

# CS410 Tech Review:

## Text classification with TensorFlow

Fall2020 | Xinqian Xiang

### *Overview*

My final group project is to build a system to help user to identify whether the link is useful to them based on user profile. I would like to investigate some tools or examples that can prepare me better for my project.

I searched online and found TensorFlow is an open-source machine learning platform that provides not only tools, libraries, and datasets but also tutorials to help users at different levels.

### *Text Classification with TensorFlow*

Through learning the Text Classification example with TensorFlow to help me get familiar with the process.

#### 1. Set Up and Loading Data

The example is for binary classification which can be widely applied to different kind of machine learning problem. It uses the IMDB dataset from the Internet Movie Database. It contains 50,000 movie review and split half into training and half testing.

Take closer look at the data. Each line is a sentence representing the movie review and a corresponding label. The label is an integer value of either 0 or 1, where 0 is a negative review and 1 is a positive review.

#### 2. Build the Model

The neural network is created by stacking layers and it built 3 layers classifier with below process:

- 1) The first layer is a TensorFlow Hub layer. It uses a pre-trained model to map a sentence into an embedded vector([google/nlm-en-dim50/2](https://www.tensorflow.org/hub/docs/nlm-en-dim50/2)). It splits the sentence into tokens, embeds each token and then combines the embedding.
- 2) This fixed-length output vector is piped through a fully-connected (Dense) layer with 16 hidden units.
- 3) The last layer is densely connected with a single output node. This outputs logits: the log-odds of the true class, according to the model.

### 3. Loss Function and Optimizer

Since this is a binary classification problem and the model outputs a probability (a single-unit layer with a sigmoid activation), the example uses the `binary_crossentropy` loss function.

### 4. Validate Accuracy of the Model

To develop and tune our model using only the training data, it set apart 10,000 examples from the original training data.

### 5. Train and Evaluate the Model

Train, monitor and record model's loss and accuracy on the 10,000 samples from the validation set. Then check the Loss and accuracy of the model to observe the performs.

### 6. Graph of Accuracy and Loss over Iterations

Plot the history recorded during the training and validation. It shows use these data to plot the training and validation loss for comparison, as well as the training and validation accuracy.

## *Taken-away Lesson Learn*

The above 6 steps well demonstrate the process of using TensorFlow for text classification. For my final project, there are some other things I can try.

- Try different number of layers for the neural network. In the example, it demonstrated 3 layers. Maybe less training cycles will be with more layers selected at the beginning.
- Build good amount of data for training and validation. This is the hardest part of my final project. I need to find a good data source or build a dataset with a enough size for model training.
- Keep record of training and validation data for later graphic analysis. So that I can find the optimization point to prevent overfitting.

## *Reference*

[https://www.tensorflow.org/hub/tutorials/tf2\\_text\\_classification](https://www.tensorflow.org/hub/tutorials/tf2_text_classification)

<https://analyticsindiamag.com/tensorflow-vs-keras-which-one-should-you-choose/>

[https://www.tensorflow.org/tutorials/keras/text\\_classification](https://www.tensorflow.org/tutorials/keras/text_classification)

<https://tfhub.dev/google/nnlm-en-dim50/2>