Assignment 8 Language Modeling with an RNN

By Mimi Trinh

Section 1: Summary and Problem Definition

In this project, management is thinking about using a language model to classify written customer reviews, calls, and complaint logs. If the most critical customer messages can be identified, then customer support personnel can be assigned to contact those customers. Specifically, this assignment uses pre-trained word vectors and sentences as sequences of words to train language models to predict movie review sentiment (negative vs. positive comments). Implemented in Python using TensorFlow, we use recurrent neural networks (RNN) to analyze the sequences, as needed for natural language processing (NLP). Specifically, we will build four models, recommend the best one as the most relevant to the customer service function, and make suggestions to automate the customer support system that is capable of identifying negative customer feelings.

Section 2: Research Design, Measurement, and Statistical Methods

Specialized RNN models have been developed to accommodate the needs of many NLP tasks. Larger relevant vocabularies are usually associated with more accurate models, but training with larger vocabularies requires more memory and longer processing times. Thus, we will speed it up by using pre-trained word vectors and subsets of pre-trained word vectors to train the models. This is called word embeddings with popular techniques such as word2vvec, GloVe (global vectors), and FastTest to provide ways of representing words as numeric vectors.

In this assignment, we use two pre-trained word embeddings (glove.6B.50d.txt and glove.6B.100d.txt) in the downloaded zip folder and two different vocabulary sizes (10,000 and 15,000) to develop four language models. Therefore, we generate four different language models though all four use the English language since that's the language written in the movie reviews.

Section 3: Programming Work

This project uses both TensorFlow and scikit-learn packages in Python to develop the four different language models. Specifically, the following steps are taken.

- Install the Python chakin package and obtain GloVe embeddings
- Using the starter code from Canvas and train-test split approach similar to previous assignments, develop the first language model using pre-trained word vector glove.6B.50d.txt and 10,000 words in vocabular size to analyze and predict the sentiment of movie review data
- Develop three additional language models using different pre-trained word vectors (glove.6B.50d.txt) and vocabulary sizes (15,000 words)
- Compute the classification accuracy of the test set

Section 4: Results and Recommendation

Using the completely-crossed 2x2 experimental design, we generate the following results.

	GloVe.6B.50d	GloVe.6B.100d
10,000 words	0.6258	0.6378
15,000 words	0.6307	0.6148

Using the results above, we advise senior management to use model #2 with GloVe.6B.100d and 10,000 words since it has the highest accuracy on the test set. Thus, this system is the most relevant to the customer services function. In other words, considering the results of this assignment in particular, 10,000 vocabulary size and pre-trained word vector of GloVe.6B.100d are needed to make an automated customer support system that is capable of identifying negative customer feelings. However, since the accuracy is still quite low, to make language models more useful in customer service function, the data science team should continue with the project to develop models with higher accuracy by adjusting different hyperparameter settings and/or using other technologies such as word2vec and FastText instead of GloVe.