**KODERS INTERNSHIP**

**Problem Statement 1:-** Research on how to access web-view(which can’t be done using UI Automator) in Appium when we have released version of the APK and debug mode isn’t enabled so even the Chrome Dev Tools aren’t working.

**Solution:-** One major principle followed by **Appium** which provides appium an advantage over ***Calabash automation tool*** is that, in Appium it is not required to change the app under test. We automate same app which goes to production. Appium allows us to test Web-Apps and Hybrid Apps using it’s built-in support via **Chromedriver**. We can use selenium webdriver existing methods to automate webview’s inside hybrid app.

The sad part is in inspection of hybrid app the ***UIAutomatorViewer*** does not help us in inspecting elements which are under WebView.

We can inspect it in 2 ways:-

### Using Selendroid Inspector:-Using Appium tool by running appium with desired capability ***automationName*** as “***Selendroid***” only for apps which are built for Android version < 4.4.

### Steps:-

### Browse apk to launch from local folder.

### Select desired capability ***automationName*** as “***Selendroid***” and other mandatory parameters like platformname, platformversion, launch activity.

### Click play button to start server.

### App gets launched. If we closely see Appium server running logs then we can see it starts selendroid server on local host at port 8080.

* If selendroid server started properly then on hitting below URL it will return below json string

***URL :***[***http://localhost:8080/wd/hub/status***](https://localhost:8080/wd/hub/status)

jSON String : {“value”:{“supportedApps”:[],”build”:{“browserName”:”selendroid”,”version”:”0.12.0″},”os”:{“arch”:”armeabi-v7a”,”version”:”19″,”name”:”Android”},”supportedDevices”:[]},”status”:0}

### To start inspecting app we hit this url [*http://localhost:8080/inspector*](https://localhost:8080/inspector) it will automatically redirect to [*http://localhost:8080/inspector/session/88ce8684-d785-4a24-9157-80967becc60d/*](https://localhost:8080/inspector/session/88ce8684-d785-4a24-9157-80967becc60d/) by taking session id if selendroid server is running and now we get option to inspect app elements

### 2) Using Chrome Remote debugger:- Only for apps which supports Android version 4.4+ or API level >18. Here important point to note is appium works for API level >17 but chrome remote debugger works for API level >18 i.e we can use this chrome feature for API level>18.

**Steps:-**

* Type “***chrome://inspect/#devices***” in chrome browser.
* Open app with enabled webview debugger in device listed with adb.
* Click inspect link and we can inspect elements under webview.

**Article Reference:-** <https://www.toolsqa.com/mobile-automation/appium/how-to-inspect-and-automate-webview-in-hybrid-app/>

**Problem Statement 2:-** Sentiment Analysis model on Twitter US Airways Dataset.

Sentiment analysis is the interpretation and classification of emotions (positive, negative and neutral) within text data using text analysis techniques. Sentiment analysis tools allow businesses to identify customer sentiment toward products, brands or services in online feedback.

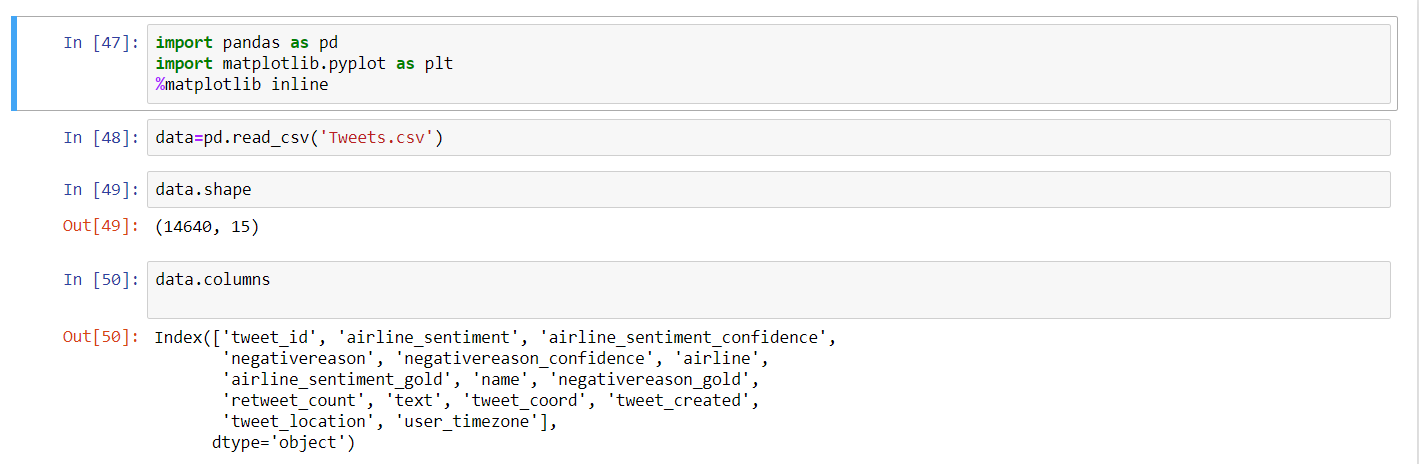
Here, I have performed sentiment analysis on Twitter US Airways Dataset

Dataset:-<https://www.kaggle.com/crowdflower/twitter-airline-sentiment/data>.

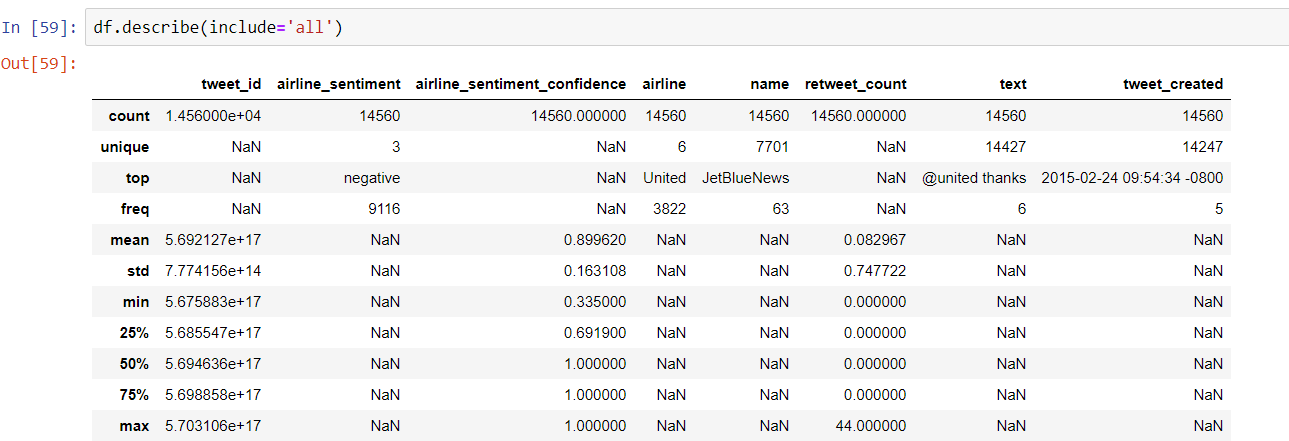
**Stage 1:-Preprocessing Stage**

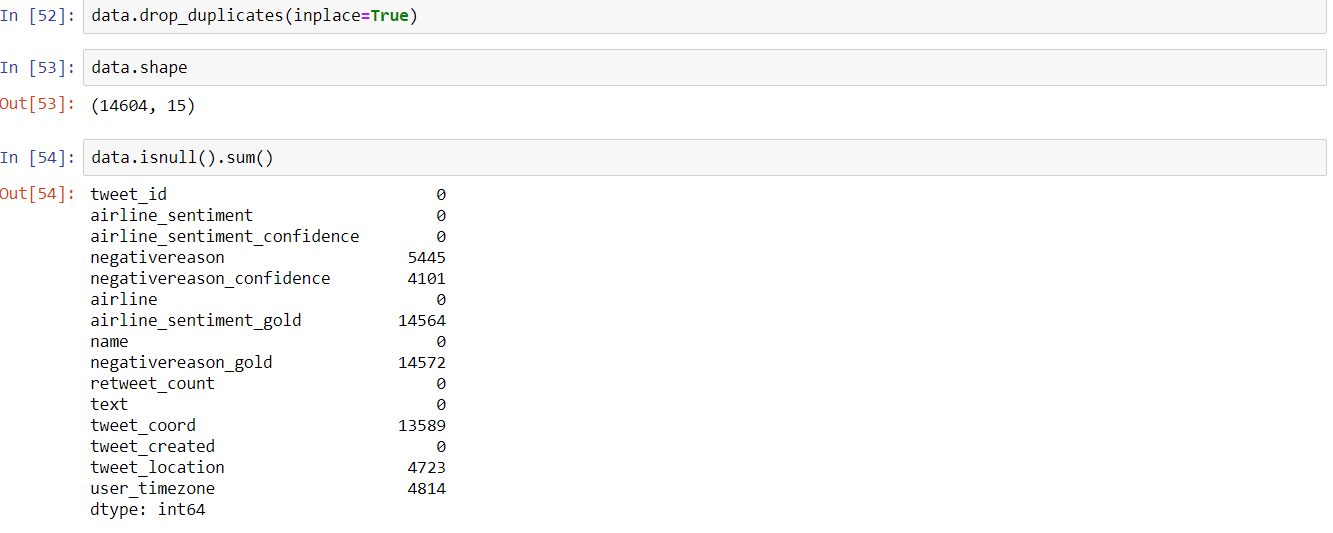
I have firstly performed all the necessary preprocessing operation on the data set to make it a pure sentiment analysis dataset which one would get in real time.

Step1:- Getting to know about the dataset by getting to know about its columns.

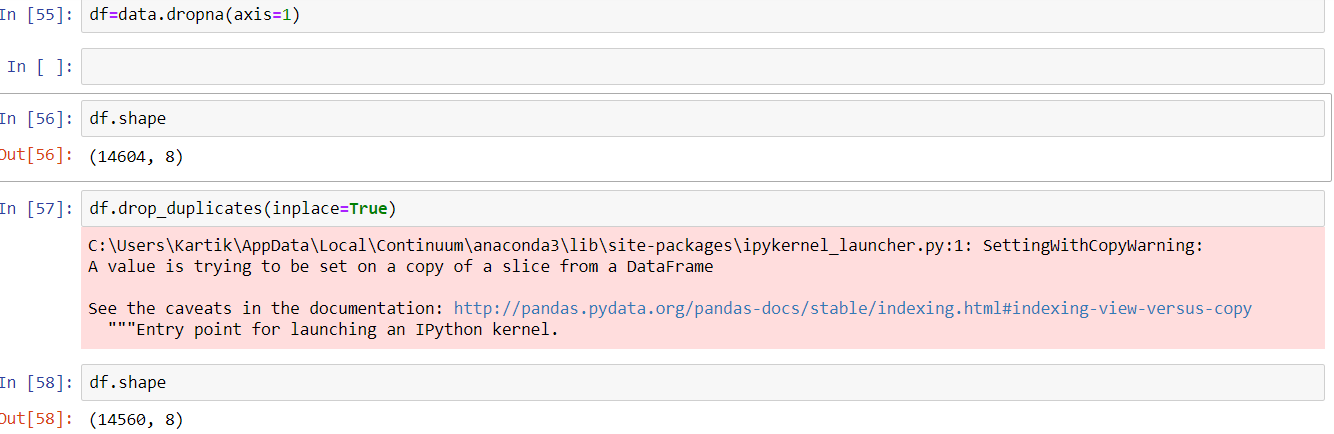
 Initially, the dataset has 14,640 rows and 15 columns.

Step2:- Driving insight about continuous and categorical variables

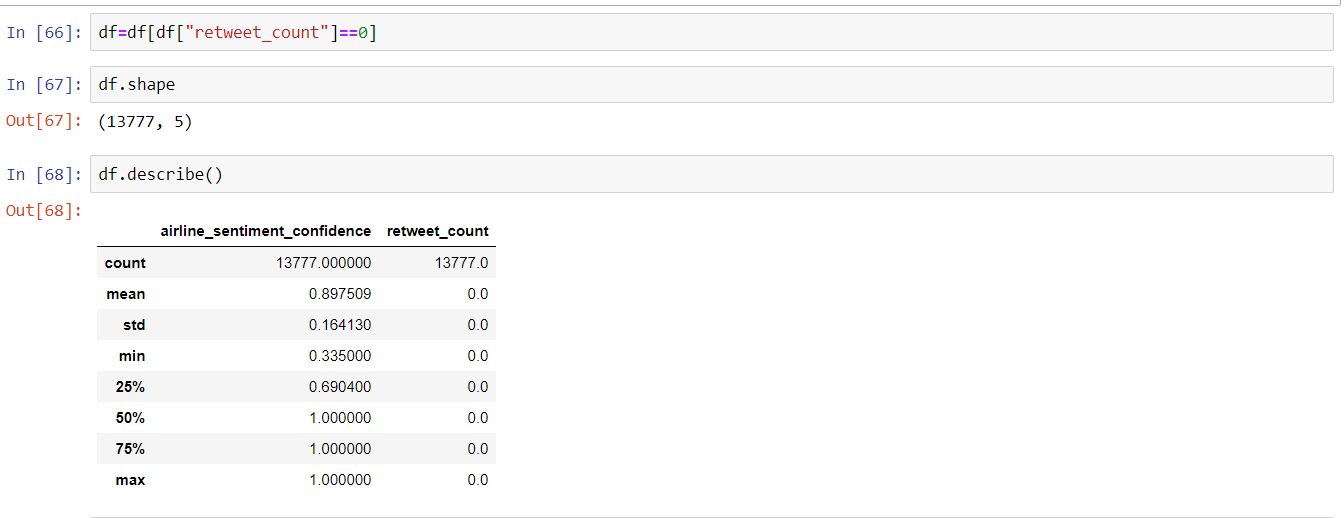
 Step 3:- Dropping duplicates and seeing for missing values in data if any.



Step 3:- Filtering data and Dropping missing values as few columns have a lot and it would prove to be unfruitful for us to substitute them using any method like mean or mode.

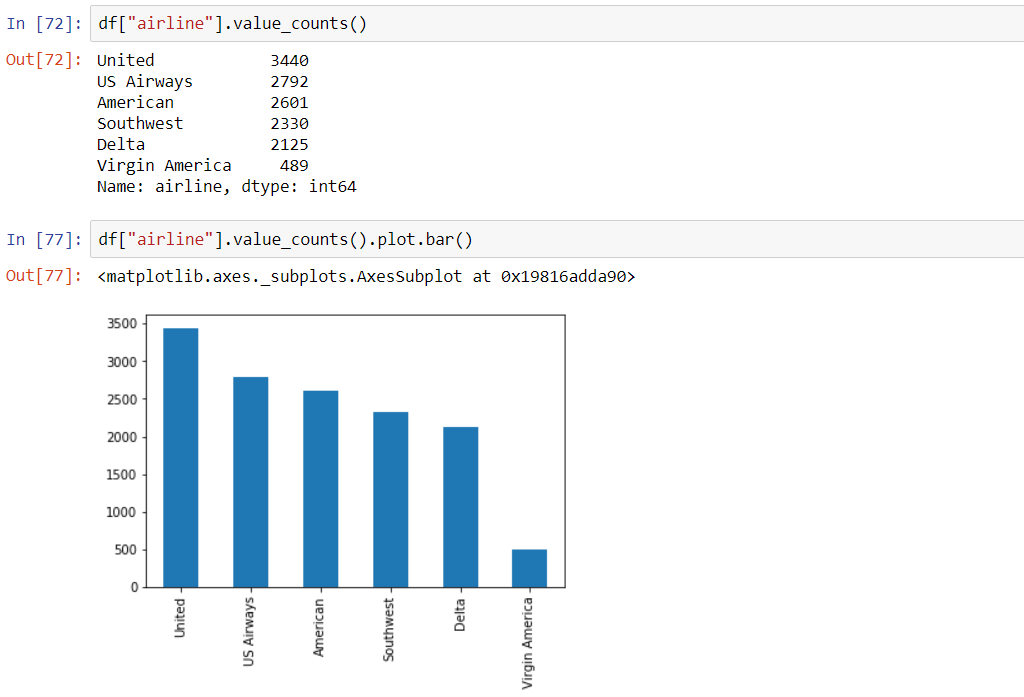


Step 4:-Filtering out data of re-tweet variables as it would just be off very less use in sentment analysis and would not be so much effective.

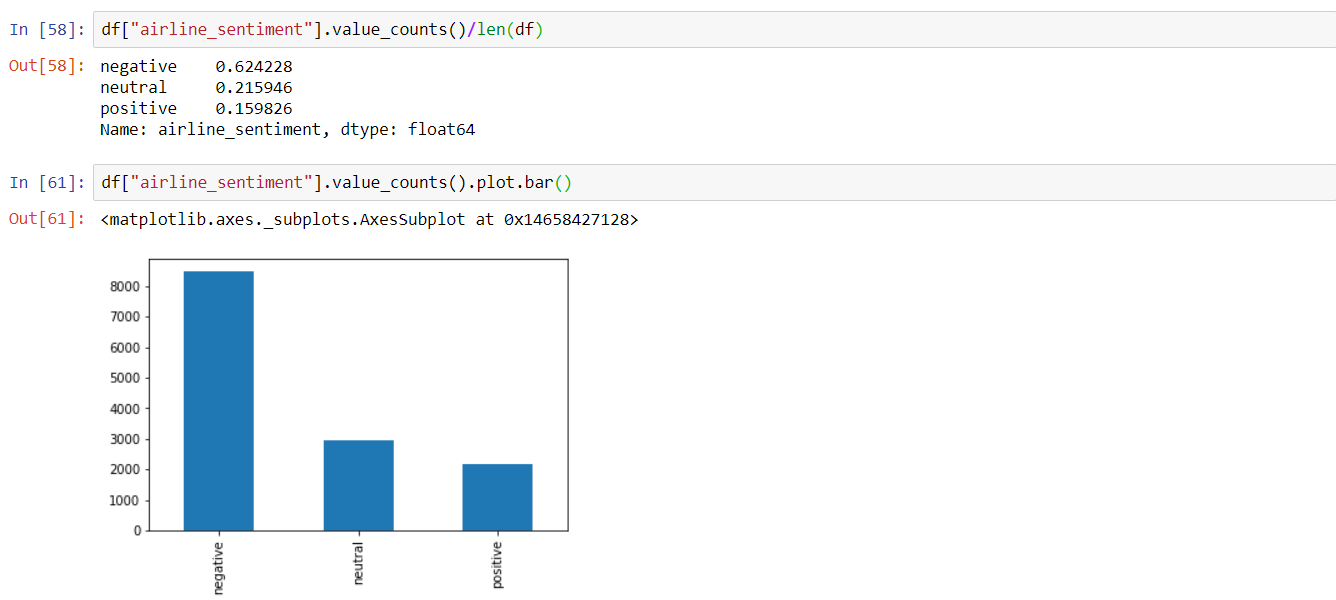


Step 5:- Determining the amount of Data of each category present .

Data insight about airlines:-



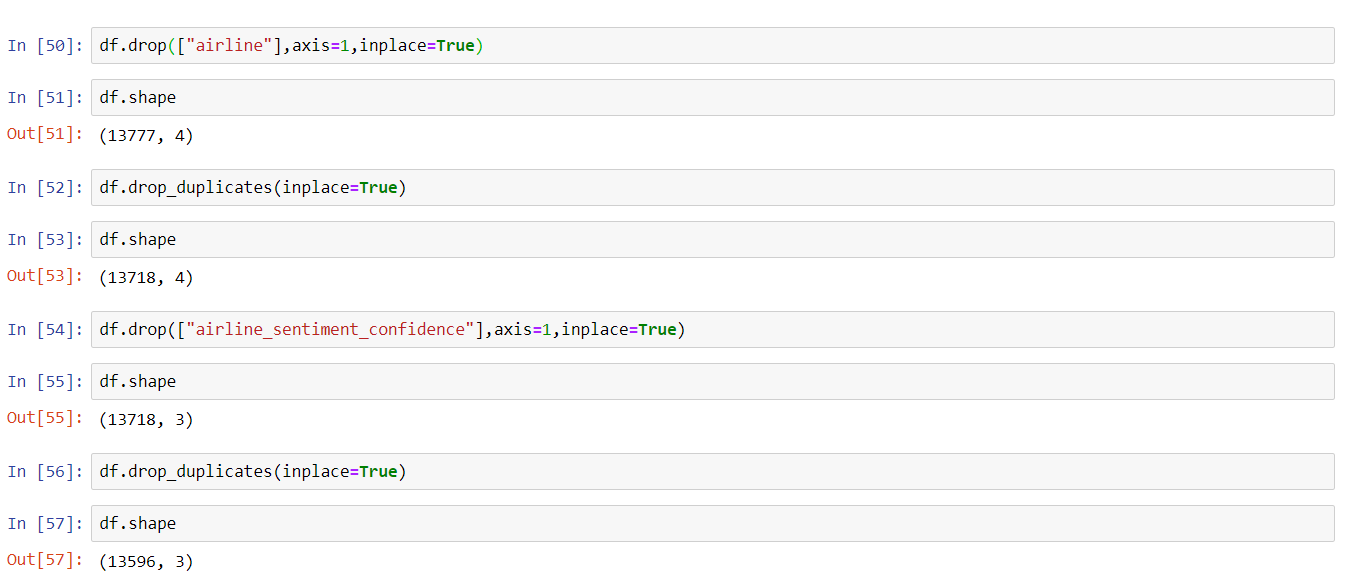
Data insight about Sentiments i.e amount of negative, positive and neutral reviews present:-



Step 6:- Now it’s time for processing of data and removal of all “@” remarks that exist in the text . First i have transformed all the text into lower case and then removed all the tweet-remark related with airline names.



Step 7:- Drop all other remaining columns as now the processing required from them is completed and we will save this in a new file of “FinalizedTweets.csv” that has 13596 rows.





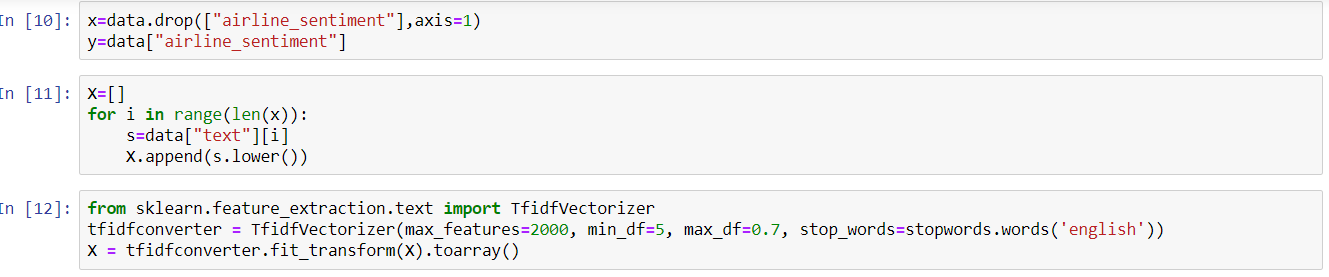
The new Dataset has 2 major columns:-

* The labelled data-“airline\_sentiment”
* The required sentiment reviews:- “text”

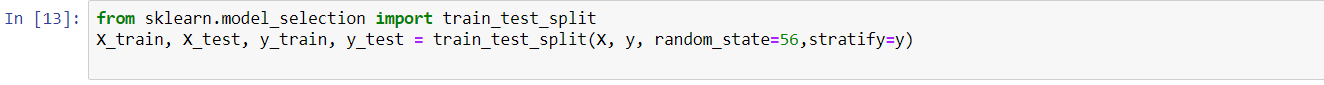
Stage 2:-**Model Development**

Here we will tools related to NLP like the NL tool kit in python and Vectorizer like TF-IDF or Count Vectorizer as the machine don’t understand text we need to provide the text to machine in the form of numbers.





Here, we split the model for the purpose of training and testing. I have used Stratified Distribution technique here so that my Labelled for every class get equally distribute in both the train and test dataset.

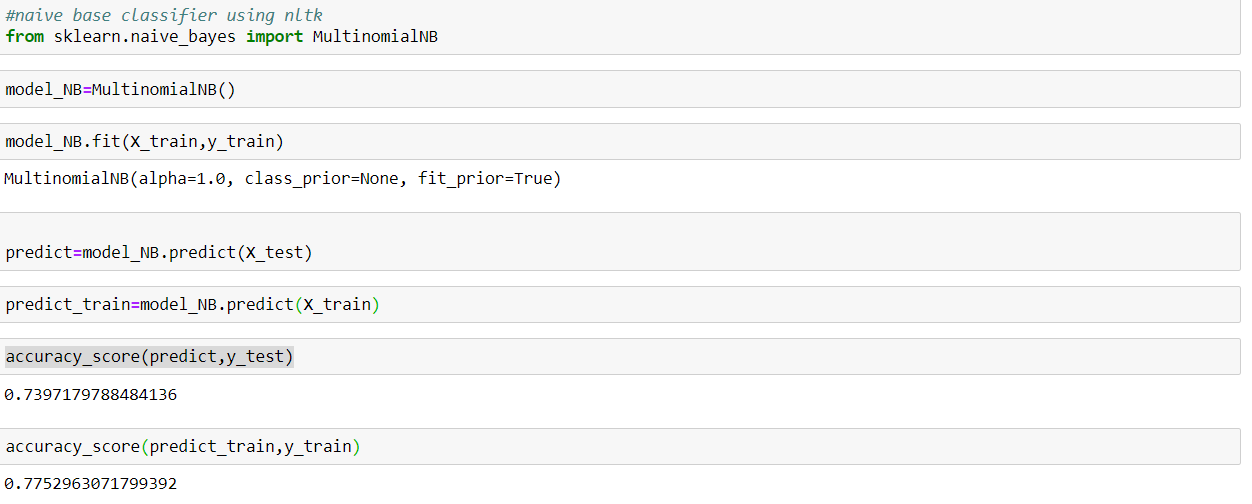
Using different Models and also cross checking for under fitting and over fitting in model:-

1. **Gaussian Naive-Bayes Classifier**:-First we prefer the simplest Naive-Bayes Classifier the Gaussian NB but as the model shows a very low accuracy of just **40.50%** so we are not using this model for evaluation and we set our approach towards other Naive-Bayes Classifier.



1. **Multinomial Naive Bayes:-**Here we use the Naive Bayes Classifier Algorithm as Multinomial NB that has been especially for multi labelled Classes as in this case.

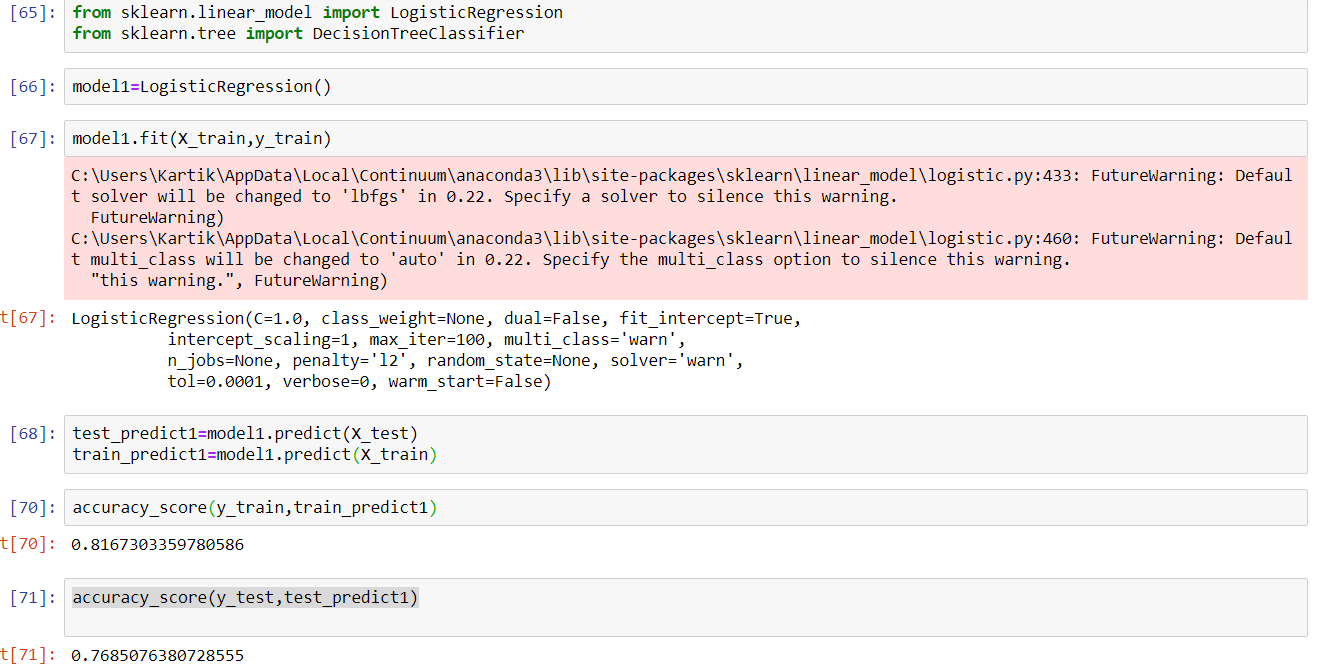
The model performs quite well with an accuracy of **73.9%** and shows no such under fitting and over fitting so we can surely say it is a good model.



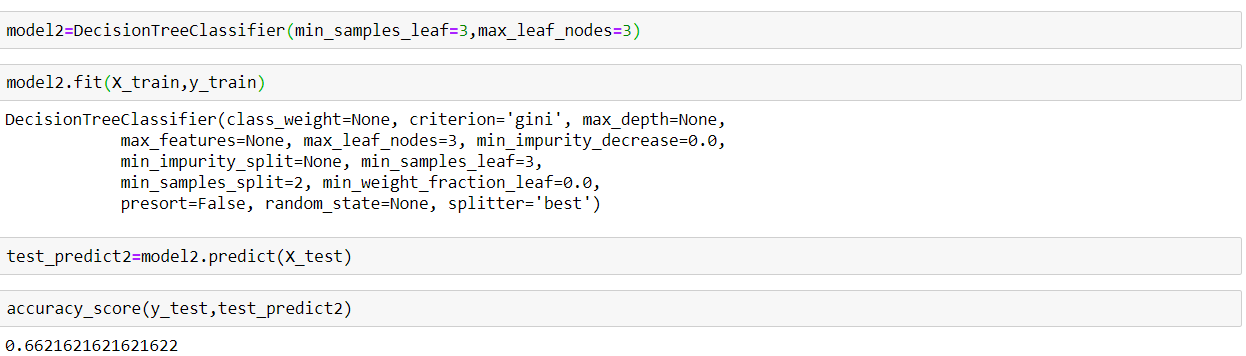
1. **KNN Classifier:-**  The nearest neighbour classifier doesn’t prove to be so helpful in this case as it shows a very low accuracy of just **41.10**%.So we don’t wish to continue with this model further.



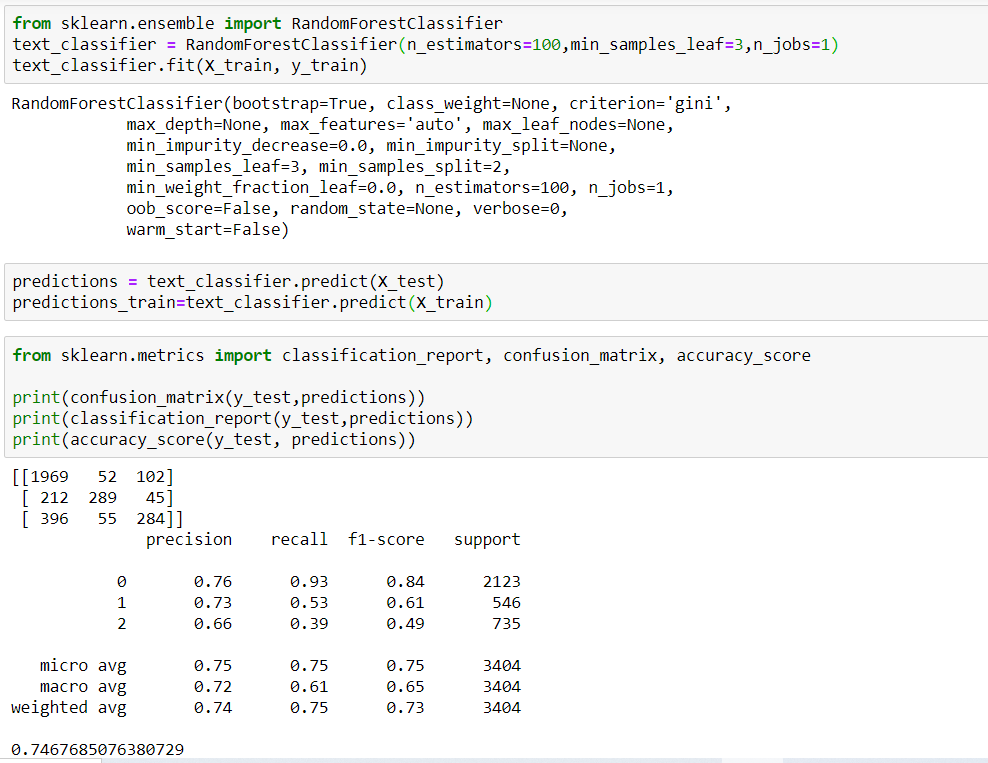
1. **Logistic Regression Classifier:-** Now we use the logistic regression classifier that has proven to be good classifier for sentiment analysis for multi-class labels. The Classifier shows an accuracy of **76.85%. Moreover,** Model fits and shows no over-fitting or under-fitting **.**



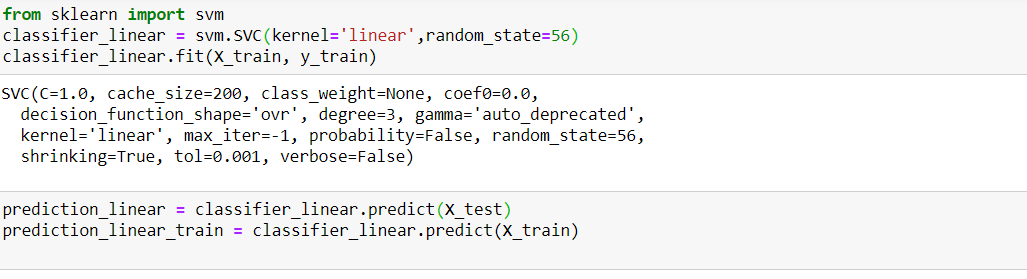
1. **Decision Tree Classifier**:- The Decision Tree Classifier that has proven to be moderate classifier for sentiment analysis for multi-class labels. The Classifier turn up with an accuracy of just **66.21%.Thus**, we reject this model as we already have model with higher accuracy**.**

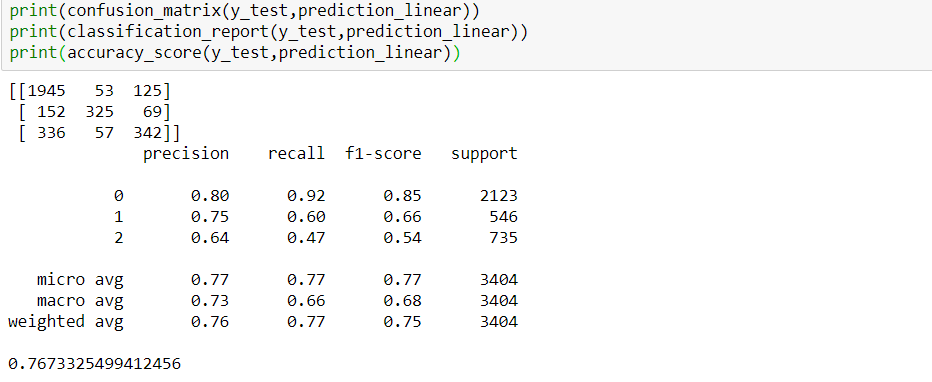


1. **Random Forest Classifier:-**The ensemble model of RandomForest has turned out to be quite good when restricted to 100 estimators and minimum leaf nodes as 3,the model shows an accuracy of **74.67%** .It even fits the Over-fitting and Under-fitting Parameters.



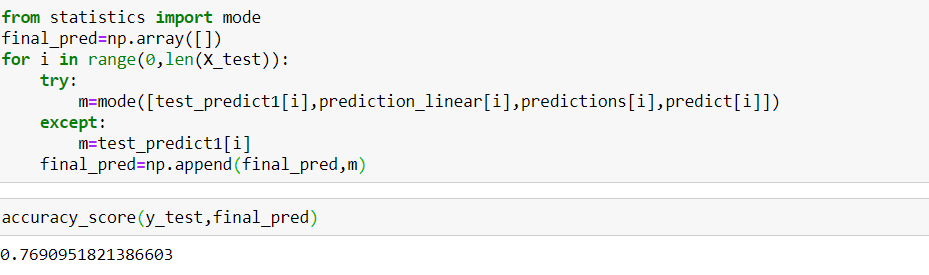
1. **Supper Vector Machines:-** The classifier has proven it self to be the best Classifier in terms of Sentiment Analysis and in our case to the model performs extremely well and gets an accuracy of **76.77%.** The model shows no over fitting and under fitting as such and thus proves to be a good model.

****

****

1. **Improving accuracy by basic ensemble technique:**

Improving accuracy by performing taking mode on your top 4 models i.e. Logistic Regression Classifier, Multinomial NB classifier, Random Forest Classifier and Supper Vector Classifier. We are thus able to achieve an accuracy of 76.909%.



**Conclusion**

The sentiment analysis model has been built with a final accuracy of 76.90% that has been achieved by implementing Ensemble technique of Mode of Random Forest Classifier, Supper Vector Machine Classifier, Multinomial Naive Bayes Classifier and Logistic Regression Classifier. The Model stands accurate and shows no over fitting and under fitting and thus proves to be the Right Model on Twitter US airlines dataset.

Link of Git-hub Repository:-

<https://github.com/Kartikarora01031999/Sentiment-Analysis-on-Twitter-US-airline-data>