सार्थक
Word Sense Disambiguation Using Head-Subcat Relationship in HPSG and Ontological Classes

D Sharma, K Vikram, Manav R Mital, A Mukerjee, A M Raina
Indian Institute of Technology, Kanpur,
Kanpur 208016,
Uttar Pradesh, India.
Email: {*deepaks,vikram,manavrm,amit,achla*}*@iitk.ac.in*

Abstract

In this paper, we describe a novel approach to Word Sense Disambiguation at the sentential level, which employs the concept of semantics and pragmatics to achieve sentence level word sense disambiguation for English sentences. It also restrict feasible syntactic parses generated by a syntactic parser. The program operates on HPSG parsed output(s) which is/are purely syntactic and *untagged* with any word senses. Then, using a simple verb-subject heuristic and word-usage restrictions expressed in the form of the head-subcat relationship extant in HPSG, it generates an ordered list of the various sense combinations of the different verbs and nouns in the sentence. The heuristic is modeled on sense restrictions which use the concept of ontological classes in the form of synsets derived from WordNet. It takes its input from a generalized bilingual parser that produces multiple syntactic parses for sentences both in English and Hindi, using HPSG. In this way, the Word Sense Disambiguator also restricts any semantically unseemly parses generated by the parser. Though, complete in itself, the disambiguator is expected to provide a basis for generating sentential information in predicate structural form, to be used in discourse representation for the सार्थक project which aims at discourse comprehension.

1 Introduction

The सार्थक project, part of a larger effort at story animation from text, is an attempt to perform discourse-level analysis on stories written for a restricted domain. In this endeavour, it becomes necessary to consider the semantics and pragmatics of the language, which requires building models of processes in the real world.

In our work we put our emphasis on sentence-level word sense disambiguation, which makes it different from general statistical techniques that use contextual information for the same. Consider the sentences: `the person barked at the baby`, and `the dog was barking`. In such situations, the varied senses for the same part-of-speech usage are traditionally disambiguated using techniques like semantic distance computation ⟦Smeaton and Quigley1996, Resnik1997⟧, semantic tree constructions ⟦Ding2001⟧ and conceptual dependencies ⟦Agirre and Rigau1996⟧. Another class of techniques ⟦Yarowsky1992⟧ is the one that uses Roget's Thesaurus, and works by considering windows of words in a text and then finds the most appropriate sense of a particular word by computing the semantic distance between that word and the other words in the window. Such statistical techniques often work on large context windows and the results are not that good on small sentences say about five words long. Such methods do not incorporate any of the extensive works available on word epistemology and ontology and it is the intent of this paper to show how this knowledge can be incorporated into the WSD process.

Our primary focus is to leverage available knowledge of the ontological classes of words. For instance, for the verb `bark`, if the subject is in the class `person` then `bark` means *to speak in an unfriendly tone*, or if the subject is in the class `animal`, then it refers *to make the sounds as those produced by a dog*.

The ontological-class based approach uses the HPSG ⟦Pollard and Sag1992⟧ which produces the form of head-subcat relationships. This paper reflects the widely held belief among linguists that the HPSG parse reflects a semantic proximity between the head and its subcat. The disambiguation lexicon uses the word senses available in WordNet ⟦Fellbaum1998⟧, and is further used to obtain the ontological classes of the word. The disambiguator is used as a back-end to the HPSG parser and serves to reject any unacceptable parses. For instance, in the case of sentence like `I saw the girl with long hair`, there are two parse structures available:

- a. `with` being the subcat of `saw` suggesting that the girl was seen with the help of long hair as an instrument of seeing, and
- b. `with` being the subcat of `girl` suggesting that the girl had long hair.

In cases such as these, the word sense disambiguator is unable to find a ontological class for "hair" whereby it may serve as an optical instrument. This rules out *a* as a possible parse. In this manner, the word sense disambiguator works as a supplement to the parser, rejecting erroneous parses.

In what follows, we give a brief overview of HPSG (section 1.1) and सार्थक (section 1.2) and the input domain (section 1.3). The WSD lexicon, restrictions, ontology and the computational data structure are dealt with in section 2. In section 3 we take up a run through example. Section 4 and 5 deal with the example results and the issue of comparison with existing models. Section 6 concludes the paper.

1.1 HPSG

HPSG is a computationally efficient parsing grammar that was developed at Stanford in the late 80's ⟦Pollard and Sag1992⟧. It reduces the complexity of language to a uniform parsing structure and a few rules ⟦Mukerjee *et al.*1999⟧. It has been applied in a number of practical systems and found quite useful at those ends. Also in systems like ALE and LKB ⟦Copestake1993⟧, it has been demonstrated to work on different languages. Recently, many researchers have started moving on to the HPSG workbench because of its rich semantics.

1.2 Overview of Saarthaka

In the designing of the project सार्थक, we have followed a structured approach (figure 1). In this disambiguator we have kept to a sentential analysis only. The information of the sentence, in predicate form, will then go to a discourse processing module that will process the whole discourse level information using the discourse representation theory. The whole approach is as follows:

- First, the syntax of each sentence is considered, and parsing is done using HPSG
- Secondly, the senses of the content words is examined for all the parses of each particular sentence. Some syntactic parses may be rejected based on word sense disambiguation.
- Thirdly, the appropriate predicate structures for each sentence are generated.
- Lastly, predicate structures are processed and unified in accordance with the discourse representation theory. To this end, a discourse representation is made with the program which is known as the ADIS (Acquired Discourse Information Structure)

1.3 Domain

For सार्थक, we have used a restricted domain of about 412 words, which describes an average family and their environment. In our corpus we have 14-15 stories, each one being of about 5 to 12 sentences, collected from a group of volunteers. We chose some of the available senses for all the domain words using WordNet 1.7 ⟦Fellbaum1998⟧. WordNet provides a word sense network ⟦Fellbaum1998⟧. Each sense of a word is mapped to a word sense node which is a group of strict synonyms called *synsets* and all synsets are linked by a variety of semantic relationships such hypernymy (ontological relationships between words). As discussed above, WordNet maintains groups of words termed as *Synsets* and each word is assigned a Synset No. corresponding to the Synset that it belongs to. These semantic similarities can be used for word sense disambiguation using class based techniques or using techniques like conceptual densities ⟦Agirre and Rigau1996⟧.

Intuitively enough, sense disambiguation can be done mainly for content words - verbs, nouns, adjectives and adverbs. At present, the implementation takes care of nouns and verbs only. All senses of verbs that are spelt out in WordNet are taken care of. The ontological tree has been borrowed from WordNet, but some of these entries have been extended by us to meet the needs of a richer ontological class. We are using the JWNL v1.1 API to WordNet.

2 WSD Lexicon

We use the head-subcat relationship, available in an HPSG parsed output ⟦Pollard and Sag1992⟧, to represent combinatorial possibilities of the various senses of different words taken from the WordNet. To return to the example considered in the introduction, following are the five senses of the verb bark available with WordNet:
1. To speak in an unfriendly tone.
2. To cover with a bark.

3. To remove the bark of a tree.
4. To make barking sounds.
5. To tan with bark tannins.
Ideally, any Human Computer Interface should have access to all the senses to be able to, at one given point of time, make out what the user is saying to it. These senses are semantically related to the subject, subcategory, etc. For instance, only a person can bark#1 or speak in an unfriendly tone or only an animal can bark#4 or make barking sounds. These proximities are captured in the WSD lexicon.

2.1 Restrictions

The WSD lexicon is quite a large lexicon that can be used to eliminate implausible sense combinations. A sample restriction included in the lexicon is:

```
bark
V
(HEAD:#V:bark_767337_1)(SUB:N:person5145)=0.5
!the person barks...
```

The entry corresponds to the verb `bark`. Each term here has a three tuple representing the restriction on the usage of the head word `bark`. The symbol '#' indicates that `bark` is the word being considered. This restriction holds on the subject and one subcategory of the word. The term `bark_767337_1` means the word 'bark' with synset no. 767337 and sense no. 1. The term `person5145` means the ontological class 'person' with synset no 5145. The second tuple is a letter like V, N, A or P etc. which means the part of speech. The letter 'X' means another embedded phrase entry as the third tuple. Finally, the term `=0.5` means that the "credibility" associated with a parses sub-tree that matches with this restriction. Note that the character '!' comments a line in the lexicon. The tuple '-' is a wild-card that can mean anything. Other sample lexical entries for the word bark are

```
(HEAD:#V:bark_935290_2)(SUB:N:organism2908)
(SUBCAT:N:woodyplant10814086)=0.9
!as in, to skin a tree
(HEAD:#V:bark_767468_4)(SUB:N:animal10787)=0.7
!the barking by an animal, such as a dog
(HEAD:#V:bark_767337_1)(SUB:N:person5145)(SUBCAT:X:
(HEAD:P:at)(SUBCAT:N:person5145))=0.95
```

These entries represent the co-occurrence *possibilities* of the various senses of the word bark with other entities, which are represented in the form of subcats and even subcats of subcats of the head bark. In the case of the example, `the person barked at the baby`, there are 5 possible senses for bark and 3 for baby (that we consider) and so, out of a combination of 15 possibilities, the program can pick out 2 most credible sense combinations, in where the bark refers to the sense implying *speaking in unfriendly tone* and baby may mean either *a young offspring* or *the youngest member of a group*. These restrictions are very expressive and easy to formulate, maintain and extend.

Take for instance `stand`, which may be either the viewpoint or the object; when we say that `Mary bought a good stand` then we know that `buy` as head can refer only to the object `stand` in its subcat. We follow a fuzzy approach in this case, if a sense combination adheres to a certain restriction, it is assigned a particular credibility associated with that restriction. These credibilities are determined statistically and are values ranging between 0 and 1. If more than one restrictions can be applied, then the maximum of those credibilities is taken. In general, the more specific a restriction is, the more credibility it carries.

In the above process a critical step is finding the subject of a head phrase. Mostly this is directly known, in some cases a simple heuristic can be used to identify the subject. This information is invariably present in the HPSG parse tree itself.

In the end, a ranking of the various possible sense combinations of a particular sentence are taken. For this purpose, k top senses corresponding to each parse of the sentence are finally shown as the output.

2.2 Ontology

We extend the ontology used in WordNet. e.g., tree_10814827_1 is a hyponym of woodyplant_10814086_1 in WordNet. In order to successfully process sentences such as `he saw the girl with long hair` we need to introduce our own classes, e.g., optical instrument. Since these synsets are not in WordNet, we assign a 9 digit number to these synsets (Note that WordNet synsets are only 8 digits long).

2.3 Computational Data Structure

The WSD algorithm searches over all possible sense instantiations of the parsed tree in order to obtain a ranking of the parses by their credibility. If a given sense combination does not match any restriction in the WSD lexicon, it is discarded. Otherwise it is given the credibility corresponding to that restriction.

We need to consider subcats that are subcats and may have their own subcats, and so on. e.g., the person barked at the baby. Here, we apply the last restriction stated for `bark`. The word `baby` appears as a subcat of the subcat and sense combination of this whole sub-tree is necessary for disambiguating the sense of the word bark.

As the WSD algorithm proceeds along the sentence, previously disambiguated words with their credibilities can be propagated for efficient analysis.

2.3.1 Parts of speech other than nouns and verbs

In case of such words, we just consider their default senses, *i.e.,* the word itself. Such words are important to our considerations because often senses of nouns and verbs can be ascertained with quite much certainty upon examination of adjoining articles or prepositions etc. *e.g.* `He sat on the chair.` Here the preposition helps disambiguate between the four senses of the word chair, and the one corresponding to the sense `seat` can be made out. Hence, the corresponding sense is given a higher credibility and the appropriate sense combination is chosen.

2.3.2 Relative clauses

The treatment of relative clauses and treatment of the *wh* markers also deserves special mention. The words *who, whom, which* etc are disambiguated, i.e., their referent is figured out and they are assigned a default sense with synset no 0. They are practically never used in disambiguation. They crop up as subjects to verbs, and in any case the subject is needed, they are simply replaced by their referents and subsequent processing is carried on normally.

2.3.3 Auxillary verbs

In this case, the connection between the auxiliary verb and its associated main verb is specified in the lexicon itself. The main verb is mentioned in the subcat of the auxiliary verb, and the corresponding credibility is kept quite high. Then the corresponding structure is optimally selected by the program.

3 Example

In this section we consider a simple run through example that briefly illustrates the mechanism of the whole WSD algorithm. Consider the input sentence to be `the person bought the stand`.

In this case, the words `stand` and `buy` are polysemous. The word `person` has only one sense and the word `the` is a determiner and has no sense (for our purpose).

The restrictions corresponding to the words `stand` and `buy` are:

- buy
  V
  (HEAD:#V:buy_1648144_1)(SUB:N:organism2908)(SUBCAT:N:object12545)=0.8
  !e.g. : he bought the stand
  (HEAD:#V:buy_501280_4)(SUB:N:organism2908)(SUBCAT:N:content4884186)=0.9
  !e.g. : he bought the idea
  (HEAD:#V:buy_1648144_1)(SUB:N:organism2908)(SUBCAT:N:-)=0.2
  !because an organism can buy anything

- stand
  N
  (HEAD:#N:stand_3448815_5)=0.2
  (HEAD:#N:stand_2378629_1)=0.1
  (HEAD:#N:stand_5146205_7)=0.07

We see that in the WSD lexicon, there are 3 senses for the noun `stand` and 2 for the verb `buy`. It implies a search space of 6 possible sense combinations.
The algorithm proceeds as follows. It first grows two branches for the senses of the head word `buy`.

At each node, there a hierarchy instantiation with the sense of word buy specified. In each of these cases, it spawns children nodes for all possible instantiations of the hierarchy corresponding to the senses assigned to the subcats of buy or their subcats. Three nodes each are grown with hierarchy instantiations with sense assignments for `stand` and `buy`. Of the total 6 leaf nodes, after applying the restrictions, all sense combinations corresponding to the 4th sense of `buy` are rejected and those with the first and fifth sense of stand are chosen. Hence, we are left with two leaf nodes. Now we recursively consider the words `person` and `stand` in a similar fashion. At the level of the word `person` since it is the only sense, it is taken as is with the subcat meanings. At the level of the noun `stand`, the restrictions for the sense numbers 1 and 5 match with the respective cases in the nodes of the search tree. The sense no 5 instantiation is given a higher credibility.

In case of multiple syntactic parses, all sense-combinations corresponding to each particular parse are investigated and output is given as a list of parses in the decreasing order of appropriateness.

4 Example results

The program was tested with a few ambiguous cases with various possibilities of verbs such as sleep, be, bark, chair, baby, stand, buy, etc.

- One of the most important functions is to choose semantically *correct* parses from a list of possible syntactic parses. For example, in the case of the two parses of the sentence `the person saw the boy with the chocolate` , the program correctly assigns a higher credibility to the parse linkage of the chocolate to the boy and not the person. This was possible because we can *see* with objects that can be classified as *opticalinstruments*.
- In cases like `this person is the new chair of our college`, the program rejects the sense in which *chair* may be interpreted as a furniture and assigns it a sense corresponding to professorship. Correspondingly, in a sentence like `the chair is in the room`, the program assigns it a most probable sense of furniture.
- It correctly identifies all the implied senses in cases of multiple senses of bark in sentences such as: `the person was barking the tree, the person barked at the boy, the dog was barking, the dog barked at the boy`, etc. These were then extended to other sentential constructs having apparent ambiguity at syntax level but none at the level of semantics, and program correctly identified the desired senses. The cases were such as: `the boy bought the idea, the boy was the baby of the house, the woman bought the stand,` etc.
- It correctly employs prepositions in identifying verb senses, instances of these were in case of sentences such as `the boy walked in the bank` and the other one `the boy walked on the bank`. This is possible because `in` applies to closed spaces with property of containership which applies to the sense of bank corresponding to the building. Similarly, the `on` preposition helps in resolving bank to be the river-side sloping ground.
- In inherently ambiguous sentences such as `the person jumped on the chair`, it gives two results with closely separated credibility whereas the correct meaning is obtained in the sentence `the boy slept on the chair, the dog sat on the chair, etc`.

- Relative clauses in cases such as `the person who was walking on the bank saw the chocolate` are handled. The presence of the word *who* does not make any difference because the subject field comes into play over here.
- The adjective is also used in certain cases to disambiguate between the various noun senses. For instance, the adjective *good* could be applied to *stand* with greater certainty if it referred to an object and the adjective *noble* could be applied to the same noun if it suggested the sense corresponding to view-point. This is analogous to the human reasoning process in some sense, and also reduces the dependency of a full verb-noun relation specification for disambiguation purposes

5 Comparison with existing models

- On of the advantages of our model is that even on small context, or a sentence, good results can be obtained. It captures compactly the considerable word usage knowledge. The WSD lexicon is an encapsulation of the multiple senses present in the language.

- A better statistical model for calculating credibility for whole parses need to be developed but that can be done only on large training data and again, by a clear demarcation of the domain.

- Another advantage is that it does not specifically rely too much on the dictionary, but uses the API interface to it. In most systems, when a word has just a single sense occurrence, it is taken to be unambiguous, ruling out the scope for any further lexical expansion. In the current system however, all content words are given the same treatment irrespective of how many senses they have.

- One weakness of this approach is that the WSD lexicon is likely to be very large for general domains.

6 Conclusions

This technique shows that linkages between rich parse structures as in HPSG with powerful lexicon and ontological class hierarchy such as WordNet may be exploited to give good results. The program and its technique is domain independent. All the information is kept in the lexicon which makes it very flexible to use. For practical use, combining rules that occur frequently with statistical method may lead to a very efficient hybrid structure. In situations with limited domains, one can consider those senses of WordNet that are relevant in the domain. For instance, in a domain of a family, a single sense of the verb *bark* is enough. This makes the program a practical computational system for even a small system and yet fulfills the needs of a specialized, domain-dependent agent.
The domain-dependency feature to improve efficiency is further justified by the fact that this particular system is to be used in conjunction with a parser and predicate generator ⟦Hess1991⟧ to construct discourse representations for a very restricted domain. In the context of our proposed usage in the story telling, the system will be used only on a very restricted domain and much of the WSD problems

will not be very complex. A fully general search can be very expensive, e.g., *The person who was walking on the bank saw a chocolate.* It works quite fast when used with the restrictive interface that works with only a limited senses of the words but runs out of memory on a small computer otherwise. In full generality, it has 3 senses of person, 10 senses for walk, 13 senses for be, 10 senses for bank, 24 senses for see and 3 senses for chocolate – total of 280800 sense combinations. Exploration of other domain dependent heuristics may further enhance performance as is the case for many contemporary systems ⟦Sutcliffe *et al.*1995⟧.

References

⟦ **Agirre and Rigau1996** ⟧ E. Agirre and G. Rigau. Word sense disambiguation using conceptual density. *In proceedings of COLING'96, pages 16–22, Copenhagen, Danmark.*, 1996. citeseer.nj.nec.com/agirre96word.html.

⟦ **Copestake1993** ⟧ Ann Copestake. The compleat lkb. Technical Report 316, University of Cambridge Computer Laboratory, 1993.

⟦ **Ding2001** ⟧ Ying Ding. The enabler for the semantic web. *Journal of Information Science*, 27, 2001.

⟦ **Fellbaum1998** ⟧ Christiane Fellbaum. *WordNet: An Electronic Lexical Database.* The MIT Press, Cambridge, 1998.

⟦ **Hess1991** ⟧ Michel Hess. Recent developments in discourse representation theory. *Communication with Men and Machines*, 1991.

⟦ **Mukerjee *et al.*1999** ⟧ Amitabha Mukerjee, Mahim Mishra, and Rajat Bhattacharjee. A natural language interface for generating video animation. Technical report, Indian Institute of Technology Kanpur, 1999.

⟦ **Pollard and Sag1992** ⟧ Carl Pollard and Ivan E. Sag. *Head-Driven Phrase Structure Grammar.* University of Chicago Press, 1992. A classic text on HPSG and its computational implementation for parsing purposes,.

⟦ **Resnik1997** ⟧ Philip Resnik. Selectional preference and sense disambiguation. In *Proceedings of the ACL SIGLEX Workshop on Tagging Text with Lexical Semantics: Why, What and How? Washington, D.C., USA*, 1997.

⟦ **Smeaton and Quigley1996** ⟧ Alan F. Smeaton and Ian Quigley. Experiments on using semantic distances between words in image caption retrieval. In *Research and Development in Information Retrieval*, pages 174–180, 1996. citeseer.nj.nec.com/smeaton96experiments.html.

⟦ **Sutcliffe *et al.*1995** ⟧ Richard F.E. Sutcliffe, Paul Boersma, Annelies Bon, Ton Donker, Michael C. Ferris, Peter Hellwig, Patrick Hyland, Heinz-Detlev Koch, Pieter Masereeuw, Annette McElligott, and et al. Beyond keywords: Accurate retrieval from full text documents, 1995. citeseer.nj.nec.com/sutcliffe95beyond.html.

⟦ **Yarowsky1992** ⟧ David Yarowsky. Word-sense disambiguation using statistical models of Roget's categories trained on large corpora. In *Proceedings of COLING-92*, pages 454–460, Nantes, France, July 1992. citeseer.nj.nec.com/yarowsky92wordsense.html.