UofL: Word Sense Disambiguation Using Lexical Cohesion

Yllias Chali
Department of Computer Science
University of Lethbridge
Lethbridge, Alberta, Canada, T1K 3M4
chali@cs.uleth.ca

Shafiq R. Joty
Department of Computer Science
University of Lethbridge
Lethbridge, Alberta, Canada, T1K 3M4
jotys@cs.uleth.ca

Abstract

One of the main challenges in the applications (i.e.: text summarization, question answering, information retrieval, etc.) of Natural Language Processing is to determine which of the several senses of a word is used in a given context. The problem is phrased as "Word Sense Disambiguation (WSD)" in the NLP community. This paper presents the dictionary based disambiguation technique that adopts the assumption of one sense per discourse in the context of SemEval-2007 Task 7: "Coarse-grained English all-words".

1 Introduction

Cohesion can be defined as the way certain words or grammatical features of a sentence can connect it to its predecessors (and successors) in a text. (Halliday and Hasan, 1976) defined cohesion as "the set of possibilities that exist in the language for making text hang together". Cohesion occurs where the interpretation of some element in the discourse is dependent on that of another. For example, an understanding of the reference of a pronoun (i.e.: he, she, it, etc.) requires to look back to something that has been said before. Through this cohesion relation, two text clauses are linked together.

Cohesion is achieved through the use in the text of semantically related terms, reference, ellipse and conjunctions (Barzilay and Elhadad, 1997). Among the different cohesion-building devices, the most easily identifiable and the most frequent type is lexical cohesion. Lexical cohesion is created by using semantically related words (repetitions, synonyms, hypernyms, hyponyms, meronyms and holonyms, glosses, etc.)

Our technique used WordNet (Miller, 1990) as the knowledge source to find the semantic relations among the words in a text. We assign weights to the semantic relations. The technique can be decomposed into two steps: (1) building a representation of all possible senses of the words and (2) disambiguating the words based on the highest score.
The remainder of this paper is organized as follows. In the next section, we review previous work. In Section 3, we define the semantic relations and their weights. Section 4 presents our two step procedure for WSD. We conclude with the evaluation.

2 Previous Work

Lexical Chaining is the process of connecting semantically related words, creating a set of chains that represent different threads of cohesion through the text (Galley and McKeown, 2003). This intermediate representation of text has been used in many natural language processing applications, including automatic summarization (Barzilay and Elhadad, 1997; Silber and McCoy, 2003), information retrieval (Al-Halimi and Kazman, 1998), and intelligent spell checking (Hirst and St-Onge, 1998).

Morris and Hirst (1991) at first proposed a manual method for computing lexical chains and first computational model of lexical chains was introduced by Hirst and St-Onge (1997). This linear-time algorithm, however, suffers from inaccurate WSD, since their greedy strategy immediately disambiguates a word as it is first encountered. Later

Proceedings of the 4th International Workshop on Semantic Evaluations (SemEval-2007), pages 476–479,
Prague, June 2007. ©2007 Association for Computational Linguistics

research (Barzilay and Elhadad, 1997) significantly alleviated this problem at the cost of a worse running time (quadratic); computational inefficiency is due to their processing of many possible combinations of word senses in the text in order to decide which assignment is the most likely. Silber and McCoy (2003) presented an efficient linear-time algorithm to compute lexical chains, which models Barzilay's approach, but nonetheless has inaccuracies in WSD.

More recently, Galley and McKeown (2003) suggested an efficient chaining method that separated WSD from the actual chaining. It performs the WSD before the construction of the chains. They showed that it could achieve more accuracy than the earlier ones. Our method follows the similar technique with some new semantic relations (i.e.: gloss, holonym, meronym).

3 Semantic Relations

We used WordNet2.1[1] (Miller, 1990) and eXtended WordNet (Moldovan and Mihalcea, 2001) as our knowledge source to find the semantic relations among the words in a context. We assigned a weight to each semantic relation. The relations and their scores are summarized in the table 1.

4 System Overview

The global architecture of our system is shown in Figure 1. Each of the modules of the system is described below.

4.1 Context Processing

Context-processing involves preprocessing the contexts using several tools. We have used the following tools:

Extracting the main text: This module extracts the context of the target word from the source xml document removing the unnecessary tags and makes the context ready for further processing.

Sentence Splitting, Text Stemming and Chunking: This module splits the context into sentences, then stems out the words and chunks those**.** We used OAK systems[2] (Sekine, 2002) for this purpose.

Candidate Words Extraction: This module extracts the candidate words (for task 7: noun, verb, adjective and adverb) from the chunked text.

4.2 All Sense Representation

Each candidate word is expanded to all of its senses. We created a hash representation to identify all possible word representations, motivated from Galley and McKeown (2003). Each word sense is inserted into the hash entry having the index value equal to its synsetID. For example, athlete and jock are inserted into the same hash entry (Figure 2).

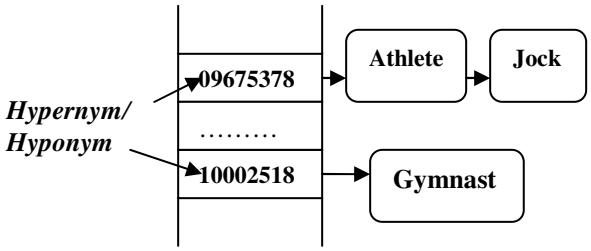

Figure 2. Hash indexed by synsetID

On insertion of the candidate sense into the hash we check to see if there exists an entry into the index value, with which the current word sense has one of the above mentioned relations. No disambiguation is done at this point; the only purpose is to build a representation used in the next stage of the algorithm. This representation can be shown as a disambiguation graph (Galley and McKeown, 2003) where the nodes represent word instances with their WordNet senses and weighted edges connecting the senses of two different words represent semantic relations (Figure: 3).

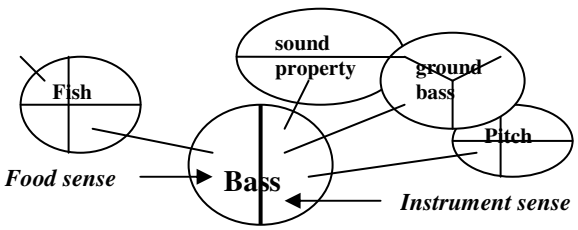

Figure 3. Partial Disambiguation graph, Bass has two senses, 1. Food related 2. Music instrument related sense. The instrument sense dominates over the fish sense as it has more relations (score) with the other words in the context.

[1] http://wordnet.princeton.edu/
[2] http://nlp.cs.nyu.edu/oak/

4.3 Sense Disambiguation

We use the intermediate representation (disambiguation graph) to perform the WSD. We sum the weight of all edges leaving the nodes under their different senses. The one sense with the highest score is considered the most probable sense. For example in fig: 3 Bass is connected with three words: Pitch, ground bass and sound property by its instrument sense and with one word: Fish by its Food sense. For this specific example all the semantic relations are of Hyponym/Hypernym type (score 0.33). So we get the score as in table 2.

In case of tie between two or more senses, we select the one sense that comes first in WordNet, since WordNet orders the senses of a word by decreasing order of frequency.

Sense	**Mnemonic**	**Score**	**Disambiguated Sense**
4928349	Musical Instrument	3*0.33 =0.99	Musical Instrument (4928349)
7672239	Fish or Food	0.33	

Table 2. Score of the senses of word "Bass"

Relation	**Definition**	**Example**	**Weight**
Repetition	Same occurrences of the word	***Weather*** is great in Atlanta. Florida is having a really bad ***weather***.	1
Synonym	Words belonging to the same synset in WordNet	Not all ***criminals*** are ***outlaws***.	1
Hypernym and Hyponym	*Y* is a hypernym of *X* if *X* is a (kind of) *Y* ***And*** *X* is a hyponym of *Y* if *X* is a (kind of) *Y*.	Peter bought a ***computer***. It was a Dell ***machine***.	0.33
Holonym And Meronym	*Y* is a holonym of *X* if *X* is a part of *Y* ***And*** *X* is a meronym of *Y* if *X* is a part of *Y*	The ***keyboard*** of this ***computer*** is not working.	0.33
Gloss	Definition and/or example sentences for a synset.	Gloss of word ***"dormitory"*** is {a college or ***university*** building containing living quarters for students}	0.33

Table 1: The relations and their associated weights

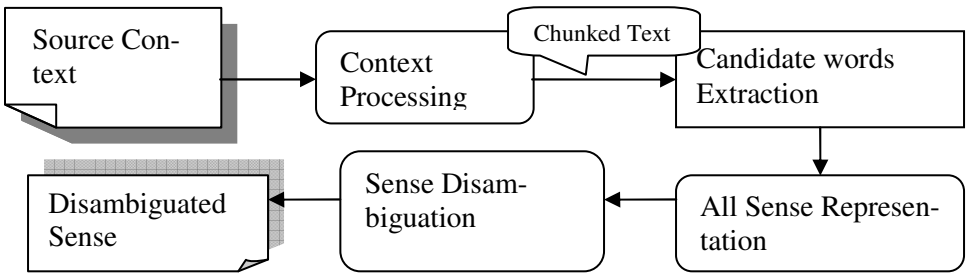

Figure 1: Overview of WSD System

5 Evaluation

In SemEval-2007, we participated in Task 7: "Coarse-grained English all-words". The evaluation of our system is given below:

Cases	Precision	Recall	F1-measure
Average	0.52592	0.48744	0.50595
Best	0.61408	0.59239	0.60304
Worst	0.44375	0.41159	0.42707

6 Conclusion

In this paper, we presented briefly our WSD system in the context of SemEval 2007 Task 7. Along with normal WordNet relations, our method also included additional relations such as repetition and gloss using semantically enhanced tool, eXtended WordNet. After disambiguation, the intermediate representation (disambiguation graph) can be used to build the lexical chains which in tern can be used as an intermediate representation for other NLP applications such as text summarization, question answering, text clustering. This method (summing edge weights in selecting the right sense) of WSD before constructing the chain (Gallery and McKeown, 2003) outperforms the earlier methods of Barzilay and Elhadad (1997) and Silber and McCoy (2003) but this method is highly dependent on the lexical cohesion among words in a context. So the length of context is an important factor for our system to achieve good performance. For the task the context given for a tagged word was not so large to capture the semantic relations among words. This may be the one of the reasons for which our system could not achieve one of the best results.

References

Barzilay, R. and Elhadad, M. 1997. Using Lexical Chains for Text Summarization. In *Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics and the 8th European Chapter Meeting of the Association for Computational Linguistics*, Workshop on Intelligent Scalable Test Summarization, pages 10-17, Madrid.

Chali, Y. and Kolla, M. 2004. Summarization techniques at DUC 2004. In *Proceedings of the Document Understanding Conference,* pages 105 -111, Boston. NIST.

Galley, M. and McKeown, K. 2003. Improving Word Sense Disambiguation in Lexical Chaining. In *Proceedings of the 18th International Joint Conference on Artificial Intelligence (IJCAI'03)*, pages 1486-1488, Acapulco, Mexico.

Halliday M. and Hasan R. 1976. Cohesion in English. *Longman*, London.

Harabagiu S. and Moldovan D. 1998. WordNet: An Electronic Lexical Database, chapter Knowledge Processing on an Extended WordNet. *MIT press*.

Hirst G. and St-Onge D. 1997. Lexical Chains as representation of context for the detection and correction of malapropisms. In *Christiane Fellbaum, editor, WordNet: An Electronic Lexical Database and Some of its Applications. MIT Press,* pages 305-332.

Morris J. and Hirst. G. 1991, Lexical Cohesion Computed by Thesaural Relations as an Indicator of the Structure of Text .*Computational Linguistics*, 17(1):21-48.

Silber H.G. and McCoy K.F. 2002. Efficiently Computed Lexical Chains As an Intermediate Representation for Automatic Text Summarization. *Computational Linguistics,* 28(4):487-496.