Lang Resources & Evaluation (2009) 43:181–208
DOI 10.1007/s10579-009-9085-0

Improving English verb sense disambiguation performance with linguistically motivated features and clear sense distinction boundaries

Jinying Chen · Martha S. Palmer

Published online: 26 February 2009

Abstract This paper presents a high-performance broad-coverage supervised word sense disambiguation (WSD) system for English verbs that uses linguistically motivated features and a smoothed maximum entropy machine learning model. We describe three specific enhancements to our system's treatment of linguistically motivated features which resulted in the best published results on SENSEVAL-2 verbs. We then present the results of training our system on OntoNotes data, both the SemEval-2007 task and additional data. OntoNotes data is designed to provide clear sense distinctions, based on using explicit syntactic and semantic criteria to group WordNet senses, with sufficient examples to constitute high quality, broad coverage training data. Using similar syntactic and semantic features for WSD, we achieve performance comparable to that of human taggers, and competitive with the top results for the SemEval-2007 task. Empirical analysis of our results suggests that clarifying sense boundaries and/or increasing the number of training instances for certain verbs could further improve system performance.

Keywords Word sense disambiguation · Sense granularity · Maximum entropy · Linguistically motivated features · Linear regression

1 Introduction

There is increasing evidence that word sense disambiguation (WSD), determining the meaning a word bears in its given context, has the potential to improve the

J. Chen (✉)
BBN Technologies, Cambridge, MA, USA
e-mail: jchen@bbn.com

M. S. Palmer
University of Colorado, Boulder, CO, USA
e-mail: mpalmer@colorado.edu

performance of many high-level natural language processing (NLP) applications that require a certain degree of semantic interpretation, such as machine translation, information retrieval (IR) and question answering (Gonzalo et al. 1998; Sanderson 2000; Stokoe 2003; Carpuat and Wu 2007; Chan et al. 2007). However, high accuracy and broad coverage (disambiguation of a large vocabulary) are two crucial prerequisites for WSD to be able to benefit NLP applications. Previous investigations into the role of WSD in IR have shown that low accuracy in WSD negated any possible performance increase from ambiguity resolution (Sanderson 1994; Stokoe 2003). As evidenced by the SENSEVAL exercises,[1] supervised WSD systems tend to perform better than unsupervised methods (Edmonds and Cotton 2001; Palmer et al. 2001; Mihalcea et al. 2004; Snyder and Palmer 2004; Pradhan et al. 2007). On the other hand, creating the necessary large-scale high-quality sense-tagged corpora is very difficult and time-consuming. In fact, many successful attempts at all-words WSD use unsupervised methods to compensate for the lack of training data.

The work we report here targets large-scale high performance word sense disambiguation and includes three major aspects: (1) developing a high-performance WSD system for English verbs by using linguistically motivated features; (2) applying this system to the first large-scale annotation effort aimed specifically at providing suitable training data for high-performance WSD, followed by discussion and analysis of these results; and (3) discussing potential future research areas for large-scale high-performance supervised WSD. The annotation effort mentioned in (2) involves annotating sufficient quantities of instances for English verbs that are linked to a sense inventory based on coarse-grained groupings of fine-grained WordNet senses (Fellbaum 1998).

We focus on verb sense disambiguation for two reasons. First, the problems caused by high polysemy are more serious for verbs, as indicated by the relatively poorer performance achieved by the best system in the SENSEVAL-2 English lexical sample task for verbs: 56.6% accuracy, in contrast with the 64.2% accuracy for all parts-of-speech (Yarowsky et al. 2001; Dang and Palmer 2002). In the coarse-grained English lexical sample task in SemEval-2007, the best system's performance on verbs are 11% lower than its performance on nouns (78 vs. 89%) (Pradhan et al. 2007). Second, accurate verb sense disambiguation is very important, not only for selecting lexical items but also, in many cases, for generating correct and elegant syntactic structures for the target side in machine translation. It is also extremely useful for NLP applications based on deep-level natural language understanding, such as the question answering systems that take full-sentence queries as input or information extraction systems that find global and/or domain-independent relational information.

In Sect. 2, we introduce our supervised WSD system built using a smoothed maximum entropy (MaxEnt) model and linguistically motivated features. We first discuss the motivation for using linguistically motivated features for WSD from the angle of reducing data sparseness for machine learning. Second, we briefly introduce the structure of our WSD system and the machine learning model it uses. We then describe in detail three specific enhancements we made to the automatic

[1] http://www.senseval.org/.

feature generation procedure of our system. With these three enhancements, our system achieved the best performance, i.e., 64.6% accuracy, for highly polysemous verbs (16.7 senses on average, based on WordNet 1.7 sense distinctions) in an evaluation using data from the English SENSEVAL-2 lexical sample task (Chen and Palmer 2005).

However, 65%, or even 70% WSD accuracy is insufficient for NLP applications. Given that human inter-annotator agreement (ITA) rates for WordNet senses tend to average just above 70%, it is unlikely that, even with vast amounts of training data, systems will be able to improve much on that score. Furthermore, if two senses of a word are difficult to distinguish, even by humans, it is a strong indicator that the sense distinctions of these two senses are not crucial for understanding sentences containing this word. Therefore, we are participating in a large-scale annotation effort that is based on grouping subtle, fine-grained WordNet senses into coherent semantic sense groups that can be readily distinguished by human annotators. This is part of the OntoNotes project, which also includes Treebanking (Marcus et al. 1994), PropBanking (Palmer et al. 2005), linking to the Omega Ontology (Philpot et al. 2005) and coreference (Hovy et al. 2006). The goal of this project is to achieve average ITA rates of 90%, in order to create training data that can support system performance in the 80+% range. In Sect. 3, we introduce and discuss our experimental results from applying our WSD system to 217 verbs from the OntoNotes data mentioned above and to the 65 SemEval-2007 verbs which came from the same data source. First, we briefly introduce our current work in OntoNotes verb sense annotation. We then show that our system performance on the 217 OntoNotes verbs approaches ITA rates, which demonstrates that automatic WSD is comparable to human performance. Our system performance is also competitive with the top system for the SemEval-2007 verbs. We discuss and analyze the experimental results on the 217 verbs with respect to three major aspects: (1) the impact of grouping fine-grained senses on ITA and system performance; (2) the impact of using linguistically motivated features for automatic disambiguation of grouped senses; (3) statistical analysis of factors that affect system performance as well as analysis of certain verbs that have system performance significantly higher or lower than their ITA rates.

In the last section, we discuss potential research areas for large-scale high-performance supervised WSD based on our experimental results presented here and our previous exploratory efforts in active learning.

2 A high-performance supervised WSD system by using linguistically motivated features

The senses of a polysemous word can only be distinguished by using its context. In practice, a WSD system instantiates the contextual information of a target word as various types of features. WSD approaches are often categorized as linguistically impoverished or linguistically rich. Linguistically rich approaches use linguistically motivated features (also called rich linguistic features), e.g., syntactic and/or semantic features, and rely heavily on sophisticated preprocessing. In contrast,

linguistically impoverished approaches want to avoid such preprocessing and therefore eschew such features.

2.1 Using linguistically motivated features

We chose a linguistically rich approach for our WSD system for two major reasons. First, linguists and computational linguists have found that a verb's meanings are closely related to (or determine) its linguistic behavior, including the syntactic alternations in which it participates and the different types of semantic arguments it can take (Levin 1993; Hanks 1996; Fellbaum et al. 2001, 2005). For example, the verb *leave* has two distinct meanings in (1a) and (1b):

(1) a. She did not ***leave*** until midnight. (*leave*: go away from a place)
b. The inflation ***left*** them penniless. (*leave*: act or be so as to become in a specified state)

When *leave* is used with the meaning *go away*, it often takes a *location* as its direct object or has its direct object dropped as in (1a), but never occurs with a small clause as in (1b). The subjects of this verb with this particular sense should be *movable* objects, either animate (such as a person or an animal) or inanimate (such as a train). In contrast, with the meaning *act so as to become a specified state*, the verb *leave* usually occurs with a small clause with the syntactic pattern *verb + noun + adjective*. In this sense the subjects tend to be events or actions which are quite different from the subjects of the *go away* sense. Therefore, (linguistically motivated) features representing specific aspects of a verb's linguistic behavior are expected to be quite useful for inferring the verb's meanings. To distinguish between the two senses of *leave* given by the above example, a machine learning algorithm using a single syntactic feature that denotes whether the verb occurs with a small clause construction can achieve fairly high accuracy. When taking into account the other senses of this verb in the corpus, the algorithm will need more features for sense disambiguation, e.g., the semantic features representing the semantic categories of the verb's subjects and direct objects and the syntactic features denoting other syntactic patterns the verb can appear in.

The second reason for using linguistically motivated features is to relieve the data sparseness problem. If we represent the data instances as feature vectors and find there are only a small percentage of non zero's in the matrix of observations (data instances) vs. features, we then face a problem that is common to machine learning and is termed "data sparseness". WSD suffers from a data sparseness problem mainly because the features, which are extracted from the context of a target word and used for classification (disambiguation), generally include lexical features (e.g., words occurring in a local context window and words associated with noun phrase (NP) arguments of verbs). Without proper generalizations, the amount of possible lexical features used by a machine learning model can be very large, but the frequencies with which they occur in the data sets can be very low. We use Eq. 1 to estimate the data sparseness of the feature vector data generated by our WSD system, where *freq* (f_j) is the frequency of the *j*th feature observed for a verb (in our

data, each feature occurs at most once for an instance), m is the total number of features and n is the total number of data instances observed for the verb. In practice, we calculate the *data sparseness rate* for each verb by using Eq. 1 and then average these values to get the final result. A large value for the data sparseness rate means a high degree of sparseness of the data.

$$data\_sparseness\_rate = 1 - \frac{\sum_{j=1}^{m} freq(f_j)}{m * n} \tag{1}$$

In practice, we often use another variable, *feature activation frequency* (i.e., on average how many instances for which a feature is active or observed), for the data sparseness analysis. The value of this variable can be easily calculated based on the value of the *data sparseness rate*, as shown in Eq. 2, and has a more straightforward interpretation.

$$feature\_activation\_frequency = \frac{\sum_{j=1}^{m} freq(f_j)}{m} = (1 - data\_sparseness\_rate) \times n \tag{2}$$

For example, the data sparseness analysis (using Eqs. 1 and 2) of the feature vector data our system generated for the 29 SENSEVAL-2 English verbs gives an average *data sparseness rate* of 0.975 and an average *feature activation frequency* of 4.63 (averaged on the 29 verbs). That means a feature tends to be active (be observed) for very few (i.e., 4.63) instances and an instance typically has a very low percentage of active features among all features observed for the whole data set ($0.025 = 1 - 0.975$). As a consequence, it is very likely that many active features (observed features) for the test data have not been seen by a machine learning model for WSD during its training phase. We call such features OOV (Out Of Vocabulary) features and use Eq. 3 to calculate the *OOV feature rate*.

$$OOV\_feature\_rate_k = \frac{\left|f \middle| f \notin F_{train,k} \wedge f \in F_{test,k}\right|}{\left|f \in F_{test,k}\right|} \tag{3}$$

where f is a feature, $F_{train,k}$ is the set of type k features for the training data, and $F_{test,k}$ is the set of type k features for the test data. In our OOV feature analysis of the SENSEVAL-2 verbs, we computed the *OOV feature rate* for the lexical features associated with verbs' NP arguments and got an *OOV feature rate* value of 0.699, which means 69.9% of such lexical features have not been seen in the training data.

The above analysis indicates that WSD faces a grave problem of data sparseness. A (probabilistic) machine learning model cannot reliably estimate the impact (prediction power) of low frequency features on the classification and cannot utilize a potentially large number of features that are observed only in the test data (i.e., OOV features). In addition, many features that are observed in the training data and unseen in the test data are used to estimate the distribution of the complete data set (including the test data) and can also weaken the impact of the truly useful features on the classification. These three factors, i.e., unreliable estimation, unknown features in the test data and underestimation of the impact of the useful features,

often result in poor performance of a machine learning model even if it achieves a fairly high training accuracy.

Using linguistically motivated features can help alleviate data sparseness and its related problems as discussed above. In the first place, these features are usually more indicative of verb sense distinctions than simple collocation features such as words next to the target verb or their POS tags.

(2) a. This horse has ***drawn*** a great big, old-fashioned wagon slowly up the hill. (*draw*:pull)
b. His speech has ***drawn*** a great deal of attention. (*draw*: get or derive)

For example, the verb *draw* has two distinct meanings in (2a) and (2b), which can be readily distinguished by the head noun of its direct object. For example, with the meaning *pull*, as in (2a), this verb often takes a concrete object (such as *wagon* or *boat* etc.) as its direct object. In contrast, with the meaning *get or derive*, as in (2b), it often takes an abstract noun (such as *attention* or *benefit* etc.) as its direct object. A disambiguation algorithm only looking at the adjacent positions of the target verb will miss the indicative features, e.g., *wagon* and *attention* in (2). If the algorithm extracts linguistically impoverished features from a broader context, the learning model has a better chance of finding relevant features. However, a broader context also introduces more irrelevant (noisy) features, e.g., *great, big*, *old-fashioned*, and *deal* in (2). As a consequence, the machine learning model needs more training data to tell which features are relevant (irrelevant). Unfortunately, usually there is less than the required amount of training data for a WSD task. An efficient way to alleviate this problem is to use more indicative features, that is, linguistically motivated features. More indicative features tend to be observed for more training and test instances. The common subspace (shared by the training and test data) constructed by these features is expected to have a reasonably high density without introducing too many noisy features into the whole feature space.

Another advantage comes from using semantic features, e.g., semantic categories of NP arguments. Semantic features are more general than lexical features (words or stem words). For example, to disambiguate the two senses of the verb *draw* in (2), in addition to using direct_object = *wagon* and direct_object = *attention* as features, we also use semantic features such as direct_object = *physical object* and direct_object = *psychological feature*. These semantic features are WordNet synsets and hypernyms. An obvious advantage of this treatment is that a learning model can handle more unknown words (words that have not occurred in the training data) provided that it has access to their semantic categories. For example, in the SENSEVAL-2 verb data, the *OOV feature rate* for the lexical features associated with verbs' NP arguments (as calculated by Eq. 3) is 0.699 and that for the corresponding WordNet semantic features (synsets and hypernyms) is only 0.280.

Using semantic features can also reduce the negative effects caused by low frequency features (data sparseness often results in many low frequency features). This method belongs to the class-based approach designated as addressing the data sparseness problem in the WSD literature (Ide and Veronis, 1998). It is expected

that, given a relatively small amount of training data, machine learning models will obtain a better estimation of their parameters for classes of words than for individual words.

In practice, semantic features are usually used in conjunction with lexical features in supervised WSD. One reason is that abandoning lexical features can lose information about subtle sense distinctions. In addition, semantic categories, from either hand crafted or automatically generated taxonomies, do not always serve our purpose. Our English WSD system uses WordNet synsets and hypernyms as semantic features associated with the verbs' NP arguments. We can calculate the *feature activation frequency* in the same way as we discussed previously (by Eq. 2). The results show that, on average, 5.7 instances are observed for each of these semantic features and 3.7 for each of the lexical features mentioned above.

Due to the reasons just discussed, we chose a linguistically rich approach for verb sense disambiguation. It is worth noting that the advantages of using linguistically motivated features can be generalized to other parts-of-speech such as adjectives and adverbs. For example, empirical study (Yarowsky 1993; Yarowsky and Florian 2002) showed that machine learning algorithms can take advantage of the nouns modified by adjectives when disambiguating the senses of these adjectives. Similarly, it is expected that adverbs derive disambiguation information from the verbs they modify. There is an exception for nouns, however. The senses of nouns generally can be distinguished well by simple collocation features from the local context and a bag of words extracted from a wider context (topical features) without using linguistically motivated features (Yarowsky and Florian 2002).

2.2 Word sense disambiguation system

Figure 1 is an overview of our WSD system. Our system uses Ratnaparkhi's (1998) MaxEnt sentence boundary detector (MXTerminator) and POS tagger (MXPost), Bikel's (2002) parsing engine, and a named entity tagger, *IdentiFinder*™ (Bikel et al. 1999), to preprocess the training and test data automatically.

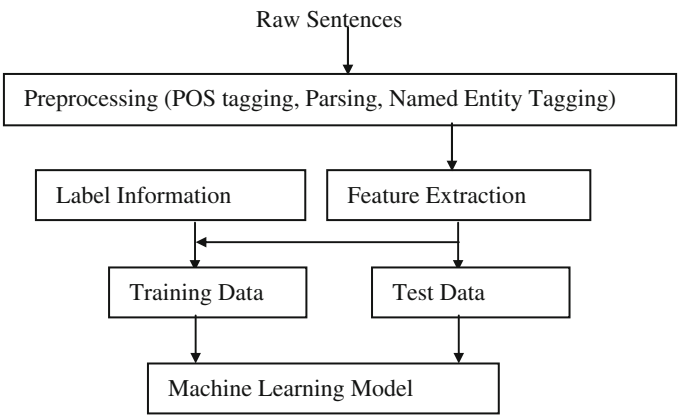

Fig. 1 Structure of our WSD system

We chose a smoothed maximum entropy (MaxEnt) model with a Gaussian prior for machine learning because a MaxEnt model combines evidence (features) from different sources naturally without assuming the independence of these features (Berger et al. 1996). The reason we chose a MaxEnt model with a Gaussian prior is that it has been shown to be less prone to overfitting than MaxEnt models with other smoothing methods (Chen and Rosenfeld 1999). This property is very appealing since, as discussed in Sect. 2.1, data sparseness is a serious problem for WSD and it often results in many low frequency features that are likely to cause overfitting for machine learning models. Furthermore, there has already been a successful application of this type of model to WSD (Dang 2004).

2.3 Three specific enhancements to feature generation

The types of features used by our original WSD system were inspired by the successful WSD system of Dang (Dang and Palmer 2002; Dang 2004). However, we implemented these features in different ways. Furthermore, we enhanced the treatment of certain rich linguistic features, which we believed would boost the system's performance. Before discussing these enhancements, we first briefly describe the basic syntactic and semantic features used by our system:

Syntactic features:

1. Is the sentence passive, semi-passive[2] or active?
2. Does the target verb have a subject or object? If so, what is the head of its subject or/and object?
3. Does the target verb have a sentential complement?
4. Does the target verb have a PP adjunct? If so, what is the preposition and what is the head of the NP argument of the preposition?

Semantic features:

1. The Named Entity tags of proper nouns (***Person, Organization*** and ***Location***) and certain types of common nouns (***Time, Date, Money*** and ***Percent***)
2. The WordNet synsets and hypernyms of head nouns of the NP arguments of verbs and prepositions

In addition to these rich linguistic features, our system also uses local collocation features (words and their POS's within a 5-word window centered by the target word) and topical features (open-class words in the two sentences preceding and following the sentence containing the target word).

To better explore the advantage of using rich syntactic and semantic features, we focused on three main enhancements: increasing the recall of the extraction of a verb's subject; unifying the treatment of semantic features of pronouns, common nouns and proper nouns; and providing a verb-specific treatment of sentential complements. These are each described in detail in (Chen and Palmer 2005) and we repeat the key content below for quick reference.

[2] Verbs that are past participles and are not preceded by *be* or*have* verbs are semi-passive.

2.3.1 Increasing subject extraction recall

To extract a subject, our original system simply checks the left NP siblings of the highest VP that contains the target verb and is within the innermost clause (see Fig. 2). This method has high precision but low recall and cannot handle the three common cases listed in (1).

(1) a. **Relative clauses**: For Republicans$_{sbj}$ [$_{SBAR}$ who began$_{verb}$ this campaign with such high hopes], ...

b. **Nonfinite clauses**: I$_{sbj}$ didn't ever want [$_{S}$ to see$_{verb}$ that woman again].

c. **Verbs within PP's**: Karipo and her women$_{sbj}$ had succeeded [$_{PP}$ in driving$_{verb}$ a hundred invaders from the isle ...]

To increase the recall, we refined the procedure of subject extraction by adding rules based on linguistic knowledge and bracketing labels that can handle relative clauses, nonfinite clauses, and verbs within prepositional phrases (PP's) as illustrated in (Chen and Palmer 2005). For example, if a clause containing the target verb has a bracketing label SBAR and an NP parent, and is headed by a relative pronoun such as *that*, *which* or *who*, then check its left NP siblings for the verb's subject.

Since the subject extraction function is embedded into the feature extraction module of our WSD system, we did not compute the precision and recall of this enhancement independently. Estimation based on SENSEVAL-2 English verb data showed that, with this enhancement, our new system extracts about 35% more subjects than before.

2.3.2 Unifying semantic features

In order to provide a more uniform treatment for the semantic features of the NP arguments of verbs and prepositions, we first merged the semantic features associated with proper nouns and common nouns. We then extended our treatment to include pronouns by adding a pronoun resolution module.

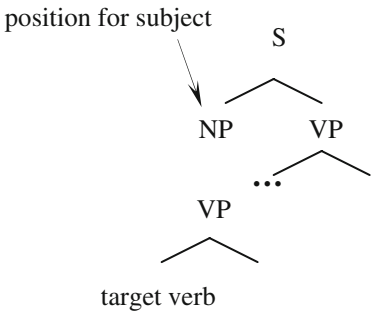

Fig. 2 Subject—left NP sibling of highest VP

2.3.2.1 Merging semantic features Our system used an automatic named entity tagger, *IdentiFinder*™, to tag proper nouns with ***Person, Organization*** and ***Location*** and common nouns with ***Date, Time, Percent*** and ***Money***. Additional semantic features are all WordNet synsets and hypernyms (each of which is represented by a unique number defined in WordNet) of the head nouns of NP arguments, i.e., the system does not disambiguate different WordNet senses of a head noun.

Previously there was no overlap between semantic features generated by the named entity tagger and by WordNet. For example, a personal proper noun only has a ***Person*** tag that has no similarity to the WordNet synsets and hypernyms associated with similar common nouns such as *specialist* and *doctor*, etc. This is likely to be a problem for WSD tasks that usually have relatively small amounts of training data, such as SENSEVAL-2. To overcome this problem, our new system associates a common noun (or a noun phrase) with each Named Entity tag (see (2)) and adds the WordNet semantic features of these nouns (or noun phrases) to the original semantic feature set.

(2) ***Person*** – someone, ***Organization*** – organization, ***Location*** – location
Time – time unit, ***Date*** – time period, ***Percent*** – percent, ***Money*** – money

2.3.2.2 Adding pronoun resolution Our original system had no special treatment for pronouns, although a rough count showed that about half of the training instances contain pronominal arguments. Lacking a high performance automatic pronoun resolution module, we adopted a hybrid approach. For personal pronouns, we simply treated them as personal proper nouns. For the rest of the pronouns including *they, them, it, themselves* and *itself*, which occur in about 13% of the training instances, we programmed a rather simple rule-based pronoun resolver. In brief, the resolver searches the parse tree for antecedent candidates similarly to Hobb's algorithm as exemplified in (Lappin and Leass 1994) and uses several syntactic and semantic constraints to filter out impossible candidates. The constraints include syntactic constraints for anaphora antecedents (Lappin and Leass 1994), number agreement, and whether the candidate is a person. The first candidate that survives the filtering is regarded as the antecedent of the pronoun and its semantic features are added to the original feature set. We do not have Gold Standard coreference annotation to be used for evaluation purposes. Also, since the pronoun resolver is built into our WSD system, it is difficult to evaluate independently.

2.3.3 Verb-specific sentential complements

Different types of sentential complements can be very useful for distinguishing certain verb senses. For example, (3a-b) show two sentences containing the verb *call* in the SENSEVAL-2 training data. *Call* has WordNet Sense 1 (*name*) in (3a) and Sense 3 (*ascribe*) in (3b). In both cases, *call* takes a small clause as its sentential complement, i.e., it has the subcategorization frame X *call* Y Z. The difference is that Z is a Named Entity when *call* has Sense 1, and Z is usually a common NP or an adjective phrase when *call* has Sense 3.

(3) a. The slender, handsome fellow was $\underline{\text{called}}_{\text{verb}}$ [S Dandy Brandon].
b. The White House is purposely not $\underline{\text{calling}}_{\text{verb}}$ [S the meeting a summit]...

Our original system used a single feature *hasSent* to represent whether the target verb has a sentential complement or not, which cannot capture subtle distinctions that are crucial to distinguishing certain verb senses but are deeply embedded in sentential complements, as described above. Therefore, we treat sentential complements in a more fine-grained, verb-specific way. For example, for the verb *call* in (3a), we have a feature *sentc_sc_nameentity* that indicates this verb takes a small clause sentential complement and Z (in X *call* Y Z) is a Named Entity. We resort to WordNet and PropBank for information about verb subcategorization frames. Another advantage of this verb-specific treatment is that it can filter out illegal sentential complements generated by the parser.

With the above three enhancements in the generation of linguistically motivated features, our system achieved higher performance than previously published best results (Dang and Palmer 2005; Lee and Ng 2002) in an evaluation using the SENSEVAL-2 English verbs with fine-grained senses (64.6% accuracy; 16.7 senses on average, based on WordNet 1.7 sense distinctions) (Chen and Palmer 2005). Further experiments indicate that the three enhancements are each individually beneficial (Chen and Palmer 2005). Since the SENSEVAL-3 data were collected over the internet and had a relatively low quality of annotation, we did not evaluate our system on that data set.

3 Towards large-scale high-performance word sense disambiguation

As introduced in Sect. 1, our major goal is to build a high performance system for large-scale WSD. We employ supervised learning with linguistically motivated features for high-performance WSD. To create large-scale sense-annotated high-quality training data, we are participating in a large-scale annotation effort that is based on grouping subtle, fine-grained WordNet senses into coherent semantic sense groups that can be readily distinguished by human annotators.

3.1 Creating a sense inventory that supports high quality annotation

The GALE OntoNotes large scale sense-annotation project has been under way for the past 3 years (Duffield et al. 2007). Our goal in this project is to create verb sense distinctions at a middle level of granularity in between fine-grained WordNet senses and syntactically based PropBank Framesets that allow us to capture as much information as possible from a lexical item while still attaining high ITA scores and high system accuracy. Building on results in grouping fine-grained WordNet senses into more coarse-grained senses that led to improved inter-annotator agreement and system performance (Palmer et al. 2004, 2007), we have developed a process for rapid sense inventory creation and annotation that also provides critical links between the grouped word senses and the Omega ontology (Philpot et al. 2005).

This process is based on recognizing that sense distinctions can be represented by linguists in a hierarchical structure, similar to a decision tree, that is rooted in very coarse-grained distinctions (PropBank Framesets) which become increasingly fine-grained until reaching WordNet (or similar) senses at the leaves. Sets of senses under specific nodes of the tree are grouped together into single entries, along with the syntactic and semantic criteria for their groupings, to be presented to the annotators. Every new coarse-level grouping created in this process has to be tested empirically by passing sample annotation in order for the clarity of the sense distinctions to be validated. The criterion for passing is roughly 90% inter-annotator agreement. ITA scores below 90% lead to a revision and clarification of the groupings by the linguist. It is only after the groupings have passed the ITA hurdle that each individual group is combined with others with the same meaning and specified as a conceptual node in the ontology. In addition to higher accuracy, we find at least a three-fold increase in annotator productivity. Each instance is tagged by two taggers and all the disagreements are adjudicated by a third tagger. To date, more than 1,400 verbs have been manually sense-grouped and sense-tagged under this project (Hovy et al. 2006). The annotated instances were drawn from the following sources: the treebanked Wall Street Journal, the Brown corpus, the English-Chinese Parallel Treebank corpus, the English Broadcast News corpus, and the English Broadcast Conversation corpus.

In the grouping process fine-grained sense distinctions listed in WordNet 2.1 (now WordNet 3.0) are collected into more coarse-grained groupings based on syntactic and semantic criteria, following standard lexicographic practice. For instance, for the verb ***call***, Sense 1: *I* ***called*** *my son David*, and Sense 12: *You can* ***call*** *me Sir* are grouped together. Other resources, including PropBank, VerbNet (based on Levin's verb classes, Levin 1993) and online dictionaries, are consulted for insights into syntactic and semantic similarities (Palmer et al. 2004, 2007; Kipper et al. 2006). As an aid to annotators, sense groupings are ordered according to saliency and frequency. Detailed comments about distinctions between the groups, including syntactic frames and semantic features as discussed below, are explicitly provided for each group. Several example sentences from WordNet or Google search results are included for further clarification of the sense groupings.

3.1.1 Syntactic criteria

Annotators have found syntactic frames, such as those defining VerbNet classes, to be useful in clarifying boundaries between sense groupings. For example, *split* was originally grouped with consideration for the units resulting from a splitting event (i.e. whether a whole unit had been split into incomplete portions of the whole, or into smaller, but complete, individual units.) This grouping proved difficult for annotators to distinguish, with a resulting ITA of 42%. Using the causative/inchoative alternation for verbs in the "*break-45.1*" Levin class to regroup the verb senses accordingly, for example, grouping together: *John* ***split*** *the log*/*The log* ***split***, resulted in higher consistency among annotators, increasing the ITA score to 95%. The presence or absence of particular types of prepositional phrases can also be

useful, easily detected criteria for separating two different sense groups (Duffield et al. 2007).

3.1.2 Semantic criteria

Similar semantic features of specific verb arguments, such as [+/−attribute], [+/−animate], and [+/−locative] are also frequently used to group senses together or to clarify distinctions. These semantic features are often associated with particular syntactic constituents, so correctly detecting the syntactic constituents is a necessary prerequisite. For example, separate senses are often distinguished by the presence of an [+animate] AGENT as the subject of one, and an [−animate] "event" or "force" as the subject of the other. For a more detailed discussion of grouping criteria, see (Duffield et al. 2007)

The annotation process begins with fifty sample sentences being given double blind annotation, and if an ITA rate of 90% or above is achieved, the verb entry is considered complete. The rest of the instances are then given double blind annotation and adjudication. Groupings that receive less than 90% ITA scores are re-grouped and re-annotated. It is sometimes impossible to get ITA scores over 85% for high frequency verbs that also have high polysemy and high entropy. These have to be carefully adjudicated to produce a gold standard. One technique for lowering the cognitive load on the annotators is to split off all verb–particle constructions and multi-word expressions involving a particular verb. Several of our most polysemous verbs have two distinct sets of groupings, one for the bare verb and one for the multi-word expressions. The final versions of the sense groupings are mapped to VerbNet and FrameNet as well as being linked to the Omega Ontology (Philpot et al. 2005).

Verbs are selected based on frequency of appearance in the WSJ corpus. The 740 most frequent verbs were grouped first. They have an average polysemy of 7 senses in WordNet which is reduced to 3.75 by grouping. The 217 verbs used in our experiments, which have fairly high frequency, have an average WordNet polysemy of 10.4 which reduces to 5.1. The WordNet senses of these verbs range from 59 to 2 senses per verb, and the groups range from 16 to 2. In addition to reducing polysemy, the clear, explicit criteria for sense distinctions improve annotator productivity up to three-fold (Palmer et al. 2004). The instances for SemEval-2007 come from the same annotated data source, and there is a 44 verb overlap between the two datasets. Our data set is larger, 217 compared to 65, and has similar amount of instances per verb.

3.2 Experimental results

We evaluate our system on two different verb datasets, both of which come from the OntoNotes project. The first one consists of the verbs that had at least 50 annotated and adjudicated instances. This amounted to a total of 217 verbs and 35,210 instances. We preprocessed the resulting corpus and ran our feature extraction module to derive features for each of these instances as described in Sect. 2. The

second set is the SemEval-2007 Lexical Sample task verb dataset, also from OntoNotes, consisting of 65 verbs and 11,280 total instances, and discussed in Sect. 3.2.2.

In both evaluations, we used the machine learning software Mallet (McCallum 2002) to build the smoothed MaxEnt model with a Gaussian prior. Based on our previous experience in using this model for nominal entity extraction, we chose the default value provided by Mallet, 1, as the value of the Gaussian prior variance parameters.

3.2.1 Experiments with the 217 OntoNotes coarse-grained English verbs

Because the complete 217 verb corpus used in the first experiments has only recently been created by our team, and thus is new to the WSD community, no other WSD systems have been evaluated against it in the past. However, there is an overlap with the SemEval-2007 data, as discussed subsequently. Therefore, we compared the performance of our system against the most frequent sense baseline in which all instances were labeled with the most frequent sense of the verb. In the process of annotating our corpus, we collected the ITA rates, which reflected the percentage of instances where both annotators agreed in their choice of senses. Because a machine learning system rarely exceeds the performance of a human annotator, ITA can be viewed as a natural way of comparing the performance of an automatic WSD system to human tagging.

In our experiments, a separate model was built for each verb. Five-fold cross validation was used for testing, where a different 20% set of the instances provides the test data each time. Effectively every instance is also a test instance at some point, so the number of test instances equals the total number of instances. The accuracy reported is on the entire set of instances. Table 1 shows the experimental results. In addition to giving the five-fold cross-validation accuracy for the smoothed MaxEnt model (column 6), the baseline accuracy (column 5) and the ITA

Table 1 Performance of the MaxEnt system for the 217 OntoNotes English verbs

Verb	Polysemy	Sense entropy	# of instances	Baseline accuracy	ME accuracy	ITA
Base	3	0.6403	121	0.6612	0.9835	0.9237
Try	5	0.2751	157	0.9299	0.9809	0.9623
Occur	4	0.5166	88	0.8182	0.9773	0.8978
Maintain	4	0.3145	147	0.9048	0.9728	0.9864
Tell	4	0.2201	513	0.9474	0.9727	0.9844
Stand	7	1.4351	324	0.3735	0.5772	0.6344
Extend	4	1.0679	66	0.4242	0.5758	0.7714
Count	4	1.3330	56	0.3393	0.5714	0.9464
Play	12	1.9377	124	0.3387	0.5323	0.4138
Draw	11	1.9206	146	0.2260	0.5000	0.7105
Average	5.1	0.8328	162	0.6803	0.8272	0.8253

(column 7), the table also provides the number of grouped senses (column 2), the number of instances (column 4), and sense entropy (column 3) which was calculated by Eq. 4:

$$-\sum_{i=1}^{n} P(sense_i) \log P(sense_i) \tag{4}$$

where n is the total number of senses in our data set; $P(sense_i)$ is the probability of the ith sense of the verb, which is estimated based on the frequency count of the senses in our data set.

Due to space limitations, we only show the five verbs with the highest system accuracies and the five verbs with the lowest system accuracies. The last row gives the average numbers (weighted by the number of instances) for all 217 verbs. As we can see from the table, our WSD system beat the most frequent sense baseline by a wide margin of at least 14 percentage points. Furthermore, its performance is slightly above the ITA rate, although the difference is not statistically significant.[3] The fact that there is no statistical significance between the performance of our WSD system and the ITA can be interpreted as meaning that the performance of our supervised WSD system is comparable to human tagging.

It is worth mentioning that Zhong et al. (2008) has reported a high accuracy (89.1%) that their system achieved in a large-scale WSD experiment with 685 OntoNotes words (including both nouns and verbs), which approached the ~90% ITA rate of the OntoNotes project.

3.2.2 Experiments with SemEval-2007 coarse-grained English verbs

Our system performance in the cross-validation setting could be better than that obtained in a typical train-test setting due to a more balanced instance distribution over verb senses in the former setting. To test our system in a typical train-test setting, we also ran it on the 65 verbs from the SemEval-2007 coarse-grained English lexical sample task, and the performance is still competitive. This allows us to compare our system with the top-ranking systems participating in this evaluation task. Table 2 summarizes the experimental results. Pradhan et al. (2007) provides the average scores (unweighted by instance frequencies) of the top 8 systems on the 65 verbs and their scores on each of the 43 selected "difficult" verbs (Table 4 in their paper). The unweighted score of our system on the 65 SemEval-2007 verbs is 83.5%, higher than the top ranking system NUS-ML (78%, Column 5 in Table 2). The weighted score of our system (84.4%) is comparable (i.e., the difference is not statistically significant) to NUS-ML (84.2%) (Cai et al. 2007). For the 43 selected difficult verbs, our system performance is also comparable to the top system, with an unweighted score 75.7% (vs. 76.4%) and weighted score 74.2% (vs. 74.0%).

[3] The significance test mentioned here and in later discussions is a statistical hypothesis test, where the two systems (sysA and sysB)' run on N test instances were regarded as two experiments each consisting of N independent trials. The null hypothesis is $p_A = p_B$, where p_X is sysX (X = A or B)'s error rate (i.e., estimated probability of making mistakes based on N observations). The significance test script we used was *signif* written by Jeff Bilmes from UC Berkeley (1996).

Table 2 Performance of the MaxEnt system for SemEval-2007 English verbs

Verbs	No. of training instances	No. of test instances	NUS-ML		MaxEnt	
			Weighted score	Unweighted score	Weighted score	Unweighted score
65 Verbs	8988	2292	84.2	78.0	84.4	83.5
43 Selected verbs	5521	1336	74.0	76.4	74.2	75.7

Among these 43 verbs, our system achieved the best performance on 15 verbs (compared with the 8 systems mentioned above).

Compared with NUS-ML, our system used more linguistically motivated features, such as those representing verbs' sentential complements and the semantic categories of the NP arguments of verbs and prepositions. Without using these rich features, our system's performance decreased by 2.4%, from 84.4% (weighted score; see Table 2) to 82.0% (not shown in Table 2). The performance drop is significant ($p < 0.02$). NUS-ML used latent dirichlet allocation to reduce the data sparseness problem that occurs when generating topical features for capturing global context information. We expect using this technique could further improve our system performance.

3.3 Discussion

In this section, we discussed the impact of grouping senses and using linguistically motivated features, and provide a more detailed analysis of our system performance based on the experimental results for the 217 OntoNotes English verbs.

3.3.1 Impact of grouping senses

Table 3 compares the data (our system accuracy, ITA, the most frequent sense baseline and the error reduction rate etc.) in our experiments using the 217 English verbs to those in our previous experiments with the SENSEVAL-2 verbs (Chen and Palmer 2005). As we can see (Column 5 in Table 3), the $error\ reduction\ rate\left(=\frac{System\ Acc.-Baseline\ Acc.}{1-Baseline\ Acc.}\right)$ for our system is 40.3% for the SENSEVAL-2 data with fine-grained senses and improves to 45.9% for the OntoNotes data with coarse-grained senses. We also investigated our system performance on two subsets of the data: one contains verb instances with the most frequent sense (called *MFS*) and the other contains the rest of the verb instances

Table 3 Baseline accuracy, system accuracy, error reduction rate, ITA, subset accuracies for two datasets with different sense granularities

Data set	Baseline accuracy	System accuracy	ITA	Error reduction rate	Acc-MFS	Acc-other
SENSEVAL-2	0.407	0.646	0.713	0.403	0.932	0.449
OntoNotes	0.680	0.827	0.825	0.459	0.955	0.558

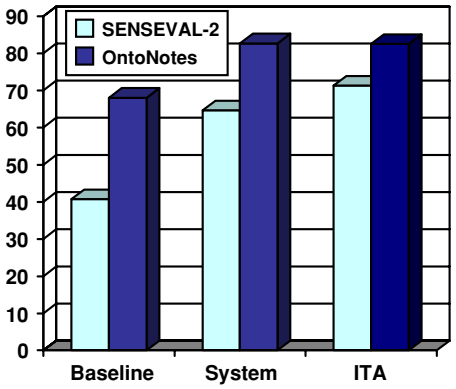

Fig. 3 Baseline accuracy, system accuracy and ITA of Senseval-2 data (fine-grained senses) vs. OntoNotes data (coarse-grained senses)

(with other senses; called *Other*). The results are given in Columns 6 (*Acc-MFS*) and 7 (*Acc-other*) in Table 3. Compared with the fine-grained SENSEVAL-2 verbs, our system's performance on the two subsets for the coarse-grained OntoNotes verbs both improved. The fact that the improvement of *Acc-other* is significant and is greater than that of *Acc-MFS* suggests that using coarse-grained senses increased the capability of our WSD system to classify non-dominant senses. This is a major reason why the *error reduction rate* of our system improves for coarse-grained OntoNotes verbs.

As shown in Table 3 and Fig. 3, the system performance improves by 18% (absolute gain) when using coarse-grained senses. This performance gain is lower than the improvement in the baseline accuracy (27%) and higher than the improvement in the ITA (9%). This result implies that less complicated learning methods (e.g., the most frequent sense heuristic) for WSD could benefit more from adopting more coarse-grained and therefore clearer sense distinctions. However, the fact that our system performance is comparable to that of humans is still very impressive. On the other hand, since the tagger (annotator) accuracy compared to the Gold Standard is usually higher than ITA,[4] there is still room for improvement in our system performance.

It is also worth mentioning that the sense groups are significantly more fine-grained than PropBank and map readily to VerbNet/FrameNet. So, they are still preserving important sense distinctions and are intended to provide an appropriate level for making semantic generalizations (Yi et al. 2007).

Navigli (2006) reported that the accuracy of the best system in the SENSEVAL-3 English all-words task improves by 12% (absolute gain) by using coarse-grained senses produced through mapping WordNet senses to sense hierarchies of the Oxford Dictionary of English. In our experiments, we focused on English verbs and used different methods for grouping WordNet senses. Despite these differences,

[4] In our case, the ITA is the number of instances whose senses were agreed upon by two annotators divided by the total number of annotated instances. The Gold Standard consists of senses adjudicated by the third person (especially for those disagreed upon by the two annotators). Since the adjudicated sense was usually the choice of one of the annotators, the tagger (i.e., annotator) accuracy compared to the Gold Standard is usually higher than the ITA.

both results are consistent and indicate that the accuracies of WSD systems benefit significantly from well-defined coarse-grained sense distinctions. Two coordinated SemEval-2007 tasks compare the coarse-grained choices of these approaches (Pradhan et al. 2007; Navigli et al. 2007).

3.3.2 Impact of linguistically motivated features

To investigate how much linguistically motivated features contributed to the high accuracy our system achieved on the OntoNotes verb data used in our experiments, we trained and tested our system by using three different feature sets: all the features (ALL); without semantic features (w/o SEM); and without semantic and syntactic features (w/o SEM + SYN). The semantic and syntactic features were listed in Sect. 2.3. We did not create a testing set for all the features without syntactic features (w/o SYN) because most of the semantic features we used, i.e., the WordNet synsets and hypernyms of head nouns of the NP arguments of verbs and prepositions, are dependent on syntactic features. If a verb instance has such semantic features, this implies that this verb has certain syntactic features, e.g., it has a subject, direct object or a PP adjunct. Therefore, we cannot get a pure feature set that includes only our semantic features but not syntactic features. Table 4 gives the results. The differences between accuracies are all significant ($p < 0.0001$).

Among the 217 verbs, the system has increased performance on 179 verbs by using linguistically motivated features and decreased performance on 36. By looking at the sense definitions as well as the mistakes made by the system on the 36 verbs, we identified two major reasons for decreased performance. First, the semantic and syntactic features can be insufficient for distinguishing the major senses of a verb. For example, the verb *treat* has two major senses in our corpus: ***treat1***: "interact with, behave towards" (as in *He* ***treats*** *his customers kindly*) and ***treat2***: "subject to a process, often chemical" (as in *The doctor* ***treated*** *her for burned tissue*). Both senses have *subject + verb + direct object* as their typical syntactic structures. In our corpus, *persons* are typical direct objects for both senses (in practice, both senses can often take inanimate direct objects). Therefore, the semantic and syntactic features we used (e.g., the semantic categories of a verbs' direct object; whether the verb is intransitive in its given context) provide few clues for distinguishing these two senses. The second reason is due to the inaccuracies in automatic preprocessing (POS tagging, parsing, named entity finding etc.) and in feature extraction. A more detailed discussion in this aspect can be found in (Chen et al. 2006).

The two reasons mentioned above also partially account for why our system did not get much improvement from using linguistically motivated features on coarse-grained senses with clearer sense boundaries than on fine-grained senses. In

Table 4 System accuracy with different feature sets for the 217 OntoNotes English verbs

	ALL	w/o SEM	w/o SEM + SYN
System accuracy	0.827	0.816	0.789

addition, by grouping fine-grained senses together for verbs, different senses of a verb, in some cases, could have more overlapping linguistically motivated feature values. For example, the verb *extend* is usually transitive in some of its WordNet fine-grained senses (e.g., WN2, WN11, WN16) and intransitive in some other WordNet senses. However, with either one of its two major grouped senses, the verb can be both transitive and intransitive. Therefore, the syntactic feature *transitivity* is less useful for distinguishing this verb's coarse-grained senses. It is worth mentioning however, that though the post processing manual analysis described above gives us rich insights into why certain verbs do not benefit from using linguistically motivated features, reliably predicting beforehand which verbs could or could not benefit from these rich features is still a difficult, unsolved problem that needs additional exploration (Chen 2006, Chap. 5).

3.3.3 Predicting WSD system performance

A WSD system performance (accuracy and recall) on a set of target words can be affected by many factors, such as characteristics of its learning algorithm, language specificity (i.e., whether the system is designed for WSD of a specific language), domain specificity (i.e., whether the system is designed for WSD in a specific domain or general domains), the sense entropies of target words, the number of training instances, etc.

In this section, we investigate the predictive capabilities of six factors: polysemy, number of instances, average number of instances (the number of instances per sense), sense entropy, baseline accuracy, and ITA of a verb. The two quantitative methods we used for this purpose are analysis of the correlation coefficient (Pearson product-moment correlation coefficient) and linear regression. In the data analysis, each of the above factors is treated as a predictor variable and the accuracy of our MaxEnt WSD system as the response variable. Since we test the system accuracy on 217 verbs, we have 217 observations (samples).

Table 5 (Row 2) gives the calculated correlation coefficient between each predictor variable and the response variable. As we can see, the *sense entropy* has the highest correlation with the *system accuracy*, followed by the *baseline*, *ITA* and *polysemy*. Note that with respect to *sense entropy* and *polysemy*, the correlation is actually negative, so the higher the entropy the lower the performance (same for *polysemy*). In contrast, the correlation between the *average number of instances* and the *system accuracy* is very low. And there is almost a lack of correlation between the *number of instances* and the *system accuracy*. However, the low correlation

Table 5 Correlation coefficient between the response variables (system accuracy, system accuracy improvement) and the predictor variables (polysemy, the number of instances, the average number of instances, baseline, ITA and sense entropy)

	Polysemy	No. of instances	Ave. no. of instances	Baseline	ITA	Sense entropy
System accuracy	−0.4518	0.0228	0.2753	0.7462	0.5758	−0.8093
System_Acc_Imp	0.2389	0.0124	−0.1316	−0.8555	−0.1675	0.7158

between the *(average) number of instances* and the *system accuracy* is not so surprising because, when we collected data for our experiments, we chose verbs that had at least 50 annotated and adjudicated instances so that the WSD system was expected to have enough training data for most verbs. Therefore, the low correlation value here might suggest that, when the number of observations (verb instances in our case) is enough (e.g., >50 in our case) for most verbs, increasing the number of observations won't boost system performance too much. Further experiments and analysis need to be conducted to verify this hypothesis.

The high correlation between the *system accuracy* and the *sense entropy* is not only due to the fact that we used the maximum entropy learning algorithm for our WSD system. In fact, we tested the correlation coefficient between the above six factors and the accuracies of our system when using other learning algorithms such as linear SVM and AdaBoost with Decision trees as base classifiers.[5] The results are all similar, with *sense entropy* having the highest correlation with *system accuracy*, followed by *baseline*, *ITA* and *polysemy*. This suggests that the *sense entropy* is likely to be a good, robust predictor for the accuracy of a WSD system that uses machine learning for sense disambiguation. These results are quite predictable, since a word with a strongly predominant sense will have very low entropy and a high baseline, whereas a word with several evenly distributed senses will have a much lower baseline and much higher entropy. Such a word is also more likely to cause confusion among both human and automatic taggers.

The predictor variables are not completely independent of each other in our analysis. In fact, some of them are highly correlated (e.g., the *sense entropy* and the *baseline*). Table 6 shows the correlation coefficient among these variables. We used an exploration-like method to choose predictor variables to best fit a linear regression model for the response variable, i.e., the *system accuracy*. The basic idea is as follows. Assume we have n variables $\{x_1, x_2, x_3, \ldots, x_n\}$, which we can choose as predictor variables for a linear model for y (the *system accuracy* in our case). We first build n single-variable linear regression models and choose the linear model, $model_1$, which best fits y (with highest R^2). We assume the predictor variable used by this model is x_i (the *sense entropy* in our case). We then build *n-1* linear models which use $\{x_i\} \times \{x_j \mid j \neq i\}$ as predictor variables and again choose the one that best fits y, $model_2$. After all x variables have been used, we select the model with the largest adjusted R^2. A shortcoming of R^2 is that it won't decrease when adding a predictor variable in the linear regression. An adjusted R^2 overcomes this shortcoming in that it can decrease in value if the contribution to the explained deviation by the additional variable is less than the impact on the degrees of freedom (Lucke and Embretson 1984). By using the method described here, we obtain the following linear regression model:

$$\begin{aligned} \textit{System Accuracy} = {} & 0.786 - 0.195 \times \textit{Sense Entropy} + 0.210 \times \textit{ITA} + 1.069 \times 10^{-4} \\ & \times \textit{Number of Inst}. \end{aligned}$$

[5] For more details about the systems developed with these other learning algorithms see Chen et al. (2007).

Table 6 Correlations between predictor variables

	Polysemy	No. of instances	Ave. no. of instances	Baseline	ITA	Sense entropy
Polysemy	1.0000	0.2960	−0.2955	−0.3487	−0.2859	**0.6209**
No. of instances		1.0000	**0.6830**	−0.0107	−0.1090	0.1173
Ave. no. of instances			1.0000	−0.2042	0.0632	−0.2571
Baseline				1.0000	0.4225	**−0.9042**
ITA					1.0000	**−0.5083**
Sense entropy						1.0000

Table 7 Best-fit linear regression model in each step of model exploration

Predictor variable added in each step	Step 1 Sense entropy	Step 2 ITA	Step 3 No. of instances	Step 4 Baseline accuracy	Step 5 Ave. no. of instances	Step 6 Polysemy
R^2	0.6549	0.6914	0.7083	0.7089	0.7089	0.7089
Adjusted R^2	0.6533	0.6885	**0.7041**	0.7034	0.7021	0.7006

with an adjusted R^2 of 0.7041 ($F = 172.4$, DF = (3,213), $p < 2.2 \times 10^{-16}$). Table 7 gives the R^2 and adjusted R^2 of models selected in each step. Figure 4 gives the corresponding learning curves.

In a later experiment, we also used the relative difference between the *system accuracy* and the *baseline* as the response variable, called *Relative_Acc_Improvement*, to investigate the predictive power of the above six factors. The results are also quite predictable. As we see in Row 3 of Table 5, there is a high, positive correlation between the *sense entropy* and the *Relative_Acc_Improvement* in our case. High entropy corresponds to a low baseline, which gives substantial head room for improving accuracy. It is worth noting that a low (Pearson product-moment) correlation coefficient value only suggests the lack of a linear relationship between two variables, such as the *Relative_Acc_Improvement* and the *ITA* in our case. It is possible that these two variables depend on each other in a non-linear way. By using the same method mentioned above, we obtained a linear regression model for *Relative_Acc_Improvement*:

$$\begin{aligned} System\_Acc\_Improvement = {} & 1.232 - 1.914 \times Baseline + 0.551 \times ITA - 0.155 \\ & \times Sense\ Entropy + 1.085 \times 10^{-4} \times Number\ of\ Inst. \end{aligned}$$

with an adjusted R^2 of 0.7804 ($F = 192.9$, DF = (4,212), $p < 2.2 \times 10^{-16}$).

3.3.4 Verbs with extremely low or high system accuracy

The correlation analysis in Sect. 3.3.3 suggests that *sense entropy*, *baseline accuracy* and *ITA* are all good predictors of system performance. In practice, *ITA* (or

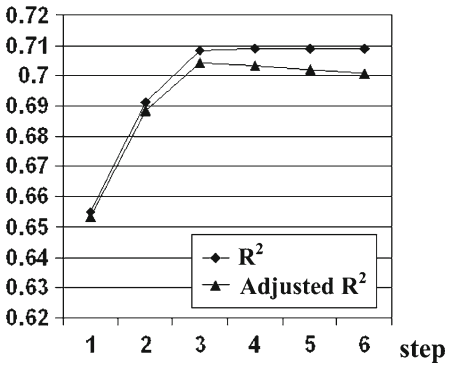

Fig. 4 Learning curves for finding a best-fit linear regression model for system accuracy

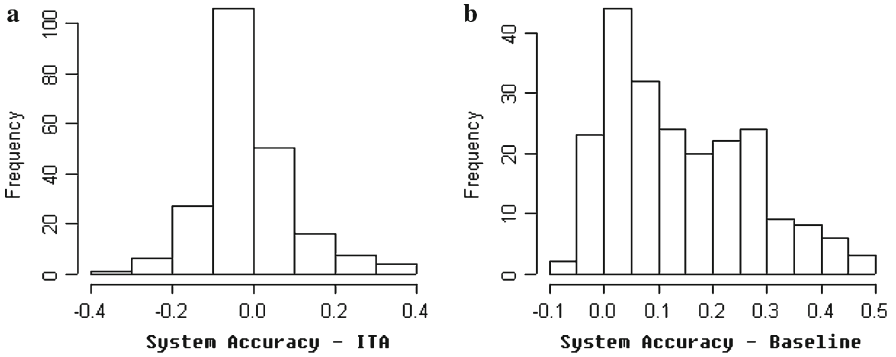

Fig. 5 Frequency distributions of difference between system accuracy and ITA and difference between system accuracy and baseline

tagger-Gold Standard agreement) is usually the upper bound of the system performance (i.e., humans are expected to do better than machines in sense disambiguation) and *baseline accuracy* is the lower bound. In this section, we discuss our analysis results on verbs with unexpectedly low or high system accuracies. The average ITA for the OntoNotes verbs is 89%, but there are still individual verbs with much lower (and much higher) ITAs. The high frequency verbs used in our experiments tend to have lower ITAs. Figure 5 shows the frequency distribution of the difference between the *System accuracy* and the *ITA*[6] (*system accuracy – ITA*, Fig. 5a), and the difference between the *System accuracy* and the *baseline accuracy* (*system accuracy – baseline*, Fig. 5b).

[6] We did not use tagger Gold Standard agreement for our analysis because the data we used were created in summer 2007, for which we did not calculate that number. The current OntoNotes data contains more instances for the 217 verbs and redefined sense groupings for some verbs that had extremely low ITA's before, so the analysis could not be redone. However, we expect that doing a comparison with tagger-Gold Standard rates instead would give very similar results, since the distribution is quite similar and the agreement rate is on average 7% higher (based on the current OntoNotes data for the 217 verbs).

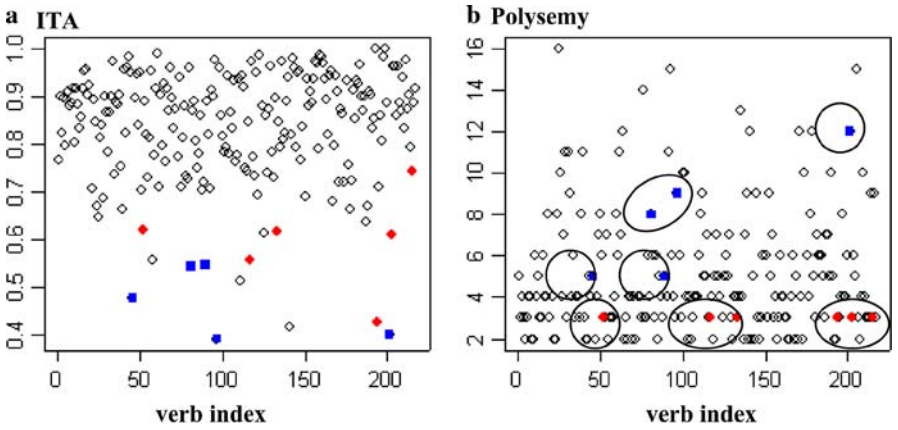

Fig. 6 Analysis of verbs with high accuracies

In our analysis, we regarded verbs with system accuracies higher than their ITAs by 0.2 or more as verbs with unexpectedly high accuracies. A total of 11 verbs were selected, including *continue*, *decide*, *fill*, *gain*, *have*, *let*, *operate*, *suffer*, *throw*, *trade*, and *worry*. Our analysis suggests that these verbs have unusually low ITAs (solid diamonds and squares in (Fig. 6a).

The lower ITAs can be caused by several factors such as high polysemy, lack of clarity between sense boundaries, and the misunderstanding of a sense by a human annotator. In Fig. 6a and b, we marked verbs with high polysemies (with 5 or more senses, including *continue*, *fill*, *gain*, *have*, and *throw*) by solid squares. Figure 6a shows that these verbs generally have extremely low ITAs. We will reexamine the sense definition of these verbs in the future and may merge certain sense groups if appropriate. The other verbs *decide*, *let*, *operate*, *suffer*, *trade*, and *worry* have only 3 senses yet still have low ITAs. A close look at the annotation data indicates that the two annotators often disagree on the first two senses of these verbs. Some of the senses simply have unclear boundaries, such as ***suffer1***: "endure or experience a painful event" and ***suffer2***: "become worse or decline, experience negative effects" which both deal with negative outcomes. The sense entries specify that ***suffer1*** is for animate, sentient beings suffering, and the sufferer in ***suffer2*** is NOT supposed to be animate, i.e., to be able to experience physical or emotional pain. ***Operate*** and ***let*** are similar, in that the main difference between the first 2 senses is that one must have an animate agent which the other does not have to be. If a careless or hurried annotator misses that single point, the senses are easy to confuse. However, recognizing that the animacy of the subject is an important feature and needs to be applied consistently is straightforward for machine learning. On the other hand, even though the entries may seem quite distinct, the instances can still be difficult to distinguish in practice. Does "*But the size, shape and role of the NSC staff have been left for each president and his national security adviser to* ***decide***," fit the "make a choice" sense of ***decide***, or the "settle, resolve or adjudicate" sense? The case is similar for ***trade***, where the difference has to do solely with whether a

specific event or a general type of activity is being described, a clear distinction which can still sometimes be hard to make. ***Worry*** is in a class by itself. Its first two senses, "be anxious or preoccupied", and "repeatedly handle or manipulate something", are very clearly separated. However, the adjudicated instances indicate that every instance marked sense 2 actually belongs in sense 1 (with the exception of 2 that belong in sense 3), a surprising case of systematic, shared annotator error.

We used two methods to find verbs with unexpected low accuracies. For the first method, we chose verbs with system accuracies lower than their ITAs by 0.2 or more (see Fig. 5a). Verbs selected in this way include: *back, catch, count, defend, draw, order* and *treat*. Our analysis results suggest that these verbs tend to have low baselines (<0.8) and a small average number of instances per sense (<20). Due to space limitations, we omit the figure representations similar to Fig. 6 for our data analysis using this method and the second method as described below.

For the second method, we chose verbs with system accuracies lower than their baseline accuracies (see Fig. 5b). Verbs selected in this way include: *complain*, *defend*, *force*, *order*, *ride*, *seem*, *stay*, *talk* and *throw*. These verbs can be further divided into two sets based on their baseline accuracies: verbs with baseline accuracies higher than 0.8 (set I: *complain*, *force*, *seem*, *stay* and *talk*) and verbs with baseline accuracies lower than 0.8 (set II: *defend*, *order*, *ride* and *throw*). Our analysis results suggest that the verbs with lower baseline accuracies and even lower system accuracies (set II verbs) tend to have a small average number of instances per sense (<20). The verbs selected by both methods (except set I verbs selected by method 2) are verbs with high priorities for which we'd like to improve our system accuracies further in the future by increasing sense annotated data and/or improving the feature sets used by our machine learning model for these verbs.

3.3.5 WSD system vs. human annotators

Our experiments show that the average overall accuracy of our WSD system is very similar to that of humans tagging the same data. However, there are interesting differences with respect to which types of verbs are most challenging for humans vs. which ones are most challenging for the system. The analysis on verbs with system accuracies higher than their ITAs, as discussed in Sect. 3.3.4, suggests that the system seems to be able to perform more consistently than humans, although it cannot access many types of world knowledge that humans can access. As a result, verb sense distinctions that respond well to meticulously consistent application of criteria tend to get high performance from the system, while human annotators could make mistakes on these senses due to carelessly missing some important distinguishing points (a human considers many more types of information than the machine at a given point in time). When sense distinctions rely on subtle interpretations, world knowledge or information from a broad context, they tend to get lower system performance (such as the verb *treat* discussed in Sect. 3.3.2). An instance-level analysis showed that, among the instances whose senses were disagreed upon by two annotators, 57% were predicted correctly by the system. Among the instances predicted by the system incorrectly, 65% were agreed upon by the annotators. This result also indicates that system and human do differ in their

disambiguation performance at more fine-grained levels (i.e., verbs and verb instances), though their overall average performances are similar.

4 Conclusions and future work

This paper presents our efforts at achieving large-scale high-performance verb sense disambiguation. Our supervised WSD system uses linguistically motivated features and a smoothed MaxEnt model for machine learning. Many of these features have also been used by other successful supervised systems (Lee et al. 2004; Agirre and Edmonds 2007). Our data analysis showed that using linguistically motivated features such as semantic features helped to relieve the data sparseness problem (characterized by the *data sparseness rate*, the *feature activation frequency* and the number of OOV features). We further enhanced our system's treatment of the linguistically motivated features in three specific ways by using linguistic knowledge and rules and automatic pronoun resolution. With these three enhancements our system achieved higher performance than previously published best results in an evaluation using the SENSEVAL-2 English verbs with fine-grained senses. Further experiments suggested that all the three enhancements improved performance (Chen and Palmer 2005).

Supervised WSD tasks generally suffer from an insufficiency of sense-tagged training data. Very fine-grained sense distinctions often cause low inter-annotator agreement and slow down the annotation of large-scale corpora. We address this problem by grouping fine-grained WordNet senses into sense groups with clearly explicated sense distinctions and annotating large numbers of instances with these coarse-grained grouped senses. Following standard lexicographic practice, the groupings are defined by using syntactic and semantic criteria. The automatic system features have been explicitly chosen to capture the same types of information (Palmer et al. 2007).

We evaluated our system performance on 217 verbs annotated by this approach and also on the 65 SemEval-2007 verbs from the same source. The experimental results show that our system achieved a very high accuracy of 82.7% on the 217 verbs, which is close to our ITA. We also achieved results competitive with the top results for the SemEval-2007 verbs.

Our analysis of the experimental results on the larger set suggests several areas we can explore in the future for improving high-performance WSD. The statistical analysis of factors that affect system performance suggests that sense entropy, ITA, and the most frequent sense baseline accuracy are all good predictors of our system performance. By finding verbs that had system performance much higher than ITA rates (0.2 absolute gain), we identified several sense definition entries that were lacking clear boundaries and could be clarified. The modified sense entries are expected to improve ITA as well as system performance. Analysis of verbs that had very low system accuracies (lower than ITAs by at least 0.2, or lower than their baseline accuracies) suggested that increasing the amount of sense-annotated data for these verbs could boost system performance further. Our previous work in active learning for WSD (Chen et al. 2006) suggests that active learning works well for

learning coarse-grained verb senses. In the future, we will also experiment with applying active learning to improve the productivity of our annotation effort. In addition, we could add to our system the latent dirichlet allocation technique used by NUS-ML for reducing the data sparseness problem that occurs when generating topical features for capturing global context information.

Zhong et al.'s (2008) recent work suggests that WSD systems trained with coarse-grained senses could suffer $\sim 10\%$ performance drop when being applied to a new domain. They also showed that using a domain adaptation technique combined with active learning (i.e., selecting a small amount of data from the new domain to enhance the system's training set) could improve their system performance on the new domain efficiently. In the future, we want to explore this area by testing and increasing our system's capability to extend to different domains.

Acknowledgements We gratefully acknowledge the support of the National Science Foundation Grant NSF-0415923, Word Sense Disambiguation, and Defense Advanced Research Projects Agency (DARPA/IPTO) under the GALE program, DARPA/CMO Contract No. HR0011-06-C-0022, subcontract from BBN, Inc. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

References

Agirre, E., & Edmonds, P. (2007). Word sense disambiguation: Algorithms and applications. *Text, Speech and Language Technology Series* (Vol. 33). Springer, Netherland. ISBN: 978-1-4020-6870-6.

Berger, A. L., Della Piertra, S. A., & Della Pietra, V. J. (1996). A maximum entropy approach to natural language processing. *Computational Linguistics, 22*(1), 39–71.

Bikel, D. M. (2002). Design of a multi-lingual, parallel-processing statistical parsing engine. In *Proceedings of HLT 2002*, San Diego, CA.

Bikel, D. M., Schwartz, R., & Weischedel, R. M. (1999). An algorithm that learns what's in a name. *Machine Learning, 34(1–3). Special Issue on Natural Language Learning*.

Cai, J. F., Lee, W. S., & Teh, Y. W. (2007). NUS-ML: Improving word sense disambiguation using topic features. In *Proceedings of the 4th international workshop on Semantic Evaluations (SemEval 2007)*, Prague, Czech Republic (pp. 249–252).

Carpuat, M., & Wu, D. (2007). Improving statistical machine translation using word sense disambiguation. In *Proceedings of the 2007 joint conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)* (pp. 61–72).

Chan, Y. S., Ng, H. T., & Chiang, D. (2007, June). Word sense disambiguation improves statistical machine translation. In *Proceedings of the 45th annual meeting of the association of computational linguistics*, Prague, Czech Republic. Association for Computational Linguistics (pp. 33–40).

Chen, J. (2006). *Towards high-performance word sense disambiguation by combining rich linguistic knowledge and machine learning approaches*. PhD Thesis, University of Pennsylvania.

Chen, J., Dligach, D., & Palmer, M. (2007). Towards large-scale high-performance English verb sense disambiguation by using linguistically motivated features. In *Proceedings of the international conference on semantic computing (ICSC 2007)*. Irvine, CA.

Chen, J., & Palmer, M. (2005, October 11–13). Towards robust high performance word sense disambiguation of English verbs using rich linguistic features. In *Proceedings of the 2nd international joint conference on natural language processing*, Jeju Island, Korea.

Chen, S. F., & Rosenfeld, R. (1999). A Gaussian prior for smoothing maximum entropy models. Technical Report CMU-CS-99-108, CMU.

Chen, J., Schein, A., Ungar, L., & Palmer, M. (2006). An empirical study of the behavior of word sense disambiguation. In *Proceedings of NAACL-HLT 2006*, NY, 2006.

Dang, H. T. (2004). *Investigations into the role of lexical semantics in word sense disambiguation*. PhD Thesis, University of Pennsylvania.

Dang, H. T., & Palmer, M. (2005, June 26–28). The role of semantic roles in disambiguating verb senses. In *Proceedings of the 43rd annual meeting of the association for computational linguistics*, Ann Arbor, MI.

Dang, H. T., & Palmer, M. (2002). Combining contextual features for word sense disambiguation. In *Proceedings of the SIGLEX/SENSEVAL workshop on WSD: Recent successes and future directions*, in conjunction with ACL-02, Philadelphia.

Duffield, C. J., Hwang, J. D., Brown, S. W., Dligach, D., Vieweg, S. E., Davis, J., & Palmer, M. (2007). Criteria for the manual grouping of verb senses. In *Linguistics annotation workshop, held in conjunction with ACL-2007*, Prague, The Czech Republic.

Edmonds, P., & Cotton, S. (2001). SENSEVAL-2: Overview. In *Proceedings of SENSEVAL-2: 2nd international workshop on evaluating WSD systems*. ACL-SIGLEX, Toulouse France.

Fellbaum, C. (1998). *WordNet—an electronic lexical database*. Cambridge, MA/London: The MIT Press.

Fellbaum, C., Delfs, L., Wolff, S., & Palmer, M. (2005). Word meaning in Dictionaries, corpora, and the speaker's mind. In G. Barnbrook, P. Danielsson, & M. Mahlberg (Eds.), *Meaningful texts: The extraction of semantic information from monolingual and multilingual corpora* (pp. 31–38). Birmingham, UK: Birmingham University Press.

Fellbaum, C., Palmer, M., Dang, H. T., Delfs, L., & Wolf, S. (2001, June 2, 3). Manual and automatic semantic annotation with WordNet. In *SIGLEX workshop on WordNet and other lexical resources (NAACL-01)*, Invited talk, Pittsburgh, PA.

Gonzalo, J., Verdejo, F., Chugur, I., & Cigarran, J. (1998). Indexing with WordNet synsets can improve text retrieval. In *Proceedings of the COLING/ACL'98 workshop on usage of WordNet for NLP*, Montreal, Canada.

Hanks, P. (1996). Contextual dependencies and lexical sets. *The International Journal of Corpus Linguistics, 1*, 1.

Hovy, E., Marcus, M., Palmer, M., Ramshaw, L., & Weischedel, R. (2006). OntoNotes: The 90% solution. In *Proceedings of HLT-NAACL06*, New York.

Ide, N., & Veronis, J. (1998). Introduction to the special issue on word sense disambiguation: The state of the art. *Computational Linguistics, 24*(1), 140.

Kipper, K., Korhonen, A., Ryant, N., & Palmer, M. (2006). Extensive classifications of English verbs. In *Proceedings of the 12th EURALEX international congress*, Turin, Italy.

Lappin, S., & Leass, H. (1994). An algorithm for pronominal anaphora resolution. *Computational Linguistics, 20*(4), 535–561.

Lee, Y. K., & Ng, H. T. (2002). An empirical evaluation of knowledge sources and learning algorithms for word sense disambiguation. In *Proceedings of the conference on Empirical Methods in Natural Language Processing (EMNLP)* (pp. 41–48).

Lee, Y. K., Ng, H. T., & Chia, T. K. (2004). Supervised word sense disambiguation with support vector machines and multiple knowledge sources. In *Proceedings of SENSEVAL-3: Third international workshop on the evaluation of systems for the semantic analysis of text*, Barcelona, Spain (pp. 137–140).

Levin, B. (1993). *English verb classes and alternations: A preliminary investigation*. Chicago: University of Chicago Press.

Lucke, J. F., & Embretson, S. (1984). The biases and mean squared errors of estimators of multinormal squared multiple correlation. *Journal of Educational Statistics, 9*(3), 183–192. doi:10.2307/1165005.

Marcus, M., Kim, G., Marcinkiewicz, M. A., MacIntyre, R., Ferguson, M., Katz, K., et al. (1994). The Penn Treebank: Annotating predicate argument structure. In *Proceedings of the ARPA'94 HLT workshop*.

McCallum, A. K. (2002). *MALLET: A machine learning for language toolkit*. http://www.cs.umass.edu/~mccallum/mallet.

Mihalcea, R., Chklovski, T., & Kilgarriff, A. (2004, July). The Senseval-3 English lexical sample task. In *Proceedings of Senseval-3: The third international workshop on the evaluation of systems for the semantic analysis of text*, Barcelona, Spain.

Navigli, R. (2006, July 17–18). Meaningful clustering of senses helps boost word sense disambiguation performance. In *Proceedings of the 21st international conference on computational linguistics and the 44th annual meeting of the ACL*, Sydney, Australia (pp. 105–112).

Navigli, R., Litkowski, K. C., & Hargraves, O. (2007, June). SemEval-2007 Task 07: Coarse-grained English all-words task. In *Proceedings of SemEval*, held in conjunction with *ACL 2007*, Prague, Czech Republic.

Palmer, M., Babko-Malaya, O., & Dang, H. T. (2004). Different sense granularities for different applications. In *Proceedings of the 2nd workshop on scalable natural language understanding systems (HLT/NAACL 2004)*. Boston, MA.

Palmer, M., Dang, H., & Fellbaum, C. (2007, June). Making fine-grained and coarse-grained sense distinctions, both manually and automatically. *Journal of Natural Language Engineering, 13*(2), 137–163.

Palmer, M., Fellbaum, C., Cotton, S., Delfs, L., & Dang, H. T. (2001, July 5–6). English tasks: All-words and verb lexical sample. In *Proceedings of SENSEVAL-2: Second international workshop on evaluating word sense disambiguation systems*. Toulouse, France.

Palmer, M., Gildea, D., & Kingsbury, P. (2005). The proposition bank: A corpus annotated with semantic roles. *Computational Linguistics, 31*, 1. doi:10.1162/0891201053630264.

Philpot, A., Hovy, E., & Pantel, P. (2005). The omega ontology. In *Proceedings of the ONTOLEX workshop at the International Conference on Natural Language Processing (IJCNLP05)*. Jeju Island, Korea.

Pradhan, S., Loper, E., Dligach, D., & Palmer, M. (2007, June). SemEval-2007 task-17: English lexical sample, SRL and all words. In *Proceedings of SemEval*, held in conjunction with *ACL 2007*, Prague, Czech Republic.

Ratnaparkhi, A. (1998). *Maximum entropy models for natural language ambiguity resolution*. Ph.D. Thesis, University of Pennsylvania.

Sanderson, M. (1994). Word sense disambiguation and information retrieval. In *Proceedings of the 17th International ACM SIGIR*, Dublin, Ireland

Sanderson, M. (2000). Retrieving with good sense. *Information Retrieval, 2*(1), 49–69.

Snyder, B., & Palmer, M. (2004, July). The English all-words task. In *Proceedings of Senseval-3: The third international workshop on the evaluation of systems for the semantic analysis of text*. Barcelona, Spain.

Stokoe, C., Oakes, M. P., & Tait, J. (2003). Word sense disambiguation and information retrieval revisited. In *Proceedings of the 26th annual international ACM SIGIR conference on research and development in information retrieval*, Toronto, Canada.

Yarowsky, D. (1993). One sense per collocation. In *Proceedings of the 5th DARPA speech and natural language workshop*.

Yarowsky, D., Cucerzan, S., Florian, R., Schafer, C., & Wicentowski, R. (2001). The Johns Hopkins SENSEVAL2 system description. In *Proceedings of SENSEVAL-2: 2nd international workshop on evaluating WSD systems*, Toulouse France.

Yarowsky, D., & Florian, R. (2002). Evaluating sense disambiguation across diverse parameter spaces. *Journal of Natural Language Engineering, 8*(4), 293–310.

Yi, S.-t., Loper, E., & Palmer, M. (2007, April). Can semantic roles generalize across genres? In *Proceedings of NAACL 2007*, Rochester, NY.

Zhong, Z., Tou Ng, H., & Chan, Y. S. (2008, October). Word sense disambiguation using OntoNotes: An empirical study. In *Proceedings of EMNLP 2008*, Waikiki, Honolulu, HI.