Assignment 2

CPSC 599/601: Practical Natural Language Processing

February 1, 2024

This Assignment will evaluate your skills in web scraping and text representation using Word2vec. Include in your submission the code used to generate answers as a Jupyter Notebook or Python program file as well as any files generated. Make sure it is clear what code answers each question.

This Assignment is meant to be completed individually. You may discuss the questions at a high level with other students but the final work submitted must be your own. Please reference any external resources you use to complete this Assignment using ACM referencing format.

Exercise 1 (10 marks)

For this exercise, you will be scraping information from Quotes to Scrape. Write a function to extract the quotes and author names from each page of the site. Write the results into a table that collects the information as separate columns OR store the results as a dictionary if you prefer. Design your method so that it would work regardless of the number of pages the site actually has. (**10 marks**)

Exercise 2 (20 marks)

A) Use the following corpus of D1 and D2 to manually calculate TF-IDF vectors for all the terms in Document 2 the same way scikit-learn does (by default, but assume no application of L2/Euclidean normalization) as discussed in class. (**10 marks**)

Manually

D1: "the cat leaps onto the fence, surprising both the dog and the nearby fox"

D2: "the dog barks loudly and the fox dashes away"

B) Read in the reviews of IMDB_Dataset.csv.

1. Convert the table of reviews into a bag-of-words (BOW) matrix using scikit-learn. (**2 marks**)

use skit learn

2. Do the same thing for each bigram of the text. (**2 marks**)
3. Calculate the TF-IDF of these documents using scikit-learn. (**2 marks**)
4. Describe 2 reasons why one hot encoding is rarely used for text representation anymore. Are there other vector space model representations that also share the issues you described? Name them. (**4 marks**)

Exercise 3 (20 marks)

A) Assume you have to make your own training set to pass as input to Word2Vec instead of just a list of lists of text. Write a windowing function that can take text as input and make 1 training set in the CBOW format as seen in slide 5 of the Word2Vec slides. Make the window size a parameter of your function that can be changed. Test your code with file nlptext.txt. Note that you do not have to train a Word2Vec model. (**6 marks**)

B) Suppose you are training a Word2Vec model with the text in part A. (**10 marks**)

1. Given that you set your hyperparameters for Word2Vec as vector_size=50, window=3, and min_count=5, what will the dimensions of the embedding matrix be and why (i.e. what do the dimensions of the embedding matrix represent)? What will the dimensions be for the context matrix? Assume you do not do any preprocessing of the text other than simple word tokenization with NLTK.
2. How would the answer to these questions change if vector_size=50, window=5, and min_count=5?
3. How would the answer to these questions change if vector_size=100, window=3, and min_count=5?

C) What is the purpose of negative sampling when training Word2Vec? (**2 marks**)

D) Why does CBOW train faster than Skip-gram? Hint: Think about how each model is updated. (**2 marks**)

Exercise 4 (20 marks)

SpaCy uses a modified version of Word2Vec to get token representations called FastText. FastText is essentially the same as Word2Vec, except that instead of operating on tokens of entire words, it operates on sets of characters.

Ex in W2 Vec

A) Import fasttext from Gensim and get the representation of at least two out of vocab words when you train a model with corpus = [["horse", "pulled", "cart"], ["dog", "say", "woof"]] and the most similar words to it. Find a word that is not part of this corpus vocabulary that will work with this model, ie. will be represented. Will any word work with this fasttext model? Why or why not? (**8 marks**)

find 2 words -> find word in corpus similar to them

B) Use pretrained Word2Vec and FastText models word2vec-google-news-300 and fasttext-wiki-news-subwords-300, respectively (you may need to load them individually and remove after the analysis is done as they are quite large). Load these models and come up with at least four examples to compare syntactic (2 examples) and semantic (2 examples, hint: how can you make analogies with Gensim?) representations between the two models. Compare and contrast the results. (**8 marks**)

C) What do you think the potential benefits are of using FastText over Word2Vec and vice versa? (**4 marks**)