Jingru Ma & Moran Wang

Final Project

Capstone

Augest 16th 20018

**Detect a News is Fake or Not**

**Abstract**

With the development of social media, people can get and send more and more information from internet. People get news from internet for their daily life. However, many news in the internet including bias and other bad impact on people. When people read this bias news online, they don’t realize it and will spread this bias news to others. Our project may help people to detect the reality of the news. Detection fake news is a hard topic. We plant to use LSTM and tensor flow to build the model to do detection.

**Literature reviews**

In article “I trained fake news detection Al with > 95% accuracy, and almost went crazy.”, the author talks about what is achievement for others to detect the news. In that blog, the author get the high accuracy and he indicates that to improve the accuracy of the prediction, we may get more real news data because fake news has many different characteristics. It is hard to detect. But the characteristic of real news is more clearly than fake news.

The blog ‘Keras’ indicates that how to use Keras package in the python to implement Tensorflow and provides us many examples about how to use it.

The blog of “SMOTE.” and “SMOTE with Imbalanced Data.” talk about how to use SMOTE package as a oversampling method in python. SMOTE is Synthetic Minority Oversampling Technique. This will generate new instance from the existing minority class. “The new instances are not just copies of existing minority cases; instead, the algorithm takes samples of the *feature space* for each target class and its nearest neighbors, and generates new examples that combine features of the target case with features of its neighbors. This approach increases the features available to each class and makes the samples more general.” (SMOTE)

The blog of ‘Keras LSTM tutorial’ talks about what is LSTM model and how to apply it in python. LSTM model is a sequential model that can be used in predict time and predict the content that have order.

In the website of “GloVe: Global Vectors for Word Representation.” we download the GloVe dictionary.

“keras-team/keras.” is the website how others to load GloVe dictionary to the python.

We don’t know how others solve this problem because we didn’t see any code to solving this problem online. We just saw one blog that indicates that he uses Natural language processing to do it. And then he finds that trying to detect the fake news is not as good as detecting the real news. So, he finally tries to gathering real news information and detect if a news a real. We don’t know the details about how he solves this problem. However, we also use Natural language processing approach this problem. We use LSTM algorithm to build the model because we think the contents in the news have order.

**Detail of your approach**

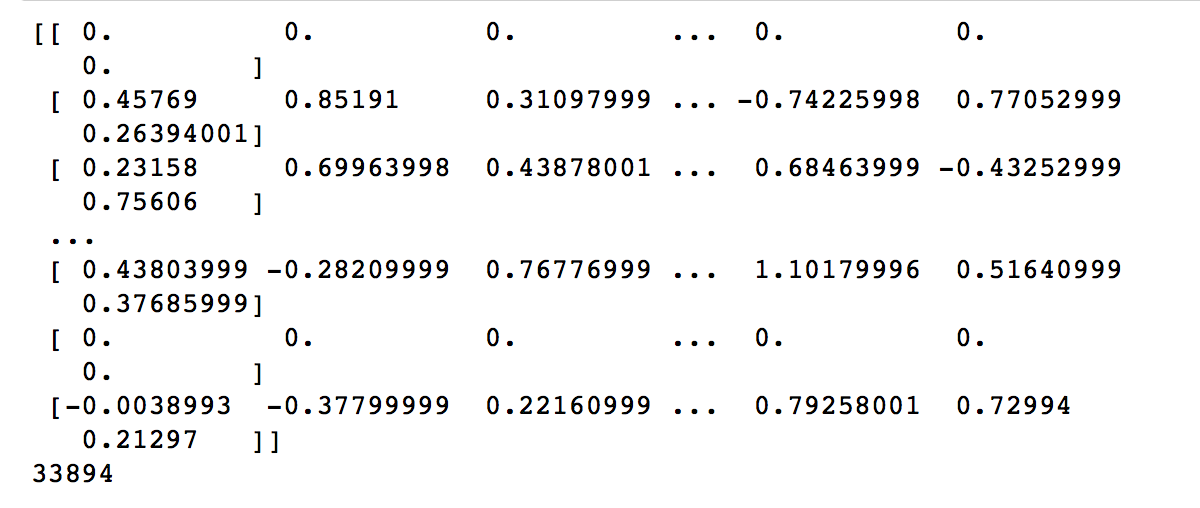
The basic idea is converting a news into vector and fitting this vector to TensorFlow model by adding LSTM layer. For the first part, we read data from json files and csv files and store them into panda DataFrame. Then padding these data into integer. After that, we load GloVe dictionary. We load GloVe as dictionary that includes the words as the key and vectors as the value. We then use GloVe to convert the padding integer to vector.

For the second part, we start to train the model. We use packages from keras to build model. We set the model is sequential. And then, we add embedding layer to the model and use LSTM into the model.

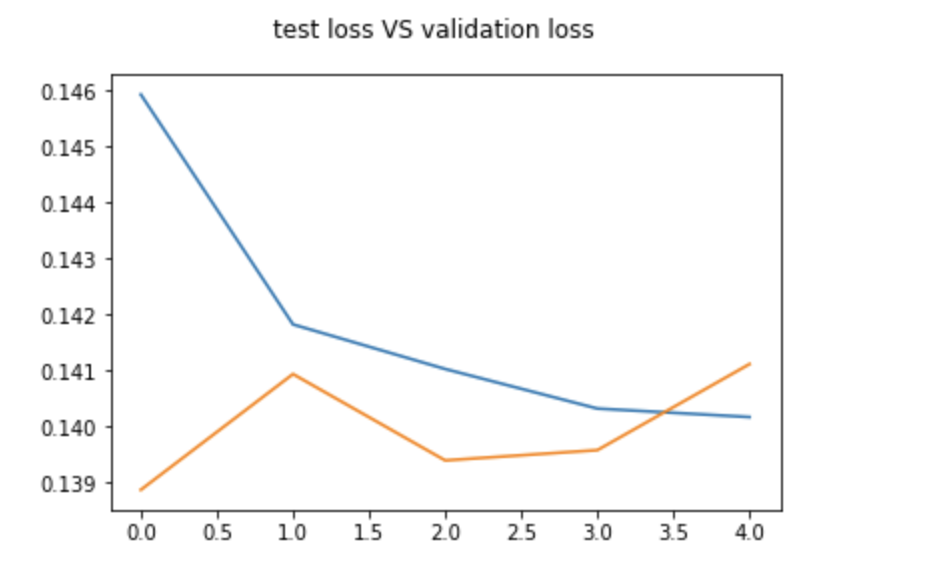
For the third part, we found the data is imbalance. The ratio of fake to real news is 88:1. In order to solve this problem, we use weighted cross entropy to calculate the loss. We increase the weight of falsely predicting the real news and decrease the weight of falsely predicting the false news. Therefore, in order to minimum the loss, the machine will try to avoid to falsely predict the real news.

In the last part, we extract a new news content from BBC website and padding it. We predict this news. The prediction result is close to 1.

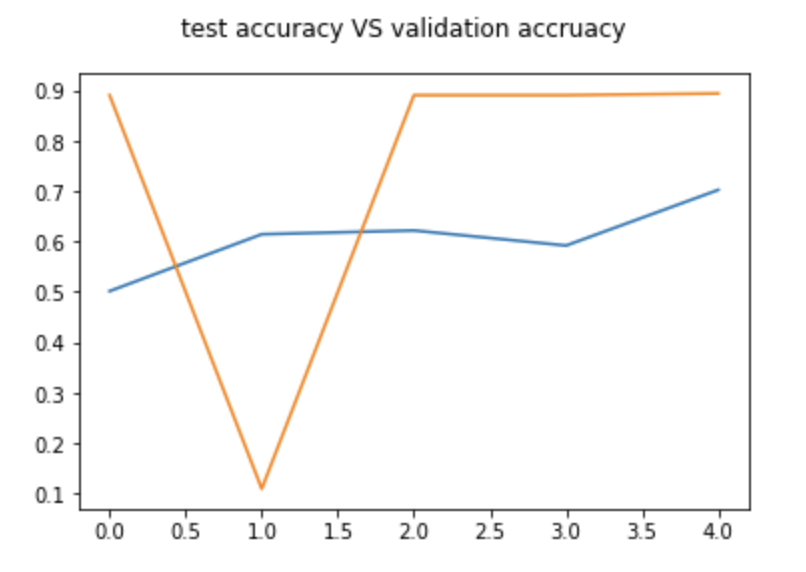
**Experiment detail**

****

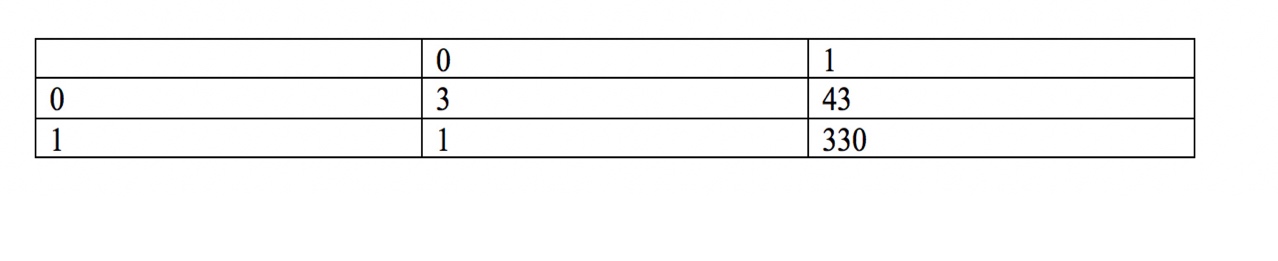
Since the data we got are all characters, to train them, we need to convert them into vectors. The figure above shows the matrix we have converted.



This graph shows the test loss and validation loss, the blue line is test loss and the orange line is validation loss. As we can see, the test loss keep decreasing from 0.146 to 0.140. The loss from validations set goes up at beginning and then goes down and then goes up when training and finally close to 00.141.



The blue line is test accuracy and the orange line is validation accuracy. This graph shows the accuracy of the model from test set is increasing from 0.5 to 0.7. The accuracy of validation set firstly decrase to 0.1 and then goes back t0 0.9.



The table we got above is confusion matrix of test data and prediction data. The accuracy we got on test data is 0.88 which is the same as the accuracy we got from model. evaluate.

**Error analysis**

Due to the problem of computer, we were not able to train much more enough data now. Therefore, the accuracy we got is not high. For next time, we will use the computer with nice GPU. In addition, the number of real and fake news we have are not balanced, we need to add more real news next time. Also, we will add more batch size, eposide and layer in the future.

**Conclusion**

We extract a new news from BBC website and padding it. We found the prediction result is close to 1. The prediction result is fake. We are not solving this problem very well. We are not realized there are too many types of fake news and it is hard to find all the characteristic of these fake news. There is two ways to improve our project. The first way we think is gathering more real news data. The second way is changing the way we deal with class imbalance like using weighted cross entropy.

“I trained fake news detection Al with > 95% accuracy, and almost went crazy.” *Toward Data Science,* 11/01 <https://towardsdatascience.com/i-trained-fake-news-detection-ai-with-95-accuracy-and-almost-went-crazy-d10589aa57c>

“Keras.” *TensorFlow,* 8/08/2018 <https://www.tensorflow.org/guide/keras>

“SMOTE with Imbalanced Data.” *Kaggle* <https://www.kaggle.com/qianchao/smote-with-imbalance-data>

“SMOTE.” *Microsoft Azure.* 01/09/2018.

https://docs.microsoft.com/en-us/azure/machine-learning/studio-module-reference/smote

“GloVe: Global Vectors for Word Representation.” 08/2014. <https://nlp.stanford.edu/projects/glove/>

“Keras LSTM tutorial – How to easily build a powerful deep learning language mode.” Adventures in Machine Learning, 02/03/2018 <http://adventuresinmachinelearning.com/keras-lstm-tutorial/>

“keras-team/keras.” *GitHub*. 09/07. <https://github.com/keras-team/keras/blob/master/examples/pretrained_word_embeddings.py>

“Keras Weighted categorical\_crossentropy.” Gist, [gist.github.com/wassname/ce364fddfc8a025bfab4348cf5de852d](http://gist.github.com/wassname/ce364fddfc8a025bfab4348cf5de852d).