data frame has a column for a feature value that does not include in the column names of the anchor data frame, such feature values are discarded.

7.2.3 Substitutable Analysis

The correspondence analysis is performed in the function whose arguments include (1) the substitutable preposition (i.e., the anchor), (2) the feature being used to measure the senses, and (3) the senses (i.e., identifying the prepositions and the senses) which have the anchor preposition as a substitution. In other words, we identify all the senses that have one of the anchor preposition's as a putative substitution preposition. These senses can be assessed quickly, less than a minute. The core of this analysis is performed in the function **CA** ("**C**orrespondence **A**nalysis") in the library FactoMineR in R.

The script begins by reading the preposition dictionary in PDEP, with comma-separated preposition, sense number, and definitions; this dictionary will be used for printing the results. The function begins by printing the anchor contingency table, as described above in section 7.2.1. This can be used to compare the features of the target (preposition, sense). The list of the targets (prepositions and their senses) is read into a data frame to iterate through the targets.

In the iteration, we get the target's entry in the preposition dictionary to obtain its definition from the preposition dictionary. We print the target name, the preposition, the sense, and the definition to serve as the sense that is being examined. We next obtain the target supplementary row feature, as described above in section 7.2.2. This row is now appended to the anchor data frame, now suitable for locating the target sense in a display of the anchor senses.

At this point, the correspondence analysis is now called with its function CA, with the appended data frame, identifying the supplementary row, getting the results of the analysis, which can be examined. In particularly, the results provide coordinates for each preposition sense and each feature, for the anchor tabulation and the supplementary row. We can determine the distances between the supplementary target ("T") and each of the anchor senses. These distances can be ordered in a distance data frame so that we can identify which are the closest senses. With this function, we now print the results. We print the contingency table of the anchor preposition. Then, for each target, we print the target preposition, its sense number, and its definition. We print the supplementary row so that it can be com-

pared with the anchor tabulations. Finally, we print the two closest anchor senses, showing the distance, the anchor preposition, the anchor sense, and its definition.

7.2.4 Lexicographic Benefits for Correspondence Analysis

The steps described above provide methodical procedures for characterizing preposition behavior. The above is only a beginning of what might be available. The example used above, for behind and its putative uses as a substitutable preposition, was selected simply as the first obvious example, based on the preposition abaft. The results were very positive and accomplished very quickly, taking only 5 or 10 seconds. It is not known how well the methods apply generally.

The example examines only 2 of the potential 119 feature combinations. Generally, it was quite easy to examine other combinations, for example, such as the lemma, the WordNet lexical name, or the immediate hypernym. It is not immediate to lay out any general steps that might be best or better. There are some difficulties in some of feature combinations. Further examination is required to see the problematic cases.

As indicated above, the analysis seems best to use the OEC corpus, since the example sentences provide the best distinctions between the senses for a given preposition. However, in several cases, the set of sentences was not complete. For example, the preposition about has only 36 sentences for 5 senses, 20 for one sense, 5 each for another 3 senses, and only 1 sentence for the other sense. The unequal number of sentences likely diminishes the correspondence analysis for other senses that may have about as a substitutable preposition.

7.3 Digraph after Substitutable Prepositions

(In development) In the basic data for the sense tagging, potential substitutable prepositions have been identified. These possible substitutes can be examined with the idea that the null hypotheses that have similar senses. As an initial possibility, we can use the preposition digraph.

7.4 Sense without Substitutes

There are 261 senses that have an empty *opreps* field. There are 192 distinct definitions in these senses. Most (64) of the 69 duplicative senses arise from what are characterized as **Tributary** prepositions that are merely orthographic variants of some other preposition and can substitute for any sense

of that preposition. These prepositions have been given sense inventories corresponding to the base preposition. This class contains 24 senses under 24 prepositions.[16]

Others are *as stated in* from *according to (4(n)), under cover of (3(n))*; *expressing the time when an event takes place* for *about 6(n)* and *around (6(n))*; *identifying the person or thing affected by or receiving something* for *onto (6(n))* and *to 8(3)*; *so as to see or be seen from* for *in sight of (1(1))* and *within sight of (1(1))*; *to be replaced by* for *in favor of* and *in favour of*; and *within reach of; close to attaining* for *in sight of (2(1a))* and *within sight of (2(1a))*

Classes Merged (In development) Where is the preposition digraph? See how much this can be used to identify substitutes. The graphs are in C:/Research/Preps/Digraphs. Where is the data from which these graphs were created?

8 Supersenses

Many senses have been characterized with supersenses (clusters or relations). With other corpus instances, the supersenses and other senses can be examined with reference corpora. Note that this is also similar to the previous discussion on substitutable prepositions (Section 7). Allows similar examination of the PDEP field for **supersenses**, also allowing the examples used in the guidelines for supersenses in Schneider et al. (2017)

9 Multiword Expressions

About 70 senses were added to PDEP that were added to the sense inventories of the prepositions, based on their occurrence the CPA corpus. Many of these senses corresponded to multiword expressions (MWEs), with their own entries in the ODE dictionary (Stevenson and Soanes (2003)).

10 Reviewing the Corpora Tagging

The tagging of individual instances can be assessed against whatever sets of features have been characterized as similar, enabling a more quantitative CA

[16] *'cept (0), 'gainst (8), 'mongst (0), 'pon (4), afore (0), agin (7), amidst (0), betwixt (4), fore (0), frae (7), neath (0), nigh (0), o' (14), o'er (3), outta (4), outwith (0), sans (0), thro' (4), thru (4), thwart (0), till (0), toward (1), upon (4), while (0)* : See `https://www.clres.com/db/classes/ClassTributary.php`

test. It is possible to compare independent tagging against the dictionary definitions for each of the senses.

11 Related Work

(In development) Other research that has developed preposition disambiguation. How this relates to what others have done on this topic.

References

Herve Abdi and Lynne J. Williams. Correspondence analysis. In Neil Selkind, editor, *Encyclopedia of Research Design.* Sage, Thousand Oaks, CA, 2010.

Vít Baisa, Jane Bradbury, Silvie Cinková, Ismaïl El Maarouf, Adam Kilgarriff, and Octavian Popescu. SemEval-2015 task 15: A CPA dictionary-entry-building task. In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, pages 315–324, Denver, Colorado, June 2015. Association for Computational Linguistics. doi: 10.18653/v1/S15-2053. URL `https://www.aclweb.org/anthology/S15-2053`.

Guillaume Desagulier. Clustering methods. In *Corpus Linguistics and Statistics with R: Quantitative Methods in the Humanities*, pages 239–294. Springer International Publishing AG, 2017.

Dylan Glynn. Correspondence analysis: Exploring data and identifying patterns. In Dylan Glynn and Justyna Robinson, editors, *Corpus Methods for Semantics: Quantitative studies in polysemy and synonymy*, pages 443–486. John Benjamins Publishing Company, 2014.

Michael Greenacre. *Correspondence analysis in practice, Third Edition.* CRC press, Boca Raton, FL, 2017.

Dirk Hovy, Stephen Tratz, and Eduard Hovy. What's in a preposition? dimensions of sense disambiguation for an interesting word class. In *Coling 2010: Posters*, pages 454–462, Beijing, China, August 2010. Coling 2010 Organizing Committee. URL `https://www.aclweb.org/anthology/C10-2052`.

Karolina Krawczak and Dylan Glynn. Operationalising construal: A corpus-based study in cognition and communication constructions. *Jezikoslovlje*, 20(1):1–30, 2019. URL `https://hrcak.srce.hr/219568`.

Ken Litkowski. The preposition project corpora. Technical Report 13-01, CL Research, Damascus, MD 20872 USA, April 2013. URL http://www.clres.com/online-papers/TPPCorpora.pdf.

Ken Litkowski. Pattern Dictionary of English Prepositions. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1274–1283, Baltimore, Maryland, June 2014. Association for Computational Linguistics. URL http://www.aclweb.org/anthology/P14-1120.

Ken Litkowski. Identifying the least common subsumer ontological class. Technical Report 16-01, CL Research, Damascus, MD 20872 USA, February 2016a. URL http://www.clres.com/online-papers/LCSClasses.pdf.

Ken Litkowski. Pattern dictionary of english prepositions. In Mona Diab, Aline Villavicencio, Marianna Apidianaki, Valia Kordoni, Anna Korhonen, Preslav Nakov, and Mark Stevenson, editors, *Essays in Lexical Semantics and Computational Lexicography - In Honor of Adam Kilgarriff*. Springer, 2016b. URL http://www.clres.com/online-papers/LitkowskiPDEP.pdf.

Ken Litkowski. Feature ablation for preposition disambiguation. Technical Report 16-02, CL Research, Damascus, MD 20872 USA, May 2016c. URL http://www.clres.com/online-papers/PSDFeatAbl.pdf.

Ken Litkowski. Honing the sketch engine prepositions. Technical Report 19-01, CL Research, Damascus, MD 20872 USA, 2019. URL http://www.clres.com/online-papers/HoneSkE.pdf.

Ken Litkowski and Orin Hargraves. The Preposition Project. In *Proceedings of the Second ACL-SIGSEM Workshop on The Linguistic Dimensions of Prepositions and their Use in Computational Linguistics Formalisms and Applications*, pages 171–179, Colchester, England, University of Essex, April 2005. Association for Computational Linguistics.

Kenneth C. Litkowski. Digraph analysis of dictionary preposition definition. In *Proceedings of the ACL-02 Workshop on Word Sense Disambiguation: Recent Successes and Future Directions*, pages 9–16. Association for Computational Linguistics, July 2002. doi: 10.3115/1118675.1118677. URL http://www.aclweb.org/anthology/W02-0802.

Kenneth C. Litkowski and Orin Hargraves. Coverage and inheritance in the preposition project. In *Proceedings of the Third ACL-SIGSEM Workshop on Prepositions*, 2006. URL https://www.aclweb.org/anthology/W06-2106.

Kenneth C. Litkowski and Orin Hargraves. Semeval-2007 task 06: Word-sense disambiguation of prepositions. In *Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval-2007)*, pages 24–29, Prague, Czech Republic, June 2007. Association for Computational Linguistics. URL http://www.aclweb.org/anthology/S/S07/S07-1005.

James McCracken. Rebuilding the Oxford dictionary of English as a semantic network. In *Proceedings of the Workshop on Enhancing and Using Electronic Dictionaries*, pages 69–72, Geneva, Switzerland, August 29th 2004. COLING. URL https://www.aclweb.org/anthology/W04-2114.

Barbara McGillivray, Christer Johansson, and Daniel Apollon. Semantic structure from correspondence analysis. In *Coling 2008: Proceedings of the 3rd Textgraphs workshop on Graph-based Algorithms for Natural Language Processing*, pages 49–52, Manchester, UK, August 2008. Coling 2008 Organizing Committee. URL https://www.aclweb.org/anthology/W08-2007.

Nathan Schneider, Vivek Srikumar, Jena D. Hwang, and Martha Palmer. A hierarchy with, of, and for preposition supersenses. In *Proc. of The 9th Linguistic Annotation Workshop*, pages 112–123, Denver, Colorado, USA, June 2015.

Nathan Schneider, Jena D. Hwang, Vivek Srikumar, Meredith Green, Abhijit Suresh, Kathryn Conger, Tim O'Gorman, and Martha Palmer. A corpus of preposition supersenses. In *Proc. of LAW X – the 10th Linguistic Annotation Workshop*, pages 99–109, Berlin, Germany, August 2016.

Nathan Schneider, Jena D. Hwang, Archna Bhatia, Na-Rae Han, Vivek Srikumar, Tim O'Gorman, and Omri Abend. Adposition supersenses v2. *CoRR*, abs/1704.02134, 2017. URL http://arxiv.org/abs/1704.02134.

Vivek Srikumar and Dan Roth. Modeling semantic relations expressed by prepositions. *Transactions of the Association for Computational Linguistics*, 1:231–242, 2013a. doi: 10.1162/tacl_a_00223. URL https://www.aclweb.org/anthology/Q13-1019.

Vivek Srikumar and Dan Roth. An inventory of preposition relations, 2013b.

Angus Stevenson and Catherine Soanes, editors. *The Oxford Dictionary of English (ODE)*. Clarendon Press, Oxford, 2003.

STHDA. Ca - correspondence analysis in r: Essentials, 2017. URL `http://www.sthda.com/english/articles/31-principal-component-methods-in-r-practical-guide/113-ca-correspondence-analysis-in-r-essentials/`.

Stephen Tratz. *Semantically-enriched parsing for natural language understanding*. PhD thesis, University of Southern California, 2011.

Stephen Tratz and Eduard Hovy. A fast, accurate, non-projective, semantically-enriched parser. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 1257–1268, Edinburgh, Scotland, UK., July 2011. Association for Computational Linguistics. URL `https://www.aclweb.org/anthology/D11-1116`.

Phillip M Yelland. An introduction to correspondence analysis. *The Mathematica Journal*, 12(1):86–109, 2010.

A Correspondence Analysis Techniques

Correspondence analysis has several varieties, grouped into two major types, simple and multiple. Simple CA examines contingency tables; multiple CA works with (See Greenacre (2017), Abdi and Williams (2010), Desagulier (2017) and Glynn (2014).)

A.1 Simple Correspondence Analysis

The singular value decomposition (SVD) factors the standardized residual matrix into a diagonal matrix known as the singular values, explained the variance of the contingency table . The table has the original size of m x n. The diagonal matrix has *n-1* singular values, in decreasing magnitude. The sum of the singular values is known as the total inertia, constituting 100 percent over the *n-1* dimensions. A figure, such as in Figure 1, shows the percentage of each dimension that is covered. A biplot shows the results for two of those dimensions, identifying how much of the inertia (the total variance) is covered for each. In general, the hope is that the total of the

first two dimensions will cover at least 80 percent of the variance. (See also STHDA (2017).)

After this basic statistic, CA allows considerable analysis. The first step creates a correspondence matrix (CM), dividing each cell by the sum of all the cells, so that the sum of the cells of the CM is equal to 1. This matrix is used to compute a matrix of standardized residuals, which is used to perform a singular value decomposition (SVD) into its factorizations. This permits a plot of the rows and columns of the original table, in this case shown in Figure 1. The figure visualizes how the senses and the parts of speech relate to one another. There are several packages for correspondence analysis, particularly in R, one in Python, and others in various statistical software. They can also perform the components in CA or can be implemented by developing the computations.

A.2 Supplementary Points

In a contingency table, as described in a A.1, the rows and columns establish the principal axes and the basis for the plots. These axes are viewed as *active*. Each active has a different force of attraction - profiles farther from the average have more "leverage" in orienting the map. Sometimes, we wish to examine points that have no mass at all (i.e., their contribution to the inertia is zero). Such points are called *supplementary points* or *passive*. There are three common situations: an additional column, an additional row, or another row which is the sum of two rows. In these situations, the procedure is to add the supplementary rows or columns, as if they were to be analyzed as part of the contingency table, but then label them as supplementary. In the analysis, it is still possible to compute what inertia the supplementary points would have and we can show how these points relate to the original table, i.e., to determine the closest points of the original table. (See Greenacre (2017), pp. 89-96 and 263-264 and Yelland (2010).)

A.3 Multiple Correspondence Analysis

B Word-Finding Rules

This appendix characterizes the analysis of the word-finding rules tabulated in the feature selection tables. Each subsection deals with a particular rule. The word-finding rules fall into two groups: words pertaining to the governor and words pertaining to the complement.

B.1 Governor (h)

The governing token of the preposition in the dependency parse.

B.2 Verb or Head to the Left (l)

The first token with a lower index in the dependency parse that has a verb part of speech or a noun, pronoun, or adjective label.

B.3 Head to the Left (hl)

The first token with a lower index in the dependency parse that has a noun, verb, pronoun, or adjective label and which is identified as the head of the prepositional phrases.

B.4 Verb to the Left (vl)

The first token to the left that has a verb label.

B.5 Word to the Left (wl)

The first token to the left, if the preposition is not the first word.

B.6 Syntactic Preposition Complement (c)

Views the preposition as a head in the dependency parse, and examines its children for tokens identified as preposition objects or complements.

B.7 Heuristic Preposition Complement (hr)

Examines tokens that follow the preposition, looking at the part of speech to identify the most likely furthest complement (examining nouns, adjectives, pronouns, gerunds, and some specific words such as “some” or “each”.

C Feature Extraction Rules

This appendix characterizes the analysis of the feature extraction rules tabulated in the feature selection tables. Each subsection deals with a particular rule. In these discussions, significant rules indicate that the observed frequencies for a feature are different for the expected frequencies. The indicated feature occurs many more times than occurring when it is significantly different.

C.1 Immediate WordNet Hypernyms (h)

WordNet immediate hypernym: the lemmas in the immediate WordNet hypernyms (a large number of values, with perhaps 10 values for each token) (113,686 in the three corpora)

C.2 All WordNet Hypernyms (ah)

all WordNet hypernyms (h): the lemmas in all WordNet hypernyms, up to 15 levels in the WordNet hierarchy (doubles the numbers for WordNet immediate hypernyms) (189,240 in the three corpora)

C.3 Affixes (af)

A feature that characterizes prefixes and suffixes present in the token (such as numerical prefixes and disease suffixes), (there are 27 possible affixes that are checked; they occur relatively frequently). (3717 in the three corpora)

C.4 Capitalized Word (c)

Whether the word is capitalized: the single value *true* when the token is capitalized, generally a low frequency feature (396 in the three corpora).

C.5 All WordNet Gloss Words (g)

All words in all glosses of all the senses in WordNet of the token (by far, the largest set of features, as much as half of all features) (388,825 in the three corpora)

C.6 Lemma (l)

The lemma for the token, if identifiable (generally equal to the number of instances, but with more duplicated values than the **w** values) (5,945 in the three corpora, noting that TPP had the highest number, indicating that the FrameNet focus on specific words and lemmas)

C.7 Word (w)

The token itself (generally equal to the number of instances, with some occurring multiple times, with 4308 significance instances). (4,308 in the three corpora)

C.8 Word Class (wc)

Word class (**wc**) has one of four values, *noun*, *verb*, *adjective*, or *adverb*. This feature extraction rule is significant in a relatively small cases (834 in the three corpora).

C.9 WordNet Lexical Name (ln)

One of 40 values (with many occurring for a particular token, with several occurrences for each token, reflecting WordNet polysemy), (11,044 in the three corpora)

C.10 Part of Speech (pos)

One of 37 values (generally only about half of these occur for a given preposition, but usually covering all instances), (3,111 in the three corpora)

C.11 Rule Itself (ri)

A feature with the sole value *rulefired* added when a word-finding rule is successful in finding a token (generally succeeds for all word-finding rule and for all instances, but there are usually many exceptions). This occurred 20 times in the three corpora.

C.12 WordNet Immediate Synonym (s)

The lexemes in the WordNet synsets for the token (a large number of possible values, with perhaps as many as 15 occurrences for each instance for each token) (71,811 in the three corpora)

C.13 WordNet All Synonyms (as)

Extends the immediate WordNet synsets to include all derived forms and morphological variants, with more than double times the number of values and occurrences. (192,241 in the three corpora)

C.14 Pattern Dictionary of English Verbs (cpa)

A feature generated only when the head (**h**-B.1) is in the pattern dictionary of English **verbs** and has the preposition as part of its specification, as described in Baisa et al. (2015), a low frequency feature, occurring only for the more common prepositions. (8 in the three corpora)

C.15 FrameNet Entry (fn)

A feature generated only when the head (**h**-B.1) is in the FrameNet dictionary and has the preposition as a frame element realization, a low frequency feature, occurring only for the more common prepositions. (372 in the three corpora)

C.16 VerbNet Entry (vn)

A feature generated only when the head (**h**-B.1) is in the VerbNet dictionary and has the preposition as part of its specification, a low frequency feature, occurring only for the more common prepositions. (43 in the three corpora)

C.17 Oxford Noun Hierarchy (o)

A feature generated only for preposition complements (**hr**-B.7) and governors (**h**-B.1) that are nouns, where the noun is accessed in the Oxford Dictionary of English noun hierarchy to identify its immediate hypernyms, as described in McCracken (2004), a moderately frequent feature, with the possibility of multiple hypernyms for a token. (4,008 in the three corpora)