Homework 4: Bias in Language Models

Based on a Model AI Practicum designed by
Prof. Ameet Soni and Prof. Krista Thomason of Swarthmore College

Introduction

This homework will introduce you to *word embeddings*, mappings from words (or phrases) to vectors of numbers. These types of methods aim to provide a representation of words that incorporate the context in which words get used. For example, while "motel" and "hotel" are distinct words, we understand them to be fairly similar in meaning; a computer, however, would not know this if given only the words. By placing these two words close to each other (i.e. assigning similar vector values) a word embedding model would provide an algorithm clues to the underlying meaning of these words. A typical application that uses word embeddings is a language translator (e.g., Google Translate).

To learn these embeddings, algorithms such as GloVe and word2vec use corpora, or large data sets of examples of human language, to train the model. For example, you will use models trained either on Twitter or Wikipedia. In this homework, you will get to see the usefulness of these word embeddings. But you will also show that, by training on natural data sets, these models also incorporate cultural biases. Before continuing, please read the following paper for more background:

"Semantics derived automatically from language corpora contain human-like biases". Caliskan, Aylin; Bryson, Joanna J.; Narayanan, Arvind. *Science*, Vol. 356, No. 6334, 14.04.2017, p. 183-186.

Instructions

Both code and data needed for this assignment are available on Piazza under "Resources." Please download those. Before beginning the assignment, first work through the provided code and make sure it works (you should be using Python3, and you may need to change the first line in each of the Python files – you don't need to do any programming for this homework, though). You should refer to the `README.md` file in your code directory for details on each of these steps.

1. Follow the **Setup** instructions in the `README.md` file. This will provide you with pretrained word embedding models; download the Twitter and Wikipedia models. If given a choice for dimensionality of the word vectors, you can use what you want – the smaller ones will make your experiments run faster.

2. Next, follow the instructions to run `findSimilarWords.py`. This program provides a simple proof of concept – do word embedding models do a good job of mapping similar words to a similar location in the embedding (i.e., are similar words close to each other)? Try several search terms on each of the data sets e.g., "dog", "baseball", "mason", etc. and verify that the resulting words are similar to the search term.

3. Finally, run `weatTest.py` according to the provided instructions to perform the evaluation in the Caliskan, Bryson, and Narayanan paper. This program performs a Word Embedding Association Test to detect biases in the word embedding models. You will need to provide four word lists - one each for the two concepts (e.g., European names and African names) and one each for the attribute you are testing for a bias (e.g. pleasant and unpleasant).

Once you have completed these steps, you are able to move on to the assignment. This is a good time to pause and to clear up any confusions or to restate what you have learned to this point.

Assignment Instructions

This is an individual assignment, but again, discussion of the overall problem and your ideas is encouraged. **The only deliverable is your final writeup (2 pages maximum.** Any copying of text in your writeup, or in choices of novel wordlists in (2) below will be treated as an honor code violation.

Using the tools provided for testing associations with pairs of words, explore the implicit bias that can be encoded in natural language word embeddings. First, begin by replicating some of the pairings from the readings (e.g., European names vs. African names when paired with pleasant/unpleasant words). Then, respond to the following two prompts.

1. Rerun the pairings by varying the underlying training corpus used to learn the word embeddings. Discuss what impact, if any, this has on introducing biases into the trained word embedding model. Be sure to try each of Wikipedia and Twitter on both a task related to race/ethnicity and a task related to gender. Is there a noticeable change in each case? Provide some quantitative data to justify your conclusions. Focus more on the reported effect sizes rather than the images WEAT generates.

2. Propose a new set of WEAT pairings to see if there are additional biases (e.g., religion, nationality, class). Generate a list of words for your target pair and use one of the existing attribute pairs (unpleasant/pleasant, family/career, male/female) or create your own. Describe your hypothesis (including how you created your lists of words) and then your findings. Discuss the implications of your results (e.g., what does this tell us about the algorithm, training corpus, and/or the experimental design itself). **Include the lists of words for your target pair in your writeup itself.**

You do not need to submit any code for this assignment, only your report.