Words sense disambiguation 2

Anna Hjalmarsson & Preben Wik

1.0 The basic Idea of SenseRelate

The SenseRelate project at the University of Minnesota implements an unsupervised learning algorithm for word sense disambiguation. Word sense disambiguation is concerned with assigning the correct sense to a target word based on the context in which it occurs. The idea behind SenseRelate is that words that occur together in a sentence are semantically related. Quillian [3] introduced the idea that a dictionary can be represented as a semantic network. Each node in the network represents a word and its neighboring nodes represents words used to define the concept of the original word in a dictionary. The neighboring nodes of these nodes are in turn the words that define the concepts of these words. Quillian's work is an early example of utilizing *gloss overlaps*, words that share words in dictionary definitions. SenseRelate introduce extended gloss overlaps in a lexical database as a measure for semantic relatedness.

1.1 Measure of semantic relatedness and similarity

SenseRelate utilizes the Lesk algorithm to measure gloss overlap. The algorithm attempts to identify the most likely meaning for a word in a given context based on the semantic relations between the target word and its neighboring words in a sentence. The word is disambiguated by calculating which sense's definition shares most words with the definitions of the neighboring words. The SenseRelate algorithm returns a measure of *relatedness*, a similarity score. Relatedness is a function that takes two senses of a word as input and outputs a real number. The output is the number of words that overlap in the two senses, and the larger number; the more related are the two word senses. The algorithm calculates the relatedness between each neighboring word and all the possible senses of the target word. An advantage of gloss based measures is that they can be used to compare concepts of different part of speech. As used in Pedersen et al [3] the extended gloss overlap measure weighs phrasal matches more heavily than word overlaps. An n word overlap is assigned the score of n^2. If several senses receive the same highest score all senses are reported and no attempt is made to select a single sense.

1.2 WordNet

SenseRelate utilizes WordNet (http://wordnet.princeton.edu/), a machine readable dictionary with information about nouns, verbs, adjectives and adverbs. A word is represented by a synonym set (synset). Each synset has a gloss that defines the concept that it represents. Synonyms constitute a single synset with the same gloss. The semantic relations between the synsets are explicitly defined in WordNet such as *synonymy*, *is-a*, *part-of* or *antonymy*. In general these relations only connect word senses that are used in the same part of speech. For example the most commonly used relation for nouns is is-a. The more general concepts in WordNet are higher in the hierarchy than more specific ones. The path lengths in WordNet can consequently be interpreted differently depending on where they occur. However, the Lesk algorithm works best when the path lengths

have a relatively consistent interpretation. Therefore Pedersen et. al have incorporated a number of correcting factors when calculating the path lengths. An extended gloss overlap was also developed to overcome the limitations of too short definitions. Instead of only trying to find overlaps between the two concepts being compared the extended gloss overlap also search for overlaps among the concepts to which they are related.

2.0 Our experiments evaluating its performance

In is difficult to evaluate WSD results. For instance, different types of texts can be involved, including both highly technical or domain specific texts where sense use is limited as opposed to general texts where sense use may be more variable. WordNet is generic and all kinds of different texts can be used for evaluation. The most frequent heuristic baseline [4] is when WSD results are compared to a WSD which always selects the most frequent sense for a specific word. However, this will still depend on which dictionary is used, and the material that the probabilities are based on. A second approach is to use pre-tagged test material [2]. Still, there is no total agreement between human judges. We evaluated SenseRelate on a number of sentences. We selected sentences we knew were difficult to disambiguate, as well as a regular paragraph from an online newspaper.

Wsd.pl is a 'convenience-script' over the WordNet-SenseRelate-AllWords-0.04 package, were you can specify a file to be disambiguated. The output from wsd.pl is the list of words from the input file, in their base form, with the sense number from WordNet attached to it. In order to see whether SenseRelate has done a correct wsd, one has to look the words up in WordNet one at a time, and see if the correct sense definition corresponds with the one selected by SenseRelate. For example the famous Groucho Marx sentence:

> *"Time flies like an arrow. Fruit flies like a banana"*
> – got the result:
> `Time#n#5 fly#n#1 like#a#1 an#n#1 arrow#n#2`
> `Fruit#n#1 fly#n#1 like#a#1 a#n#5 banana#n#1`

We see immediately that the two tokens of 'flies' have been incorrectly classified the same way, but we cannot tell which sense is chosen. A check in WordNet on the word 'fly' yields:

The noun fly has 5 senses (first 4 from tagged texts)

1. (6) fly -- (two-winged insects characterized by active flight)
2. (1) tent-fly, rainfly, fly sheet, fly, tent flap -- (flap consisting of a piece of canvas that can be drawn back to provide entrance to a tent)
3. (1) fly, fly front -- (an opening in a garment that is closed by a zipper or buttons concealed by a fold of cloth)
4. (1) fly, fly ball -- ((baseball) a hit that flies up in the air)
5. fly -- (fisherman's lure consisting of a fishhook decorated to look like an insect)

The verb fly has 14 senses (first 9 from tagged texts)

1. (33) fly, wing -- (travel through the air; be airborne; "Man cannot fly")
2. (9) fly -- (move quickly or suddenly; "He flew about the place")
3. (5) fly, aviate, pilot -- (fly a plane)
4. (3) fly -- (transport by aeroplane; "We fly flowers from the Caribbean to North America")
5. (2) fly -- (cause to fly or float; "fly a kite")
6. (2) fly -- (be dispersed or disseminated; "Rumors and accusations are flying")
7. (2) fly -- (change quickly from one emotional state to another; "fly into a rage")

8. (1) fly, fell, vanish -- (pass away rapidly; "Time flies like an arrow"; "Time fleeing beneath him")
9. (1) fly -- (travel in an airplane; "she is flying to Cincinnati tonight"; "Are we driving or flying?")
10. fly -- (display in the air or cause to float; "fly a kite"; "All nations fly their flags in front of the U.N.")
11. flee, fly, take flight -- (run away quickly; "He threw down his gun and fled")
12. fly -- (travel over (an area of land or sea) in an aircraft; "Lindbergh was the first to fly the Atlantic")
13. fly -- (hit a fly)
14. vanish, fly, vaporize -- (decrease rapidly and disappear; "the money vanished in las Vegas"; "all my stock assets have vaporized")

The adj fly has 1 sense (no senses from tagged texts)

1. fly -- ((British informal) not to be deceived or hoodwinked)

The two-winged insect sense of the word has been selected. This was wrong for the first sentence, but right for the second sentence.

Another feature with wsd.pl is that the input-files can be of different formats. Either raw text or PoS tagged text. The example below demonstrates that having a text that is already Part-of-speech tagged can help in the disambiguation process (or at least affect the process). In the untagged example '*flies*' has been incorrectly classified as a noun and then given sense 1.

> *An airplane flies high and fast.*
> `An#n#1 airplane#n#1 fly#n#1 high#a#2 and fast#a#1`

And then the same sentence already PoS-tagged:

> *An/DT airplane/NN flies/VBZ high/JJ and/CC fast/RB*
> `An airplane#n#1 fly#v#3 high#a#4 and fast#r#1`

The correct sense of the word should have been `fly#v#1` (see above), and the word '*high*' has for some reason been changed from:
high#a#2 (being at or having a relatively great or specific elevation or upward extension (sometimes used in combinations like `knee-high'); "a high mountain"; "high ceilings"; "high buildings"; "a high forehead"; "a high incline"; "a foot high")
to:
high#a#4. high, high-pitched -- (used of sounds and voices; high in pitch or frequency)

Perhaps the most common example used to explain word-sense disambiguation in English, is the word *bank*. In Pedersen [3]:

> [For example, suppose we wish to disambiguate *bank*, in the sentence *I sat on the bank of the lake*. Suppose that *bank*1 is defined as *financial institution that accepts deposits and channels the money into lending activities*, and *bank*2 is defined as *sloping land especially beside a body of water*. Suppose that *lake* is only defined with one sense, *a body of water surrounded by land*. There are no overlaps between *bank*1 and the sense of *lake*, but there are are two content words that overlap between *lake* and *bank*2, *body* and *water*. Thus, the Lesk algorithm would determine that *bank*2 is the appropriate sense in this context.]

We tried Pedersen's example sentence, *"I sat on the bank of the lake"* as well as a sentence with an alternative sense *"I deposited my cash in the bank of America"*.

The result was not in favor of the performance of the SenseRelate package:

I sat on the bank of the lake
`I#n#1 sit#v#6 on#r#3 the bank#n#1 of the lake#n#1`

I deposited my cash in the bank of America
`I#n#1 deposit#v#2 my cash#n#1 in#n#3 the bank#n#1 of America#n#1`

The example used by Pedersen [3] when they say "suppose Bank1 is defined as... and Bank2 is defined as....and lake is defined as.... " are taken directly from the WordNet definitions.

(bank#n#1 = [a financial institution that accepts deposits...])

The only discrepancy between the quote above and our WordNet definitions is that *lake* has two more senses:

2. lake -- (a purplish red pigment prepared from lac or cochineal)
3. lake -- (any of numerous bright translucent organic pigments)

However, neither definition seems to have any overlap with the definition of "Bank1". How come the first sentence didn't manage to define *bank* as "Bank2"? If Pedersen et. al are using the Lesk algorithm as described one would expect the outcome to be differently. Perhaps the correcting factors Pedersen et. al have incorporated when calculating the path lengths have corrupted the results?

2.1 Word sense disambiguation of a news paper text

As a part of the evaluation a short item from New York Post (http://www.nypost.com/) was disambiguated using SenseRelate. We included the text as a part of the evaluation in order to study how well SenseRelate work on more general material such as news paper corpus. The text used was:

> **Providence Detective Killed by Own Gun**
> PROVIDENCE - A Providence detective was killed with his own gun at police headquarters Sunday by a suspect who was not handcuffed and managed to get hold of the weapon, the police chief said. The killing of James Allen, a 27-year veteran, comes after a series of attacks that have raised concerns about the security of those who work in the criminal justice.

WordNet only includes information about nouns, verbs, adjectives and adverbs. The assignment of senses to words which part of speech tags were not available in WordNet appears to be problematic. These words were often assigned senses with the wrong part of speech tags. *At* in "…at police quarters" is assigned the sense of: "the atomic number of the element astatine" and both occurrences of *who* are in tagged as the World Health Organization. The proper noun *James* is assigned the sense: "occlude, jam, block". These problems can possibly be avoided if the text is pre-tagged. SenseRelate allows for this option. If a part-of-speech tagger had been used in advance and *at* had been assigned the correct tag (preposition) the word would not have been assigned any sense at all. This would be the correct choice based on that fact that information about prepositions is not

available in WordNet. Apart from this problem SenseRelate appears to have a number of difficulties with the news paper text. This table presents some of them:

Word	Sense assigned	Correct sense
criminal	felon, crook, outlaw, malefactor	relating to crime or its punishment; "criminal court"
work	workplace	activity directed toward making or doing something
headquarters	military unit consisting of a commander and the headquarters staff	central office, main office, home office, home base
attack	military an offensive against an enemy	the act of attacking
security	a formal declaration that documents a fact of relevance to finance and investment	the state of being free from danger or injury
concern	business concern, business organization	something that interests you because it is important or affects you;
raise	raise from a lower to a higher position	call forth emotions, feelings, and responses
come	move toward, travel toward	come to pass; arrive, as in due course

The causes of the problems are not always obvious. The WSD of *criminal* and *work* will probably benefit from pre-part-of-speech tagging. The correct senses of these words would probably be found if they had been pre-tagged with the correct parts of speech (verb and adjective). When it comes to *headquarters*, *attack*, *security* and *concern* SenseRelate appears to have found gloss overlaps where they should not have been in this specific context. It appears that the boundaries between different senses of homographs are not always as clear as for the *bank*, the example in [3]. SenseRelate would actually have performed better here if the gloss overlap between *headquarters* and something related to the military had not been found since headquarters would have been assigned the most frequent sense, “main office”, which is the correct sense. *Raise* and *come* have both been assigned senses with the correct part of speech, but they are still wrong. “Concern”, in raised *concern*, could possibly contribute to an assignment of the correct sense, “call forth emotions”, but *concern* is wrongly disambiguated and no gloss overlaps are found.

3.0 Our thoughts on where it might be useful

SenseRelate is not lightning-fast. The long processing times restrict the use of SenseRelate. It is consequently not suitable for disambiguation in spoken dialogue systems, or other systems that require real-time use.

An airplane flies high and fast.
The truck travels treacherously.
Submarines travel slowly and silently.

```
An#n#1 airplane#n#1 fly#n#1 high#a#2 and fast#a#1
The truck#n#1 travel#v#1 treacherously#r#1
Submarine#n#1 travel#v#1 slowly#r#1 and silently#r#1
```

This takes 3 min to complete. Pre-tagged material is faster:

An/DT airplane/NN flies/VBZ high/JJ and/CC fast/RB
The/DT truck/NN travels/VBZ treacherously/RB
Submarines/NNS travel/VBP slowly/RB and/CC silently/RB

```
An airplane#n#1 fly#v#3 high#a#4 and fast#r#1
The truck#n#1 travel#v#1 treacherously#r#1
Submarine#n#1 travel#v#1 slowly#r#1 and silently#r#1
```

(0.40 min)

Time flies like an arrow. Fruit flies like a banana.
(1.40 min).

The frequency hierarchy in WordNet is probably not appropriate for disambiguating very domain-specific corpora since the frequencies in these texts will be differently distributed. According to Pedersen [3] the extended gloss overlap are supposed to look out for this using a number of correcting factors. Still, the results in our evaluations implies that SenseRelate often choose the most frequent sense of a word. Whether this is caused by the frequency hierarchy in WordNet, too short definitions in WordNet or that the correcting factors in the extended gloss overlap is not working satisfactory is difficult to say.

4.0 References

[1] Banerjee, S., and Pedersen, T., "Extended gloss overlaps as a measure of semantic relatedness" In *Proceedings of the Eighteenth International Conference on Artificial Intelligence (IJCAI-03),* Acapulco, Mexico, August 2003.

[2] Ide, N., and Véronis, J., "Word Sense Disambiguation: The State of the Art", *Computational Linguistics*, Vol 14, Part 1, 1998

[3] Pedersen, T., Banerjee, S., and Patwardhan, S., "Maximizing Semantic Relatedness to Perform Word Sense Disambiguation, 2003

[4] Ng, Hwee Tou, & Zelle, John "Corpus-Based Approaches to Semantic Interpretation in Natural Language Processing", *AI Magazine, 18(4) (pp. 45 – 64), Special Issue on Natural Language Processing*, 1997

5.0 Appendix: Installing and running SenseRelate

Installing and getting these packages to run was quite time consuming. There are several dependencies that will briefly be described below.

We started with a windows installation, but after a while, reading the install instructions on some of the accompanying packages, decided to try on a Linux installation instead. The reason being that they only mention how to install it in Unix. This was not trivial either however, since the WordNet installation that must be installed in the bottom to try out the packages in the assignment. They mentioned that it had been successfully compiled on RedHat, but it did not successfully install in our Linux distribution. After some futile attempts, we gave up, and seriously considered switching assignments. We decided to give it a last try, and went back to the windows installation, since we had successfully installed WordNet on windows (just double-click and go!).

Some of the files from four different distributions had to be copied into specific places in the Perl distribution. Which files to use, and where to place them was not obvious, and not documented. It was resolved after several tests trying a script that resulted in errors.

From the WordNet-QueryData-1.37 distr. the file:
    QueryData.pm
Was copied into: C:\Perl\site\lib\WordNet
(The folder WordNet was created in: C:\Perl\site\lib)

From the WordNet-SenseRelate-AllWords-0.04 distr. the file:
    WordNet-SenseRelate-AllWords-0.04\lib\WordNet\SenseRelate\AllWords.pm
Was copied into : C:\Perl\site\lib\WordNet\SenseRelate

From the WordNet-Similarity distr. the files:
    WordNet-Similarity\lib\vectorFile.pm
    WordNet-Similarity\lib\stem.pm
    WordNet-Similarity\lib\get_wn_info.pm
    WordNet-Similarity\lib\WordNet\lesk-relation.dat
    WordNet-Similarity\lib\WordNet\Similarity.pm
    WordNet\Similarity\ all the files (14)
Were copied into C:\Perl\site\lib\WordNet\

Text-Similarity-0.02
All the files in Text-Similarity-0.02\lib\Text
    Similarity\Overlaps.pm
    OverlapFinder.pm
    Similarity.pm
Were copied into: C:\Perl\site\lib\Text

We found two test-scripts - WordNet-SenseRelate-AllWords.t and wsd.t. The two test-scripts were run successfully (after they had been moved up one step in the directory hierarchy first), giving us an indication that everything needed were installed. Looking at the code of the test-scripts also gave us some useful hints as to how to use the modules.