An Unsupervised Verb Class Disambiguation

Jianguo Li
Department of Linguistics
The Ohio State University
1712 Neil Ave, Columbus, OH, 43201
jianguo@ling.ohio-state.edu

Abstract

We present an unsupervised learning method for disambiguating verbs that belong to more than one Levin verb class (1993) when occurring in a particular syntactic frame. We used examples that contain unambiguous verbs in each verb class as the training data for ambiguous verbs in that class. A Naive Bayesian classifier was employed for the disambiguation task using context words as features. Our experiments suggest that our unsupervised learning method does not match the supervised one in disambiguating Levin verbs, but it consistently outperforms the random baseline model.

1 Introduction

Much research in lexical acquisition has concentrated on verb classification (Schulte im Walde, 2000; McCarthy, 2000; Merlo and Stevenson, 2001). Levin (1993) argues that verbs that exhibit the same diathesis alternation - alternations in the realization of their argument structure - can be assumed to share certain semantic components and to form a semantically coherent class. Lately, there have been several efforts that aim to disambiguate verbs that belong to more than one Levin class when occurring in a certain subcategorization frame. More precisely, given a verb in a frame, the task is to assign it to one of its possible Levin classes. For example, consider the verb *call* in the following two sentences:

(1) He called me a fool.

(2) He called me a taxi.

The verb *call* is ambiguous between the class of DUB and GET when occurring in the double object frame. We therefore need to automatically identify *call* as a DUB verb in sentence (1) and a GET verb in sentence (2).

Word sense disambiguation is usually cast as a problem in supervised learning, where a verb class disambiguator is induced from a corpus of manually sense-tagged texts. The context within which the ambiguous word occurs is typically represented by a set of linguistically-motivated features from which a learning algorithm induces a representative model that performs the disambiguation task. One classifier that has been extensively used is the Naive Bayesian classifier. A Naive Bayesian classifier assumes that all the feature variables representing a problem are conditionally independent given the value of the classification variable. In the task of verb class disambiguation described above, the features $(a_1, a_2, ..., a_n)$ represent the context surrounding the ambiguous verbs and the probability of a verb *v* belonging to to a class *c* when occurring in a frame *f* given its context can be expressed as

$$P(c, f, v|a_i) \approx P(c, f, v) \prod_{i=1}^{n} P(a_i|c, f, v) \quad (1)$$

Equation (1) has two components: the prior probability $P(c, f, v)$ and the posterior probability $\prod_{i=1}^{n} P(a_i|c, f, v)$. Lapata and Brew (2004) develops an informative prior to estimate a distribution over Levin verb classes for a given verb in a given frame. Its prior model is able to generate class preference for an ambiguous verb. Consider the verb *call* again. It is ambiguous between the class of DUB and GET. The model predicts DUB to be the predominant class. The model's outcome is considered correct given that corpus tokens also reveal a preference for the class DUB. To compute the posterior probability, it uses contextual features (e.g. word collocation) extracted from a manually sense-tagged corpus. Its experiments suggest that the informative prior helps achieve better verb class disambiguation.

In this paper, we focus on only the derivation of the posterior probability. We propose a method for disambiguating Levin verbs without relying on a manually sense-tagged corpus for training the disambiguator.

2 Experiment

2.1 Motivation

As noted above, Levin (1993) has classified verbs according to their syntactic behavior. Verbs that show similar diathesis alternation are assumed to share certain semantic components and to form a coherent semantic class. On the other hand, many scholars have shown that it is possible to automatically identify words that are semantically similar to a given word based on the context words (Rohde et al., 2004) or the syntactic collocation patterns of the words (Lin, 1998). The underlying assumption of this approach is that words that share more context words tend to be semantically closer to each other than words that share fewer context words. Faced with these two different approaches to identifying semantically similar words, we attempt to address the following two questions in this study:

1. Are the semantic components shared by verbs in a Levin class semantically correlated with their contexts words? In other words, do verbs belonging to the same Levin class also tend to share their context words in addition to their syntactic behavior?

2. Can we use the context words of unambiguous verbs in each Levin class to disambiguate ambiguous verbs in that class?

Rohde et al. (2004) provides a method for calculating semantic distance between two words based on their context words. Its experiments indicate that verbs belonging to the same Levin class tend to cluster together, even though not as tightly as nouns, suggesting that verbs within the same Levin class do tend to share their context words, at least to certain extent. Based on this result, we propose a learning method which uses data

class	ambiguous verbs	unambiguous verbs
DUB	*call, make vote*	*anoint, baptize, brand, christen consecrate, crown, decree, dub name, nickname, pronounce, rule stamp, style, term*
GET	*call, find leave, vote*	*book, buy, cash, catch charter, choose, earn, fetch gain, gather, hire, keep order, phone, pick, pluck procure, pull, reach, rent reserve, save, secure, shoot slaughter, steal, win*

Table 1: DUB and GET class

containing unambiguous verbs as the training data for verb class disambiguation. Consider the verb *call* again, it is ambiguous between the class of DUB and GET when occurring in the double object frame. However, most verbs in these two classes are not ambiguous, as shown in Table 1. For an unambiguous verb, we know for sure the class it belongs to without even examining the sentences in which it occurs. To disambiguate *call* in a double object frame, we used all sentences identified as a double object frame that contain an unambiguous verb in the class DUB as the training data for the class DUB and did the same for the class GET. This offers us an unsupervised learning method given that we do not need to manually tag our training data. In this paper, we test our unsupervised learning method on two frames: the double object frame and the transitive frame.

2.2 Constructing Training Data

We parsed the whole BNC using the parser described in Charniak (2000) and then extracted the double object frames and transitive frames from the parser's output. However, the frames obtained this way tend to contain a lot of noise. For example, Charniak's parser does not differentiate between NP arguments and NP adjuncts. The data for the double object frame is likely to be polluted by NP adjuncts. For this reason, we, following Lapata (1999), employed linguistically-motivated heuristics and corpus-based statistical tests to filter our false instances of the double object frame and transitive frame.

2.2.1 Guessing the Double Object Frame

We developed linguistically-motivated heuristics and corpus-based statistical tests to determine whether syntactic cues derived from the output of Charniak's parser are true instances of the double object frame.

Linguistic Heuristics: We applied the following linguistic heuristics to cues of the double object frame.

1. **Reject** if the direct object is a nominative or reflexive pronoun (e.g. *tells* [*Carol*] [***she***], ask [*the subjects*] [***themselves***]).

2. **Reject** if the the head noun in the direct object is a temporal or locative NP or the word "way" (e.g. *fed* [*the boy*] [***yesterday***], *feed* [the boy][***this way***]) [1].

3. **Accept** if the indirect object is an accusative or reflexive or indefinite pronoun (e.g. *gave* [***them***] [*another chance*], *made* [***herself***] [*a snack*], *cook* [***everyone***] [*a meal*])

4. **Accept** if the structure of the direct object is "$MOD^{+}Noun^{+}$" (e.g. *bought* [*Mary*] [***a beautiful beach house***]) [2].

5. **Cannot decide** if the cue's surface structure is "$MOD^{*}NounNoun^{+}$" (e.g. *offer* [***a free bus***] [***service***]). These cases receive further processing.

6. **Reject** in all other cases.

Compound Nouns: To deal with the **cannot decide** case, we employed the process proposed in Lapata (1999). This detects if a noun sequence is a compound noun or a sequence that should be parsed as a pair of noun phrases. Lapata's compound noun detection process works as follows: It only applies to noun sequences of length 2 and 3. They were compared against a compound dictionary compiled from WordNet (Miller and William, 1991). Noun sequences that are listed in the WordNet as compound nouns were rejected as instances of the double object frame. For sequences of length 2 not listed in the WordNet, we estimated log-likelihood ratio score (G-score) (Dunning, 1993) to estimate the lexical association between two nouns. We assumed that two nouns cannot be double NP arguments if they are lexically associated. Therefore, cues were rejected as instances of the double object frame if they contain two nouns whose G-score has a *p*-value less than 0.05. For sequences of length 3 not listed in WordNet, we first determined the bracketing (either [[n_1 n_2] n_3], or [n_1 [n_2 n_3]]) and then computed the G-score between the single noun and the 2-noun sequence. Again cues whose G-score has a *p*-value less than 0.05 were rejected as instances of double object frame. This is exactly what Lapata (1999) did, with slightly different input.

2.2.2 Guessing the Transitive Frame

We applied several linguistic heuristics to cues of the transitive frames [3].

1. **Reject** if the verb is followed by a nominative pronoun (e.g. *Japanese suppliers assume* [***they***]).

2. **Reject** if the verb is followed by a noun phrase whose head noun is a temporal or locative NP (e.g. *play* [***each week***]).

3. **Reject** if the cue's structure is [V-NP-S] or [V-NP-INF], or [V-NP-ADJP] and the verb licenses the frame according to Levin's classification. (e.g. *told* [*me*] [***that he is leaving***], *asked* [*him*] [*to leave*], *keep* [*it*] [***cold***]).

4. **Accept** in all other cases.

2.2.3 Evaluation

The performance of the linguistic heuristics for identifying the double object frame was evaluated by randomly selecting 1,000 tokens accepted as instances of the double object frame and 1,000 tokens rejected as instances of the double object frame by the heuristics. These selected tokens were examined by a human judge. The examination showed that the "Reject" and "Accept" heuristics achieve an accuracy of 93.6% and 91.8% respectively. We further evaluated the compound noun detection procedure by randomly selecting 500 tokens which the compound detection accepted as compound nouns. Again, these tokens were inspected by a human judge. Our examination showed that 492 tokens are compound nouns, giving us an accuracy of 98.4% for compound noun detection. These results are comparable to what Lapata (1999) reports.

---

[1] The list of temporal NPs includes the days of the week, the months and seasons of the year, and expressions containing the word *year, month, week, day, morning, afternoon, evening, night, weekend, moment, hour, minute, yesterday, today, tomorrow, and tonight*.

[2] MOD represents any prenominal modifier including cardinal, determiner, adjective or numeral or ordinal, predeterminer, possessive pronoun and possessive NP.

[3] We included the double object frames rejected by our heuristics in the previous section, among the inputs to this stage, since we assume that they are probably transitives.

Again, we evaluated the performance of our heuristics for guessing transitive frames by randomly selecting 1,000 tokens rejected as the transitive frame and another 1,000 tokens accepted as the transitive frame by our heuristics. These selected tokens were examined by a human judge. The “Reject” and “Accept” heuristics achieve an accuracy of 91.5% and 89.8% respectively. Again, these results are comparable to what Lapata (1999) reports.

The evaluation of our heuristics for guessing the double object and transitive frame shows that our heuristics are very accurate in distinguishing correct cues from incorrect cues. As a result, we are reasonably confident that the frames generated by Charniak’s parser are more accurate after being screened with our linguistic heuristics. However, the training data obtained this way is still noisy, which potentially has a negative effect on the performance of our unsupervised learning method.

2.3 Classifier and Feature Space

2.3.1 A Naive Bayesian Classifier

As noted above, we employed a naive Bayesian classifier for our disambiguation task. Within a Naive Bayesian approach, the probability of a verb *v* belonging to to a class *c* when occurring in a frame *f* given its context can be expressed as

$$P(c, f, v|a_i) \approx P(c, f, v) \prod_{i=1}^{n} P(a_i|c, f, v) \quad (2)$$

If we choose the prior $P(c, f, v)$ to be uniform, (2) can be further simplified to:

$$P(c, f, v|a_i) \approx \prod_{i=1}^{n} P(a_i|c, f, v) \quad (3)$$

2.3.2 Feature Space

As common in word sense disambiguation, we represented the contexts of the ambiguous verbs as collocations. Collocations are words that are frequently adjacent to the verbs to be disambiguated. We considered 12 word collocations. For example, a collocation **L2R4** represents two words to the left and four words to the right of an ambiguous verb. Collocations are represented as lemmas.

2.4 Results and Discussion

Our test data consists of 1500 example sentences involving 18 ambiguous verbs and two frames (8 verbs for the double object frame and 10 verbs for the transitive frame). The test data was provided by Mirella Lapata and disambiguated by two human annotators. We compared the performance of the following three modes:

- **unsupervised learning**: the classifier was trained on the training data that contains sentences of all unambiguous verbs in each class as described in the previous section and tested on the 1500 test examples.

- **supervised learning**: the classifier was trained and tested using 10-fold cross-validation on the 1500 test examples.

- **random baseline**: we randomly assigned to each test example a class of all possible classes the verb to be disambiguated belongs to when occurring in the given frame, using a uniform distribution. We did this 100 times and averaged their results.

mode	average accuracy	highest accuracy	lowest accuracy
unsupervised	54.0%	62.5%(L0R2)	49.7%(L1R0)
supervised	60.7%	71.8%(L2R4)	47.9%(L1R0)
baseline		44.9%	

Table 2: Results for the double object frame

mode	average accuracy	highest accuracy	lowest accuracy
unsupervised	58.2%	65.6%(L0R2)	49.7%(L2R0)
supervised	64.7%	74.9%(L2R4)	54.8%(L1R0)
baseline		43.8%	

Table 3: Results for the transitive frame

The results are summarized in Table 2 and 3. The average accuracy was obtained by averaging the accuracy over all 12 collocations. We also report the highest or lowest accuracy for each frame. For example, using the collocation L0R2 (see Table 2) our unsupervised learning achieves the best performance 62.50% for disambiguating ambiguous verbs in the double object frame. As shown in Table 2 and 3, our unsupervised learning does not match the supervised learning in disambiguating Levin verbs. However, it consistently outperforms the random baseline model, suggesting that verbs belonging to the same Levin class do share their context words. Our unsupervised method could be used in the absence of a manually sense-tagged corpus.

To see how our unsupervised learning performs on each individual verb, we compared

verb	unsupervised learning	random baseline
call	63.5%	50%
cook	55.6%	50%
find	58.8%	50%
leave	32.0%	33.3%
make	70.5%	50%
pass	40.5%	33.3%
save	54.5%	50%
write	56.7%	50%

Table 4: Results for individual verbs in the double object frame

the accuracy for disambiguating each individual verb achieved by the unsupervised learning method against that achieved by a random baseline model. Again the accuracy for each individual verb achieved by the unsupervised learning was obtained by averaging the results over all 12 collocations. The random baseline for each individual verbs was obtained by dividing 1 over the number of possible classes the verb to be disambiguated belongs to when occurring in a particular frame. For example, the verb *call* belongs to two classes when occurring in the double object frame. The accuracy achieved by the random baseline model for *call* is therefore 50%. Our comparison reveals that the unsupervised learning method achieves an accuracy higher than the random baseline model for 17 verbs out of all the 18 verbs we tested. The only exception is the verb *leave* in the double object frame. We summarized the results for each individual verb in the double object transitive frame in Table 4. As shown in Table 4, our unsupervised method achieves an accuracy of 32.0% for the verb *leave*, which is lower than the random baseline. However, this result might be explained by the fact that our two human annotators also had a hard time disambiguating *leave* in the double object frame. As a matter of fact, the inter-annotator agreement measured by Kappa coefficient (Cohen, 1960) for *leave* is the lowest among all 8 verbs we tested for the double object frame.

Overall, our unsupervised learning does a relatively good job disambiguating Levin verbs. However, there is one problem that this unsupervised learning cannot deal with. Recall that our unsupervised learning relies on sentences containing unambiguous verbs to disambiguate ambiguous verbs. There are some cases where we cannot find in our training data any sentences containing any unambiguous verb belonging to a particular class that is involved in our disambiguation task. For example, the verb *feed* is ambiguous between the class GIVE and FEEDING when occurring in the double object frame. The class FEEDING has 6 verbs in it: *bottlefeed, breastfeed, feed, forcefeed, handfeed, spoonfeed*. Although the verb *feed* in its FEEDING sense frequently occurs in the double object frame (e.g. *Mary fed the baby some soy milk.*), we could not find *bottlefeed, breastfeed, forcefeed, handfeed or spoonfeed* with a double object frame in our training data. As a consequence, it does not make much sense for us to apply our unsupervised learning method to the verb *feed*. To solve this problem, we could build a prior model as in Lapata and Brew (2004) which is relatively accurate in generating the predominant class for a verb in a given frame and use the predominant class generated by the prior model as the default class for the test examples containing *feed*.

3 Conclusions and Future Work

In this paper, we experimented with an unsupervised learning method to disambiguate ambiguous Levin verbs. Our experiments show that the unsupervised learning method is not as accurate as the supervised one. One reason is that we relied on a statistical parser for identifying double object and transitive frames in constructing training data for our unsupervised learning method. Even though we applied a set of linguistically-motivated heuristics to filter out false instances of the frames we are interested in, the training data obtained this way is still noisy in that some false instances of the double object and transitive frames are included in the training data. However, we believe that the major reason for our results is that Levin verb classification is based on diathesis alternations, not context words. Nevertheless, since our unsupervised learning method performs consistently better than the random baseline, we believe that the semantic components shared by verbs within the same Levin class are semantically correlated with their context words.

Several avenues can be taken for future research. First, instead of using all context words as features, we could use the arguments (subjects, direct objects and indirect objects) selected by the verbs (Lin, 1998). For example, consider *The man wrote me a very long letter.*, we say *write* has three features: subj(*man*), direct_obj(*letter*) and indirect_object(*me*). Intuitively, these arguments should be more discriminative than words like *the*, *a*, and *very*. Next, we only tested our unsupervised learning method on two of Levin's subcategorization frames. It remains to be shown that this unsupervised learning method works equally well for other frames. Finally, we also want to try a state-of-the-art word sense disambiguator (e.g. support vector machine) and experiment it with both the supervised and unsupervised learning methods.

References

Eugene Charniak. 2000. A maximum-entropy-inspired parser. In *Proceedings of the 2000 Conference of the North American Chapter of the Association for Computation Linguistics*, pages 132–139.

Jacob Cohen. 1960. A coefficient of agreement for nominal scales. *Educational and Psychological Measurement*, 20:37–46.

Ted Dunning. 1993. Accurate methods for the statistics of surprise and coincidence. *Computational Linguistics*, 19(1):61–74.

Mirella Lapata and Chris Brew. 2004. Verb class disambiguation using informative priors. *Computational Linguistics*, 30(1):45–73.

Mirella Lapata. 1999. Acquiring lexical generalizations from corpora: A case study for diathesis alternations. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics*, pages 397–404.

Beth Levin. 1993. *English Verb Classes and Alternations: A Preliminary Investigation*. University of Chicago Press, Chicago, 1st edition.

Dekang Lin. 1998. Automatic retrieval and clustering of similar words. In *COLING-ACL 98*.

Diana McCarthy. 2000. Using semantic preference for identifying verbal participation in role switching alternations. In *Proceedings of the 1st Conference of the North American Chapter of the Association for Computational Linguistics*, pages 58–63.

Paola Merlo and Suzanne Stevenson. 2001. Automatic verb classification based on statistical distribution of argument structure. *Computational Linguistics*, 27(3):373–408.

George Miller and Charles William. 1991. Contextual correlates of semantic similarity. *Language and Cognitive Processes*, 6(1):1–28.

Douglas Rohde, Laura Gonnerman, and David Plaut. 2004. An improved method for deriving word meaning from lexical co-occurrence. *Cognitive Science*. submitted.

Sabine Schulte im Walde. 2000. Clustering verbs semantically according to alternation behavior. In *Proceedings of the 18th International Conference on Computational Linguistics*, pages 747–753.