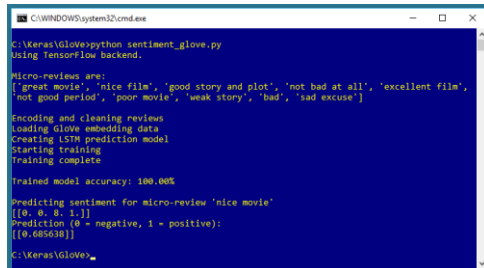


Predictive Analytics World / Deep Learning World

Exercises - GloVe Embeddings

1. Locate and download the GloVe embedding data, in particular, file glove.6B.100d.txt based on 2014 Wikipedia text. Write a Python program that sets up 10 hard-coded micro reviews (four words max length each, five positive and five negative) and creates a sentiment analysis model based on the GloVe embeddings. For example:



```
C:\WINDOWS\system32\cmd.exe
C:\Keras\GloVe>python sentiment_glove.py
Using TensorFlow backend.

Micro-reviews are:
['great movie', 'nice film', 'good story and plot', 'not bad at all', 'excellent film',
 'not good period', 'poor movie', 'weak story', 'bad', 'bad excuse']

Encoding and cleaning reviews
Loading glove embedding data
Creating LSTM prediction model
Starting training
Training complete

Trained model accuracy: 100.00%

Predicting sentiment for micro-review 'nice movie'
[[0. 0. 1.]]
Prediction (0 - negative, 1 - positive):
[[0.685638]]

C:\Keras\GloVe>
```

2. Modify your program for multi-class analysis where negative = (1,0,0), neutral = (0,1,0), positive = (0,0,1). Increase the max length of the micro reviews to five or six, and add five hard-coded neutral micro-reviews to the training data.

3. Which statement is most accurate?

- a.) Using global embeddings like GloVe is most appropriate when creating NLP prediction systems for domain-specific problems, such as legal text, or medical text.
- b.) Using global embeddings like GloVe is most appropriate when creating NLP prediction systems for generic problems, such as systems like Apple Siri or Amazon Alexa.
- c.) Using global embeddings like GloVe is most appropriate for initializing the weights of an embedding layer.

4. Which statement is most accurate?

- a.) The Keras Tokenizer fit() method encodes words according to order of first appearance.
- b.) The Keras Tokenizer fit() method encodes words according to their frequency.
- c.) The Keras Tokenizer fit() method encodes words randomly, controlled by NumPy random class.

5. Which statement is most accurate?

- a.) In general, most NLP neural systems, including Keras Tokenizer, use 0 for padding.
- b.) In general, most NLP neural systems, including Keras Tokenizer, use 0 for out-of-vocabulary.
- c.) In general, most NLP neural systems, including Keras Tokenizer, use 0 for stop-words.

6. Which statement is most accurate?

- a.) In general, most NLP neural systems by default place padding at the beginning of a sentence.
- b.) In general, most NLP neural systems by default place padding at the end of a sentence.
- c.) In general, most NLP neural systems place half of padding at beginning, half at end of a sentence.