Text Analytics Homework 2
Megan Hazlett
October 2020
Github: https://github.com/MSIA/mlh9188_msia_text_analytics_2020

Question 1

After completing the lab assignment, experiment with 2 or more sets of word2vec model parameters, for example, different embedding sizes, CBOW vs. skip-gram models, etc. Provide a paragraph or two of qualitative evaluation of your embeddings (no need to provide a quantitative evaluation). For example, you can evaluate the embeddings by hand-picking ~10 words and manually reviewing/evaluating their closest neighbors in terms of cosine similarity.

To evaluate different models of word2vec parameters, I used the following 10 words: olympian, athletes, winter, history, jeopardy, demon, child, marxist, california, and religion. I looked at the top 5 cosine similar words for each of these evaluation words across 4 models (see CSV files). The 4 models I chose to compare were default skip gram, default CBOW, skip gram with size of embedding 50, and CBOW with size of embedding 50.

I found that the models with the default size of embeddings (100) tended to identify more accurate words in cosine similarity than the models with size 50. I also found that in general, for this data set, the skip gram performed better than CBOW. I noticed that the skip gram picked up on more subtle cultural references than CBOW. For example, for California, skip gram's top word was "dreamin", referring to the popular phrase "california dreamin". In comparison, CBOW's top word for California was Florida. I found that across the board, the models struggled most with the word "demon". All of the words picked by the models did not make sense. Humorous examples include "bouncy", "fashionista" and "yeehaw".

Question 2
Read the required embeddings papers (word2vec, Bert, optionally Elmo) and compare the approaches in less than one page, preferably in a table format. You are free to decide what aspects to compare. For example, you can include information such as: learning model details, summary of word context approaches, corpus size requirements, computational requirements, ease of installation/use of source code, date of publication and number of google scholar citations :), … etc.

	Distributed Representations of Words and Phrases and their Compositionality	**BERT: Pre-training Deep Bidirectional Transformers for Language Understanding**	**Deep Contextualized Word Representations**
Authors	Tomas Mikolov, Ilya Sutskever, Kai Chen. Greg Corrado, Jeffrey Dean	Jacob Devlin, Ming-Wei Chang, Kenton Lee, Kristina Toutanova	Matthew E Peters, Mark Beumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, Luke Zettlemoyer
Number of citations	22k	9K	4k
Project name	Word2vec	BERT	ELMo, biLM
Date of publication	2013	2018	2018
Inspiration	Skip gram but, allows it to work on phrases	Previously pre-trained NPL tasks. ELMo algorithm.	Past works of deep contextualized word representations (CoVe, Context2vec, LSTM). Different than traditional embeddings because involved the whole sentence.
Techniques tested	Hierarchical softmax, NCE, Negative sampling, downsampling frequent words	BERT has two steps: pretraining on unlabeled data (with masked LM and next sentence prediction) and fine tuning.	biLM: Uses forward and backwards linear models to predict a word. ELMo: Combination of layer representations in biLM
Results	Hierarchical softmax works best with downsampling. It does not perform well without it. Negative sampling had good success. Word vectors have a linear relationship.	Bidirectional architecture is very beneficial. Using unsupervised pre-training is integral.	Shows good success on question answering, textual entailment, semantic role labeling, conference resolution, named entity extraction, and sentiment analysis.
Corpus size requirements	Large training data is crucial (much more accurate the more data you use)	Scaling model sizes leads to large improvements	pretraining on a large text corpus
Comparative works	Skip gram performs much better than other methods, but this is likely due to large training set; however it is still a lot faster	ELMo	CoVe