**Write up – On explaining the sourced code**

**Title: Context classification/ Multi-class text classification**

**Author: Jayathilaga Ramajayam**

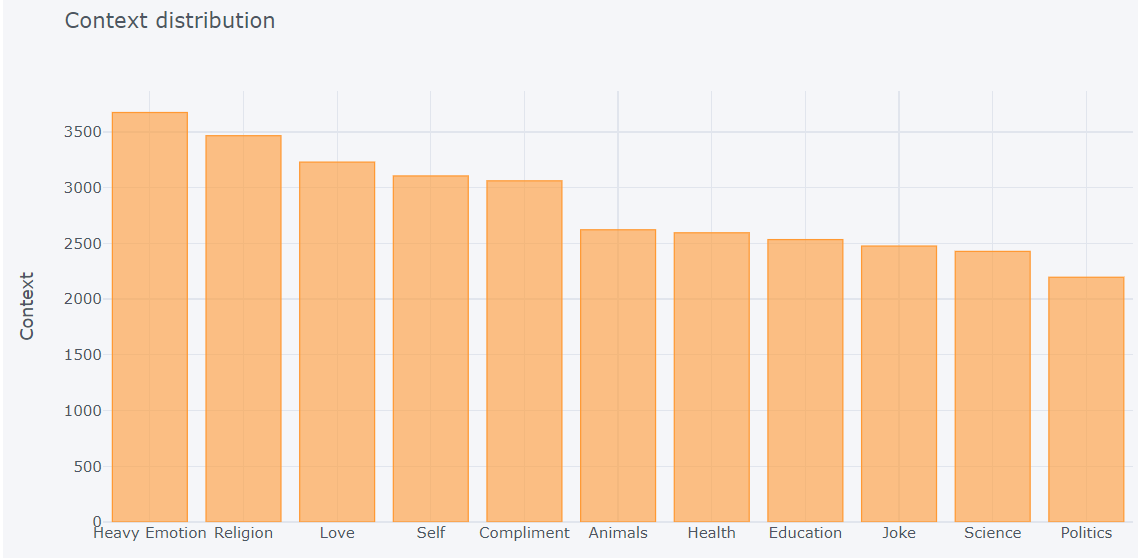
**Data:**

The dataset is related to different semantics or subjects. The classification goal is to predict whether the sentence belongs to particular topic. It includes **31386 records and 2 fields**

**Input variables:** Text

**Predict variable (desired target):** Context/Topic

**Data exploration:**



**Create dummy variables:**

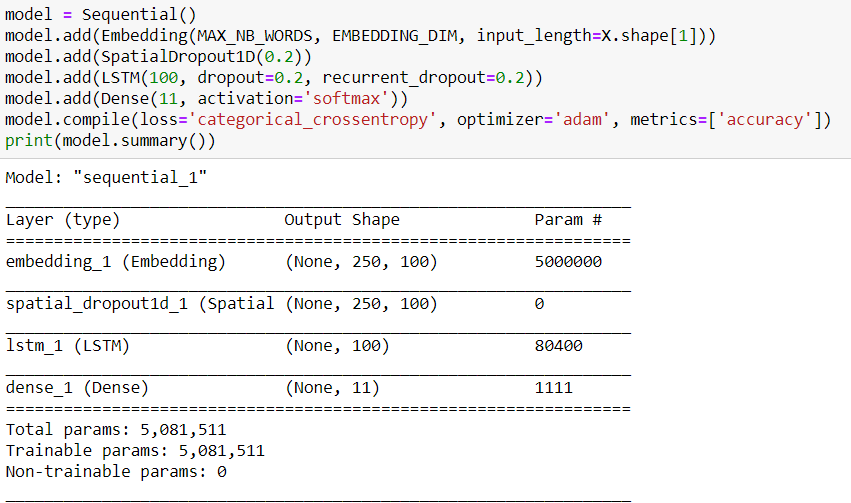
That is variables with only two values, zero and one.

**Pre-Processing Text data:**

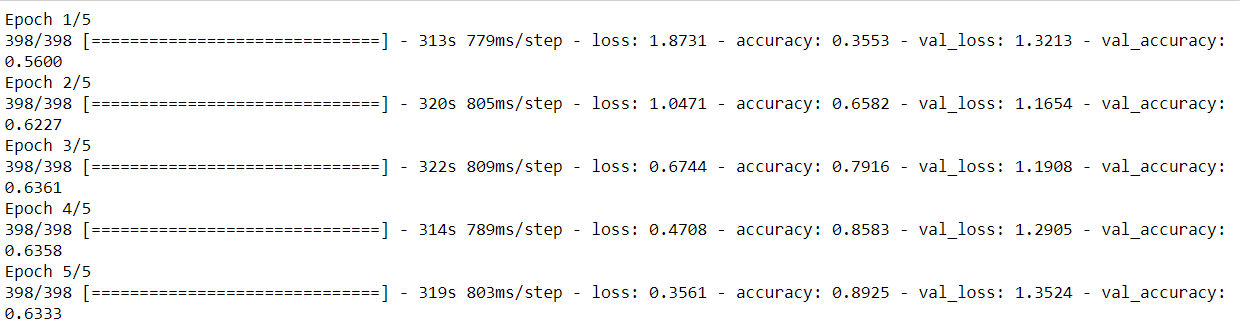
1. Convert to lowercase.
2. Remove punctuations, number
3. Removed stop words
4. Identify root word using Lemmatization
5. Generated tokens using keras Tokenizer
6. Filtered by pad\_sequence using keras preprocessing

**LSTM:**

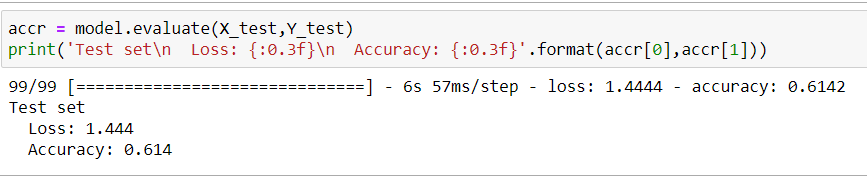
1. Vectorizer consumes text, by turning each text into either a sequence of integers or into a vector.
2. Limit the data set to the top 5,0000 words.
3. Set the max number of words in each text at 250.
4. Truncate and pad the input sequences so that they are all in the same length for modeling.
5. The first layer is the embedded layer that uses 100 length vectors to represent each word.
6. SpatialDropout1D performs variational dropout in NLP models.
7. The next layer is the LSTM layer with 100 memory units.
8. The output layer must create 11 output values, one for each class.
9. Activation function is softmax for multi-class classification.
10. Because it is a multi-class classification problem, categorical\_crossentropy is used as the loss function.



**Model Training:**

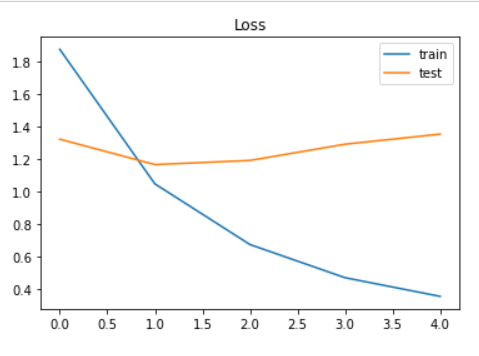


**Test Accuracy:**

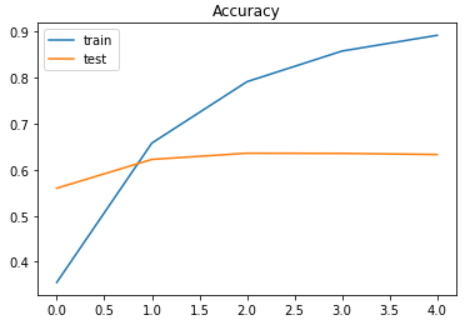


The plots suggest that the model has a little over fitting problem, more data may help, but more epochs will not help using the current data.

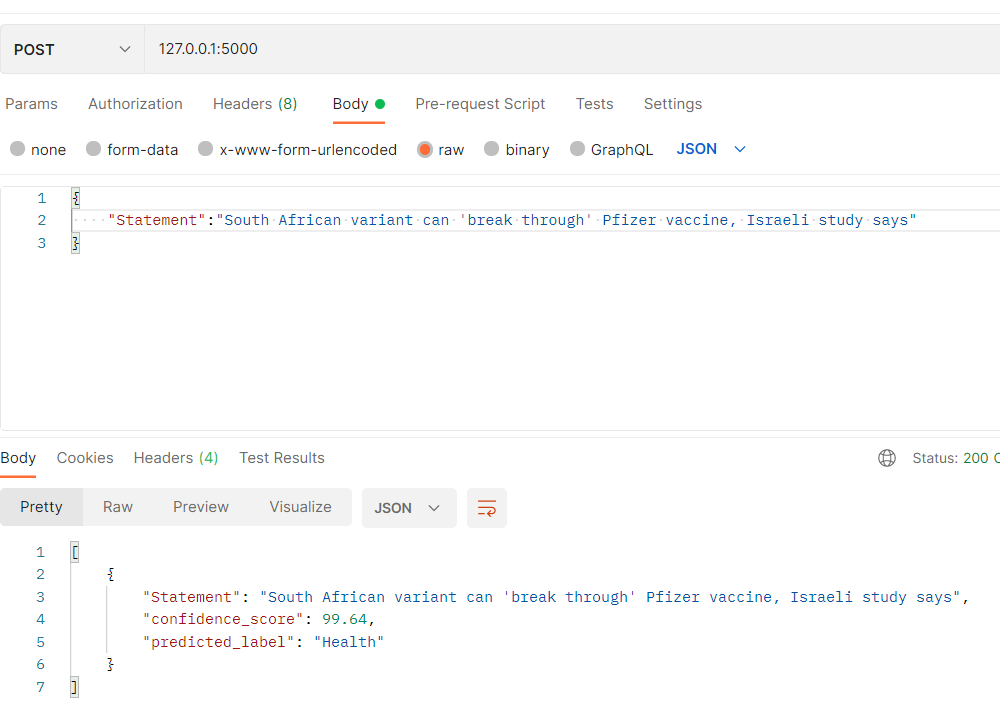
**Loss Curve:**



**Accuracy Curve:**



**Flask API - Postman**



**How to Run:**

1. Model is trained in Multi-Class Text Classification LSTM.ipynb
2. LSTM is saved as model
3. Saved tokenizer as tokenizer.pickle
4. To Run the Flask Service code
   1. Python final.py
   2. To visualize use Postman as shown above