Models performance analysis

- ➔ Larger embeddings, like glove-wiki-gigaword-300, outperform smaller embeddings like glove-wiki-gigaword-200. This relationship seems to be described by a logarithmic curve, implying that the marginal performance gain tends to diminish while adding embeddings. For example, there is more gain when increasing embeddings from 20 to 125 than from 200 to 300. Finally this is not suprising as more embeddings grossly means more categories, hence finer grained criterias to distinguish terms based on their surroundings.

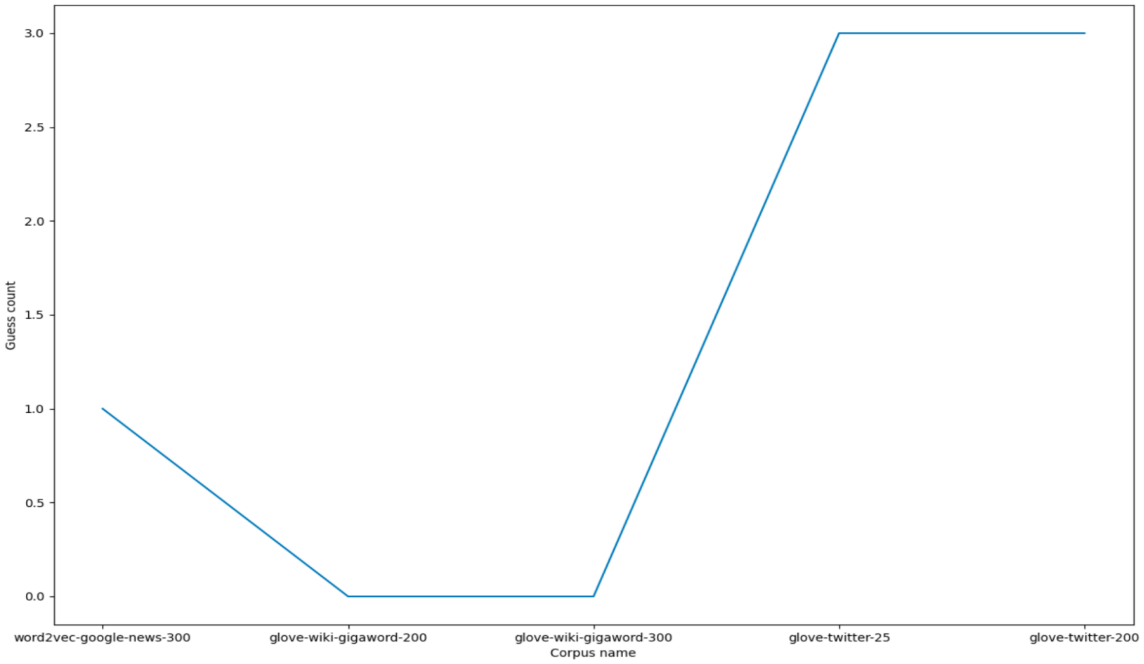
- ➔ The gold standard has 85.57% accuracy on the test. I would like to point out that the score of some annotators in the class seems suspiciously low (20-30% accuracy, hence the large standard deviation) and should therefore be classified as outliers. Such an operation would likely bring the overall accuracy over 90%. Ignoring this we can say that models on the left-hand side (checkout analysis.csv for exact values) have relatively the same accuracy as the gold standard. In fact, they all did slightly better than us. On the right hand side, twitter and random models did much worse than us.

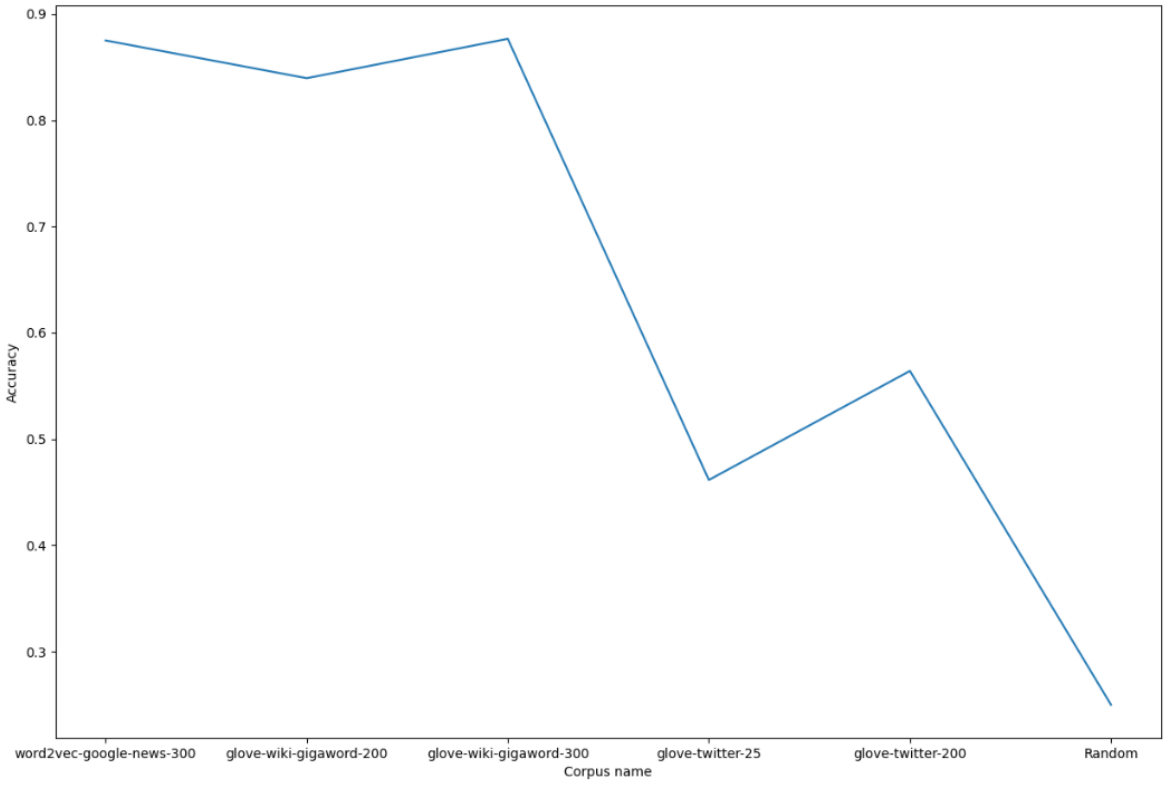
- ➔ Twitter underperforms relatively to its embedding size. To derive this conclusion I compared glove-twitter-300 with glove-wiki-gigaword-200. A possible explanation for this result could be that Twitter's corpus is smaller. Another explanation, the most likely in my opinion, is that Twitters' corpus has a vocabulary less relevant to the synonym test. Since the synonym test evaluates the model on many researched words that are far more likely to appear in articles rather than in Tweets we can say that the glove-twitter models will have fewer training examples with some the words included in the test. From the fundamentals of deeplearning, a lower amount of training data usually implies worse predictions since that the weights are perfected/adjusted over iterations on the training data. This reasoning is based on the premise that more frequent appearances of a word in a corpus gives the model more opportunities to train on its surroundings and adjust its output embeddings to give a better representation of the term. The first graph reinforces this theory by showing the model trained on the twitter corpus made more guesses that than the others (and vice versa since the wikipedia model did none and had best performance). My argument was based on "less" training but guesses are the most explicit form of lack of training data.

- ➔ The random model gave expected results that are close to the mathematical expectancy of 25% valid answers on the test.

- ➔ How to improve the models: my recommandation would be diversidified corpus. News articles, wikipedia pages and tweets all tends to have similiarities publications that resemble each other within the same corpus. A given word is likely used in similar contexts in tweets but could be used differently in a news article. Allowing the model to see a bigger set of usages would allow to refine the embeddings and make better predictions.

**Graphs on the next page


3.0
2.5
2.0
1.5
1.0
0.5
0.0
Guess count
word2vec-google-news-300
glove-wiki-gigaword-200
glove-wiki-gigaword-300
glove-twitter-25
glove-twitter-200
Corpus name


0.9
0.8
0.7
0.6
0.5
0.4
0.3
Accuracy
word2vec-google-news-300
glove-wiki-gigaword-200
glove-wiki-gigaword-300
glove-twitter-25
glove-twitter-200
Random
Corpus name