Sense Disambiguation Using Semantic Relations and Adjacency Information

Anil S. Chakravarthy
MIT Media Laboratory
20 Ames Street E15-468a
Cambridge MA 02139
anil@media.mit.edu

Abstract

This paper describes a heuristic-based approach to word-sense disambiguation. The heuristics that are applied to disambiguate a word depend on its part of speech, and on its relationship to neighboring salient words in the text. Parts of speech are found through a tagger, and related neighboring words are identified by a phrase extractor operating on the tagged text. To suggest possible senses, each heuristic draws on semantic relations extracted from a Webster's dictionary and the semantic thesaurus WordNet. For a given word, all applicable heuristics are tried, and those senses that are rejected by all heuristics are discarded. In all, the disambiguator uses 39 heuristics based on 12 relationships.

1 Introduction

Word-sense disambiguation has long been recognized as a difficult problem in computational linguistics. As early as 1960, Bar-Hillel [1] noted that a computer program would find it challenging to recognize the two different senses of the word "pen" in "The pen is in the box," and "The box is in the pen." In recent years, there has been a resurgence of interest in word-sense disambiguation due to the availability of linguistic resources like dictionaries and thesauri, and due to the importance of disambiguation in applications like information retrieval and machine translation.

The task of disambiguation is to assign a word to one or more senses in a reference by taking into account the context in which the word occurs. The reference can be a standard dictionary or thesaurus, or a lexicon constructed specially for some application. The context is provided by the text unit (paragraph, sentence, etc.) in which the word occurs.

The disambiguator described in this paper is based on two reference sources, the Webster's Seventh Dictionary and the semantic thesaurus WordNet [12]. Before the disambiguator is applied, the text input is processed first by a part-of-speech tagger and then by a phrase extractor which detects phrase boundaries. Therefore, for each ambiguous word, the disambiguator knows the part of speech, and other phrase headwords and modifiers that are adjacent to it. Based on this context information, the disambiguator uses a set of heuristics to assign one or more senses from the Webster's dictionary or WordNet to the word. Here is an example of a heuristic that relies on the fact that conjoined head nouns are likely to refer to objects of the same category. Consider the ambiguous word "snow" in the sentence "Slush and snow filled the roads." In this sentence, the tagger identifies "snow" as a noun. The phrase extractor indicates that "snow" and "slush" are conjoined head words of a noun phrase. Then, the heuristic uses WordNet to identify the senses of "slush" and "snow" that belong to a common category. Therefore, the sense of "snow" as "cocaine" is discarded by this heuristic.

The disambiguator has been incorporated into two information retrieval applications which use semantic relations (like A-KIND-OF) from the dictionary and WordNet to match queries to text. Since semantic relations are attached to particular word senses in the dictionary and WordNet, disambiguated representations of the text and the queries lead to targeted use of semantic relations in matching.

The rest of the paper is organized as follows. The next section reviews existing approaches to disambiguation with emphasis on directly related methods. Section 3 describes in more detail the heuristics and adjacency relationships used by the disambiguator.

2 Previous Work on Disambiguation

In computational linguistics, considerable effort has been devoted to word-sense disambiguation [8]. These approaches can be broadly classified based on the reference from which senses are assigned, and on the method used to take the context of occurrence into account. The references have ranged from detailed custom-built lexicons (e.g., [11]) to standard resources like dictionaries and thesauri like Roget's (e.g., [2, 10, 14]). To take the context into account, researchers have used a variety of statistical weighting and spreading activation models (e.g., [9, 14, 15]). This section gives brief descriptions of some approaches that use on-line dictionaries and WordNet as references.

WordNet is a large, manually-constructed semantic network built at Princeton University by George Miller and his colleagues [12]. The basic unit of WordNet is a set of synonyms, called a *synset*, e.g., [go, travel, move]. A word (or a word collocation like "operating room") can occur in any number of synsets, with each synset reflecting a different sense of the word. WordNet is organized around a taxonomy of *hypernyms* (A-KIND-OF relations) and *hyponyms* (inverses of A-KIND-OF), and 10 other relations. The disambiguation algorithm described by Voorhees [16] partitions WordNet into *hoods*, which are then used as sense categories (like dictionary subject codes and Roget's thesaurus classes). A single synset is selected for nouns based on the hood overlap with the surrounding text.

The research on extraction of semantic relations from dictionary definitions (e.g., [5, 7]) has resulted in new methods for disambiguation, e.g., [2, 15]. For example, Vanderwende [15] uses semantic relations extracted from LDOCE to interpret nominal compounds (noun sequences). Her algorithm disambiguates noun sequences by using the dictionary to search for predefined relations between the two nouns; e.g., in the sequence "bird sanctuary," the correct sense of "sanctuary" is chosen because the dictionary definition indicates that a sanctuary is an area for birds or animals.

Our algorithm, which is described in the next section, is in the same spirit as Vanderwende's but with two main differences. In addition to noun sequences, the algorithm has heuristics for handling 11 other adjacency relationships. Second, the algorithm brings to bear both WordNet and semantic relations extracted from an on-line Webster's dictionary during disambiguation.

3 Sense Disambiguation with Adjacency Information

The input to the disambiguator is a pair of words, along with the adjacency relationship that links them in the input text. The adjacency relationship is obtained automatically by processing the text through the Xerox PARC part-of-speech tagger [6] and a phrase extractor. The 12 adjacency relationships used by the disambiguator are listed below. These adjacency relationships were derived from an analysis of captions of news photographs provided by the Associated Press. The examples from the captions also helped us identify the heuristic rules necessary for automatic disambiguation using WordNet and the Webster's dictionary. In the table below, each adjacency category is accompanied by an example. 39 heuristic rules are used currently.

Adjacency Relationship	Example
Adjective modifying a noun	Express train
Possessive modifying a noun	Pharmacist's coat
Noun followed by a proper name	Tenor Luciano Pavarotti
Present participle gerund modifying a noun	Training drill
Noun noun	Basketball fan
Conjoined nouns	A church and a home
Noun modified by a noun at the head of a following "of" PP	Barrel of the rifle
Noun modified by a noun at the head of a following "non-of" PP	A mortar with a shell
Noun that is the subject of an action verb	A monitor displays information
Noun that is the object of an action verb	Write a mystery
Noun that is at the head of a prepositional phrase following a verb	Sentenced to life
Nouns that are subject and object of the same action	The hawk found a perch

Given a pair of words and the adjacency relationship, the disambiguator applies all heuristics corresponding to that category, and those word senses that are rejected by all heuristics are discarded. Due to space considerations, we will not describe the heuristic rules individually but

294

instead identify some common salient features. The heuristics are described in detail in [3].

- Several heuristics look for a particular semantic relation like hypernymy or purpose linking the two input words, e.g., "return" is a hypernym of "forehand."

- Many heuristics look for particular semantic relations linking the two input words to a common word or synset; e.g., a "church" and a "home" are both buildings.

- Many heuristics look for analogous adjacency patterns either in dictionary definitions or in example sentences, e.g., "write a mystery" is disambiguated by analogy to the example sentence "writes poems and essays."

- Some heuristics look for specific hypernyms such as person or place in the input words; e.g., if a noun is followed by a proper name (as in "tenor Luciano Pavarotti" or "pitcher Curt Schilling"), those senses of the noun that have "person" as a hypernym are chosen.

The disambiguator has been used in two retrieval programs, ImEngine, a program for semantic retrieval of image captions, and NetSerf, a program for finding Internet information archives [3, 4]. The initial results have not been promising, with both programs reporting deterioration in performance when the disambiguator is included. This agrees with the current wisdom in the IR community that unless disambiguation is highly accurate, it might not improve the retrieval system's performance [13].

References

1. Bar-Hillel, Yehoshua. 1960. "The Present Status of Automatic Translation of Languages," in *Advances in Computers*, F. L. Alt, editor, Academic Press, New York.

2. Braden-Harder, Lisa. 1992. "Sense Disambiguation Using On-line Dictionaries," in *Natural Language Processing: The PLNLP Approach*, Jensen, K., Heidorn, G. E., and Richardson, S. D., editors, Kluwer Academic Publishers.

3. Chakravarthy, Anil S. 1995. "Information Access and Retrieval with Semantic Background Knowledge" Ph.D thesis, MIT Media Laboratory.

4. Chakravarthy, Anil S. and Haase, Kenneth B. 1995. "NetSerf: Using Semantic Knowledge to Find Internet Information Archives," to appear in *Proceedings of SIGIR'95*.

5. Chodorow, Martin. S., Byrd, Roy. J., and Heidorn, George. E. 1985. "Extracting Semantic Hierarchies from a Large On-Line Dictionary," in *Proceedings of the 23rd ACL*.

6. Cutting, Doug, Julian Kupiec, Jan Pedersen, and Penelope Sibun. 1992. "A Practical Part-of-Speech Tagger," in *Proceedings of the Third Conference on Applied NLP.*

7. Dolan, William B., Lucy Vanderwende, and Richardson, Steven. D. 1993. "Automatically Deriving Structured Knowledge Bases from On-line Dictionaries," in *Proceedings of the First Conference of the Pacific Association for Computational Linguistics*, Vancouver.

8. Gale, William, Church, Kenneth. W., and David Yarowsky. 1992. "Estimating Upper and Lower Bounds on the Performance of Word-sense Disambiguation Programs," in *Proceedings of ACL-92*.

9. Hearst, Marti. 1991. "Noun Homograph Disambiguation Using Local Context in Large Text Corpora," *Proceedings of the 7th Annual Conference of the UW Centre for the New OED and Text Research*, Oxford, England.

10. Lesk, Michael. 1986. "Automatic Sense Disambiguation: How to Tell a Pine Cone from an Ice Cream Cone," in *Proceedings of the SIGDOC Conference*

11. McRoy, Susan. 1992. "Using Multiple Knowledge Sources for Word Sense Discrimination," in *Computational Linguistics*, 18(1).

12. Miller, George A. 1990. "WordNet: An On-line Lexical Database," in *International Journal of Lexicography*, 3(4).

13. Sanderson, Mark. 1994. "Word Sense Disambiguation and Information Retrieval," in *Proceedings of SIGIR'94*.

14. Yarowsky, David. 1992. "Word Sense Disambiguation Using Statistical Models of Roget's Categories Trained on Large Corpora," in *Proceedings of COLING-92*, Nantes, France.

15. Vanderwende, Lucy. 1994. "Algorithm for Automatic Interpretation of Noun Sequences," in *Proceedings of COLING-94*, Kyoto, Japan.

16. Voorhees, Ellen. M. 1993. "Using WordNet to Disambiguate Word Senses for Text Retrieval," in *Proceedings of SIGIR'93*.