1. What you learned in the ICP:

In this ICP I learned how to analyzing twitter dataset and cleaning it from all unnecessary words/columns. Also, to build and evaluate a model using classification report.

1. ICP description what was the task you were performing:

I performed the sentiment analysis task by cleaning and preprocessing the dataset as the following: removing punctuations and stop words, Tokenization, TFIDF vectors..etc. Also, I built a model that can predict if the tweet is positive or negative as well as used the classification report to measure the quality of predictions.

1. Challenges that you faced:

There was no big challenge in this task. However, I spent most of my time to clean and organize the data.

1. Screen shots that shows the successful execution of each required step of your code:

Start reading the file using the link that has been provided by the instructor.

Graphical user interface, text, application, email

Description automatically generated

Here: I read only the tweet column, and print the length of the column. Then, I removed all punctuations, numbers and @user and save it in a new column named as “clean\_data”.

Graphical user interface, text, application, email

Description automatically generated

I’ve split the data suing .split method and saved that on the frequency distribution class in order to print the most common words. Then, I’ve converted the “clean\_data” column to string to complete performing the preprocessing process.

Graphical user interface, text, application

Description automatically generated

I tokenized the dataset by sentences and words and print the most common words.

Graphical user interface, text, application, email

Description automatically generated

Plot the above words using matplotlib library. However, since I got […] that mean still the first cleaning process did not clean everything. Therefore, I’ve removed all punctuations marks again in order to remove all unnecessary words.

Chart

Description automatically generated with low confidence

Print the fdist to ensure that all the punctuations marks have removed and visualize it using plot.

Chart

Description automatically generated with medium confidence

In this step I should remove the stop words since the list above is meaningless of the most common words. The words length was decreased from **242462** to **153016** words.

Graphical user interface, text, application, Teams

Description automatically generated

Plot the first 20 common words. The word “Day” has mentioned more than 1600 time.

Chart

Description automatically generated with low confidence

I visualized the data after cleaned using the cloud word.

Graphical user interface, text

Description automatically generated

Data is cleaned and ready to build the model. I used the scikit-learn TFIDF to convert all the clean tweet to numerical in order to be readable by the ML algorithm. Also, I’ve split print the report classification to measure our model. The accuracy of the model is 97%.

Application, table

Description automatically generated with medium confidence

In the last step, I gave the model a random text to predict if that text is positive or negative.

Graphical user interface, text, application, email

Description automatically generated

1. Video link:

<https://drive.google.com/file/d/1dxJPS3xaaKymsGAeaS1BfaG4P0vvRPzJ/view?usp=sharing>