G_012

Using Ontological Knowledge to Disambiguate Unknown Words in Semantic Tagging

フィンチ アンドリユー†‡ 隅田 英一郎†‡

Andrew Finch Eiichirō Sumita

Abstract

This paper presents a detailed study of the integration of knowledge from hierarchical word ontologies into a maximum-entropy-based tagging model that simultaneously labels words with both syntax and semantics. Our findings show that ontological information can lead to strong improvements in overall system accuracy, and in particular increased accuracy for words not seen in the training data.

2. Introduction

Part-of-speech (POS) tagging has been one of the fundamental areas of research in natural language processing for many years. Most of the prior research has focussed on the task of labeling text with tags that reflect the words' syntactic role in the sentence. In parallel to this, the task of word sense disambiguation (WSD), the process of in which semantic sense the word is being used, has been actively researched. This paper addresses a combination of these two fields, that is: labeling running words with tags that comprise, in addition to their syntactic function, a broad semantic class that signifies the semantics of the word in the context of the sentence, but does not necessarily provide information that is sufficiently fine-grained as to disambiguate its sense. This differs from what is commonly meant by WSD in that although each word may have many "senses" (by senses here, we mean the set of semantic labels the word may take), these senses are not specific to the word itself but are drawn from a vocabulary applicable to the subset of all types in the corpus that may have the same semantics. The problem of how to deal with out of vocabulary word (OOV's) is central to the task of semantic tagging because the single most useful feature in tagging is the identity of the word being tagged. When this word has not occurred in the training data we are deprived of the key information we need to identify its semantics.

In order to mitigate this problem, we draw on research from several related fields, and exploit publicly available linguistic resources, namely the WordNet database [5], and a large corpus of unannotated text. Our aim is to simultaneously disambiguate the semantics of the words being tagged while tagging their POS syntax. We treat the task as fundamentally a POS tagging task, with a larger, more ambiguous tag set. However, as we will show later, the `*n*-gram' feature set traditionally employed to perform POS tagging, while basically competent, is not up to this challenge, and needs to be augmented by features specifically targeted at semantic disambiguation.

†独立行政法人 情報通信研究機構
‡ＡＴＲ音声言語コミュニケーション研究所

3. Related Work

Our work is a synthesis of POS tagging and WSD, and as such, research from both these fields is directly relevant here.

The basic engine used to perform the tagging in these experiments is a direct descendent of the maximum entropy (ME) tagger of Ratnaparkhi [15] which in turn is related to the taggers of Kupiek [6] and Merialdo [11]. The ME approach is well-suited to this kind of labeling because it allows the use of a wide variety of features without the necessity to explicitly model the interactions between them.

The literature on WSD is extensive. For a good overview we direct the reader to Nancy and Jean [13]. Typically, the local context around the word to be sense-tagged is used to disambiguate the sense Yarowsky [19], and it is common for linguistic resources such as WordNet [9][12][14], or bilingual data [8] to be employed as well as more long-range context. An ME-system for WSD that operates on similar principles to our system [17] was based on an array of local features that included the words/POS tags/lemmas occurring in a window of +/-3 words of the word being disambiguated. Lamjiri [7] also contributed an ME-based system that used a very simple set of features: the article before; the POS before and after; the preposition before and after, and the syntactic category before and after the word being labeled. The features used in both of these approaches resemble those present in the feature set of a standard *n*-gram tagger, such as the one used as the baseline for the experiments in this paper.

The semantic tags we use can be seen as a form of semantic categorization acting in a similar manner to the semantic class of a word in the system of Lamjiri [7]. The major difference is that with a left-to-right beam-search tagger, labeled context to the right of the word being labeled is not available for use in the feature set.

Although POS tag information has been utilized in WSD techniques (e.g. Suarez [17]), there has been relatively little work addressing the problem of assigning a part-of-speech tag to a word together with its semantics, despite the fact that the tasks involve a similar process of label disambiguation for a word in running text.

4. Experimental Data

The primary corpus used for the experiments presented in this paper is the ATR General English Treebank. This consists of 518,080 words (approximately 20 words per sentence, on average) of text annotated with a detailed semantic and syntactic

(_(Please_RRCONCESSIVE mention_VVIVERBAL-ACT this_DD1
coupon_NN1DOCUMENT when_CSWHEN ordering_VVGINTER-ACT

OR_CCOR ONE_MC1WORD FREE_JJMONEY FANTAIL_NN1ANIMAL SHRIMPS_NN1FOOD

Figure 1: Example sentences from the training corpus.

tagset.

To understand the nature of the task involved in the experiments presented in this paper, one needs some familiarity with the ATR General English Tagset. For detailed presentations, see Black et al. [3][2][1]. An apercu can be gained, however, from Figure 1, which shows two sample sentences from the ATR Treebank (and originally from a Chinese take-out food flier), tagged with respect to the ATR General English Tagset. Each verb, noun, adjective and adverb in the ATR tagset includes a semantic label, chosen from 42 noun/adjective/adverb categories and 29 verb/verbal categories, some overlap existing between these category sets. Proper nouns, plus certain adjectives and certain numerical expressions, are further categorized via an additional 35 "proper—noun" categories. These semantic categories are intended for any "Standard-American-English" text, in any domain. Sample categories include: "physical.attribute" (nouns/adjectives/adverbs), "alter" (verbs/verbals), "interpersonal.act" (nouns/adjectives/adverbs/verbs/verbals), "orgname" (proper nouns), and "zipcode" (numericals). They were developed by the ATR grammarian and then proven and refined via day-in-day-out tagging for six months at ATR by two human "treebankers", then via four months of tagset-testing-only work at Lancaster University (UK) by five treebankers, with daily interactions among treebankers, and between the treebankers and the ATR grammarian. The semantic categorization is, of course, in addition to an extensive syntactic classification, involving some 165 basic syntactic tags.

The test corpus has been designed specifically to cope with the ambiguity of the tagset. It is possible to correctly assign any one of a number of `allowable' tags to a word in context. For example, the tag of the word *battle* in the phrase "a legal battle" could be either **NN1PROBLEM** or **NN1INTER-ACT**, indicating that the semantics is either a problem, or an interpersonal action. The test corpus consists of 53,367 words sampled from the same domains as, and in approximately the same proportions as the training data, and labeled with a set of up to 6 allowable tags for each word. During testing, only if the predicted tag fails to match any of the allowed tags is it considered an error.

5. Hierarchical Ontologies

The contribution of this paper is to consider the effect of features derived from hierarchical sets of words. The primary advantage is that we are able to construct these hierarchies using knowledge from outside the training corpus of the tagger itself, and thereby glean knowledge about rare words. In these experiments we use the human annotated word taxonomy of hypernyms (IS-A relations) in the WordNet database, and an automatically acquired ontology made by clustering words in a large corpus of unannotated text.

We have chosen to use hierarchical schemes for both the automatic and manually acquired ontologies because this offers the opportunity to combat data-sparseness issues by allowing features derived from all levels of the hierarchy to be used. The process of training the model is able to decide the levels of granularity that are most useful for disambiguation. For the purposes of generating features for the ME tagger we treat both types of hierarchy in the same fashion. One of these features is illustrated in Figure 2. Each predicate is effectively a question which asks whether the word (or word being used in a particular sense in the case of the WordNet hierarchy) is a descendent of the node to which the predicate applies. These predicates become more and more general as one moves up the hierarchy. For example in the hierarchy shown in Figure 3, looking at the nodes on the right hand branch, the lowest node represents the class of **apple trees** whereas the top node represents the class of all **plants**.

We expect these hierarchies to be particularly useful when tagging out of vocabulary words (OOV's). The identity of the word being tagged is by far the most important feature in our baseline model. When tagging an OOV this information is not available to the tagger. The automatic clustering has been trained on 100 times as much data as our tagger, and therefore will have information about words that tagger has not seen during training. To illustrate this point, suppose that we are tagging the OOV *pomegranate*. This word is in the WordNet database, and is in the same synset as the `fruit' sense of the word *apple*. It is reasonable to assume that the model will have learned (from the many examples of all fruit words) that the predicate representing membership of this **fruit** synset should, if true, favor the selection of the correct tag for fruit words: **NN1FOOD**. The predicate will be true for the word *pomegranate* which will thereby benefit from the model's knowledge of how to tag the other words in its class. Even if this is not so at this level in the hierarchy, it is likely to be so at some level of granularity. Precisely which levels of detail are useful will be learned by the model during training.

5.1 Automatic Clustering of Text

We used the automatic agglomerative mutual-information-based clustering method of Ushioda [18] to form hierarchical clusters from approximately 50 million words of tokenized, unannotated text drawn from similar domains as the treebank used to train the tagger. Figure 2 shows the position of the word *apple* within the hierarchy of clusters. This example highlights both the strengths and weaknesses of this approach. One strength is that the process of clustering proceeds in a purely objective

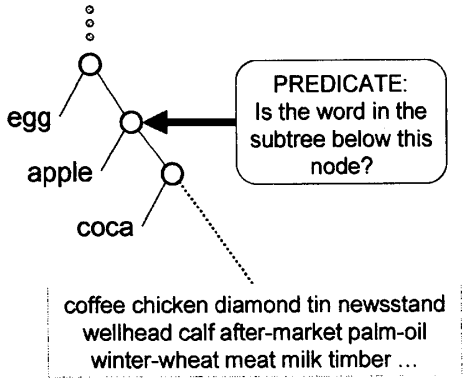

Figure 2: Dendrogram from automatically acquired clustering process.

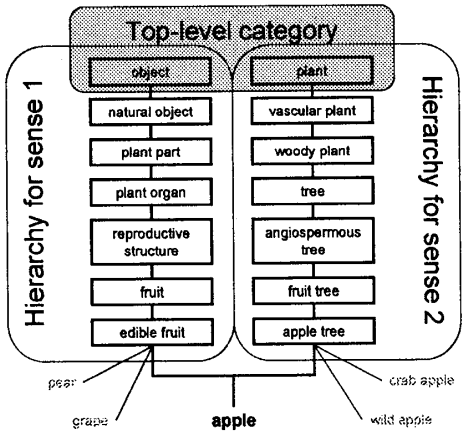

Figure 3: WordNet hierarchy derived from hypernym taxonomy.

fashion and associations between words that may not have been considered by a human annotator are present. Moreover, the clustering process considers all types that actually occur in the corpus, and not just those words that might appear in a dictionary (we will return to this later). A major problem with this approach is that the clusters tend to contain a lot of noise. Rare words can easily find themselves members of clusters to which they do not seem to belong, by virtue of the fact that there are too few examples of the word to allow the clustering to work well for these words. This problem can be mitigated somewhat by simply increasing the size of the text that is clustered. However the clustering process is computationally expensive. Another problem is that a word may only be a member of a single cluster; thus typically the cluster set assigned to a word will only be appropriate for that word when used in its most common sense.

Approximately 93% of running words in the test corpus, and 95% in the training corpus were covered by the words in the clusters (when restricted to verbs, nouns, adjectives and adverbs, these figures were 94.5% and 95.2% respectively). Approximately 81% of the words in the vocabulary from the test corpus were covered, and 71% of the training corpus vocabulary was covered.

5.2 WordNet Taxonomy

For this class of features, we used the hypernym taxonomy of WordNet Fellbaum [5]. Figure 3 shows the WordNet hypernym taxonomy for the two senses of the word *apple* that are in the database. The set of predicates query membership of all levels of the taxonomy for all WordNet senses of the word being tagged. An example of one such predicate is shown in the figure.

Only 63% of running words in both the training and the test corpus were covered by the words in the clusters. Although this figure appears low, it can be explained by the fact that WordNet only contains entries for words that have senses in certain parts of speech. Some very frequent classes of words, for example determiners, are not in WordNet. The coverage of only nouns, verbs, adjectives and adverbs in running text is 94.5% for both training and test sets. Moreover, approximately 84% of the words in the vocabulary from the test corpus were covered, and 79% on the training corpus. Thus, the effective coverage of WordNet on the important classes of words is similar to that of the automatic clustering method.

6. Experimental Results

The results of our experiments are shown in Table 1. The task of assigning semantic and syntactic tags is considerably more difficult than simply assigning syntactic tags due to the inherent ambiguity of the tagset. To gauge the level of human performance on this task, experiments were conducted to determine inter-annotator consistency; in addition, annotator accuracy was measured on 5,000 words of data. Both the agreement and accuracy were found to be approximately 97%, with all of the inconsistencies and tagging errors arising from the semantic component of the tags. 97% accuracy is therefore an approximate upper bound for the performance one would expect from an automatic tagger. As a point of reference for a lower bound, the overall accuracy of a tagger which uses only a single feature representing the identity of the word being tagged is approximately 73%.

The overall baseline accuracy was 82.58% with only 30.58% of OOV's being tagged correctly. It is immediately apparent from Table 1 that there is a strong response to the new features based on the ontological hierarchies. Performance for both clustering techniques was quite similar, with the WordNet taxonomical features being slightly more useful, especially for OOV's. One possible explanation for this is that overall, the coverage of both techniques is similar, but for rarer words, the MI clustering can be inconsistent due to lack of data (for an example, see Figure 2: the word *newsstand* is a member of a cluster of words that appear to be commodities), whereas the WordNet clustering remains consistent even for rare words. It seems reasonable to expect, however, that the automatic method would do better if trained on more data. Furthermore, all uses of words can be covered by automatic clustering, whereas the common use of the word *apple* as a company name is beyond the scope of WordNet.

#	Model	Accuracy	OOV's	Nouns	Verbs	Adj/Adv
1	Baseline	82.58	30.58	67.47	74.32	70.99
2	Automatically Acquired Ontology	83.71	35.08	71.89	75.83	75.34
3	WordNet	83.90	36.18	72.28	76.29	74.47

Table 1: Results of the tagging experiments (all figures quoted as percentages).

7. Conclusion

We have described a method for simultaneously labeling the syntax and semantics of words in running text. We develop this method starting from a state-of-the-art maximum entropy POS tagger which itself outperforms previous attempts to tag this data [3]. We augment this tagging model with word class membership information of the word being tagged. We define the features in such a manner that the granularity of the word classes used is automatically selected by the model. Our experimental results show that large gains in performance are obtained.

Both hierarchical ontology-based approaches increased overall performance, but with particular emphasis on OOV's, the intended target for this feature set. Visual inspection the output of the tagger on held-out data suggests there are many remaining errors arising from special cases that might be better handled by models separate from the main tagging model. In particular, numerical expressions and named entities cause OOV errors that the techniques presented in this paper are unable to handle. In future work we would like to address these issues, and also evaluate our system when used as a component of a WSD system, and when integrated within a machine translation system.

References

[1] E. Black and A. Finch. 2001. Developing and proving effective broad-coverage semantic-and-syntactic tagsets for natural language: The ATR approach. In *Proceedings of ICCPOL-2001.*

[2] E. Black, S. Eubank, H. Kashioka, R. Garside, G. Leech, and D. Magerman. 1996a. Beyond skeleton parsing: producing a comprehensive large–scale general–english treebank with full grammatical analysis. In *Proceedings of the 16th Annual Conference on Computational Linguistics*, pages 107–112, Copenhagen.

[3] E. Black, S. Eubank, H. Kashioka, and J. Saia. 1996b. Reinventing part-of-speech tagging. *Journal of Natural Language Processing (Japan)*, 5:1.

[4] Ezra Black, Andrew Finch, and Hideki Kashioka. 1998. Trigger-pair predictors in parsing and tagging. In *Proceedings, 36th Annual Meeting of the Association for Computational Linguistics, 17th Annual Conference on Computational Linguistics*, Montreal, Canada.

[5] C. Fellbaum. 1998. WordNet: An Electronic Lexical Database. MIT Press.

[6] J. Kupiec. 1992. Robust part-of-speech tagging using a hidden markov model. *Computer Speech and Language*, 6:225–242.

[7] A. K. Lamjiri, O. El Demerdash, and L.Kosseim. 2004. Simple features for statistical word sense disambiguation. In *Proc. ACL 2004 – Third International Workshop on the Evaluation of Systems for the Semantic Analysis of Text (Senseval-3)*, Barcelona, Spain, July. ACL-2004.

[8] C. Li and H. Li. 2002. Word translation disambiguation using bilingual bootstrapping.

[9] Xiaobin Li, Stan Szpakowicz, and Stan Matwin. 1995. A wordnet-based algorithm for word sense disambiguation. In *IJCAI*, pages 1368–1374.

[10] Mitchell P. Marcus, Beatrice Santorini, and Mary Ann Marcinkiewicz. 1994. Building a large annotated corpus of english: The penn treebank. *Computational Linguistics*, 19 (2):313–330.

[11] B. Merialdo. 1994. Tagging english text with a probabilistic model. *Computational Linguistics*, 20(2):155–172.

[12] Rada Mihalcea and Dan I. Moldovan. 1998. Word sense disambiguation based on semantic density. In *Sanda Harabagiu, editor, Use of WordNet in Natural Language Processing Systems: Proceedings of the Conference*, pages 16–22. Association for Computational Linguistics, Somerset, New Jersey.

[13] I. Nancy and V. Jean. 1998. Word sense disambiguation: The state of the art. *Computational Linguistics*, 24:1:1–40.

[14] G. Ramakrishnan and B. Prithviraj. 2004. Soft word sense disambiguation. In *International Conference on Global Wordnet (GWC 04)*, Brno, Czeck Republic.

[15] A. Ratnaparkhi. 1996. A maximum entropy part-of-speech tagger. In *Proceedings of the Empirical Methods in Natural Language Processing Conference.*

[16] R. Rosenfeld. 1996. A maximum entropy approach to adaptive statistical language modelling. *Computer Speech and Language*, 10:187–228.

[17] A. Suarez. 2002. A maximum entropy-based word sense disambiguation system. In *Proc. International Conference on Computational Linguistics.*

[18] A. Ushioda. 1996. Hierarchical clustering of words. In *Proceedings of COLING 96*, pages 1159–1162.

[19] D. Yarowsky. 1993. One sense per collocation. In *Proceedings of ARPA Human Language Technology Workshop.*