Olga Redko
CS366
Assignment 4: Part 2

1. wsd_context_features results in higher accuracy scores for hard.pos, line.pos, and serve.pos.

- hard.pos:
- wsd_context_features accuracy: 0.8950
- wst_word_features accuracy: 0.8593
- line.pos:
- wsd_context_features accuracy: 0.7373
- wst_word_features accuracy: 0.7157
- serve.pos:
- wsd_context_features accuracy: 0.8345
- wst_word_features accuracy: 0.7386

I think that wsd_context_features might be more accurate because it emphasizes taking into account more immediate contextual clues if *m* is not set to a very high number, like the default number 3 (which often spans less than the length of a whole sentence). wsd_word_features, on the other hand, takes all the words of a sentence into account to use as context clues, including words that may be quite distant from the target word. These distant words are often less relevant to the target word than closer words, and they may even be associated with a different sense of the target word. That is why the accuracy may be lower for wsd_context_features in many cases. It might not be fair to compare the accuracy of the classifiers across different target words because the kinds of contexts in which different target words may appear in may vary drastically, with some target words having remarkably consistent (nearby) surrounding words and others not having consistent (nearby) surrounding words.

2. See #1.

- hard.pos:
- Random baseline accuracy: 1/3 (0.333333333333)
- Majority baseline accuracy: 0.797369028386799
- line.pos:
- Random baseline accuracy: 1/6 (0.16666666666667)
- Majority baseline accuracy: 0.0899662325132658
- serve.pos:
- Random baseline accuracy: 1/4 (0.25)
- Majority baseline accuracy: 0.4143444495203289

3.

- Setting the number to less than 300 decreases the model's accuracy. Reducing the context window to a number smaller than 3 increases the model's accuracy.
- Adding stopwords increases accuracy for wsd_word_features but has no effect on wsd_context_features.
- Adding the words "harder" and "hardest" to the stopwords list decreased accuracy, and this is the effect I expected since I can imagine such words being helpful for distinguishing the sense of "hard." Certain phrases that use hard or harder tend to use a certain sense of "hard" more often than others. For example, let's compare "He tried hard, but she tried harder (hard has to do with putting effort in this instance)" with "He

made the chemicals hard, but she made it harder (hard has to do with physical durability in this instance).” At least based on my own personal experiences, I have noticed “harder” being used in the sense of putting effort like in the former sentence more often than in the sense of physical durability like in the latter sentence.

4. Error Analysis:

Note: right guesses for a sense / (wrong guesses + right guesses for a sense) = accuracy

- hard.pos:

- HARD1: 643 / (11 + 643) = 0.983180428
- HARD2: 73 / (51 + 73) = 0.588709677
- HARD3: 60 / (29 + 60) = 0.674157303

HARD2 had the lowest accuracy score, so it seems to be the hardest sense for the model to estimate.

- line.pos:

- cord: 44 / (44 + 44) = 0.5
- division: 59 / (21 + 59) = 0.7375
- Formation: 54 / (31 + 54) = 0.635294118
- phone: 56 / (45 + 56) = 0.554455446
- product: 342 / (25 + 342) = 0.931880109
- text: 57 / (52 + 57) = 0.52293578

cord had the lowest accuracy score, so it seems to be the hardest sense for the model to estimate.

- serve.pos:

- SERVE10: 331 / (17 + 331) = 0.951149425
- SERVE12: 213 / (51 + 213) = 0.806818182
- SERVE2: 134 / (36 + 134) = 0.788235294
- SERVE6: 73 / (41 + 73) = 0.640350877

SERVE6 had the lowest accuracy score, so it seems to be the hardest sense for the model to estimate.

- interest.pos:
- interest_1: 33 / (1 + 33) = 0.970588235
- interest_2: 3 / (251 + 3) = 0.0118110236
- interest_3: 7 / (3 + 7) = 0.7
- interest_4: 12 / (6 + 12) = 0.666666667
- interest_5: 43 / (10 + 43) = 0.811320755
- interest_6: 105 / 105 = 1

Interest_2 had the lowest accuracy score, so it seems to be the hardest sense for the model to estimate.

A pattern I've noticed is that when the hardest word sense is the correct label in an errors.txt file, the surrounding words tend to be associated with other senses more. For example, in
"a little fortress of kindness for each other in a HARD world ."
The classifier guessed HARD3 (hard meaning "resisting weight or pressure") when the label was actually HARD2 (hard meaning "dispassionate"). I'm assuming that the classifier predicted "hard" as having the sense HARD3 because physical inanimate objects like the "world" are often

especially associated with that sense of hard. I don’t think there’s a very easy way to improve the model when it comes to such issues other than to maybe feed it more training data or explicitly specifying such exceptions.