

CHAPTER 6

EXPERIMENTS ON DISTRIBUTIONAL VERB SIMILARITY

This chapter describes a series of experiments that deal with various aspects of assigning and assessing lexical similarity scores to a set of English verbs on the basis of their distributional context in the English gigaword corpus. These experiments simultaneously varied four parameters that influence distributional lexical similarity with respect to five different verb classification schemes. The verb classifications considered were Levin (1993), VerbNet, FrameNet, Roget's Thesaurus, and WordNet. The parameters considered were choice of feature set, measure of lexical similarity, feature weighting, and feature selection. The purpose of this series of experiments is to examine interactions between these parameters with respect to the five verb schemes mentioned above and described in Chapter 5.

The remainder of this chapter is organized as follows. Section 6.1 describes the set of verbs and corpus used in the experiments. Section 6.2 describes two measures for evaluating distributional lexical similarity used in the experiments. Section 6.3 discusses the feature sets and procedures for extracting features from the corpus. Section 6.4 describes the experiments performed.

6.1 Data Set

6.1.1 Verbs

The set of verbs used in the following experiments was selected from the union of Levin, VerbNet, and FrameNet verbs that occurred at least 10 times in the English gigaword corpus (i.e., were tagged as verbs at least 10 times by the Clark and Curran CCG parser; details of the parsing procedure are in Section 6.3.2). Roget and WordNet contain many more items than each of Levin, VerbNet, and FrameNet, so in order to maintain an approximately equal number of verbs in each verb scheme, we restricted the selection of verbs from Roget and WordNet to ones that appear in either Levin, VerbNet, or FrameNet. This selection procedure resulted in a total of 3937 verbs; the number of items per verb scheme is shown in Table 6.1.

Verb Scheme	Total Num. Verbs	Num Verbs Included in Exps.
Levin	3004	2886
VerbNet	3626	3426
FrameNet	2307	2110
WordNet	11529	3762
Roget	≈14000	2879

Table 6.1: Number of verbs included in the experiments for each verb scheme

Following Curran and Moens (2002)’s work on automatic thesaurus extraction, we do not distinguish between senses of verbs in the evaluation for two reasons. First, because we aggregate all occurrences of a verb into a single context vector, the extracted items represent a conflation of senses. Second, items that are ostensibly classified as belonging to only one class in, e.g., Levin or FrameNet rarely belong to only one class in practice. For example, one of the most frequent verbs in the English gigaword corpus is *add*, which Levin places exclusively in the MIX class (e.g.,

combine, join, link, merge, etc.). However, in the English gigaword corpus, this verb is used most often as a synonym for *say* (e.g., “*I don’t think I’ll really fully realize the impact until I swear in,*” *Bush added.*), and FrameNet places it exclusively in the STATEMENT class. Because of the recognized difficulties in establishing an inventory of senses for verbs in particular and words in general (e.g., Manning and Schütze, 1999: 229-231), we conflated senses in the verb schemes and defined items as neighbors as follows.

1. Levin, VerbNet, FrameNet: two items are neighbors if the intersection of the classes they belong to is non-empty; e.g., they share at least one sense which puts them in the same class. For example, for VerbNet *link* \in {MIX, TAPE} and *harness* \in {BUTTER, TAPE} are neighbors because $\{\text{MIX, TAPE}\} \cap \{\text{BUTTER, TAPE}\} = \{\text{TAPE}\}$.
2. Roget, WordNet: two words are neighbors if either is listed as a synonym of the other.

Table 6.2 shows the average number of neighbors per verb in our study for each of the verb schemes using these criteria. Table 6.3 contains the baselines that

	Levin	VerbNet	FrameNet	Roget	WordNet
Mean (Std. Dev.)	86.3 (85.0)	103.5 (120.5)	40.4 (41.9)	31.9 (20.1)	12.6 (10.9)
Max	513	669	248	185	76
Median	49	48	23	39	10
Min	2	1	1	4	1

Table 6.2: Average number of neighbors per verb for each of the five verb schemes

indicate the chance that two verbs in our study selected at random are neighbors. For all five schemes, the baseline is less than 3%.

Verb Classification	Baseline
Levin	0.029
VerbNet	0.028
FrameNet	0.018
Roget	0.006
WordNet	0.001

Table 6.3: Chance of randomly picking two verbs that are neighbors for each of the five verb schemes

6.1.2 Corpus

The English gigaword corpus (Graff, 2003) is composed of nine years of newspaper text (1994–2002) from four distinct international sources of English newswire: Agence France Press English Service, Associated Press Worldstream English Service, The New York Times Newswire Service, and The Xinhua News Agency English Service. This text covers a wide spectrum of subjects and is not tied to any particular domain, although it is skewed towards political and economic news.

6.2 Evaluation Measures

As discussed in Chapter 5, framing the task of extracting distributionally similar verbs in terms of an information retrieval task or thesaurus construction enables the use of evaluation measures commonly used in those domains. Following representative work on automatic thesaurus extraction such as Lin (1998a); Curran and Moens (2002), and Weeds (2003) we utilize measures of precision and inverse rank score in evaluating the results of the experiments reported here. These measures are presented in the following sections along with a discussion of their key characteristics.

6.2.1 Precision

Following the methodologies for evaluating distributional lexical similarity reported in, e.g., Lin (1998a), Curran and Moens (2002), and Weeds (2003), one evaluation measure that we report here is precision at k , where k is a fixed, usually low level of retrieved results. We report precision at k for $k = 1, 5, 10$. However, Manning et al. (2008: 148) point out that the highest point on the precision recall curve can be of no less interest than mean single point summaries such as F1, R-precision, or mean average precision. For the purposes of comparing feature sets and distance measures across verb schemes, we report microaveraged maximum precision (MAXP), defined as the point on the precision recall curve at which precision is the highest. We compute maximum precision for each individual verb and report the average of these values. It is always the case in our study that the trends reported for MAXP also hold for $k = 1, 5, 10$.

When precision is high and k is relatively large, this indicates that many same class items are clustered within the most highly ranked neighbors of a target verb (e.g., *appeal* in Figure 6.1). Low precision values associated with large k indicate that very few of the distributionally most similar items belong to the same class as the target (*enshrine* in Figure 6.1). High precision and small k suggest that only a few of the actual same-class items are contained within the set of highly ranked empirical neighbors (*reply* in Figure 6.1), or that the size of the class is small. The relative size of the class is shown in the precision curve by those portions of the curve that jag upwards, indicating that a cluster of same-class items has been retrieved at some lower value of k . However, precision alone does not account for the overall distribution of matches within the ranked set of results. A measure that does a better

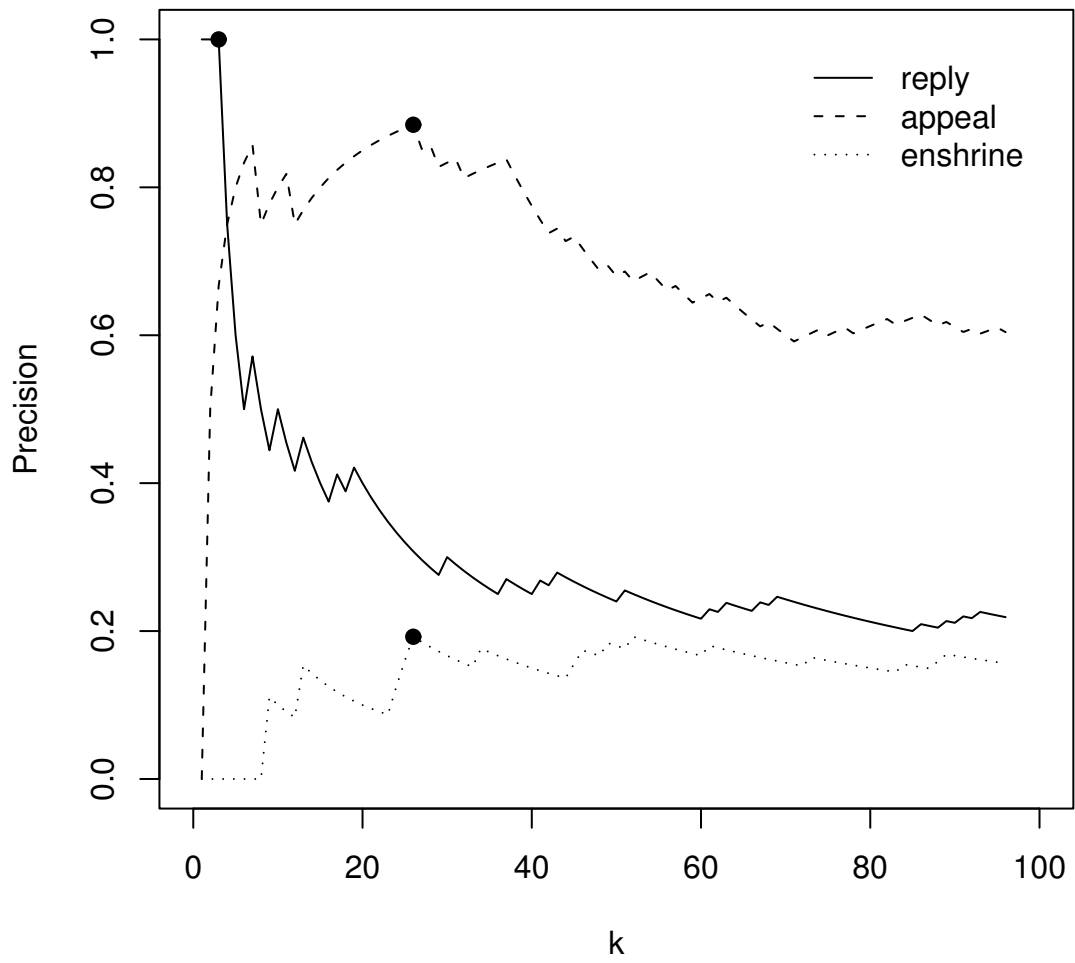


Figure 6.1: Precision at levels of k for three verbs

job of accounting for the relative positions of matches within the total set of results is the inverse rank score.

6.2.2 Inverse Rank Score

Following Curran and Moens (2002); Gorman and Curran (2006), we also evaluate distributional similarity in terms of inverse rank score, which is the sum of the inverse of the rank of each same class item in the top ranked m items:

$$\text{INVR} = \sum_{k=1}^m \frac{x_k}{k}$$

where x_k is an indicator variable defined as

$$x_k = \begin{cases} 1 & \text{if the class of the } k^{\text{th}} \text{ item matches the target class} \\ 0 & \text{otherwise.} \end{cases}$$

For example, if items at rank 2, 3, and 5 match the target class, the inverse rank score is $\frac{1}{2} + \frac{1}{3} + \frac{1}{5} = 1.03$. In the experiments reported here, only the 100 most highly ranked items were retained, so the maximum INVR score is 5.19. INVR is a useful measure because it distinguishes between result lists that contain the same number of same-class items but rank them differently. INVR assigns higher scores to result lists in which same-class items are highly ranked. For example, a result list containing 5 matches at ranks 1, 2, 3, 5, 8 receives an INVR score of 2.16; another list containing the same 5 items at rank 3, 4, 5, 6, 7 receives an INVR score of 1.09.

As with MAXP, discretion is required in interpreting INVR in the current study. For example, if one word has five synonyms and another has ten synonyms, and both sets are returned as the highest ranked items, the inverse rank score of

the second word will be higher than the score for the first word without indicating a difference in the quality of the ranked synonyms. Similarly, a word with many lowly ranked synonyms can receive a higher INVR score than a word with only a few highly ranked synonyms. For example, a word with only five synonyms ranked in positions 2–6 would receive an INVR score of 1.45; a word with eleven synonyms in positions 1, 15–24 would receive an INVR score of 1.48.

Finally, because INVR is sensitive to the number of matched items, we cannot use it to compare across verb schemes that assign different numbers of neighbors to each verb. In this case, we only report measures of precision.

6.3 Feature Sets

This section describes the feature sets used here for assessing distributional verb similarity¹. We evaluated four different feature sets for their effectiveness in extracting classes of distributionally similar verbs: Syntactic Frames, Labeled Dependency Relations, Unlabeled Dependency Relations, and Lexicalized Syntactic Frames. Syntactic frames contain mainly syntactic information, whereas the other three feature sets encode varying combinations of lexical and syntactic information. Each of these feature types has been used extensively in previous research on automatic Levin verb classification.

6.3.1 Description of Feature Sets

Syntactic Frames. Syntactic frames have been used extensively as features in early work on automatic verb classification due to their relevance to the alternation behaviors which are crucial for Levin’s verb classification (e.g., Schulte im Walde, 2000; Brew and Schulte im Walde, 2002; Schulte im Walde and Brew, 2002; Korhonen et al., 2003). Syntactic frames provide a general feature set that can in principle be applied to distinguishing any number of verb classes. However, using syntactic information alone does not allow for the representation of semantic distinctions that are also relevant in verb classification. Work in this area has been primarily concerned with verbs taking noun phrase and prepositional phrase complements. To this end, prepositions

¹Portions of Section 6.4.4 were co-authored with Jianguo Li.

have played an important role in defining relevant syntactic frames. However, only knowing the identity of prepositions is not always enough to represent the desired distinctions.

For example, the semantic interpretation of the syntactic frame NP-V-PP(*with*) depends to a large extent on the NP argument selected by the preposition *with*. In (1), the same surface form NP-V-PP(*with*) corresponds to three different underlying meanings. However, such semantic distinctions are totally lost if lexical information is disregarded.

- (1) a. I ate with *a fork*. [INSTRUMENT]
b. I left with *a friend*. [ACCOMPANIMENT]
c. I sang with *confidence*. [MANNER]

Lexicalized Frames. This deficiency of unlexicalized subcategorization frames has led researchers to incorporate lexical information into the feature representation. One possible improvement over subcategorization frames is to enrich them with lexical information. Lexicalized frames are usually obtained by augmenting each syntactic slot with its head noun (2).

- (2) a. I ate with *a fork*. [INSTRUMENT] \rightarrow NP(*I*)-V-PP(*with:fork*)
b. I left with *a friend*. [ACCOMPANIMENT] \rightarrow NP(*I*)-V-PP(*with:friend*)
c. I sang with *confidence*. [MANNER] \rightarrow NP(*I*)-V-PP(*with:confidence*)

The analysis of feature sets previously used in automatic verb classification suggests that both syntactic and lexical information are relevant in determining meaning of Levin verbs (e.g., Li and Brew, 2008). This agrees with the findings in previous studies on WSD (Lee and Ng, 2002) that although syntactic information on its own is not very informative in automatic word sense disambiguation, its combination with lexical information results in improved disambiguation. The next two feature types focus on various ways to mix syntactic and lexical information.

Dependency relations. Recall that subcategorization frames are limited as verb features in the properties of verb behaviors they tap into. Lexicalized frames, with potentially improved discriminatory power, suffer from increased exposure to data sparsity. One way to overcome data sparsity is to break lexicalized frames into dependency relations. Dependency relations contain both syntactic and lexical information (3).

- (3) a. SUBJ(*I*), PP(*with:fork*)
- b. SUBJ(*I*), PP(*with:friend*)
- c. SUBJ(*I*), PP(*with:confidence*)

However, since we augment prepositional phrases with the head nouns selected by prepositions, as in PP(*with:fork*), the data sparsity problem still exists. We therefore break all prepositional phrases in the form PP(preposition:noun) into two separate dependency relations: PP(preposition) and PP-noun, as shown in (4).

- (4) a. SUBJ(*I*), PP(*with*), PP-*fork*
- b. SUBJ(*I*), PP(*with*), PP-*friend*
- c. SUBJ(*I*), PP(*with*), PP-*confidence*

Although dependency relations have proved effective in a range of lexical acquisition tasks such as word sense disambiguation (McCarthy, Koeling, Weeds, and Carroll, 2004), construction of a lexical semantic space (Padó and Lapata, 2007), and detection of polysemy (Lin, 1998a), their utility in automatic verb classification has not been as thoroughly examined.

Unlabeled Dependency Relations. In order to further examine the separate contributions of lexical and syntactic information, we removed the syntactic tag from the labeled dependency relations, leaving a feature set that consists only of lexical items that were selected on the basis of their structural relation to the verb. However, the distinction between, e.g., Subject, Object, and Prepositional Object is no longer explicitly represented in the unlabeled feature set. The representation of the examples above using this feature set is shown in (5).

- (5) a. I ate with a fork \rightarrow {I, with, fork}
b. I left with a friend \rightarrow {I, with, friend}
c. I sang with confidence \rightarrow {I, with, confidence}

6.3.2 Feature Extraction Process

The experiments reported here used Clark and Curran’s (2007) CCG parser, a log-linear parsing model for an automatically extracted lexicalized grammar, to automatically extract the features described above from the English gigaword corpus (Graff, 2003). The lexicalized grammar formalism used by the parser is combinatory categorial grammar (CCG) (Steedman, 1987; Szabolcsi, 1992), and the grammar is automatically extracted from CCGbank (Hockenmaier and Steedman, 2005). The parser produces several output formats; we use grammatical relations (Briscoe, Carroll, and

Watson, 2006) and employ a post-processing script to extract four types of grammatical relations that are relevant to verbs: Subject-Type, Object-Type, Complement-Type, and Modifier-Type.

The primary feature type that we extract from the parser’s output is lexicalized syntactic frame. Syntactic frames are defined in terms of the syntactic constituents used in the Penn Treebank (Marcus et al., 1993) style parse trees. For example, a double object frame exemplified by a sentence like *Sam handed Tom the flute* can be represented as NP1-V-NP2-NP3. A lexicalized syntactic frame augments the structural information represented by a syntactic frame with the lexical head of each constituent, e.g., NP1(*Sam*)-V(*hand*)-NP2(*Tom*)-NP3(*flute*).

Extracting Subject-Type Relations. Table 6.4 illustrates the three types of Subject-Type relations extracted from the parser’s output. The first column indicates the relation, the second column contains an example of the relation, the third column contains representative output from the parser, and the fourth column contains the lexicalized frame that is extracted as a result of processing the parser’s output.

Each relation is represented by the parser as a quadruple, with the first element in the quadruple always containing the name of the relation. The order of the other elements depends on the type of relation. For Subject-Type, the verb is always the second element of the quadruple. Each lexical entry in the parser’s output is indexed according to its position in the input sentence.

This index also points to each item’s position in a lemmatized, part-of-speech tagged representation of the sentence that is also part of the parser’s output. In order to extract features from the *nsubj* relation, we combine the lemmatized form of the verb with the lemmatized form of the third element in the quadruple. Similarly, in order to extract features from the *xsubj* and *csbj* relations, we combine the lemmatized

form of the verb with the lemmatized form of the fourth element in the quadruple. If the fourth element is ‘_’, we do not lexicalize the relation, which is the same thing as lexicalizing it with a null element.

Grammatical Relation	Parser Output	Extracted Feature
non-clausal subject <i>Kim left</i>	(ncsubj left Kim _)	SUBJ(<i>Kim</i>)-V(<i>leave</i>)
unsaturated clausal subject <i>leaving matters</i>	(xsubj matters leaving _)	SUBJ(<i>NONE</i>)-V(<i>matter</i>)
saturated clausal subject <i>that he came matters</i>	(csubj matters came that)	SUBJ(<i>that</i>)-V(<i>matter</i>)

Table 6.4: Examples of Subject-Type relation features

Extracting Object-Type Relations. Table 6.5 illustrates the three types of Object-Type relations extracted from the parser’s output. Object-Type relations are represented as triples; the verb is always the second element, and the object is always the third element. In order to extract features from the Object-Type relation, we combine the lemmatized form of the verb with the lemmatized form of the third element in the quadruple.

Grammatical Relation	Parser Output	Extracted Feature
direct object <i>likes her</i>	(dobj likes her)	V(<i>like</i>)-DOBJ(<i>her</i>)
second object <i>gave Kim toys</i>	(obj2 gave toys)	V(<i>give</i>)-IOBJ(<i>toy</i>)
indirect object <i>flew to Paris</i>	(iobj flew to)	V(<i>fly</i>)-PP(<i>to</i>)

Table 6.5: Examples of Object-Type relation features

Extracting Complement-Type Relations. Table 6.6 illustrates the three types of Complement-Type relations extracted from the parser’s output. Prepositional phrase complement type relations (*pcomp*) are represented as triples; the verb is the second element, and the preposition is the third element. *xcomp* relations are represented as quadruples; the verb is the third element and the lexical head of the prepositional phrase is the fourth element. In order to extract features from the *pcomp* relation, we combine the lemmatized form of the verb with a “PP” label and the third element in the *pcomp* triple. In order to extract features from the *xcomp* relation, we combine the lemmatized form of the verb with the lemmatized form of the third element in the quadruple with a “GER” label indicating that the third element represents the lexical head of a gerundive.

Grammatical Relation	Parser Output	Extracted Feature
PP complement <i>pass by the shop</i>	(pcomp pass by)	V(<i>pass</i>)-PP(<i>by</i>)
unsaturated VP complement <i>enjoy running</i> <i>hate to go</i>	(xcomp _ enjoy running) (xcomp to hate go)	V(<i>enjoy</i>)-GER(<i>run</i>) V(<i>hate</i>)-GER(<i>go</i>)
clausal complement <i>knew that you left</i>	(xcomp that knew left)	V(<i>know</i>)-GER(<i>leave</i>)

Table 6.6: Examples of Complement-Type relation features

Extracting Adjunct-Type Relations. Table 6.7 illustrates the three types of Adjunct-Type relations extracted from the parser’s output. Adjunct-Type relations are represented as quadruples; the verb is always the third element and the modifying item is always the fourth element. In order to extract Adjunct-Type relations, we combine the lemmatized form of the verb with the lemmatized form of the third element in the quadruple. The label of the relation is obtained by index into the

lemmatized, part-of-speech tagged representation of the sentence that is also part of the parser’s output and comes from the part-of-speech tag of the the third element in the quadruple.

Grammatical Relation	Parser Output	Extracted Feature
non-clausal modifier		
<i>sit on a table</i>	(nmod _ sit on)	V(<i>sit</i>)-PP(<i>on</i>)
<i>left early</i>	(nmod _ left early)	V(<i>leave</i>)-ADVP(<i>early</i>)
unsaturated clausal modifier		
<i>entered smiling</i>	(xmod _ entered smiling)	V(<i>enter</i>)-GER(<i>smile</i>)
<i>left to catch her</i>	(xmod _ left to)	V(<i>leave</i>)-INFV(<i>to</i>)
<i>returned alive</i>	(xmod _ returned alive)	V(<i>return</i>)-ADJP(<i>alive</i>)
clausal modifier		
<i>when he came, Kim left</i>	(cmod _ left when)	V(<i>leave</i>)-S(<i>when</i>)

Table 6.7: Examples of Adjunct-Type relation features

The process of extracting lexicalized syntactic frames from CCG output is illustrated in Table 6.8 for the input sentence *Two men broke the door with a hammer.*

- Identify verbs in the grammatical relations output by the parser by index into the lemmatized, part-of-speech tagged representation of the sentence. For example, *broke* is identified as a verb by its index of 2, which points to the element *broke|break|VBD*.
- Identify *dobj* dependents of prepositions among the dependency relations. For example, *with* is identified as a preposition by its Penn Treebank-style part of speech tag *IN*, and *hammer* is identified as its object: PP(*with*)-*hammer*.
- Identify dependents of the verb by extracting items from the grammatical relations whose index points to the verb. For example, *door* and *with* are identified as direct and indirect objects of *broke*, respectively; *men* is identified as its subject. The lemmatized form of each item is combined along with its

grammatical relation to for a lexicalized syntactic frame: SUBJ(*man*)-V(*break*)-DOBJ(*door*)-PP(*with*)*_hammer*.

- Other feature types are adapted from this primary representation. For example, syntactic frames are obtained by removing lexical material from the frame²: SUBJ-V-DOBJ-PP(*with*). Labeled dependencies are obtained by splitting the frame and representing it as a set comprised of its individual lexicalized dependents: {SUBJ(*man*), V(*break*), DOBJ(*door*), PP(*with*)*_hammer*}. Unlabeled dependencies are retained by discarding the structural information associated with each element in the frame and retaining only the lexical heads: {*man*, *break*, *door*, *with*, *hammer*}.

Input Sentence:

Two men broke the door with a hammer

Output Relations:

(det door_4 the_3)
(dobj _ broke_2 door_4)
 (det hammer_7 a_6)
(dobj with_5 hammer_7)
(iobj broke_2 with_5)
(nsubj broke_2 men_1 _)
 (det men_1 two_0)

Output part of speech tags

Two|two|CD men|man|NNS broke|break|VBD the|the|DT door|door|NN
 with|with|IN a|a|DT hammer|hammer|NN

Table 6.8: Example of grammatical relations generated by Clark and Curran (2007)’s CCG parser

One consideration to be given to the construction of different feature sets is their scalability in terms of the potential number of features that will be generated.

²The lexical heads of prepositional phrases were retained.

The main motivation for using a large corpus like the English gigaword corpus is that relatively infrequent items may still be attested often enough to allow generalizations that would not be possible using a smaller resource. A potential downside of using such a large corpus is the bulk of data that will be generated and must be processed. Most similarity metrics run in time linear to the number of non-zero elements in two vectors being compared. Therefore, the more features, the longer the run time for finding nearest neighbors. Figure 6.2 shows the increase in the number of features as a function of the number of verb instances encountered in the English gigaword corpus.

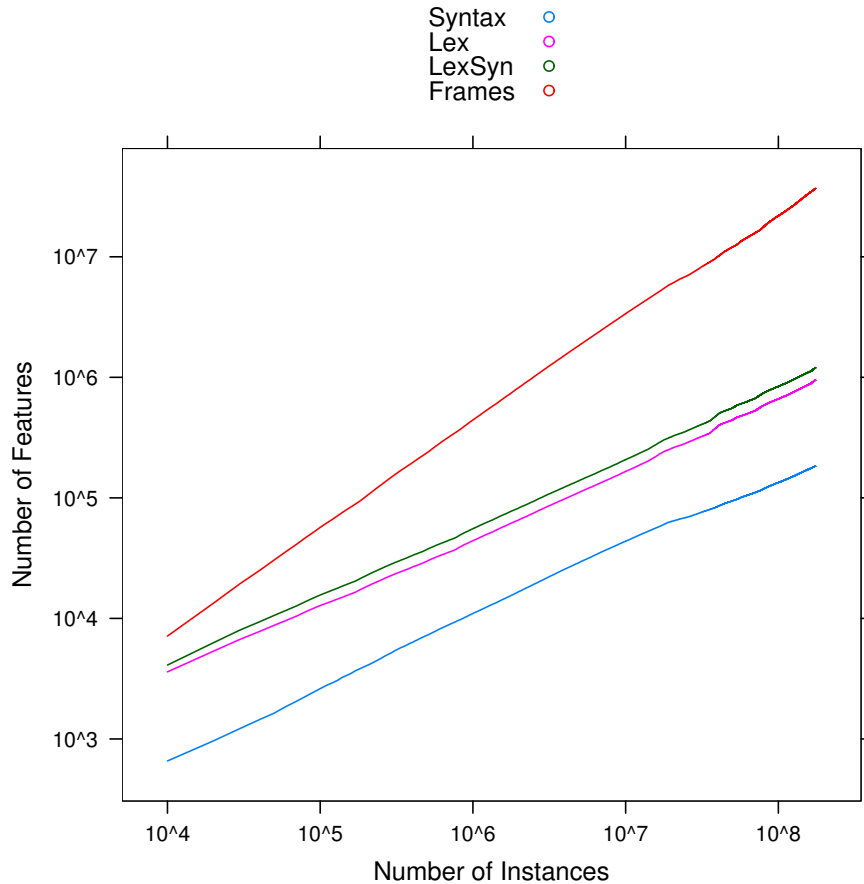


Figure 6.2: Feature growth rate on a log scale

Due to their highly specific nature, lexicalized frames constitute the largest feature set. This is because the chance of the exact combination of a verb and its lexical arguments, including prepositional phrases, occurring more than once in any corpus is very small. Therefore, most of the lexical frame features are relatively infrequent. The number of labeled and unlabeled dependency relation features is fairly close. This means that including the structural information in the feature set does not greatly impact storage or performance, and if the grammatical labels improve verb classification, that could be considered a reason for using them with relatively little downside. Eliminating lexical information in the syntactic frames results in the smallest feature set, as syntactic frames do tend to occur fairly frequently across verbs.

6.4 Experiments

This section describes a series of experiments that examine the interaction of distance measure, feature set, similarity measure, feature weighting, and feature selection on the assignment of distributionally similar verbs. The verbs used in this study come from the union of Levin, VerbNet, and FrameNet verbs that occur at least 10 times in the English gigaword corpus (3937 verbs total). The basic setup for each experiment is the same, and consists of computing a pairwise similarity matrix between all of the verbs in the study for each of the feature sets, feature weights, selected features, and distance measures under study. Each evaluation of verb distances with respect to a given verb classification scheme was restricted to only the verbs that are included in that scheme; i.e., in evaluating Levin verbs, verbs which did not occur in Levin’s classification were excluded as both target items and as empirical neighbors.

The following sections contain the results of the experiments and are organized as follows. Section 6.4.1 contains an evaluation of distributional verb similarity with

respect to the choice of distance measure. Section 6.4.2 evaluates the effect of feature weighting on distributional verb similarity. Section 6.4.3 compares the different verb schemes described early with respect to how well their respective classifications match distributionally similar verbs, and Section 6.4.4 compares feature sets.

6.4.1 Similarity Measures

The purpose of this analysis is to examine the performance of different distance measures on identifying distributionally similar verbs. Three types of distance measure – set theoretic, geometric, and information theoretic – are compared across verb schemes and feature sets. For each type of distance measure only one feature weighting was employed: the set theoretic measures were applied to binary feature vectors, the geometric distance measures were applied to vector-length normalized count vectors, and the information theoretic measures were applied to count vectors normalized to probabilities.

6.4.1.1 Set Theoretic Similarity Measures

Table 6.9 contains the precision results of the nearest neighbor classifications for three set theoretic measures of distributional similarity, using the 50,000 most frequently occurring features of each feature type. The full set of precision results using a range of feature frequencies is given in Appendix C, Figures C.1 – C.5. Table F.1 (Appendix F) contains the corresponding inverse rank scores.

Overall, for MAXP cosine returned the best results across feature types and verb classifications (MAXP = 0.43); Jaccard’s coefficient performed close to cosine (MAXP = 0.40), and overlap performed substantially lower (MAXP = 0.10). Similarly for INVR, cosine gave the overall best results (INVR= 0.81), followed by Jaccard’s

Verb Classification	Feature Type	Distance Measure			Mean
		cosine	Jaccard	overlap	
Levin	Syntactic Frame	0.40	0.39	0.14	0.31
	Lexical	0.50	0.47	0.08	0.35
	Dep. Triple	0.57	0.56	0.10	0.41
	Lex. Frame	0.49	0.48	0.11	0.36
	Mean	0.49	0.48	0.11	
VerbNet	Syntactic Frame	0.38	0.38	0.15	0.30
	Lexical	0.48	0.45	0.08	0.34
	Dep. Triple	0.55	0.53	0.10	0.39
	Lex. Frame	0.48	0.46	0.11	0.35
	Mean	0.47	0.46	0.11	
FrameNet	Syntactic Frame	0.36	0.35	0.09	0.27
	Lexical	0.48	0.44	0.07	0.33
	Dep. Triple	0.54	0.51	0.10	0.38
	Lex. Frame	0.49	0.46	0.11	0.35
	Mean	0.47	0.44	0.09	
Roget	Syntactic Frame	0.28	0.27	0.06	0.20
	Lexical	0.52	0.47	0.08	0.36
	Dep. Triple	0.61	0.53	0.10	0.41
	Lex. Frame	0.54	0.49	0.11	0.38
	Mean	0.49	0.44	0.09	
WordNet	Syntactic Frame	0.13	0.12	0.04	0.10
	Lexical	0.23	0.20	0.06	0.17
	Dep. Triple	0.28	0.25	0.11	0.22
	Lex. Frame	0.24	0.22	0.10	0.19
	Mean	0.22	0.20	0.08	

Table 6.9: Average maximum precision for set theoretic measures and the 50k most frequent features of each feature type

coefficient (INVR= 0.74) and overlap (INVR= 0.23). In terms of verb scheme, focusing just on the cosine measure, Roget, Levin, VerbNet, and FrameNet perform nearly identically (MAXP \approx 0.48), followed by WordNet at MAXP = 0.22.

For MAXP, the best performing feature type across verb scheme and distance measure is lexically specified dependency triples. Again focusing on the cosine, across verb schemes dependency triples return around 0.51 maximum precision, followed by unlabeled dependents and lexicalized frames $\text{MAXP} \approx 0.44$, and finally syntactic frames ($\text{MAXP} \approx 0.31$). These trends are mirrored in the INVR results.

6.4.1.2 Geometric Measures

Table 6.10 contains the MAXP results of the nearest neighbor classifications for three geometric measures of distributional similarity, using the 50,000 most frequently occurring features of each feature type. The context vectors were vectors of counts, normalized by vector length. The full set of MAXP results using a range of feature frequencies are given in Appendix D, figures D.1 – D.5. Table F.2 (Appendix F) contains results of the geometric measures as evaluated by INVR.

Overall, the neighbors assigned by cosine similarity (mean $\text{MAXP} = 0.35$; mean $\text{INVR} = 0.63$) resemble the given verb classifications more than the neighbors assigned by L_1 distance (mean $\text{MAXP} = 0.25$; $\text{INVR} = 0.41$) for both evaluation measures. In terms of feature type, for MAXP frame-based features did not perform as well as lexical-based features for either distance measure. For cosine, labeled and unlabeled lexical dependents performed at a very similar rate across verb schemes (mean $\text{MAXP} = 0.40$ for lexical-only versus mean $\text{MAXP} = 0.39$ for labeled dependency triples). These trends are mirrored in the INVR results.

For L_1 , the difference between lexical-only and labeled dependency triples was more pronounced: $\text{MAXP} = 0.33$ versus $\text{MAXP} = 0.23$, respectively. These trends are mirrored in the INVR results. This difference is likely due to the fact that the labeled dependency triples form a relatively sparser feature space than the unlabeled feature space and differences in how the two measures handle zeros when

Verb Classification	Feature Type	Distance Measure		Mean
		Cosine (=Euclidean)	L ₁	
Levin	Syntactic Frame	0.38	0.40	0.39
	Lexical	0.45	0.38	0.42
	Dep. Triple	0.44	0.25	0.35
	Lex. Frame	0.35	0.17	0.26
	Mean	0.41	0.30	
VerbNet	Syntactic Frame	0.36	0.38	0.37
	Lexical	0.44	0.36	0.40
	Dep. Triple	0.43	0.25	0.34
	Lex. Frame	0.34	0.18	0.26
	Mean	0.39	0.29	
FrameNet	Syntactic Frame	0.33	0.33	0.33
	Lexical	0.44	0.36	0.40
	Dep. Triple	0.45	0.24	0.35
	Lex. Frame	0.34	0.17	0.26
	Mean	0.39	0.28	
Roget	Syntactic Frame	0.25	0.28	0.27
	Lexical	0.44	0.37	0.41
	Dep. Triple	0.45	0.26	0.36
	Lex. Frame	0.34	0.10	0.22
	Mean	0.37	0.25	
WordNet	Syntactic Frame	0.11	0.13	0.12
	Lexical	0.21	0.17	0.19
	Dep. Triple	0.20	0.13	0.16
	Lex. Frame	0.15	0.05	0.10
	Mean	0.17	0.12	

Table 6.10: Average maximum precision for geometric measures using the 50k most frequent features of each feature type

comparing two vectors. Because the calculation cosine of the angle between two vectors involves multiplying corresponding features, an element with a value of zero in one vector essentially cancels out a corresponding non-zero element in the other

vector. For L_1 , a zero element is subtracted from the corresponding non-zero element, and larger differences accumulate along the many zero dimensions. In a sparse space with many zeros, differences along non-shared dimensions overwhelm similarity along shared dimensions. Since the labeled and unlabeled dependency triples represent much of the same contextual information, but the ambient space is slightly denser for the unlabeled triples, L_1 distance performs better in that space, while the cosine is relatively unaffected.

Using geometric measures of similarity, Levin verbs are picked up slightly more often than the other classes, with MAXP of 0.41 versus MAXP = 0.39 for VerbNet and FrameNet. Roget verbs are identified slightly less often at MAXP = 0.37, while WordNet synonyms are relatively unlikely to appear in the top-ranked set of distributionally similar verbs (MAXP = 0.17).

6.4.1.3 Information Theoretic Measures

Table 6.11 contains the results of the nearest neighbor classifications for two information theoretic measures of distributional similarity, using the 50,000 most frequently occurring features of each feature type. The context vectors were vectors of probabilities of counts. When L_1 distance is applied to vectors of probabilities, the result can be interpreted as the expected proportion of events that differ between the two probability distribution (Manning and Schütze, 1999: 305), and is included here for comparison. The full set of results using a range of feature frequencies are given in Appendix E, figures E.1 – E.5. Table F.3 (Appendix F) contains the corresponding inverse rank scores.

Overall, information radius and L_1 distance performed similarly across feature types and verb schemes for both MAXP and INVR (MAXP_{inforad} = 0.45; MAXP_{L1} = 0.44; INVR_{inforad} = 0.81; INVR_{L1} = 0.87).

Verb Classification	Feature Type	Distance Measure		Mean
		Information Radius	L ₁	
Levin	Syntactic Frame	0.47	0.45	0.46
	Lexical	0.55	0.55	0.55
	Dep. Triple	0.56	0.57	0.56
	Lex. Frame	0.46	0.47	0.46
	Mean	0.51	0.51	
VerbNet	Syntactic Frame	0.45	0.44	0.45
	Lexical	0.54	0.54	0.54
	Dep. Triple	0.55	0.55	0.55
	Lex. Frame	0.45	0.45	0.45
	Mean	0.50	0.50	
FrameNet	Syntactic Frame	0.43	0.41	0.42
	Lexical	0.55	0.55	0.55
	Dep. Triple	0.57	0.57	0.57
	Lex. Frame	0.46	0.46	0.46
	Mean	0.50	0.50	
Roget	Syntactic Frame	0.39	0.35	0.37
	Lexical	0.62	0.61	0.62
	Dep. Triple	0.64	0.63	0.64
	Lex. Frame	0.37	0.33	0.35
	Mean	0.51	0.48	
WordNet	Syntactic Frame	0.17	0.16	0.16
	Lexical	0.29	0.29	0.29
	Dep. Triple	0.29	0.29	0.29
	Lex. Frame	0.17	0.15	0.16
	Mean	0.23	0.22	

Table 6.11: Average maximum precision for information theoretic measures using the 50k most frequent features of each feature type

With the information theoretic measures, for MAXP a difference in classification accuracy is observed between Roget style synonyms, which are identified substantially more often than neighbors classed by any of the other verb schemes when

labeled dependencies are used ($\text{MAXP}_{\text{inforad}} = 0.64$ versus $\text{MAXP}_{\text{inforad}} = 0.57$ for next best FrameNet). Labeled and unlabeled lexical dependency relations perform better than the two syntax based feature types.

6.4.1.4 Comparison of Similarity Measures

Across the three types of similarity measure, the relative performance of verb scheme and feature type was the same. Therefore, in order to get a sense of the differences in classification performance of the various similarity measures, this section focuses on the classification of Roget synonyms using labeled dependency triples, as this combination consistently returned the highest precision and inverse rank. Table 6.12 shows the precision values for $k = 1, 5, 10, \text{MAXP}$ and the average number of neighbors (k_{MAXP}) that resulted in the maximum precision. The relative performance of each distance measure is the same for each value of k presented in the table. Table F.4 (Appendix F) shows the corresponding inverse rank scores.

	P_1	P_5	P_{10}	MAXP	k_{max}
Set Theoretic					
binary cosine	0.47	0.30	0.22	0.61	8.1
Jaccard	0.43	0.27	0.20	0.57	8.1
overlap	0.03	0.03	0.04	0.11	26.8
Geometric					
cosine (=Euclidean)	0.32	0.20	0.15	0.45	12.9
L_1	0.19	0.10	0.07	0.26	13.1
Information Theoretic					
Information Radius	0.51	0.33	0.24	0.64	3.6
L_1	0.50	0.32	0.24	0.63	7.9

Table 6.12: Measures of precision and average number of neighbors yielding maximum precision across similarity measures

Several trends are evident from the data in Tables 6.12 and F.4. First, overlap performs substantially worse than any of the other distance measures. Secondly, binary cosine, information radius, and L_1 have very similar MAXP values. From this point of view, binary cosine can be considered an information theoretic measure in the sense that it is computing the correlation between two distributions of numbers, and the fact that it is applied to binary vectors can be considered a particular feature weighting scheme. That is, the calculation of cosine does not change when it is applied to binary vectors, only the feature weighting.

Although binary cosine, information radius, and L_1 distance all achieve the same average maximum precision, they do so at different values of k . Information radius tops out with k around 3.6, while binary cosine and L_1 are between 8.1 and 7.6. Pairwise t-tests, adjusted for multiple comparisons, show that on average, information radius tops out significantly earlier than binary cosine and L_1 distance, which are not significantly different from each other. This means that although the precision is the same, L_1 distance returns just under twice as many actual neighbors as information radius does for the same precision. For $k = 1, 5, 10$, the three measures are nearly equal.

The geometric measures, i.e., cosine and L_1 applied to normalized count vectors, return lower precision values than the information theoretic measures do. However, whereas the feature weighting for set theoretic and information theoretic measures is fixed at $\{0,1\}$ and $Prob(f)$, respectively, many other feature weightings are available to which the more general geometric measures can be applied. The next section considers the effect of feature weighting on the performance of geometric similarity measures.

6.4.2 Feature Weighting

This section considers six feature weighting schemes and their interaction with lexical similarity measures. Three of the weightings (binary, normalized, and probabilities), were considered in the context of comparing distance measures. The other three, log-likelihood, correlation, and inverse feature frequency, are introduced into this study here. The three distance measures considered are cosine, Euclidean distance, and L_1 distance.

Within verb schemes and across feature sets, the relative performance of the different feature weighting schemes remained constant. Overall, labeled dependency triples performed the best, followed by unlabeled triples, lexicalized frames, and syntactic frames.

Tables 6.13 and F.6 show precision results for verb classifications using labeled dependency triples. Across verb schemes, the trends between feature weight and distance measure hold fairly consistently. Overall, the best performing combination of feature weight and distance measure was achieved by applying the cosine to vectors weighted by inverse feature frequency: 58% of the 1-nearest neighbors computed with this combination are classified as synonyms by Roget's thesaurus, with a maximum precision of 71%. This combination performed the best for the other verb schemes as well, ranging from $MAXP = 63\%$ for Levin to $MAXP = 34\%$ for WordNet.

In terms of the interactions between feature weight and distance measure, the following tendencies are observed. For Euclidean distance, the following ranking of feature weights in terms of precision approximately holds:

normalized > probability > iff > binary, log-likelihood, correlation

Feature Weighting		Distance Measure								
		Cosine			Euclidean			L ₁		
		P ₁	MAXP	k_{max}	P ₁	MAXP	k_{max}	P ₁	MAXP	k_{max}
Levin	binary	0.43	0.57	8.8	0.19	0.29	7.7	0.19	0.29	7.7
	probability	0.31	0.44	12	0.28	0.40	14	0.43	0.57	9.5
	normalized	0.31	0.44	12	0.31	0.44	12	0.17	0.25	9.9
	log-likelihood	0.44	0.58	8.9	0.19	0.29	7.8	0.19	0.29	7.8
	correlation	0.38	0.55	10	0.20	0.27	6.4	0.10	0.15	5.6
	inv feat freq	0.49	0.63	7.7	0.27	0.38	6.1	0.19	0.28	6.4
VerbNet	binary	0.43	0.55	10	0.19	0.29	9.5	0.19	0.29	9.5
	probability	0.30	0.43	14	0.27	0.39	15	0.41	0.55	11
	normalized	0.30	0.43	14	0.30	0.43	14	0.17	0.25	14
	log-likelihood	0.41	0.56	10	0.19	0.30	9.1	0.19	0.29	9.1
	correlation	0.37	0.54	11	0.22	0.29	7.1	0.12	0.16	6.3
	inv feat freq	0.47	0.62	8.8	0.27	0.38	7.1	0.19	0.29	8.2
FrameNet	binary	0.41	0.54	7.7	0.18	0.28	4.3	0.18	0.26	4.3
	probability	0.33	0.45	10	0.29	0.40	11	0.45	0.58	8.7
	normalized	0.33	0.45	10	0.33	0.45	10	0.18	0.24	9.2
	log-likelihood	0.42	0.55	7.6	0.17	0.26	4.3	0.17	0.26	4.3
	correlation	0.42	0.57	7.6	0.16	0.22	2.3	0.02	0.06	2.6
	inv feat freq	0.49	0.61	6.8	0.27	0.36	3.5	0.18	0.26	3.7
Roget	binary	0.47	0.61	8.1	0.19	0.25	4.9	0.19	0.25	4.9
	probability	0.32	0.45	12.9	0.29	0.42	14.2	0.50	0.63	7.9
	normalized	0.32	0.45	12.9	0.32	0.45	12.9	0.19	0.26	13.1
	log-likelihood	0.48	0.62	7.7	0.19	0.26	3.7	0.19	0.26	3.7
	correlation	0.43	0.60	8.6	0.19	0.24	5.9	0.03	0.04	2.8
	inv feat freq	0.58	0.71	6.1	0.28	0.35	3.5	0.19	0.24	3.0
WordNet	binary	0.19	0.28	10	0.08	0.12	5.4	0.08	0.12	5.4
	probability	0.13	0.20	12	0.12	0.19	13	0.21	0.29	9.6
	normalized	0.13	0.20	12	0.13	0.20	12	0.09	0.13	9.9
	log-likelihood	0.19	0.29	10	0.07	0.12	4.8	0.08	0.12	4.8
	correlation	0.18	0.28	11	0.08	0.11	4.8	0.01	0.03	3.2
	inv feat freq	0.24	0.34	9.3	0.11	0.16	4.7	0.08	0.11	4.7

Table 6.13: Nearest neighbor average maximum precision for feature weighting, using the 50k most frequent features of type labeled dependency triple

For Euclidean distance, the tendency for normalized vectors to produce better neighbors can be explained by the fact that Euclidean distance is quadratic in the unshared terms; normalized vectors exhibit the smallest absolute feature values of the six weights considered, so these differences will be minimized.

Interestingly, inverse feature frequency performs better than the other three weights although the average feature weight is much larger than for binary, log-likelihood, or correlation. This suggests that inverse feature frequency does a better job of capturing the relevance of the association between a verb and a feature than log-likelihood or correlation.

For L_1 distance, the following ranking of feature weights holds:

probability \gg binary, log-likelihood, iff $>$ normalized \gg correlation

Probability vectors outperform any of the other weighting methods by a substantial margin (nearly 2:1), lending credence to its interpretation as a measure of the difference between probability distributions. The differences between binary, log-likelihood, and inverse feature frequency were slight and varied unpredictably across verb schemes. Once again, weighting by correlation performed poorly.

For cosine, the following ranking of feature weights was found:

iff $>$ log-likelihood, binary, correlation \gg normalized, probability

In this combination, inverse feature frequency returned appreciably better results across the verb schemes. As with Euclidean distance, binary feature weights perform just as well as log-likelihood and correlation, which in turn outperform normal vector scalings.

These results indicate that two ingredients are needed for successful nearest neighbor identification of verbs. One is an extrinsic weighting method which models the relative strength of the association between a verb and a feature as a function of both co-occurrence frequency and its proclivity to occur with other verbs. The second consideration is an appropriate scaling of the magnitude of the feature weights. In

the case of cosine, the distance measure itself provides the scaling; in the case of the other distance measures, a scaling such as normalization improves the selection of lexical neighbors.

Given that log-likelihood and inverse feature frequency both provide functions for measuring this associational strength, it appears that the inverse feature frequency measure is a better choice than log-likelihood for this task, which attempts to model co-occurrence strength in terms of a prior asymptotic χ^2 distribution. This distribution may not be as appropriate for describing the distribution of word co-occurrences as a function which models co-occurrence distributions directly. A possible explanation for the poor performance of correlation is that in this feature space, all correlations are very small. It is likely that the average correlation between a verb and a feature are too small to be very informative. Unlike Rohde et al. (submitted) who combat this problem by taking the square root of the correlations as a post processing step, we did not make any further alternations to the correlation score.

6.4.3 Verb Scheme

One picture that consistently emerges across feature sets, distance measures, and weighting schemes is that empirically determined nearest neighbors match Roget's synonym assignments substantially more closely than they match any of the other schemes. Furthermore, WordNet synonym assignments show the lowest correspondence to empirical nearest neighbors than any of the other schemes.

However, in addition to synonymy, WordNet defines hyponymy relations between verbs, which often correspond to Roget synonyms. For example, Roget synonyms of *argue* such as *quibble*, *quarrel*, *dispute*, and *altercate* are classified as hyponyms by WordNet, and as such were not counted as matches. In these cases, WordNet provides a further refinement of verb relations that may match more closely

to distributionally similar verb assignments than its stricter definitions of synonymy. Exploring these more finely grained lexical distinctions is left for future research.

Levin, VerbNet, and FrameNet place verbs into classes based on both similarity in meaning and similarity along more schematic representations of syntactic or semantic behavior such as alternations (Levin, VerbNet) or participation in semantic frames (FrameNet). In an effort to tease apart the independent criteria of semantic and syntactic similarity, we can look at the proportion of items in a class that are independently classified as synonyms by a thesaurus, and compare this to the proportion of empirical neighbors that are either synonyms or not by the same standard.

The left column in Table 6.14 shows the average number of synonyms that are found within a verb class for Levin, VerbNet, and FrameNet.

Verb Scheme	%Synonyms in Class	%Synonyms in k -nn		
		$k = 1$	5	10
Levin	23	71	57	48
VerbNet	23	73	58	50
FrameNet	38	76	65	56

Table 6.14: Average number of Roget synonyms per verb class

This number was calculated from the number of same class items that are listed as synonyms in Roget’s online thesaurus; i.e., on average 23% of a Levin verb’s neighbors are recognized as synonyms by the thesaurus. The right column shows the average percentage of empirically determined 1-nearest neighbors that are also synonyms. For example, of the empirically determined 1-nearest neighbors that are put into the same class by Levin, 71% turn out to be synonyms by Roget’s thesaurus; this figure is slightly higher for VerbNet and FrameNet. This means that when highly similar Levin-style neighbors are identified empirically, they are over three times more

likely to be synonyms than would be expected based on the prior class probability of being synonyms; similarly for VerbNet. The fact that a greater percentage of FrameNet verbs are synonyms is to be expected given that FrameNet emphasizes semantic relatedness and does not explicitly include participation in syntactic alternations as a criterion for partitioning verbs into classes (Baker, Fillmore, and Lowe, 1998; Baker and Ruppenhofer, 2002). Also apparent in Table 6.14 is the fact that as k increases, the number of synonyms decreases for all three verb schemes. Again this points to the interpretation that highly distributionally similar items are likely to be synonyms, and that the additional grouping criteria used Levin, VerbNet, and FrameNet are not represented as well using the feature sets examined here.

The upshot of this analysis is that regardless of the four feature sets applied here, distributionally similar verbs assignments correspond to thesaurus style synonyms more than they correspond to the groupings in Levin, VerbNet, and FrameNet. As noted above, distributionally similar verbs do not correspond well to WordNet's more restrictive definitions of synonymy. This is most likely due to the fact that WordNet is conservative in assigning synonymy to verbs, and makes subtler lexical distinctions than Roget's thesaurus does.

6.4.4 Feature Set

Tables 6.15 and F.5 show the performance of the four feature sets across verb classes for the best performing feature weight/distance measure combination of inverse feature frequency and cosine. In this setting, the following ranking of feature sets in terms of precision of empirically determined nearest neighbors holds:

labeled dependencies > unlabeled dependencies > lex. frames > syntactic frames

Verb Classification	Feature Type	P ₁	P ₅	P ₁₀	MAXP	k_{max}
Levin	Syntactic Frame	0.29	0.24	0.21	0.45	11
	Lexical	0.42	0.30	0.25	0.56	8.9
	Dep. Triple	0.49	0.38	0.32	0.63	7.7
	Lex. Frame	0.37	0.29	0.25	0.52	10
VerbNet	Syntactic Frame	0.27	0.22	0.20	0.43	13
	Lexical	0.40	0.29	0.24	0.54	10
	Dep. Triple	0.47	0.36	0.30	0.62	8.8
	Lex. Frame	0.36	0.27	0.24	0.51	11
FrameNet	Syntactic Frame	0.29	0.21	0.18	0.41	7.3
	Lexical	0.45	0.30	0.24	0.57	7.2
	Dep. Triple	0.49	0.35	0.28	0.61	6.8
	Lex. Frame	0.40	0.28	0.23	0.53	8.3
Roget	Syntactic Frame	0.21	0.14	0.12	0.34	14.1
	Lexical	0.50	0.31	0.23	0.64	7.5
	Dep. Triple	0.58	0.38	0.29	0.71	6.1
	Lex. Frame	0.43	0.29	0.22	0.58	8.5
WordNet	Syntactic Frame	0.09	0.06	0.05	0.16	12
	Lexical	0.20	0.12	0.08	0.30	10
	Dep. Triple	0.24	0.14	0.10	0.34	9.3
	Lex. Frame	0.17	0.11	0.08	0.26	11

Table 6.15: Nearest neighbor precision with cosine and inverse feature frequency

For other settings of feature weight and distance measure, there was often no appreciable difference between labeled and unlabeled dependencies; the other relations between feature sets hold consistently.

The main conclusion to draw from these patterns is that lexical and syntactic information jointly specify distributional cues to lexical similarity. It is useful to differentiate between whether a verb’s argument appeared as a subject or object versus simply recording the fact that it appeared as an argument. It is likely that the lexicalized frames are overly specific and result in very large, sparse feature sets.

Other researchers have noted this problem of data sparsity, (e.g., Schulte im Walde, 2000), and have explored the additional use of selectional preference features by augmenting each syntactic slot with the concept to which its head noun belongs in an ontology (e.g. WordNet). For example, replacing a frame like *eat*, $\langle \textit{Joe}, \textit{Subj} \rangle$, $\langle \textit{corn}, \textit{Obj} \rangle$ with *eat*, $\langle \textit{Person}, \textit{Subj} \rangle$, $\langle \textit{Plant}, \textit{Obj} \rangle$ provides a level of generalization that overcomes some of the data sparsity problem. Although the problem of data sparsity can be mitigated through the use of such techniques, these features have generally not been shown to improve classification performance (Schulte im Walde, 2000; Joanis, 2002).

It is not surprising that syntactic frames perform worse than the other feature sets. A lexically unspecified syntactic frame conveys relatively little information, and any given frame may be shared by many verbs regardless of their semantic class or synonym set. The fact that syntactic information alone can achieve around 40% precision on a semantic classification task with a negligible baseline provides support for semantic theories that relate syntactic structure to verb meaning. It is interesting that syntactic frames do a better job of identifying Roget synonyms, which are not explicitly organized around syntactic behavior, than of identifying Levin neighbors, which do explicitly incorporate syntactic behavior. However, Levin’s classification involves specific syntactic alternations that preserve meaning rather than general syntactic frames that are not tied to a given semantic interpretation. The failure of syntactic frames to identify Levin neighbors more precisely is probably due to a mismatch between the information they represent and the criteria used in Levin’s original classification.

6.5 Relation Between Experiments and Existing Resources

One application of the techniques developed here would be to assist in extending existing verb schemes such as VerbNet, FrameNet, or Roget’s thesaurus by suggesting neighbors of unclassified verbs. In order to estimate the coverage of the five verb schemes studied here, we compared the number of verbs in each scheme that occur at least 10 times in the English gigaword corpus to the number of verbs in the union of the five verbs schemes. There are 7206 verbs in the union of Levin, VerbNet, FrameNet, Roget, and WordNet that occur at least 10 times in the English gigaword corpus³. Table 6.16 contains these comparisons. For each verb scheme, the average frequency of verbs included in that scheme is indicated along with the average frequency of verbs not included in that scheme.

Verb Scheme		Contained	Missing
Levin		2886	4320
	Avg. Freq	47231	23479
VerbNet		3426	3780
	Avg. Freq	44504	22558
FrameNet		2110	5096
	Avg. Freq	94374	7577
Roget		5660	1546
	Avg. Freq	61915	1151
WordNet		7110	96
	Avg. Freq	33433	351

Table 6.16: Coverage of each verb scheme with respect to the union of all of the verb schemes and the frequency of included versus excluded verbs

³The reason that there are more Roget and WordNet verbs here than in the experiments is that the experiments used the union of Levin, VerbNet, and FrameNet and extracted Roget and WordNet synonyms from those; here we are looking at the union of all five verb schemes.

For all of the verb schemes, the average token frequency of verbs included in the scheme is greater than the average frequency of excluded verbs. FrameNet covers the smallest number of verbs, but the verbs that it does contain occur on average more frequently than verbs in the other verb schemes. Levin and VerbNet show the least disparity between the average frequency of included versus excluded verbs (about 2:1), while WordNet and Roget show the greatest difference (about 95:1 and 81:1, respectively). In other words, there are many verbs that occur with relatively high frequency in the English gigaword corpus but which Levin and VerbNet do not cover.

We can take the precision results obtained on known verbs as an indication of the expected performance when using distributional similarity as a tool for assigning unknown verbs to lexical semantic classes. In this setting, we would assign an unknown verb to the class(es) of the distributionally most similar verbs in each verb scheme. Table 6.17 contains the expected proportion of correct assignments of unknown verbs to lexical semantic classes for each of Levin, VerbNet, and FrameNet. These proportions are the 1-nearest neighbor precision results using cosine similarity applied to labeled dependency triples weighted by inverse feature frequency over the 50,000 most frequent features.

Verb Scheme	Num. Classes	Baseline Accuracy			Expected Acc.
		Minimum	Average	Maximum	
Levin	191	(1) 0.01	(1.39) 0.01	(10) 0.05	0.49
VerbNet	237	(1) 0.00	(1.37) 0.01	(10) 0.04	0.47
FrameNet	321	(1) 0.00	(1.35) 0.00	(8) 0.02	0.49

Table 6.17: Expected classification accuracy. The numbers in parentheses indicate raw counts used to compute the baselines

For each verb scheme, we also indicate baseline classification accuracy. Because we have conflated verb senses, the 1-nearest neighbor precision results indicate that a verb was correctly classified if it belongs to any one of the classes that its distributionally most similar neighbor belongs to. Therefore, for each verb scheme we show three baselines:

- The most restrictive case, when the distributionally most similar known verb belongs to exactly one class, defined as 1 divided by the total number of classes in the verb scheme.
- The average case, defined as the average number of classes to which a verb belongs divided by the total number of classes.
- The least restrictive case, defined as the maximum number of classes any verb in the verb scheme belongs to divided by the total number of classes.

6.6 Conclusion

This chapter presented the results of a large-scale comparison of a variety of parameters which determine distributional lexical similarity over five lexical semantic classifications of English verbs. The main findings are summarized below.

- Of the distance measures considered here, cosine (viz. correlation coefficient) yielded the best results.
- Of the feature sets considered here, labeled dependency triples yielded the best results.
- Of the feature weightings considered here, inverse feature frequency yielded the best results.
- Using the parameters studied here, distributionally similar verb assignments correspond more closely to Roget-style synonyms than to Levin, VerbNet, or FrameNet classes or WordNet synonyms.

- The parameters explored here can be used to extend the coverage of Levin, VerbNet, and FrameNet to other verbs that occur in a large text corpus with an expected accuracy of around 49% (over a baseline accuracy of about 5%).

Based on these findings, we conclude more generally that:

- Syntactically informed lexical co-occurrence features do a better job of identifying synonyms than of identifying neighbors based on the other lexical semantic criteria that Levin, VerbNet, and FrameNet rely on (e.g., shared components of meaning such as MOTION or COVERING; participation in semantic frames).
- Extrinsic feature weightings, which quantify the association between a feature and a target verb with respect to that feature’s overall distribution among verbs, do a better job of identifying neighbors than feature weightings which do not account for overall feature distribution.
- In addition to extrinsic feature weightings, scaling a weighted context vector (i.e., via cosine or vector length normalization) improves neighbor identification.

CHAPTER 7

CONCLUSION

This dissertation addressed the problem of learning accurate and scalable lexical classifiers in the absence of large amounts of hand-labeled training data. It considered two distinct lexical acquisition tasks, both of which rely on an appropriate definition of distributional lexical similarity:

- Automatic transliteration and identification of English loanwords in Korean. For this problem, lexical similarity was defined over phonological co-occurrence features.
- Lexical semantic classification of English verbs on the basis of automatically derived co-occurrence features. For this problem, similarity was defined in terms of grammatical relations.

7.1 Transliteration of English Loanwords in Korean

The first task focused on ways to mitigate the effort of obtaining large amounts of labeled training data for transliterating and identifying English loanwords in other languages, using Korean as a case study. The key ideas that emerged from the transliteration task are:

- Consonant transliteration is highly regular and can be expressed reliably using a small number of phonological adaptation rules.

- Vowel transliteration is irregular and is heavily influenced by the orthographic forms of source words.
- These two observations can be used to constrain the predictions made by a statistical transliteration model, resulting in a model that is robust to small amounts of training data and produces a small number of transliterations per input item.

Two transliteration models were devised – a phonological rule-based model, and a statistical model that combined orthographic and phonological information. These models were applied to a set of 10,000 attested English loanwords in Korean. The rule-based model obtained 1-best transliteration accuracy of 49.2%, compared to 73.4% for the statistical model. When vowels are excluded from the output transliterations, the performance of the rule-based model and the statistical model is much closer: 89.9% for the rule-based model versus 90.8% for the statistical model. These figures underscore the variability associated with vowel transliteration.

7.2 Identification of English Loanwords in Korean

For the identification task, the basic idea involved using a rule-based system to generate large amounts of data that serve as training examples for a secondary lexical classifier. Although the precision of the rule-based output was low, on a sufficient scale it represented the lexical patterns of primary statistical significance with enough reliability to train a classifier that was robust to the deficiencies of the original rule-based output. The primary contributions of this study of loanword identification include:

- A demonstration of the suitability of a sparse logistic regression classifier to the task of automatic loanword identification.

- A highly efficient solution to the problem of obtaining labeled training data for etymological classification.
- A demonstration of the fact that automatically generated pseudo-data can be used to train a classifier that distinguishes actual English and Korean words as accurately as one trained entirely on hand-labeled data.

Three experiments were conducted which systematically varied the quantity and quality of labeled training data. The first experiment, conducted entirely on hand-labeled training data, obtained classification accuracy of 96.2%. The second experiment used the rule-based transliteration model from Chapter 3 to produce large amounts of pseudo-English loanwords that were used in conjunction with actual Korean words to train a classifier capable of identifying actual English loanwords with 95.8% accuracy. The third experiment used pseudo-English loanwords and unlabeled items that served as examples of Korean words to train a classifier that identified actual English loanwords with 92.4% accuracy.

7.3 Distributional Verb Similarity

The second lexical acquisition task considered in this dissertation was the assignment of English verbs to lexical semantic classes on the basis of their distributional context in a large text corpus. The approach to this task used the output of a statistical parser to automatically generate a feature set that was used to assign English verbs to lexical semantic classes. This study produced results on a substantially larger scale than any previously reported and yielded new insights into the properties of verbs that are responsible for their lexical categorization. A series of experiments were conducted which examined the interactions between a number of parameters that influence empirical determinations of distributional lexical similarity. The parameters examined were:

- Similarity measure. Three classes of similarity measure were considered – set theoretic, geometric, and information theoretic.
- Feature type. Four feature types based on grammatical dependencies were examined – syntactic frames, labeled and unlabeled dependency relations, and lexicalized syntactic frames.
- Feature weighting. Intrinsic weightings such as vector length normalization were compared to extrinsic weighting schemes such as correlation.
- Feature selection. Feature selection was limited to cutoff by frequency.

These parameters were further evaluated with respect to five verb classification schemes – Levin, VerbNet, FrameNet, Roget’s Thesaurus, and WordNet. The main picture that emerged from this analysis is that a combination of cosine similarity measure with labeled dependency triples and inverse feature frequency consistently yielded the best results in terms of how closely empirical verb similarities matched the labels of the five verb schemes. Performance asymptotes at around 50,000 of the most frequent features of each type.

Simultaneously considering multiple verb classification schemes allowed for a comparison of the criteria used by each scheme for grouping verbs. One of the main findings along these lines is that using the feature sets considered here, verbs within a given classification scheme that are related by synonymy are identified more reliably than verbs related by criteria such as diathesis alternations or participation in semantic frames. This approach also allowed for an examination of the relation between each verb scheme and empirically determined verb similarities. Here we saw that Roget synonyms were identified more reliably than Levin, VerbNet, and FrameNet verbs. Extrapolating the precision of empirical neighbor assignments for each of the five verb schemes to unknown verbs allowed an estimate of the expected accuracy that would be obtained for automatically extending the coverage of each

scheme to new verbs. Using the best performing combination of parameters mentioned in the preceding paragraph, Roget synonyms were correctly identified 58% of the time; Levin, VerbNet, and FrameNet verbs obtained approximately 49% accuracy, and WordNet synonyms were correctly identified 24% of the time. Together, these findings indicate that we should pay closer attention to the relation between the various criteria used in each verb scheme and which of those criteria are primarily reflected in the feature sets commonly used in automatic verb classification.