Proceedings of the 8th Conference on Natural Language Learning (CoNLL 2004), pp. 81-88.

Calculating Semantic Distance between Word Sense Probability Distributions

Vivian Tsang and **Suzanne Stevenson**
Department of Computer Science
University of Toronto
{vyctsang,suzanne}@cs.toronto.edu

Abstract

Semantic similarity measures have focused on individual word senses. However, in many applications, it may be informative to compare the overall sense distributions for two different contexts. We propose a new method for comparing two probability distributions over WordNet, which captures in a single measure the aggregate semantic distance of the component nodes, weighted by their probability. Previous such measures compute only the distributional distance, and do not take into account the semantic similarity between WordNet senses across the distributions. To incorporate semantic similarity, we calculate the (dis)similarity between two probability distributions as a weighted distance "travelled" from one to the other through the WordNet hierarchy. We evaluate the measure by applying it to the acquisition of verb argument alternation knowledge, and find that overall it outperforms existing distance measures.

1 Introduction

Much attention has recently been given to calculating the similarity of word senses, in support of various natural language learning and processing tasks. Such techniques apply within a semantic hierarchy, or ontology, such as WordNet. Typical methods comprise an edge-distance measurement over the two sense nodes being compared within the hierarchy (Leacock and Chodorow, 1998; Rada et al., 1989; Wu and Palmer, 1994). Other approaches instead assume a probability distribution over the entire sense hierarchy; similarity is captured between individual senses by a formula over the information content (negative log probabilities) of relevant nodes (e.g., Jiang and Conrath, 1997; Lin, 1998).

The latter case assumes that there is a single WordNet probability distribution of interest, which is estimated by populating the hierarchy with word frequencies from an appropriate corpus (e.g, Jiang and Conrath, 1997). But some problems more naturally give rise to multiple conditional probability distributions estimated from counts that are conditioned on various contexts, such as different corpora or differing word usage within a single corpus. Each of these contexts would yield a distinct WordNet probability distribution, or what we will call a *sense profile*. In this situation, instead of asking how similar are two senses within a single sense profile, one may want to know how similar are two sense profiles—i.e., two (conditional) distributions across the entire set of nodes.

This question could be important to a number of applications. When two sets of WordNet frequency counts are conditioned on differing contexts, a comparison of the resulting probability distributions can give us a measure of the degree of semantic similarity of the conditioning contexts themselves. These conditioning contexts may be any relevant ones defined by the application, such as differing sets of documents (to support asking how similar various document collections are), or differing usages of words within or across document collections (to support asking questions about the similarity of various words in their usages). For example, we foresee comparing the sense profile of the objects of some verb in a particular set of documents to that of its objects in another set of documents, as an indicator of differing senses of the verb across the collections.

We have developed a general method for answering such questions, formulating a measure of the distance between probability distributions defined over an ontological hierarchy, which we call "sense profile distance," or SPD. SPD is calculated as a tree distance that aggregates the individual semantic distances between nodes in the hierarchy, weighted by their probability in the two sense profiles. SPD can be calculated between two probability distributions over any hierarchy that supports a user-supplied semantic distance function. (In fact, the two sense profiles need not strictly be probability distributions—the measure is well-defined as long as the sum of the values of the two sense profiles is equal.)

We demonstrate our method on a problem that arises in lexical acquisition, of determining whether two different argument positions across syntactic usages of a verb are assigned the same semantic role. For example, even though *the truck* shows up in two different syntactic positions, it is the Destination of the action in both of the sentences *I loaded the truck with hay* and *I loaded hay onto the truck*. Automatic detection of such *argument alternations* is important to acquisition of verb lexical semantics (Dang et al., 2000; Dorr and Jones, 2000; Merlo and Stevenson, 2001; Schulte im Walde and Brew, 2002; Tsang et al., 2002), and moreover, may play a role in automatic processing of language for applied tasks, such as question-answering (Katz et al., 2001), information extraction (Riloff and Schmelzenbach, 1998), detection of text relations (Teufel, 1999), and determination of verb-particle constructions (Bannard, 2002). We focus on this problem to illustrate how our general method works, and how it aids in a particular natural language learning task.

As in McCarthy (2000), we cast argument alternation detection as a comparison of sense profiles across two different argument positions of a verb. Our method differs, however, in two important respects. First, our measure can be used on any probability distribution, while McCarthy's approach applies only to a very narrow form of sense profile known as a *tree cut*.[1] The dependence on tree cuts greatly limits the applicability of her measure in both this and other problems, since only a particular method can be used for populating the WordNet hierarchy with probability estimates. Second, our approach provides a much finer-grained measure of the distance between the two profiles. McCarthy's method rewards probability mass that occurs in the same subtree across two distributions, but does not take into account the distance between the classes that carry the probability mass. Our new SPD method integrates a comparison of probability distributions over WordNet with a node distance measure. SPD thus enables us to calculate a more detailed comparison over the probability patterns of WordNet classes. As our results indicate, this has advantages for argument alternation detection, but more importantly, we think it is crucial for generalizing the method to a wider range of problems.

[1] A tree cut for tree T is a set of nodes C in T such that every leaf node of T has exactly one member of C on a path between it and the root (Li and Abe, 1998). As a sense profile, a tree cut will have a non-zero probability associated with every node in C, and a zero probability for all other nodes in T. Figure 1 in Section 3 has examples of two tree cuts.

In the next section, we present background work on comparing sense profiles, and on using them to detect alternations. In Section 3, we describe our new SPD measure, and show how it captures both the general differences between WordNet probability distributions, as well as the fine-grained semantic distances between the nodes that comprise them. Section 4 presents our corpus methodology and experimental set-up. In Section 5, we evaluate SPD against other distance measures, and evaluate the different effects of our experimental factors, such as the precise distance functions we use in SPD and the division of our verbs into frequency bands. By classifying the frequency bands separately, our method achieves a combined accuracy of 70% overall on unseen test verbs, in a task with a baseline of 50%. We summarize our findings in Section 6 and point to directions in our on-going work.

2 Related Work

Our method draws on, and extends, earlier work in verb lexical semantics (Resnik, 1993; McCarthy, 2000). For example, Resnik (1993) uses relative entropy (KL divergence) to compare the sense profile over the objects of a verb to the profile over the objects of all verbs, to determine how much that verb differs from "average" in its strength of selection for an object. A drawback of this approach for generalizing to other sense profile comparisons is the assumption in relative entropy of an asymmetry between the two probability distributions.

Similarly, McCarthy (2000) uses skew divergence (a variant of KL divergence proposed by Lee, 1999) to compare the sense profile of one argument of a verb (e.g., the subject position of the intransitive) to another argument of the same verb (e.g., the object position of the transitive), to determine if the verb participates in an argument alternation involving the two positions. For example, the causative alternation in sentences (1) and (2) illustrates how the subject of the intransitive is the same underlying semantic argument (i.e., the Theme—the argument undergoing the action) as the object of the transitive:

(1) **The snow** melted.

(2) The sun melted **the snow**.

Because we demonstrate our new SPD measure on the same problem as McCarthy (2000), we provide more detail of her method here, for comparison. The first step is to create the sense profiles for the relevant verb/slot pairs (e.g., the intransitive subject of *melt*, and the transitive object of *melt*, if determining whether *melt* undergoes the causative alternation, as illustrated above). The head nouns are extracted from the syntactic slots to be compared for each verb, yielding the frequency of each noun for a verb/slot pair, which is then used to populate the WordNet hierarchy. McCarthy determines the sense profile of a verb/slot pair using a minimum description length tree cut model over the frequency-populated hierarchy (Li and Abe, 1998). The two profiles for a verb are "aligned" to permit comparison using skew divergence as a probability distance measure Lee (1999). (This step is explained in more detail in the next section, with an example.) The value of the distance measure is compared to a threshold, which determines classification of a verb as causative (the two profiles are similar) or non-causative (the two profiles are dissimilar), leading to best performance of 73% accuracy, on a set of hand-selected verbs.

In McCarthy (2000), an error analysis reveals that the best method has more false positives than false negatives—some slots are considered overly similar because the sense profiles are compared at a coarse-grained level, losing fine semantic distinctions. Moreover, as mentioned above, the method can only apply to tree-cuts, which restricts its use to a very narrow range of sense profile comparisons.

In the next section, we propose an alternative method of comparing sense profiles, which addresses each of the shortcomings of these previous measures.

3 Sense Profile Distance

Our measure of sense profile distance (SPD) is designed to meet three criteria. First, it should capture fine-grained semantic similarity between profiles. Second, it should allow easy comparison between any sense profiles as probability scores spread throughout a hierarchical ontology (such as WordNet), not just between a particular format such as tree cuts. Third, it should be a symmetric measure, making it more appropriate for a wide range of applications of sense profile comparison. To achieve these goals, we measure the distance as a tree distance between the two profiles within the hierarchy, weighted by the probability scores.

(Note that we formulate a *distance* measure, while referring to a component of semantic *similarity*. We assume throughout the paper that WordNet node distance is the inverse of WordNet similarity, and indeed the similarity measures we use are directly invertible.)

We illustrate with an example the differences between our measure and both McCarthy's (2000) method and general vector distance measures. Consider the two sense profiles in Figure 1, with $profile_1$ in square boxes, and $profile_2$ in ovals.[2] To calculate the vector distance between $profile_1$ and $profile_2$, we need two vectors of equal dimension. In McCarthy (2000), the distributions are propagated to the lowest common subsumers (i.e., the nodes labelled B, C, and D). The vectors representing the two profiles become:

$$profile_1 = [0.5, 0.2, 0.3]$$
$$profile_2 = [0.5, 0.2, 0.3]$$

Alternately, one can also increase the dimension of each profile to include all nodes in the hierarchy (or just the union of the profile nodes). The two profiles become:

$$profile_1 = [0, 0.5, 0.2, 0, 0, 0, 0, 0.2, 0.1]$$
$$profile_2 = [0, 0, 0, 0.3, 0.3, 0.2, 0.2, 0, 0]$$

In the first method (that of McCarthy, 2000), the two profiles become identical. By generalizing the profiles to the lowest common subsumers, we lose information about the semantic specificity of the profile nodes and can no longer distinguish the semantic distance between the nodes across profiles. In the second method, the information about the hierarchical structure (of WordNet) is lost by treating each profile as a vector of nodes. Hence, vector distance measures fail to capture any semantic similarity across different nodes (e.g., the value of node B in $profile_1$ is not directly compared to the value of its child nodes E and F in $profile_2$).

To remedy such shortcomings, our goal is to design a new distance measure that (i) compares the *distributional* differences between two profiles (somewhat similar to existing vector distances), and also (ii) captures the *semantic* distance between profiles. Intuitively, we can think of the profile distance as how far one profile (source) needs to "travel" to reach the other profile (destination). Formally, we define SPD as:

$$SPD(profile_{src}, profile_{dest}) = \sum_{\substack{s \in profile_{src} \\ d \in profile_{dest}}} amount(s, d) * distance(s, d) \quad (1)$$

where $amount(s, d)$ is the portion of the profile score at node s in $profile_{src}$ that travels to node d in $profile_{dest}$, and $distance(s, d)$ is the semantic distance between node s and node d in the hierarchy. For now, it can be assumed that $amount(s, d)$ is $score(s)$, the entire probability score at node s. Note that we design the distance to be symmetric, so that the distance remains the same regardless of which profile is source and which is destination. (We present our distance measures below.)

In the current example, we can propagate $profile_2$ (source) to $profile_1$ (destination) by moving its probabilities in this manner:

1. probabilities at nodes E and F move to node B
2. probability at node G moves to node C
3. probability at node D moves to nodes H and I

The first two steps are straightforward—whenever there is one destination node in a propagation path, we simply multiply the amount moved by the distance of the path ($distance(s, d)$). For example, step 1

[2] Note that these are both tree cuts, so that we can compare McCarthy's method, but keep in mind that our method—as well as traditional vector distances—will apply to any probability distribution over a tree.

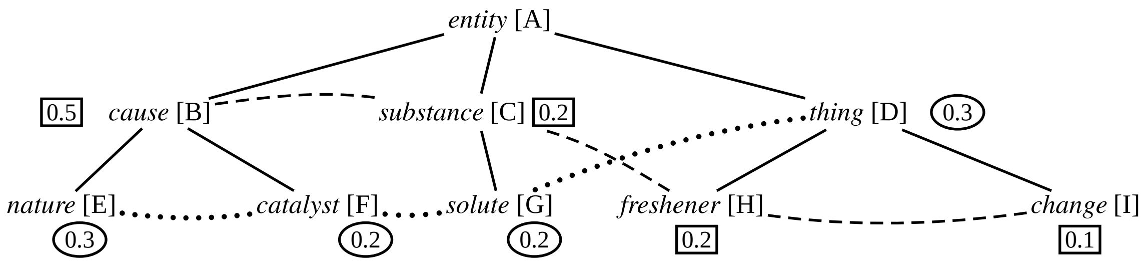

Figure 1: An example of two sense profiles; *profile₁* in square boxes, and *profile₂* in ovals. Probability values of zero are not shown. The italicized labels on nodes are WordNet classes; the single letter labels are for reference in the text.

yields a contribution to $SPD(profile_{src}, profile_{dest})$ of $score(E)\,dist(E,B) + score(F)\,dist(F,B)$.

However, the last step, step 3, has multiple destination nodes (H and I), and the probability of the source node, D, must be appropriately apportioned between them. We take this into account in the $amount$ function, by including a weight component:

$$amount(s,d) = weight(d) * portion(s) \tag{2}$$

where $weight(d)$ is the weight of the destination node d and $portion(s)$ is the portion of $score(s)$ that we are moving. (For this example, we continue to assume that the full amount of $score(s)$ is moved; we discuss $portion(s)$ further below.) The weight of each destination node d is calculated as the proportion of its score in the sum of the scores of its siblings. Thus, in step 1 above, $weight(B)$ and $weight(C)$ are both 1, and the full amount of E, F, and G are moved up. In the last step, however, the sibling nodes H and I have to split the input from node D: node H has weight $score(H)/(score(H) + score(I)) = 0.2/(0.2+0.1) = 2/3$, and node I analogously has weight $1/3$.[3]

Hence, the SPD propagating from $profile_2$ to $profile_1$ can be calculated as:

$$\begin{aligned} SPD(profile_2, profile_1) &= score(E)dist(E,B) \\ &+ score(F)dist(F,B) + score(G)dist(G,C) \\ &+ \frac{2}{3}score(D)dist(D,H) \\ &+ \frac{1}{3}score(D)dist(D,I) \end{aligned}$$

For simplicity, we designed this example such that the two profiles are very similar. As a result, we end up propagating the *entire* source profile by propagating the full score of each of its nodes. In practice, for most profile comparisons, we only move the portion of the score at each node necessary to make one profile resemble the other. Hence, $portion(s)$ in the formula for $amount(s,d)$ in equation 2 captures the difference between probabilities at node s across the source and destination profiles.

So far we have discussed very little the calculation of semantic distance between profile nodes (i.e., $distance(s,d)$ in equation 1). Recall that one important goal in designing SPD is to capture semantic similarity between WordNet nodes. Naturally, we look to the current research comparing semantic similarity between word senses (e.g., Budanitsky and Hirst, 2001). We choose to implement two straightforward methods. For one, we invert (to obtain distance) the WordNet similarity measure of Wu and Palmer (1994), yielding:

$$d_{wp}(n_1, n_2) = \frac{depth(n_1) + depth(n_2)}{2depth(LCS(n_1, n_2))}, \tag{3}$$

where $LCS(n_1, n_2)$ is the lowest common subsumer of n_1 and n_2. The other method we use is the simple edge distance between nodes, d_{edge}.[4]

Thus far, we have defined SPD as a sum of propagated profile scores multiplied by the distance "travelled" (equation 1). We have also considered propagating other values as a function of profile scores. Let's return to the same example but redistribute some of the probability mass of $profile_2$: node E goes from a probability of 0.3 to 0.45, and node F goes from 0.2 to 0.05. As a result, the distribution of the scores at the node B subtree is more skewed towards node E than in the original $profile_2$.

For both the original and modified $profile_2$, SPD has the same value because we are moving a total probability mass of 0.5 from E and F to B, with the same semantic distance (since E and F are at the same level in the tree). However, we consider that, at the node B subtree, $profile_1$ is less similar to the skewed $profile_2$ than to

[3] We have described the algorithm as moving one profile to another. Conceptually, there are cases, as illustrated in the example, where we are propagating profile scores downwards in the hierarchy. Moving scores downwards can be computationally expensive because one may need to search through the whole subtree rooted at the source node for destination nodes. We implemented an alternative by moving all the scores upwards. Since we keep track of the source and destination nodes, the two methods are equivalent.

[4] We also implemented the WordNet edge distance measure of Leacock and Chodorow (1998). Since it did not influence our results, we omit discussion of it here.

the original, more evenly distributed $profile_2$. To reflect this observation, we can propagate the "inverse entropy" in order to capture how evenly distributed the probabilities are in a subtree. We define an alternative version of $amount(s, d)$ as:

$$amount_e(s, d) = weight(d) * entropy_{inv}(s) \quad (4)$$

where we replace $portion(s)$ with inverse entropy, $entropy_{inv}(s)$, which we define as:

$$entropy_{inv}(s) = -\frac{1}{portion(s)\log_2 portion(s)} \quad (5)$$

By propagating inverse entropy, we penalize cases where the distribution of source scores is "skewed." In this work, we will experiment with both methods of propagation (with and without inverse entropy).

4 Materials and Methods

4.1 Corpus Data

Our materials are drawn from a 35M-word portion of the British National Corpus (BNC). The text is parsed using the RASP parser (Briscoe and Carroll, 2002), and subcategorizations are extracted using the system of Briscoe and Carroll (1997). The subcategorization frame entry of each verb includes the frequency count and a list of argument heads per slot. The target slots in this work are the subject of the intransitive and the object of the transitive.

4.2 Verb Selection

We evaluate our method on the causative alternation in order for comparison to the earlier method of McCarthy (2000). We selected target verbs by choosing semantic *classes* (not individual verbs) from Levin (1993) that are expected to undergo the causative alternation. The target verbs are selected randomly from these classes. We refer to these as causative verbs. For both our development and test sets, we chose filler verbs randomly, as long as the verb classes they belong to do not allow a subject/object alternation as in the causative. Verbs must occur a minimum of 10 times per syntactic slot to be chosen.

Note that we did not hand-verify that individual verbs allowed or disallowed the alternation, as McCarthy (2000) had done, because we wanted to evaluate our method in the presence of noise of this kind.

In a pilot experiment on a smaller, domain-specific corpus (6M words, medical domain) (Tsang and Stevenson, 2004), we randomly picked 18 causative verbs and 18 filler verbs for development and 20 causative verbs and 20 filler verbs for testing. In this pilot experiment, SPD is consistently the best performer in both development and testing. SPD achieves a best accuracy of 69% in development and 65% in testing (chance accuracy of 50%).

Given more data (35M words) in our current experiments, we randomly select additional verbs to make up a total of 60 causative verbs and 60 filler verbs, half of these for development and half for testing. Each set of verbs is further divided into a high frequency band (with at least 450 instances of one target slot), a medium frequency band (with between 150 and 400 instances in one target slot), and a low frequency band (with between 20 and 100 instances of one target slot). Each band has 20 verbs (10 causative and 10 non-causative). For each of the development and testing phases, we experiment with individual frequency bands (i.e., high, medium, and low band, separately), and with mixed frequencies (i.e., all verbs). To compare with our earlier results, we also experiment on the pilot development verbs (36 verbs). Note that in the BNC, these verbs are not evenly distributed across the bands, hence we can only experiment with the mixed frequencies condition.

4.3 Experimental Set-Up

Using (verb,slot,noun) tuples from the corpus, we experimented with several ways of building sense profiles of each verb's target argument slots (Resnik, 1993; Li and Abe, 1998; Clark and Weir, 2002).[5] In both our pilot experiment and current development work, we found that the method of Clark and Weir (2002) overall gave better performance, and so we limit our discussion here to the results on their model. Briefly, Clark and Weir (2002) populate the WordNet hierarchy based on corpus frequencies (of all nouns for a verb/slot pair), and then determine the appropriate probability estimate at each node in the hierarchy by using χ^2 to determine whether to generalize an estimate to a parent node in the hierarchy.

We compare SPD to other measures applied directly to the (unpropagated) probability profiles given by the Clark-Weir method: the probability distribution distance given by skew divergence (skew) (Lee, 1999), as well as the general vector distance given by cosine (cos). These are the measures (aside from SPD) that performed best in our pilot experiments.

It is worth noting that the method of Clark and Weir (2002) does not yield a tree cut, but instead generally populates the WordNet hierarchy with non-zero probabilities. This means that the kind of straightforward propagation method used by McCarthy (2000) is not applicable to sense profiles of this type.

To determine whether a verb participates in the causative alternation, we adopt McCarthy's method of using a threshold over the calculated distance measures, testing both the mean and median distances as possible thresholds. In our case, verbs with slot-distances

[5]Although Resnik's measure is not a probability distribution, his method for populating the WordNet hierarchy from corpus counts does yield a probability distribution.

Pilot Verbs	Development Verbs				
all	all	high	med	low	high-med
0.75 SPD cos	0.67 SPD skew	0.7 SPD skew cos	0.7 SPD skew	0.7 SPD	0.75 SPD

Table 1: The best development accuracy along with the measure(s) that produce that result, using a median threshold. SPD refers to SPD without entropy, using either d_{wp} or d_{edge}. “all”, “high”, “high-med”, “med”, and “low” refer to the different frequency bands.

Unseen Test Verbs				
all	high	med	low	avg(h,m,l)
0.67 skew	0.7 SPD$_{d_{wp}}$	0.8 SPD$_{d_{wp}}$ SPD$_{d_{edge}}$ skew	0.6 SPD$_{d_{wp}}$ SPD$_{d_{edge}}$	0.7 SPD$_{d_{wp}}$

Table 2: The best accuracy achieved in testing, along with the measure(s) that produced the result, using a median threshold. SPD refers to SPD without entropy, using the indicated node distance measure. “all”, “high”, “med”, and “low” refer to the different frequency bands. “avg(h,m,l)” refers to the average accuracy of the three frequency bands.

below the threshold (smaller distances) are classified as causative, and those above the threshold as non-causative. In both our pilot and development work, median thresholds consistently fare better than average thresholds, hence we narrow our discussion here to using median only. Using the median also has the advantage of yielding a consistent 50% baseline. Accuracy is used as the performance measure.

5 Experimental Evaluation

We evaluate the SPD method on sense profiles created using the method of Clark and Weir (2002), with comparison to the other distance measures (skew and cos) as explained above. In the calculation of SPD, we compare the two node distance measures, d_{wp} (Wu and Palmer, 1994) and d_{edge}, and the two ways of propagating sense profiles, without entropy ($e0$) and with entropy ($e1$), as described in Section 3. These settings are mentioned when relevant to distinguishing the results. Recall that in all experiments the random baseline is 50%.

5.1 Development Results

On both the original pilot verbs (36 verbs) and the extended development set (60 verbs), SPD performs better than or as well as the other measures. The top performance in each experimental condition is compiled in Table 1. On the pilot verbs, our measure achieves a best accuracy of 75%. On the development verbs, SPD (without entropy, using either d_{wp} or d_{edge}) is also the best or tied for best at classifying all verbs, and verbs in each frequency band. No other measure performs consistently as well as SPD.

We find that SPD with entropy does not work as well as SPD without entropy. However, it is often second best (with the exception for high frequency verbs).

There is some difference in the SPD performance on all verbs between the pilot and development sets. Recall that the pilot set contains 36 verbs from an earlier experiment (which are not evenly distributed among the frequency bands); and the development set contains 60 verbs (the pilot set plus additional verbs, with each band containing 20 verbs). We compare the two verb sets and discover that, in the pilot set, high and medium frequency verbs outnumber the low frequency verbs. To better compare with the pilot verb results, we run the experiment on only the high and medium frequency development verbs. See the “high-med” column under “Development Verbs” in Table 1. SPD (using d_{edge}) remains the best performer with accuracy of 75%, equalling the best performance for the pilot set.

5.2 Test Results

Table 2 shows the best results in the testing phase. Again, SPD has the most consistent performance. Here, similarly to the development results, SPD is the best (or tied for best) at classifying the verbs in the individual frequency bands. However, in classifying all verbs together, it is not the best; it is the second best at 63%.

As in the development results, SPD measures without entropy ($e0$) fair better than those with entropy ($e1$). However, unlike the development results, $e1$ does not do well at all. To examine $e1$’s poor performance, we do a pairwise comparison of the actual classification of the two SPD methods. In all cases, many causative verbs that are classified correctly (i.e., small profile distance) by $e0$ are no longer correct using $e1$ (i.e., they are now classified as large profile distance). By propagating entropy instead of probability mass, the distance between profiles is incorrectly amplified for causative verbs. Since this phenomenon is not observed in the development set, it is unclear under what circumstances the distance is amplified by entropy propagation. We conclude that simple propagation of profile mass is the more consistent method of the two for this application.

Recall that we also experiment with two different node distance measures (d_{wp} and d_{edge}). Though not identical, the performance between the two is remarkably similar. In fact, the actual classifications themselves are very similar. Note that Wu and Palmer (1994) designed their measure such that shallow nodes (i.e., less specific senses) are less similar than nodes that are deeper in the WordNet hierarchy, a property that is lacking in the edge distance measure. We hypothesized that our sense profiles are similar in terms of depth, so that taking relative depth into account in the distance measure has little impact. Comparing the sense profiles of groups of verbs reveals that, with one exception (non-causative development verbs), the difference in depth is not statistically significant (paired t-test). In the case that is statistically significant, the average difference in depth is less than two levels.

For comparison, we replicate McCarthy's method on our test verbs, using tree cuts produced by Li and Abe's technique, which are propagated to their lowest common subsumers and their distance measured by skew divergence. Recall that we do not hand-select our causative verbs to ensure they undergo the causative alternation, and therefore there is more noise in our data than in McCarthy's. In the presence of more noise, her method performs quite well in many cases; it is best or tied for best on the development verbs, medium frequency (70%) and on the test verbs, all verbs (67%), high frequency (80%), and medium frequency (80%). However, it does not do well on low frequency verbs at all (below chance at 40%). We suspect the problem is twofold, arising from the dependence of her method on tree cut models (Li and Abe, 1998). The first problem is that one needs to generalize the tree cut profiles to their common subsumers to use skew divergence. As a result, as we mentioned earlier, semantic specificity of the profiles is lost. The second problem is, as Wagner (2000) points out, less data tends to yield tree cuts that are more general (further up in the hierarchy). Therefore, low frequency verbs have more general profiles, and the distance between profiles is less informative. We conclude that McCarthy's method is less appropriate for low frequency data than ours.

5.3 Frequency Bands

Somewhat surprisingly, we often get better performance with the frequency bands individually than we do with all verbs together. By inspection, we observe that low frequency verbs tend to have smaller distances between two slots and high frequency verbs tend to have larger distances. As a result, the threshold for all verbs is in between the thresholds for each of these frequency bands. When classifying all verbs, the frequency effect may result in more false positives for low frequency verbs, and more false negatives for high frequency verbs.

We examine the combined performance of the individual frequency bands, in comparison to the performance on all verbs. Here, we define "combined performance" as the average of the accuracies from each frequency band. We find that SPD without entropy attains an averaged accuracy of 70%, an improvement of 3% over the best accuracy classifying all verbs together. Separating the frequency bands is an effective way to remove the frequency effect.[6]

Stemming from this analysis, a possible refinement to separating the frequency bands is to use a different classifier in each frequency band, then combine their performance. However, we observe that the best SPD performer in one frequency band tends to be the best performer in other bands (development: SPD without entropy, d_{edge}; test: SPD without entropy, d_{wp}). There does not seem to be a relationship between verb frequency and various distance measures.

6 Conclusions

We have proposed a new method for comparing WordNet probability distributions, which we call sense profile distance (SPD). Given any pair of probability distributions over WordNet (which we call a sense profile), SPD captures in a single measure the aggregate semantic distance of the component nodes, weighted by their probability. The method addresses conceptual problems of an earlier measure proposed by McCarthy (2000), which was limited to tree cut models (Li and Abe, 1998) and failed to distinguish detailed semantic differences between them. Our approach is more general, since it can work on the result of any model that populates WordNet with probability scores. Moreover, the integration of a WordNet distance measure into the formula enables it to take semantic distances directly into account and better capture meaningful distinctions between the distributions.

We have shown that SPD yields practical advantages as well, in demonstrating improved performance in the ability to detect a verb alternation through comparison of the sense profiles of potentially alternating slots. SPD achieves a best performance of 70% accuracy (baseline 50%) on unseen test verbs, and no other measure we tested performed consistently as well as it did. By comparison, McCarthy (2000) attained 73% accuracy on her set of hand-selected test verbs in a similar task; however, when applied to our randomly selected verbs, our replication of her method achieved an overall performance of 67%, and performed very poorly on low frequency verbs.

In our on-going work, we are exploring other applications of SPD, such as assessing document collection similarity, in which such an aggregate semantic distance measure has the potential to reveal meaningful distinctions. In this type of task, sense profiles over other, more domain-specific, ontologies may prove to be useful. In our presentation here, we have described SPD as a measure over sense profiles in WordNet, but clearly the

[6] Another method is to use some type of "expected distance" as a normalizing factor (Paola Merlo, p.c.). However, it is yet unclear how we would calculate this number.

method is general enough to apply to any hierarchical ontology. Indeed, a sense profile—a set of scores over the hierarchy—need not even form a probability distribution. The only requirements for the method are that a meaningful distance measure be definable over nodes in the hierarchy, and that for any two profiles being compared, the sum of their scores is equal (the latter being trivially true for probability distributions, which sum to 1).

7 Acknowledgments

We thank Diana McCarthy (U. of Sussex) for providing the tree cut acquisition code, and Ali Shokoufandeh (Drexel U.) and Ted Pedersen (U. of Minnesota) for helpful discussion. We gratefully acknowledge the support of NSERC and OGS of Canada.

References

C. Bannard. 2002. Statistical techniques for automatically inferring the semantics of verb-particle constructions. Master's thesis, University of Edinburgh, Edinburgh, UK.

T. Briscoe and J. Carroll. 1997. Automatic extraction of subcategorization from corpora. In *Proceedings of the 5th Applied Natural Language Processing Conference*, p. 356–363, Washington, D.C.

T. Briscoe and J. Carroll. 2002. Robust accurate statistical annotation of general text. In *Proceedings of the Third International Conference on Language Resources and Evaluation*, p. 1499–1504, Las Palmas, Canary Islands.

A. Budanitsky and G. Hirst. 2001. Semantic distance in WordNet: An experimental, application-oriented evaluation of five measures. In *Proceedings of the NAACL Workshop on WordNet and Other Lexical Resources*, p. 29–34.

S. Clark and D. Weir. 2002. Class-based probability estimation using a semantic hierarchy. *Computational Linguistics*, 28(2):187–206.

H. T. Dang, K. Kipper, and M. Palmer. 2000. Integrating compositional semantics into a verb lexicon. In *Proceedings of the Eighteenth International Conference on Computational Linguistics*, Saarbrucken, Germany.

B. J. Dorr and D. Jones. 2000. Acquisition of semantic lexicons: Using word sense disambiguation to improve precision. In E. Viegas, editor, *Breadth and Depth of Semantic Lexicons*, p. 79–98. Kluwer Academic Publishers, Norwell, MA.

J. Jiang and D. Conrath. 1997. Semantic similarity based on corpus statistics and lexical taxonomy. In *Proceedings on the International Conference on Research in Computational Linguistics*, p. 19–33, Taiwan.

B. Katz, J. Lin, and S. Felshin. 2001. Gathering knowledge for a question answering system from heterogeneous information sources. In *Proceedings of the Workshop on Human Language Technology and Knowledge Management*, Toulouse, France.

C. Leacock and M. Chodorow. 1998. Combining local context and WordNet similarity for word sense identification. In C. Fellbaum, editor, *WordNet: An Electronic Lexical Database*, p. 265–283. MIT Press.

L. Lee. 1999. Measures of distributional similarity. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics*, p. 25–32.

B. Levin. 1993. *English Verb Classes and Alternations: A Preliminary Investigation*. University of Chicago Press.

H. Li and N. Abe. 1998. Generalizing case frames using a thesaurus and the MDL principle. *Computational Linguistics*, 24(2):217–244.

D. Lin. 1998. An information-theoretic definition of similarity. In *Proceedings of International Conference on Machine Learning*, Madison, Wisconsin.

D. McCarthy. 2000. Using semantic preferences to identify verbal participation in role switching alternations. In *Proceedings of Applied Natural Language Processing and North American Chapter of the Association for Computational Linguistics (ANLP-NAACL 2000)*, p. 256–263, Seattle, WA.

P. Merlo and S. Stevenson. 2001. Automatic verb classification based on statistical distributions of argument structure. *Computational Linguistics*, 27(3):393–408.

R. Rada, H. Mili, E. Bicknell, and M. Bletmer. 1989. Development and application of a metric on semantic nets. *IEEE Transactions on Systems, Man, and Cybernetics*, 19(1):17–30, January/February.

P. Resnik. 1993. *Selection and Information: A Class-Based Approach to Lexical Relationships*. Ph.D. thesis, University of Pennsylvania, Philadelphia, PA.

E. Riloff and M. Schmelzenbach. 1998. An empirical approach to conceptual case frame acquisition. In *Proceedings of the Sixth Workshop on Very Large Corpora*, p. 49–56.

S. Schulte im Walde and C. Brew. 2002. Inducing German semantic verb classes from purely syntactic subcategorisation information. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, Philadelphia, PA.

S. Teufel. 1999. *Argumentative Zoning: Information Extraction from Scientific Articles*. Ph.D. thesis, University of Edinburgh, Edinburgh, UK.

V. Tsang and S. Stevenson. 2004. Using selectional profile distance to detect verb alternations. To appear in the HLT/NAACL 2004 Workshop on Computational Lexical Semantics.

V. Tsang, S. Stevenson, and P. Merlo. 2002. Crosslinguistic transfer in automatic verb classification. In *Proceedings of the 19th International Conference on Computational Linguistics*, Taipei, Taiwan.

A. Wagner. 2000. Enriching a lexical semantic net with selectional preferences by means of statistical corpus analysis. In *Proceedings of the ECAI-2000 Workshop on Ontology Learning*, Berlin, Germany.

Z. Wu and M. Palmer. 1994. Verb semantics and lexical selection. In *Proceedings of the 32nd Annual Meeting of the Association for Computational Linguistics*, p. 133–138, Las Cruces, New Mexico.