

Lexical level – Sentence Context.

Author: Ravi Kiran Holur Vijay, Computer Science department, Columbia University, NY.

Word ambiguity induced by context of use can happen when a word can have multiple interpretations based on the context in which it is used. Here, the required context is provided only by the sentence in which the word appears. This happens when a word can have different interpretations that are dependent on only the other words in the sentence. On a general note, this might include polysemes and homonyms. E.g. Financial bank vs. River bank.

Setting up a test bed.

We consider a couple of manually transcribed game sessions of Battlefield 2. A perl program is run on each of these two textual files to obtain values for two metrics. The two metrics considered are:

- **Score output of a WSD algorithm.**
- **Standard Deviation of the frequencies of various senses from WordNet for the given word.**

Here are the details of each of the two metrics:

- **Score output of a WSD algorithm:** Since we don't have a really large corpora, I tried some unsupervised dictionary based WSD algorithms. Specifically, I implemented the Simplified Lesk and WordNet Lesk algorithms for the purpose of measuring this metric. The Simplified Lesk is described in the book **“Word Sense Disambiguation by Eneko Agirre, Philip Glenn Edmonds”** and also in the paper **“Evaluating variants of the Lesk Approach for Disambiguating Words”** by **Florentina Vasilescu, Philippe Langlais and Guy Lapalme**. The WordNet lesk is described in the paper **“Adapting the Lesk Algorithm for Word Sense Disambiguation to WordNet”** by **Satanjeev Banerjee**. Both of these algorithms take a context and a target word as input and gives the most probable sense for the target word as output, using WordNet as the dictionary. But, we needed a score using which we could determine how probable the best probable sense was. For this purpose, we take the score as the number of overlaps of the sense with maximum overlaps and normalize it. In the remaining sections, this metric is referred to as **“Score”**.
- **Standard Deviation of the frequencies of senses from WordNet for the given word:** This was motivated by the rationale that if a word has a sense that is used very rarely and another sense that is used very often, this might lead to misunderstanding if the other person doesn't know about the use of this rare sense. For example, if we lookup **“fire”** in WordNet, we get frequencies 616, 106, 99, 30 etc. This shows a lot of deviation and hence the use of fire might lead to misunderstanding. I tried using standard deviation to quantify this variance. This can translate to **“if a word has greater standard deviation, it means it has greater chances of causing misunderstanding”**. In the remaining sections, this metric is referred to as **“SensesSD”**.

Some observations.

The implementation was executed on the transcripts of two different sessions of the Battlefield 2 experiments conducted at the IRT lab in Columbia University, after manually transcribing the voice recordings to text files.

Here are some of the observations:

- If the score is less than 0 or more than 1 (like really, sorry, nice, left, stop, right, back, there etc.), we can see that the words might lead to potential misunderstanding more readily.
- Now, if the score is between 0 and 1, we need to look at SensesSD to decide if the word might lead to potential misunderstanding. Some words like here, there, work, fire, want, think etc. are in this category, with low scores (less than .15) and high SensesSD (greater than 50). Please take a look at the included tables to verify this.
- So, basically we need to identify words with low scores and high SensesSD, which might give us an approximate measure for the misunderstanding it can cause in the given context (sentence).



