**ICP3**

**BIG DATA ANALYTICS AND APPLICATIONS(CS5542)**

*Sravani Niharika Garapati (16328843)*

*Nikhila Chirumamilla (16323180)*

**QUESTION:**

Use the Twitter Data and perform sentiment Analysis on that data using one of the Deep Learning classifiers (Keras Model) for text.

**What I have learned from this ICP:**

In this ICP, professor asked us to perform sentimental analysis on the twitter data by cleaning, preprocessing and building model and executing the model using one of the Deep learning models i.e., Keras Model. The functions which I learned while doing cleaning and pre processing are listed below

**Sklearn:** It is the most helpful library in python for machine learning, providing fast tools for machine learning and statistical modeling, including regression, classification, and clustering, through a python interface.

**Nltk:** It is a library that features which support statistical language processing.

**StopWords**: The words that appear frequently in texts in any language and contribute no meaning to the content.

**Stemming:** Reducing the word to their base word forms

**Lemmatization:** It's like stemming, but it's more powerful and uses rules to generate meaningful base words.

**Tfidf Vectorizer:** It learns the vocabulary, tokenizes the words, and emphasizes the most interesting terms, i.e., those with a high frequency.

**Pos\_tag ():** It's a feature that assigns words in a text to a specific part of speech (noun, verb, adjective, and so on) depending on their definition and context.

**Sequential ():** It is one of the easiest way to build model in keras which allows us to build model layer by layer.

**Epochs ():** It specifies number of passes of the entire training data set a particular algorithm completed.

We used layers such as Conv1d, GlobalMaxPooling1D, Dense which are suitable for the problem

**ICP Description about the Task:**

In this task, we need to perform sentiment analysis on the given twitter data using one of the Deep Learning classifier (Keras Model) for text. This is done in order of below mentioned steps:

* As a first step, we imported Pandas and used pandas to read the data.
* We cleaned the data in a variety of ways, including deleting twitter handles (@user), hashes (#), and numericals, as well as stop words.
* We performed Stemming and lemmatization which were used to return all of the words to their original form.
* To identify the parts of speech, I used Pos tagging on all the words (noun, verb, adjective, etc.).
* Then, using Tfidf vectorizer, we converted all of the data into vectors to determine how much data was there and could be used for analysis in the following steps.
* Split the filtered data into training and test sets to fit the model.
* Used Tokenizer and tokenized words using texts\_to\_sequences () function
* In this problem, I used Sequential model which is easiest, and which allows to build model layer by layer
* Added layers such as Conv1d, GlobalMaxPooling1D, Dense which I felt would be best suit this problem
* Then, we compiled the model and fit the model to data set with epochs 5
* And, at last measured the accuracy of the model.

**Challenges faced in this ICP:**

The challenges which we faced while doing this ICP are listed below:

* It became quite difficult for us to clean the data by deleting all punctuation, numericals, twitter handles, and any other symbols that were included.
* It took lot of time to decide and add layers for the model.

**Screenshots of Execution of Code:**

Graphical user interface, text, application

Description automatically generated

*Displaying the Loaded Data*

*Table

Description automatically generated with medium confidence*

*Pre-Processing the Data*

*Table

Description automatically generated with low confidence*

*A picture containing text

Description automatically generated*

*A picture containing text

Description automatically generated*

*Table

Description automatically generated*

*Removing Stop Words*

*Text

Description automatically generated*

*Performing Stemming and Lemmatization*

*Graphical user interface, text, application

Description automatically generated*

*Function to remove stop words and perform stemming & Lemmatization*

*Graphical user interface, text, application

Description automatically generated*

*Calling the function to apply functionalities*

*Graphical user interface, text

Description automatically generated with medium confidence*

*POS tagging*

*Graphical user interface, text, application, email

Description automatically generated*

*Counting Frequencies*

*Chart, line chart

Description automatically generated*

*TFIDF Vectorization*

*Graphical user interface, text, application, email

Description automatically generated*

*Splitting the data and performing the tokenization*

*Graphical user interface, text, application, email

Description automatically generated*

*Building the model and adding layers*

*Text

Description automatically generated*

*Compiling the Model*

*Table

Description automatically generated*

*Fitting the model to the data*

*Table

Description automatically generated with medium confidence*

*Specifying accuracy of the model*

*Graphical user interface, text, application, chat or text message

Description automatically generated*

**Video Link**

https://youtu.be/AiAwHaOFtdM