COURSEWORK 2

Data Wrangling Report

ABSTRACT

This report discusses about how word embedding can be done using GloVE model.

Katikireddy Pramod(40451384)

40451384@live.napier.ac.uk

Contents

1. Data Preprocessing Used:

Before making the model, we have to improve the information by handling the sentences. We utilize Natural Language Tool Kit module for performing few stages, Firstly for each sentence we Tokenize the sentence into set of words, After Tokenizing we evacuate the stop words, as stop words are action words, article and so forth which are very little accommodating in finding the context of sentence. The last step is lemmatization and stemming, lemmatization is the way toward changing any word to its root word structure. Stemming is the way toward Lessing the articulation in words to their root words regardless of whether the words is legitimate or not. This processing is essential as it removes unwanted data from the dataset as there will be no useless data the neural network will be more accurate.

2. Model Preparation:

In our model first step is embedding the text data into vectors for processing of the data. We have chosen GloVE embedding here instead of word2vec and Elmo because glove embedding is easy to use and implement and perform better than word2vec. We obtain the embedding matrix of unique words from our dataset. We are using pre trained model i.e glove.6B.100d.txt file which contains a dictionary for every words as key and vectors as value. We generate embedding matrix by using this dictionary. We add this matrix into the first embedding layer of our neural network which uses it as the weight.

Model summary :

Used a flatten layer to flatten the output of embedding layer i.e to make the input to the required shape then we added a dense layer with shape as 3 as we are having 3 labels category.

```
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
embedding_8 (Embedding)      (None, 300, 300)          3652800
_________________________________________________________________
flatten_3 (Flatten)          (None, 90000)             0
_________________________________________________________________
dense_5 (Dense)              (None, 3)                 270003
=================================================================
Total params: 3,922,803
Trainable params: 270,003
Non-trainable params: 3,652,800
_________________________________________________________________
```

Model Evaluation:

We have considered metrics such as accuracy and confusion matrix for our model. Accuracy defines the how much percentage of predictions are correctly done by the model. Confusion matrix gives the summary of predictions done by the model.

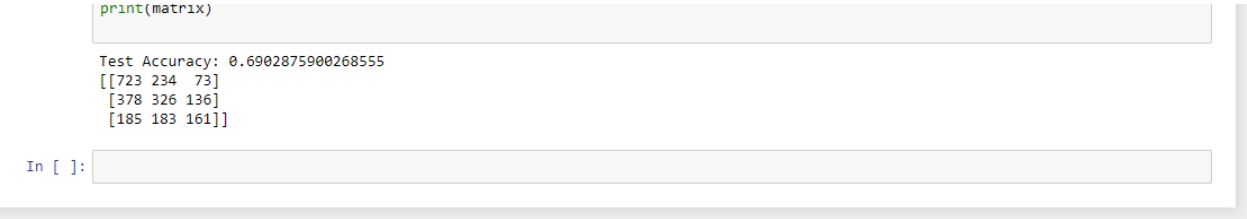
We can improve the model by removing the symbols and numbers in preprocessing of the data. We can also train the model on more data to improve the accuracy.

3. Critical Evaluation:

Training accuracy of model was 81.8 %, performance can be improvised by training the model on large dataset. We have used train test split method for validation of the model. Adding of hidden layers in our network may increase the accuracy of the model.

4. Conclusion:

Training accuracy: 81.8 %. Testing accuracy 69.02%.



Confusion matrix:

```
[[723 234  73]
 [378 326 136]
 [185 183 161]]
```

5. Appendix:

List of modules to be pre-installed:

- tqdm, re, nltk, copy, ZipFile, wget,Scikit learn, keras, genism, os, Tensorflow, Numpy, Pandas.