Gloss-based Algorithm for Disambiguation

Ganesh Ramakrishnan
Department of Computer Science
IIT Bombay
India - 400076
hare@cse.iitb.ac.in

B. Prithviraj
Department of Computer Science
IIT Bombay
India - 400076
prithvir@cse.iitb.ac.in

Pushpak Bhattacharyya
Department of Computer Science
IIT Bombay
India - 400076
pb@cse.iitb.ac.in

Abstract

The task of word sense disambiguation is to assign a sense label to a word in context. We explore a method of sense disambiguation through a process of "comparing" the current context for a word against a repository *contextual clues* or *glosses* for each sense of each word. These *glosses* are all compiled using WordNet and are of various types like *hypernymy* glosses, *descriptive glosses etc.*. The "comparison" could be done in a variety of ways that could include/exclude *stemming*, expansion of one gloss type with another gloss type, *etc*. The results show that the system does best when stemming is used and glosses are expanded.

1 Introduction

We have formulated a gloss (WordNet glosses) based algorithm for disambiguation of words. Different types of glosses, based on different types of relations in WordNet like *hypernymy, holonymy etc.* are used. The main idea behind this approach is to use the context to find the correct sense of the word using its gloss.

2 Gloss

Querying WordNet for the noun *boy* gives the following output:

The entry for *boy* sense no.1 has synonyms - { *male child, boy* }, gloss - { *a youthful male person* } and examples - { *the baby was a boy, she made the boy brush his teeth every night, most soldiers are only boys in uniform* }. The gloss in our algorithm would refer to all these 3 entries. So for example our gloss for *male#n#1 (n is the part of speech and 1 denotes the sense no.)* would be the set of words - { *male child boy a youthful male person the baby was a boy she made the boy brush his teeth every night most soldiers are only boys in uniform* }.

Senses of *boy* The noun boy has 4 senses (first 4 from tagged texts)

1. male child, boy – (a youthful male person; "the baby was a boy"; "she made the boy brush his teeth every night"; "most soldiers are only boys in uniform").
2. boy – (a friendly informal reference to a grown man; "he likes to play golf with the boys").
3. son, boy – (a male human offspring; "their son became a famous judge"; "his boy is taller than he is").
4. boy – (offensive term for Black man; "get out of my way boy").

Figure 1: Noun senses for *boy*

3 Types of Gloss

There can be different types of glosses depending on the relations in WordNet.

1. *Lesk :* These glosses contain the synonyms, examples and the WordNet gloss of a sense of the word and the same attributes of its immediate hypernym. Consider the sense 3 of *boy*,

   son, boy – (a male human offspring; ”their son became a famous judge”; ”his boy is taller than he is”)
   => male offspring, man-child – (a child who is male)

   Lesk gloss for sense 3 of noun *boy* would be - {*son boy male human offspring their became a famous judge his is taller than he man-child child who* }

2. *Lin :* They contain the synonyms of the word together with its hypernyms. Consider the sense 3 of noun *boy*,

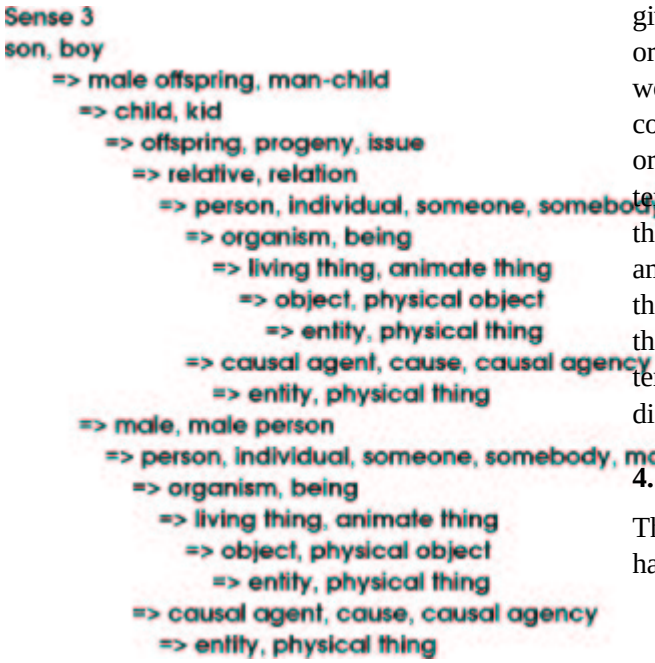

Figure 2: Hypernyms of boy#n#3

.

So the *Lin* gloss for boy#n#3 is - {*son boy male offspring male-child child kid offspring progeny issue relative relation person individual someone somebody mortal human soul organism being living thing animate object physical entity causal agent cause agency person*}

3. *Lin-Lesk-hyper :* It contains both the *Lin* and the *Lesk* gloss for a word.

4. *Lin-Lesk-Holo :* It contains the *Lin* gloss, *Lesk* gloss and the Holonyms for a word.

4 Main Algorithm

The basic idea is to find the content words in the context of the ambiguous word and then find their intersection(common words in the context and the gloss) with the gloss of each sense of the word. The scores are based on the intersections. The senses are then ordered with respect to their scores. So soft word sense disambiguation is done.

During initialization we first find the frequency of the words occurring in the WordNet glosses. The inverse document frequency (idf) is taken as the inverse of the frequency of the words. Now given a document we take a window of 1 sentence or more from it. In this window we select one word at a time and treat the rest of the words as context words. The context can be taken as it is or they also can be expanded to their glosses. Intersection is found between this set of words and the gloss of each sense of the word to be disambiguated. The score is found from the idf of the common words. The senses are given out in their order of scores. There are several parameters which can be changed in the algorithms. We discuss them below.

4.1 Parameters

The algorithm has several parameters and each one has an influence on the result.

1. **GlossType :** This shows the type of gloss being used in the algorithm. It can be *lin, lesk, lin-lesk-hyper* or *lin-lesk-holo*.

2. **Stemming :** Sometimes the words in the context are related semantically with the gloss of the ambiguous word but they may not be in the same morphological form. For example, suppose that the context contains the word *Christian* but the gloss of the word contains the word *Christ*. The base form of both the words is *Christ* but since they are not in the same morphological form they will not

be treated as common words during intersection. Stemming of words may prove useful in this case as after stemming both will give the same base form.

3. **FullContextGlossExpansion :** It shows whether the gloss of the context words should be taken or not. If set true then the gloss of all the senses of the context words would also be included for intersection.

4. **WindowSize :** The window size can be 1 sentence, 2 sentence etc and may also be 1 paragraph, 2 paragraph etc. It shows the total context window. The words to be disambiguated are taken from this window one by one while the rest of the words serve as context word for the ambiguous word.

5 Experimental Results

The program was evaluated against Semcor and was also used in Senseval-3 competition. We present the results in this section.

5.1 Results for Semcor

For experiments we chose the Semcor 1.7 corpus. It has been manually tagged using WordNet 1.7 glosses. ReRank1 denotes the percentage of cases where the highest scoring sense is the correct sense while ReRank2 denotes the percentage of cases where out of the first two highest senses one is correct. Note that we take the first sense of the word if the score is 0 for all the senses.

Stemming	WindowSize	FullGloss	POS	ReRank1	R
No	1 Sent	true	n	50.3%	
No	1 Sent	true	v	29.1%	
No	1 Sent	false	n	71.4%	
No	1 Sent	false	v	41.5%	
No	2 Sent	true	n	47.7%	
No	2 Sent	true	v	26.4%	
No	2 Sent	false	n	49.1%	
No	2 Sent	false	v	24.9%	
No	3 Sent	false	n	47.3%	
No	3 Sent	false	v	25.5%	

Table 1: Results for *Lin* glosses

Stemming	WindowSize	FullGloss	POS	ReRank1	R
Yes	1 Sent	true	n	62.2%	
Yes	1 Sent	true	v	36.6%	
No	2 Sent	true	n	57.04%	
No	2 Sent	true	v	34.2%	
Yes	2 Sent	true	n	45.8%	
Yes	2 Sent	true	v	22.8%	
Yes	2 Sent	false	n	58.13%	
Yes	2 Sent	false	v	34.03%	
Yes	3 Sent	false	n	54.7%	
Yes	3 Sent	false	v	31.4%	
Yes	3 Sent	true	n	47.7%	
Yes	3 Sent	true	v	24.4%	

Table 2: Results for *Lesk* glosses

5.2 Results for Senseval-3 task

Senseval is an online competition for evaluation of the strengths and weaknesses of WSD programs with respect to different words and different languages. The third Senseval competition (Senseval-3) is taking place currently. The task we attempted was disambiguation of WordNet glosses. The input was given in xml format and was pos-tagged. We used *Lesk* glosses and sentence window size of 1 sentence. The results are presented in the table **??**.

The results of our gloss based disambiguation system show that an optimal configuration of the parameters is essential to get good results. Most of the time *lesk* glosses together with stemming give better results than other. But it may be worthwhile to find out the weightage for different types

Stemming	WindowSize	FullGloss	POS	ReRank1	ReRank2	Total Words
No	1 Sent	true	n	43%	61.5%	36014
No	1 Sent	true	v	21.4%	35.8%	17705
Yes	1 Sent	true	n	41.3%	59.3%	7676
Yes	1 Sent	true	v	21.1%	36%	3651
No	2 Sent	false	n	53.6%	74.9%	4203
No	2 Sent	false	v	29.7%	50.6%	2032
No	3 Sent	false	n	50.9%	73.1%	3694
No	3 Sent	false	v	29%	47.8%	1796

Table 3: Results for *lin-lesk-hyper* glosses

Stemming	WindowSize	FullGloss	POS	ReRank1	ReRank2	Total Words
No	1 Sent	true	n	49.18%	71.5%	8004
No	1 Sent	true	v	26.37%	43.8%	3860
No	2 Sent	false	n	62.75%	79.7 %	23938
No	2 Sent	false	v	37.5%	58.6 %	10862
No	2 Sent	true	n	48.2%	73.2%	4051
No	2 Sent	true	v	26%	43.3%	1947
No	3 Sent	true	n	48.5%	74.3%	2886
No	3 Sent	true	v	25%	43.5%	1372
No	3 Sent	false	n	61.08%	77.75%	5737
No	3 Sent	false	v	35.6%	54.7%	2815

Table 4: Results for *lin-lesk-holo* glosses

of glosses and use all of them together. The reason behind some of the high scores is that when there are no common words between the gloss and the context words, the score is zero and so the first sense(which is the most frequently used sense) is taken as the correct sense.

Stemming	WindowSize	FullGloss	POS	ReRank1	R
Yes	1 Sent	false	n	72.9%	
Yes	1 Sent	false	v	43.5%	
Yes	1 Sent	true	n	65.1%	
Yes	1 Sent	true	v	26.2%	

Table 5: Senseval-3 task of disambiguation of WordNet glosses