MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE, GWALIOR

(A Govt. Aided UGC Autonomous & NAAC Accredited Institute Affiliated to RGPV, Bhopal)
NAAC Accredited with A++ Grade

Project Report
on
MINOR PROJECT - 1

(Sentiment Analysis for Tweets: A Random Forest Approach)

A project report submitted in partial fulfilment of the requirement for the degree of

BACHELOR OF TECHNOLOGY

In

ENGINEERING MATHEMATICS & COMPUTING

Submitted by:

Sr. No.	Name	Enrolment No
1	Anshika Singh	0901MC211012
2	Arin Tiwari	0901MC21015
3	Ayush Sharma	0901MC211020
4	Aayushi Bhargava	0901MC211021
5	Divyansh Savita	0901MC211026

Faculty Mentor:

Dr. DK Jain (Faculty Coordinator, DEPT. of MAC, MITS, Gwalior)

Submitted to:

Department of Engineering Mathematics & Computing

MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE

GWALIOR-474005

3rd Year- 5th Semester

July-Dec 2023

turnitin

Similarity Report ID: oid:28506:45715783

PAPER NAME

edited minor project (2).docx

WORD COUNT	CHARACTER COUNT
3923 Words	22628 Characters
PAGE COUNT	FILE SIZE
27 Pages	873.8KB
SUBMISSION DATE	REPORT DATE
Oct 31, 2023 1:53 PM GMT+5:30	Oct 31, 2023 1:53 PM GMT+5:30

13% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.

- 6% Internet database
- Crossref database
- 13% Submitted Works database
- 3% Publications database
- Crossref Posted Content database

Excluded from Similarity Report

- Bibliographic material

31. Oct. 2023

(ARIN TIWARI)

(ANSHIKA SINGH)

(AYUSH SHARMA)

(Divyansh Savita)

(Ayushi Bhargava)

Summary

MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE
GWALIOR

(A Govt. Aided UGC Autonomous & NAAC Accredited Institute Affiliated to RGPV, Bhopal)
NAAC Accredited with A++ Grade

CERTIFICATE OF GUIDE

This is certified that **Anshika Singh** (0901MC211012), **Arin Tiwari** (0901mc211015), **Ayush Sharma** (0901MC211020), **Aayushi Bhargava** (0901MC211021), **Divyansh Savita** (0901MC211026) has submitted the project titled Sentiment Analysis for Tweets: A Random Forest Approach under the mentorship of **Dr. DK Jain**, in partial fulfilment of the requirement for the award of the degree of Bachelor of Technology in **Engineering Mathematics & Computing** from Madhav Institute of Technology and Science, Gwalior.

Dr. DK Jain
Professor and Faculty coordinator,
Engineering Mathematics & Computing

Dr. Vikas Shine
Professor and Head,
Department of MAC

MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE, GWALIOR

(A Govt. Aided UGC Autonomous & NAAC Accredited Institute Affiliated to RGPV, Bhopal)
NAAC Accredited with A++ Grade

DECLARATION

I hereby, declare that the work is presented in this project report, for the partial fulfilment of the requirement for the award of the degree of Bachelor of Technology in Engineering Mathematics & Computing at Madhav Institute of Technology & Science, Gwalior is an authentic and original record of my work under mentorship of Dr. DK Jain, faculty Coordinator, DEPT. of MAC, MITS, Gwalior.

I declare that, I have not submitted the matter embodied in this report for the award of any degree or diploma anywhere else.

Date: 22/11/2023
Place: Gwalior

Anshika Singh(0901MC211012)
Arin Tiwari(0901MC211015)
Ayush Sharma(0901MC211020)
Ayushi Bhargava(0901MC211021)
Divyansh Savita(0901MC211026)

3rd Year- 5th Semester
Engineering Mathematics & Computing

MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE
GWALIOR

(A Govt. Aided UGC Autonomous & NAAC Accredited Institute Affiliated to RGPV, Bhopal)
NAAC Accredited with A++ Grade

ACKNOWLEDGEMENT

The full semester project has proved to be pivotal to my career. I am thankful to my institute, **Madhav Institute of Technology and Science** to allow me to continue my disciplinary/interdisciplinary project as curriculum requirement, under the provisions of the Flexible Curriculum Scheme (based on AICTE Model Curriculum 2018), approved by the academic council of the institute. I extend my gratitude to the Director of the institute, **Dr. R. K. Pandit**, and the Dean of Academics, **Dr. Manjaree Pandit** for this.

I would sincerely like to thank my department of engineering mathematics & computing, for allowing me to explore this project. I humbly thank Dr. Vikas Shinde, Professor, and Head, Department of Engineering Mathematics & Computing, for his continued support during the course of this engagement, which eased the process and formalities involved.

I an sincerely thankful to my faculty mentors. I am grateful to guidance of **Dr. DK Jain**, faculty coordinator, DEPT. of MAC, Gwalior, for his continued support and guidance throughout the project. I am also very thankful to the faculty and staff of the department.

Anshika Singh(0901MC211012)
Arin Tiwari(0901MC211015)
Ayush Sharma(0901MC211020)
Ayushi Bhargava(0901MC211021)
Divyansh Savita(0901MC211026)

3rd Year- 5th Semester
Engineering Mathematics & Computing

Abstract

The project helps to develop a sentiment analysis tool tailored specifically for social media data, with a focus on analysing tweets. Twitter serves as an ideal data source due to its diverse opinions and emotions. The project uses Natural Language Processing (NLP) techniques to process and analyse the textual data, extracting significant features. To enhance predictive accuracy, the Random Forest classifier, which is mainly used to handle complex data relationships, is chosen as the core modelling technique, leveraging ensemble learning principles.

Apart from sentiment categorization (positive, negative, or neutral), the system will also classify tweets into three distinct categories, allowing for a more strong understanding of sentiment expressions. The system will further incorporate a recommendation engine that utilises sentiment analysis results to suggest similar tweets. By identifying tweets with similar emotional tones, users can explore a broader range of content aligned with their interests and sentiments.

The project's success relies on the acquisition of a comprehensive dataset of tweets covering various topics and emotions, which will be helping us for both training and testing the sentiment model. Rigorous evaluation metrics will be functional to ensure the model's accurateness and consistency across a spectrum of sentiment categories. This project addresses the need for an advanced sentiment analysis tool tailored to social media data, contributing to a more nuanced understanding of online sentiment expressions and facilitating content discovery for users.

Table of Contents

Problem Statement

The main purpose of this project is to create a robust sentiment analysis tool tailored for social media data, specifically tweets. Twitter provides a rich dataset of diverse opinions and emotions, making it an ideal source for sentiment analysis. NLP techniques will be applied to preprocess and analyze the text data, extracting salient features.

Random Forest classifiers are chosen because they can handle complex relationships in data and provide accurate sentiment predictions. This ensemble learning technique combines multiple decision trees to improve the model's predictive performance.

In addition to sentiment categorization, the system will also categorize tweets as positive , negative, or neutral. This finer granularity initiate more nuanced understanding of sentiment expressions.

The recommendation engine will utilize sentiment analysis results to suggest similar tweets. This will be achieved by identifying tweets with similar emotional tones, helping users discover a wider variety of content that matches their interests and feelings.

A comprehensive dataset of tweets covering various topics and emotions will be collected for training and testing the sentiment analysis model. The model will be evaluated thoroughly using different metrics to ensure that it is accurate and reliable for a wide range of sentiment categories..

Introduction

In the era where social media platforms like Twitter serves as a dynamic stage for the expression of diverse opinions and emotions, the need for a robust sentiment analysis tool has never been more pressing. The purpose of this project is to address this need by crafting a tailored sentiment analysis tool, finely tuned with the unique characteristics of social media data, particularly tweets. Twitter, with its vast repository of succinct yet expressive text, offers an ideal canvas for sentiment analysis, where the challenge lies in deciphering the subtleties of human emotions within 280 characters.

To unravel the intricate tapestry of sentiment within these tweets, we will harness the power of Natural Language Processing (NLP) techniques. These methods will be working to preprocess and analyse the text data, teasing out the most pertinent features that underlie the sentiments expressed. But our approach doesn't stop there; we are eager to leverage the Random Forest classifier, chosen for its remarkable ability to unravel complex relationships within data, delivering highly accurate sentiment predictions. This ensemble learning technique amalgamates numerous decision trees, thereby elevating the predictive prowess of our model.

Moreover, we recognize the necessity for a more nuanced understanding of sentiment in social media discourse. Thus, our system will not merely categorise tweets as positive or negative; it will also identify and classify tweets into a third, vital category: neutrality. This three-fold categorization will empower users and analysts alike to grasp the full spectrum of sentiment expressions, painting a richer picture of public sentiment on Twitter.

But we don't intend to stop at sentiment analysis alone. We envision a recommendation engine that harnesses the insights garnered from sentiment analysis. By identifying tweets with similar emotional tones, we aim to offer users a curated experience, suggesting tweets that align

with their interests and sentiments. This will not only enhance user engagement but also broaden their horizons, exposing them to a wider array of content that resonates with their emotional state.

To achieve these ambitious goals, we will assemble a comprehensive dataset of tweets spanning various topics and emotions, carefully curated for training and testing our sentiment analysis model. To confirm the utmost correctness and reliability across diverse sentiment categories, we will employ precise evaluation metrics. In doing so, we embark on a journey to unlock the hidden emotions within the Twitterverse, creating a potent tool for understanding, analysing, and harnessing sentiment on one of the world's most influential social media platforms.

What do you mean by Sentiment analysis?

Sentiment analysis , is a part of natural language processing (NLP) in data science and machine learning that focuses on evaluating emotional quality shown in a piece of writing. The aim of sentiment analysis in a data science project is to inevitably classify a given text , or to assign a numerical score to represent the sentiment intensity, such as a sentiment score between -1 (very negative) and 1 (very positive). Sentiment analysis has numerous practical applications, including:

- Understanding customer sentiment about products or services.
- Analysing public opinion on social media.
- Monitoring news sentiment for financial markets.
- Filtering and moderating user-generated content.
- Identifying emerging trends or issues in online discussions.
- Automating the categorization of feedback and reviews.

Random forest algorithm

Random Forest is a popular ensemble learning algorithm which is used for both arrangement and regression work. It combines the predictions of multiple decision trees to produce a more robust and correct prediction. Random Forest is known for its flexibility, ease of use, and capability to handle both small and large datasets effectively.

1. Reduced Overfitting: The randomness presented for sampling and feature selection aids prevent overfitting, making Random Forest less prone to memorising the training data and more robust to noise.

2. High Accuracy: Random Forest tends to provide accurate estimates because it aggregates the outcomes from multiple decision trees, reducing the effect of individual tree errors.

3. Feature Importance: Random Forest can calculates the priorities of each feature in making predictions, that can be valuable for feature selection and understanding the data.

4. Robustness: It can handle misplaced values and outliers effectively.

Random Forest is extensively used in several streams, containing classification problems like, sentiment analysis, and medical diagnosis, as well as regression work like deciding house prices, stock market trends. It is a versatile algorithm that often delivers excellent results with minimal hyperparameter tuning.

Sentiment Analysis

Sentiment analysis, It uses Natural language processing to understand the emotional characteristics of a text. The primary goal is to understand and categorize the subjective opinions or emotions conveyed within the text.
Here's an explanation of sentiment analysis:

1. Text Input→ Sentiment analysis begins with a text input, which can range from a short tweet to a lengthy article or review. The input could be in the form of written text, articles, posts, or any other textual content.

2. Text Preprocessing→ Text is often cleaned and prepared for analysis by breaking it into words or phrases, removing common words, and handling punctuation. This step helps simplify the text for analysis.

3. Feature Extraction→ Identify the most important words and phrases in the text, as these will be essential for building a model which can understand and categorize sentiment.

4. Sentiment Classification Models→ ML model for sentiment analysis can classify data in several categories. Common models include Naive Bayes, Support Vector Machines ,Recurrent Neural Networks ,and Transformers. And the one we are using in this project is Random Forest .

5. Training Data→ Sentiment analysis model is trained on labeled data,

where each text is labeled with a sentiment .During training, the model learns how to recognize patterns and associations between words and sentiments.

6. Supervised Learning→ Sentiment analysis model is trained on labeled data, where each text is labeled with a sentiment .During training, the model learns how to recognize patterns and associations between words and sentiments.

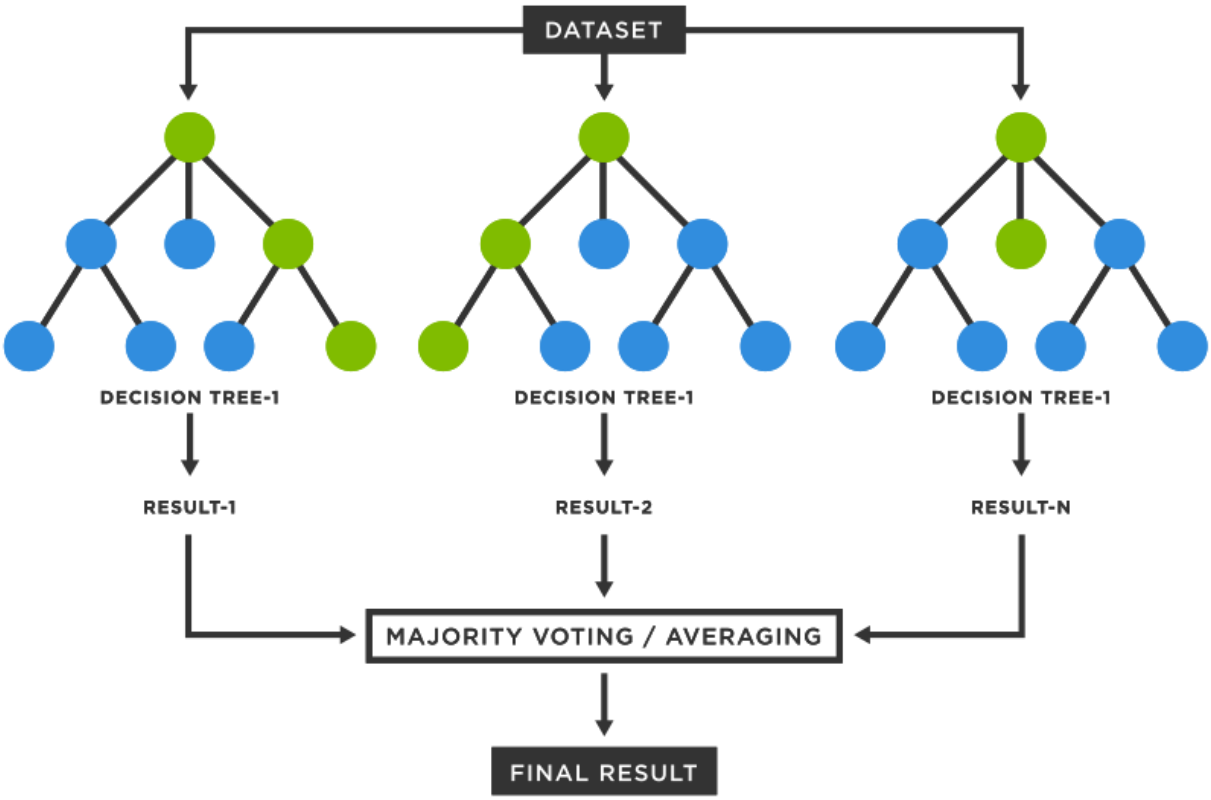
8. Domain-Specific Adaptation→ Sentiment analysis models can adapted into specific domains or industries to improve accuracy. This adaptation includes training of model using data relevant to particular domain, ensuring it understands domain-specific language and sentiments.

9. Challenges→ Sentiment analysis is challenging because it must recognize the noise of normal language, containing sarcasm, irony. Understanding the nuanced meanings in language can be complex, and models may struggle with accurately interpreting subtle expressions.

10. Applications→ Sentiment analysis is used in several industries, like customer service, digital marketing, product development, brand management,market research. Businesses use sentiment analysis to understand public opinion and make better decisions.

In summary, sentiment analysis leverages machine learning and Natural Lanuage Processing method to examine and categorize sentiments expressed in textual content, providing valuable insights for businesses, researchers, and individuals across different domains.

Random Forest



Random Forest is an ensemble algorithm that combines many decision trees to make predictions for both classification and regression problems. Here's a breakdown:

1. Bootstrapped Sampling→ Random Forest employs bootstrapped sampling, where It resamples the training data with replacement. This creates diverse datasets for training individual trees.

2. Random Feature Selection→ Randomly considering a subclass of features at every node of a decision tree diversifies the trees and introduces randomness.
3. Tree Construction, For each tree in the forest, recursively grow the tree by choosing the feature and split point that minimizes the Gini impurity (for classification) and mean squared error (for regression).
   - Continue this process until an ending condition is achieved, such as reaching a max depth or having a node with a minimum number of samples.

4. Ensemble Learning→It mixes the predictions of multiple decision trees to create a final result. For classification, it could use a common vote, and for regression, it could use an mean of the individual tree predictions.

5. Decorrelation of Trees→ The entropy in reboot feature selection helps decorrelate the individual trees. This is crucial for the usefulness of group, as correlated trees might provide similar predictions and lack diversity.

7. Handling Categorical Variables→It can handle categorical variables. For each split part, it considers a subdivision of features that includes both continuous and categorical variables.

8. Parallelization→ The training of each trees is not dependent of each other, making Random Forest easily parallelizable. This allows for well-planned training, mainly when dealing with a bigger no. of trees.

9. Feature Importance→ It provides an amount of feature importance by researching how much each feature contributes to the removal in impurity or error across all trees. This data can be used for feature selection and considerate the impact of different features on predictions.

10. Tuning Hyperparameters→ Random Forest has hyperparameters, like count of trees, max depth of trees, the size of feature subsets. Adjusting these hyperparameters is necessary for improving model performance.

Confusion Matrix

Focus on the predictive capability of a model

		PREDICTED CLASS	
		Class=Yes	Class=No
ACTUAL CLASS	Class=Yes	a	b
	Class=No	c	d

a: TP (true positive)

b: FN (false negative)

c: FP (false positive)

d: TN (true negative)

A confusion matrix: it is a table used for improve the work of a classification algorithm. It sum up results of predictions, indicating number of TP, TN, FP, and FN instances. These variable are used to compute different performance metrics.

	Predicted Positive	Predicted Negative
Actual Positive	TP	FN
Actual Negative	FP	TN

Accuracy Metric

Ratio of true positives and true negatives to the sum of true positives, true negatives, false negatives, and false positives

		PREDICTED CLASS	
		Class=Yes	Class=No
ACTUAL CLASS	Class=Yes	a (TP)	b (FN)
	Class=No	c (FP)	d (TN)

$$\text{Accuracy} = \frac{a+d}{a+b+c+d} = \frac{TP+TN}{TP+TN+FP+FN}$$

The accuracy metric: It is a calculation of all-inclusive correctness of classification model. It is the ratio of acceptably predicted instances to the total no of instances.

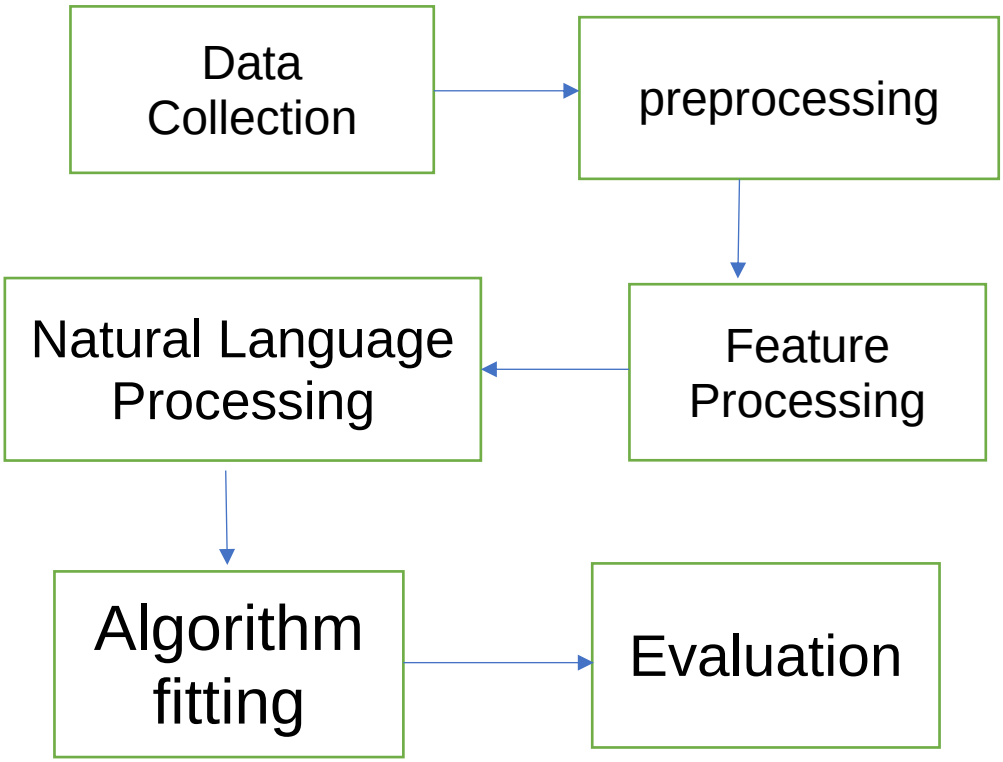
Accuracy = (Number of correct prediction)/(Total number of prediction)

Here's a brief of the terms in the formula:

(TP)→Results which were truly positive and were correctly predicted as positive.
(TN)→ Results which were truly negative and were correctly predicted as negative.
(FP)→ Results which were truly negative but were incorrectly predicted as positive.
(FN)→ Results which were truly positive but were incorrectly predicted as negative.

Accuracy offers a general calculation of how model is performing across all classes. However, it might not be the best metric in cases of not balanced datasets, where one class notably exceeds the other. In such situations, precision, recall, F1 score, or might provide further perceptive evaluations.

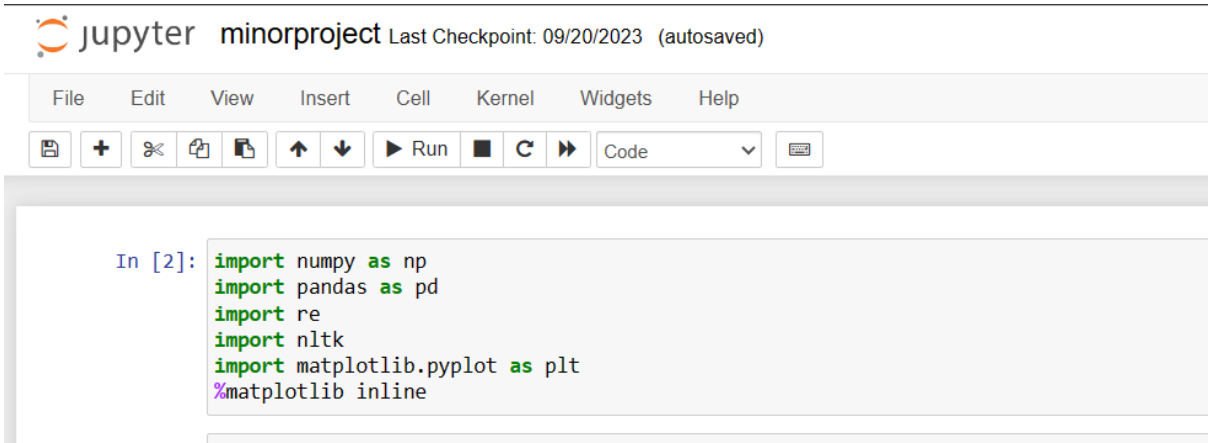
Coding explanation



Data Collection : data is collected from GitHub in form of csv , it is a Twitter data about different Airlines which has 15 columns and has 14639 rows , containing tweets about airlines which we will use train our model and then test the model

Preprocessing :

1. Libraries are imported that will help us in this process i.e Numpy , pandas ,re ,NLTK ,Matplotlib, seaborn



2. Data is read using pandas and stored in the variable named “airline_tweets”

```
In [4]: airline_tweets=pd.read_csv(r'C:\Users\DELL\Downloads\Tweets.csv')
        airline_tweets.head()
        #airline_tweets.tail()
```

Out[4]:

	tweet_id	airline_sentiment	airline_sentiment_confidence	negativereason	negativereason_confidence	airline	airline_sentiment_gold	name
0	570306133677760513	neutral	1.0000	NaN	NaN	Virgin America	NaN	cairdin
1	570301130888122368	positive	0.3486	NaN	0.0000	Virgin America	NaN	jnardino
2	570301083672813571	neutral	0.6837	NaN	NaN	Virgin America	NaN	yvonnalynn
3	570301031407624196	negative	1.0000	Bad Flight	0.7033	Virgin America	NaN	jnardino
4	570300817074462722	negative	1.0000	Can't Tell	1.0000	Virgin America	NaN	jnardino

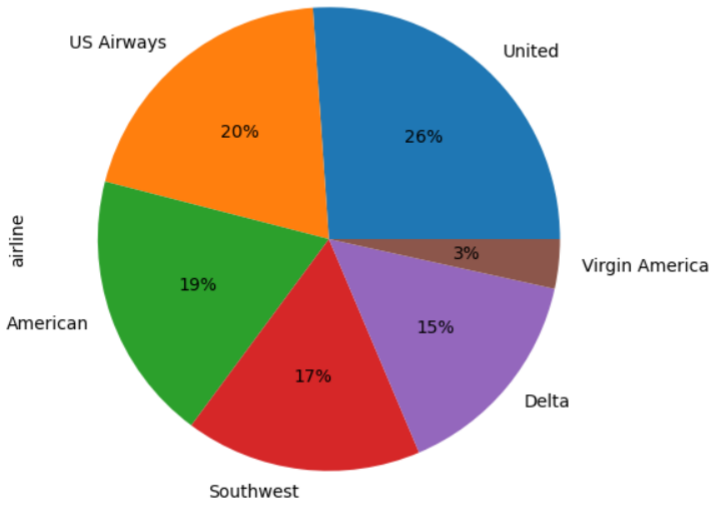
3. Fixing the plot Size , to get a better view and the graphs can be seen in organized manner .

```
In [34]: plot_size=plt.rcParams['figure.figsize']
         print(plot_size[0])
         print(plot_size[1])
         plot_size[0]=8
         plot_size[1]=6
         plt.rcParams['figure.figsize']=plot_size

         8.0
         6.0
```

4. Plotting the first graph i.e Pie Chart showing the number of tweets specified for the particular Airlines

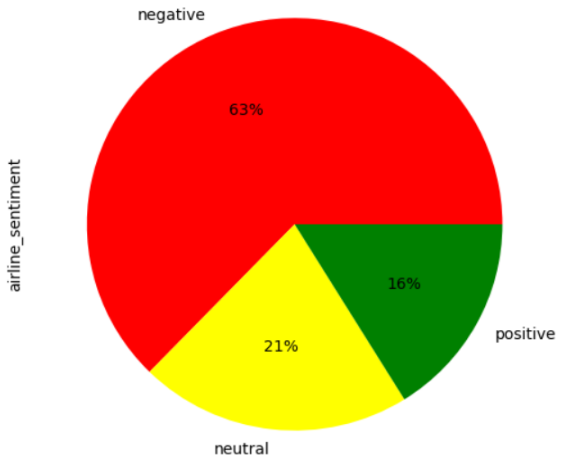
```
In [35]: airline_tweets.airline.value_counts().plot(kind='pie',autopct="%1.0f%%")
Out[35]: <AxesSubplot: ylabel='airline'>
```



It can be seen in the graph , that the most number of tweets are for the United Airlines i.e 26% , and the least are on the Virgin America Airlines i.e 3%.

5. Plotting the second graph i.e Pie Chart showing the number of tweets classified as Positive,Negative,Neutral

```
In [36]: airline_tweets.airline_sentiment.value_counts().plot(kind='pie',autopct="%1.0f%%",colors=['red','yellow','green'])
Out[36]: <AxesSubplot: ylabel='airline_sentiment'>
```



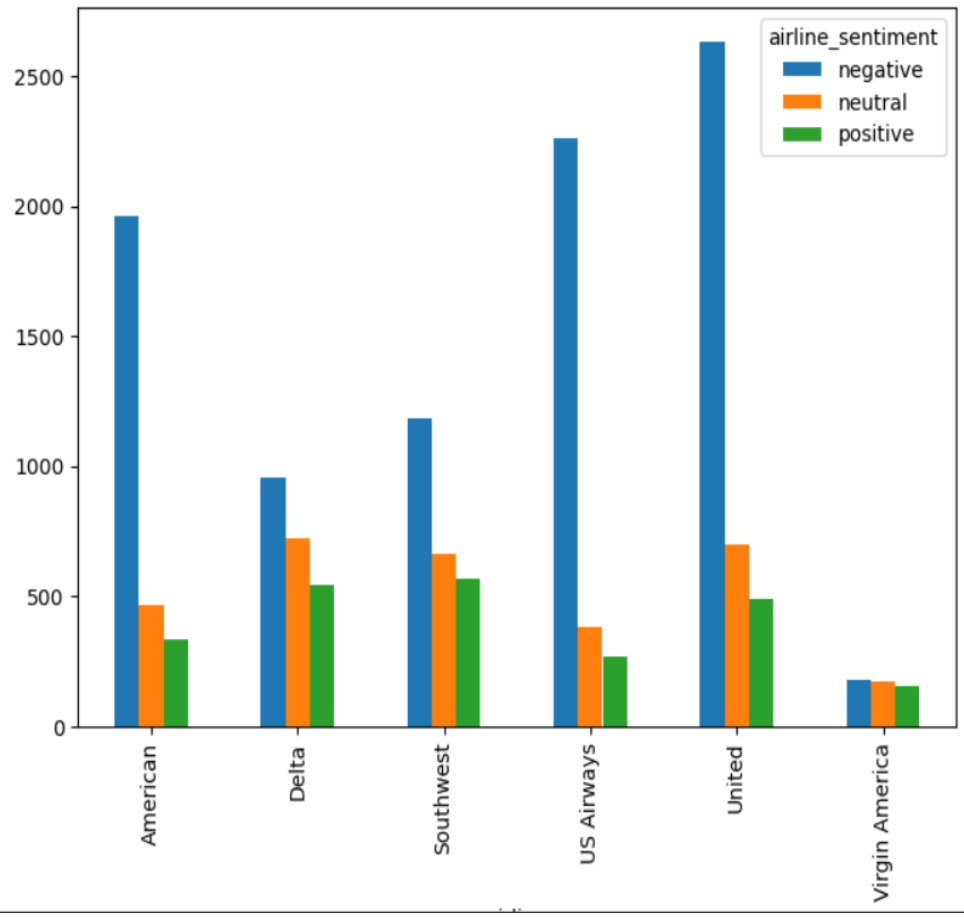
We can see that there are mostly negative tweets of about 63% and least

are positive tweets of about 16% .

6. Plotting the third graph i.e Grouped Bar Graph showing the number of positive , negative, neutral tweets of each of the Airlines .

```
In [37]: airline_sentiment=airline_tweets.groupby(['airline','airline_sentiment']).airline_sentiment.count().unstack()
         airline_sentiment.plot(kind='bar')
```

```
Out[37]: <AxesSubplot: xlabel='airline'>
```

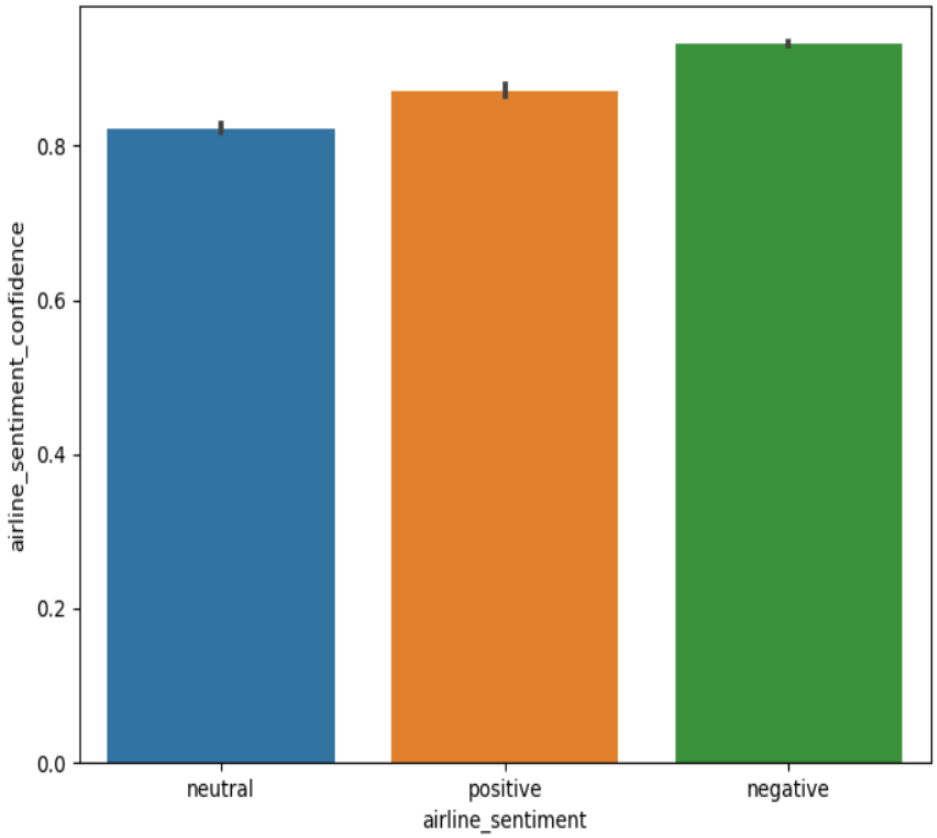


We can see that most of the negative tweets are of the United Airlines and least are of the Vrigin America , whereas the most positive tweets are of Southwest and the least are of Virgin America .

7. Plotting the fourth graph i.e Bar Graph showing the numbers of tweets classified in three categories neutral , positive, negative

```
In [38]: import seaborn as sns
         sns.barplot(x='airline_sentiment',y='airline_sentiment_confidence',data=airline_tweets)
```

```
Out[38]: <AxesSubplot: xlabel='airline_sentiment', ylabel='airline_sentiment_confidence'>
```



We can see here negative has the most confidence and the least is neutral.

Feature Processing-

1. Classify features and labels of the data for further processing

```
In [39]: features=airline_tweets.iloc[:,10].values
         labels=airline_tweets.iloc[:,1].values
```

2. Now we process the features of the tweets by removing all the special characters, every single characters, single characters which are in the start, changing multiple spaces with single space, removing prefixed B and finally converting every character to lower case, then storing them in a data structure.

```
In [40]: processed_features=[]
         for sentence in range(0,len(features)):
             #remove all the special characters
             processed_feature=re.sub(r'\W',' ',str(features[sentence]))
             #remove all single characters
             processed_feature=re.sub(r'\s+[a-zA-Z]\s+',' ',processed_feature)
             #remove single character from the start
             processed_feature=re.sub(r'\^[a-zA-Z]\s+',' ',processed_feature)
             #substituting multiple spaces with single space
             processed_feature=re.sub(r'\s+',' ',processed_feature,flags=re.I)
             #remove prefixed B
             processed_feature=re.sub(r'^b\s+',' ',processed_feature)
             #converting to lower
             processed_feature=processed_feature.lower()

             processed_features.append(processed_feature)
```

Natural Language Processing –

1. Download the Stopwords corpus from the NLTK library

```
In [31]: nltk.download('stopwords')

         [nltk_data] Downloading package stopwords to
         [nltk_data]     C:\Users\DELL\AppData\Roaming\nltk_data...
         [nltk_data]   Unzipping corpora\stopwords.zip.

Out[31]: True
```

2. Now we leverages NLTK and scikit-learn to preprocess text data. It employs tf-idf Vectorizer for feature extraction, considering 2500 terms, with stopwords removed, and filters terms based on document frequency (min_df=7, max_df=0.8). The resulting features are used for machine learning tasks.

```
In [41]: from nltk.corpus import stopwords
         from sklearn.feature_extraction.text import TfidfVectorizer

         vectorizer = TfidfVectorizer(max_features=2500, min_df=7, max_df=0.8, stop_words=stopwords.words('english'))
         processed_features = vectorizer.fit_transform(processed_features).toarray()
```

Algorithm Fitting –

1. We split our processed data in train data & test data by train_test_split feature of scikit Library

```
In [42]: from sklearn.model_selection import train_test_split
         X_train,X_test,y_train,y_test=train_test_split(processed_features,labels,test_size=0.2,random_state=0)
```

2. Now fitting the processed train data in the algorithm

```
In [18]: from sklearn.ensemble import RandomForestClassifier
         text_classifier=RandomForestClassifier(n_estimators=200,random_state=0)
         text_classifier.fit(X_train,y_train)

Out[18]: RandomForestClassifier(n_estimators=200, random_state=0)
```

3. Now predicting the sentiment of the test data and showing predictions

```
In [19]: predicitons=text_classifier.predict(X_test)

In [22]: print(predicitons)

         ['negative' 'negative' 'negative' ... 'negative' 'negative' 'negative']
```

Evaluation-

Using Classification report , confusion matrix , and accuracy score to evaluate the Algorithm

```
In [48]: from sklearn.metrics import classification_report,confusion_matrix,accuracy_score
         print(confusion_matrix(y_test,predicitons))
         print(classification_report(y_test,predicitons))
         print(accuracy_score(y_test,predicitons))
```

```
[[1723  108   39]
 [ 326  248   40]
 [ 132   58  254]]
              precision    recall  f1-score   support

    negative       0.79      0.92      0.85      1870
     neutral       0.60      0.40      0.48       614
    positive       0.76      0.57      0.65       444

    accuracy                           0.76      2928
   macro avg       0.72      0.63      0.66      2928
weighted avg       0.75      0.76      0.74      2928

0.7599043715846995
```

the confusion matrix shows the following: Positive, Negative , Neutral called as class A ,class B ,class C respectively

The element at row 1, column 1 (1723) expresses the number of results that are being rightly classified as Class A.
The element at row 1, column 2 (108) expresses the number of results that are being misclassified as Class B, but it actually belongs to Class A.
The element at row 1, column 3 (39) expresses the number of results that are being misclassified as Class C, but it actually belongs to Class A.
The element at row 2, column 1 (326) expresses the number of results that are being misclassified as Class A, but it actually belongs to Class B.
The element at row 2, column 2 (248) expresses the number of results that are being rightly classified as Class B.
The element at row 2, column 3 (40) expresses the number of results that are being misclassified as Class C, but it actually belongs to Class B.
The element at row 3, column 1 (132) expresses the number of results that are being misclassified as Class A, but it actually belongs to Class C.
The element at row 3, column 2 (58) expresses the number of results that are being misclassified as Class B, but it actually belongs to Class C.
The element at row 3, column 3 (254) expresses the number of results that are being rightly classified as Class C.

Classification Report Shows :

Precision: Precision is the ratio of true positives to sum of true positives and false positives. It expresses how many of the predicted positive instances are actually positive.
For "negative": Precision = 0.79
For "neutral": Precision = 0.60
For "positive": Precision = 0.76

Recall: Recall is the ratio of true positives to sum of true positives and false negatives. It shows how many of the real positive results were correctly predicted.
For "negative": Recall = 0.92
For "neutral": Recall = 0.40
For "positive": Recall = 0.57

F1-Score: The F1-score is harmonic mean of precision and recall. It shows balance between precision and recall.
For "negative": F1-Score = 0.85
For "neutral": F1-Score = 0.48
For "positive": F1-Score = 0.65

Support: Support is known as number of occurrences of each class in true dataset.
For "negative": Support = 1870
For "neutral": Support = 614
For "positive": Support = 444

Accuracy: It is Overall accuracy of the model, which is ratio of correct predictions to the total number of predictions. In this case, it's 0.76 or 76%.

Macro Avg: It is the Mean of precision, recall, and F1-score from all classes, without considering class instability.
Macro Avg of Precision = (0.79 + 0.60 + 0.76) / 3 ≈ 0.72
Macro Avg of Recall = (0.92 + 0.40 + 0.57) / 3 ≈ 0.63
Macro Avg of F1-Score = (0.85 + 0.48 + 0.65) / 3 ≈ 0.66

Weighted Avg: This is the weighted mean of precision, recall, as well as F1-score, considering the support (i.e., number of results) for each class.

Weighted Avg Precision = (0.79 * 1870 + 0.60 * 614 + 0.76 * 444) / 2928 ≈ 0.75
Weighted Avg Recall = (0.92 * 1870 + 0.40 * 614 + 0.57 * 444) / 2928 ≈ 0.76
Weighted Avg F1-Score = (0.85 * 1870 + 0.48 * 614 + 0.65 * 444) / 2928 ≈ 0.74

Accuracy Score : The accuracy score of approximately 0.7599 (or 75.99%) indicates that the model's predictions were correct for about 75.99% of the total instances in the data. We can also say that, it means that the model made accurate predictions for roughly 76 out of every 100 results.

Future use and Scope of Sentiment Analysis For Tweets – A random forest approach

Sentiment Analysis for Tweets using a Random Forest approach has promising future applications and a wide scope across many different industries and domains. Here are some probable uses and scope for this technology:

Brand Perception and Customer Feedback:
Companies can use sentiment analysis to display public sentiment towards their products or services. This information can be used to improve offerings and express customer concerns and priority.

Market Research:
Understanding public sentiment forward specific products, brands, or industries can provide valuable insights for market research. This can help companies make informed and precise decisions on the product launches, marketing strategies, and competitive positioning.

Crisis Management and Public Relations:
During crises or emergencies, sentiment analysis can help organizations monitor public sentiment in real-life. This allows them to respond quickly and appropriately to alleviate potential damage to their reputation and image.

Political Analysis:
Sentiment analysis can be used to calculate public sentiment towards political parties, candidates, and policies. This information is important for political campaigns, strategists, and pollsters.

Customer Support and Service Improvement:
By analyzing customer feedback on social media platforms, companies can recognize common issues and areas for improvement in their products or services.

Stock Market and Financial Analysis:
Sentiment analysis can be used to gauge market sentiment forward specific stocks or sectors. This information can be useful for making investment decisions.

Social Media Marketing:
Marketers can use sentiment analysis to evaluate and improve the effectiveness of their campaigns. It helps in understanding how consumers are reacting to specific marketing messages or promotions.

Product Development:
Sentiment analysis can be used to collect insights into customer preferences, choices and opinions which can inform the development of new products or features.

Event Monitoring and Trend Analysis:
Tracking sentiment during events, conferences, or product launches can provide immediate feedback on how well they are being received by the audience.

Election Prediction:
Sentiment analysis of tweets related to political candidates can be used to predict election outcomes. This has been demonstrated in several studies.

Healthcare and Pharma:
Sentiment analysis can be used to monitor public sentiment towards healthcare policies, pharmaceutical products, and medical treatments. This data is crucial for understanding public perception, choices and acceptance.

Customer Churn Prevention:
By monitoring sentiment, companies can identify customers who are dissatisfied or unhappy and take pre-decisive steps to retain them.

Sentiment-driven Chatbots and Customer Service Automation:
Sentiment analysis can be integrated into chatbots and automated customer service systems to understand and respond to customer emotions more effectively and efficiently.

Education and E-learning:
Analyzing sentiment in student feedback or reviews of educational content can help improve course materials and teaching methods.

Tourism and Hospitality:
Sentiment analysis can also be used to assess customer satisfaction with hotels, tourist destinations, and travel experiences, helping businesses make necessary improvements and changes, respectively.

Conclusion

In conclusion, the sentiment analysis system, employing Natural Language Processing (NLP) techniques and a Random Forest classifier, exhibits promising performance. The model accurately categorizes tweets into distinct emotional states of positive, negative, and neutral, showcasing its proficiency in understanding and interpreting sentiment expressions in social media text. The classification report further highlights the model's effectiveness, with accuracy, recall, and F1-Score values reflecting its ability to make precise predictions and minimize false negatives and positives Notably, the precision values for 'negative,' 'neutral,' and 'positive' sentiments are 0.79, 0.60, and 0.76, respectively. These scores demonstrate the model's precision in properly identifying each sentiment category.

Moreover, the recall values for the sentiments are 0.92, 0.40, and 0.57 for 'negative,' 'neutral,' and 'positive,' respectively. These values emphasis the model's capability to capture the true positive instances effectively. Additionally, the F1-Score, which strikes a balance between precision and recall, attains values of 0.85, 0.48, and 0.65 for the corresponding sentiments. These metrics collectively illustrate the robustness of the sentiment analysis system.

The accuracy score of approximately 75.99% further solidifies the model's commendable performance. This score indicates that the model's predictions were correct for about 76 out of every 100 results in the dataset. While accuracy is a vital metric, it's important to acknowledge that in cases of class imbalance, it may not provide a comprehensive assessment of the model's efficacy. Hence, it's imperative to consider additional metrics like precision, recall, and F1-Score to get more understanding of the model's performance, especially in scenarios where certain sentiment classes may be less prevalent.

The sentiment analysis system demonstrates a commendable ability to discern and categorize sentiments in tweets. The accuracy score of 75.99% provides a solid foundation for its effectiveness, yet the model's proficiency goes beyond mere numbers, as it successfully interprets and processes the nuanced expressions of sentiment in social media discourse. This capability holds significant potential in various applications, from market research to sentiment-driven product recommendations, offering valuable insights into the public's emotional responses on social platforms.

References

[1] https://www.simplilearn.com/

[2] https://github.com/satyajeetkrjha/kaggle-Twitter-US-Airline-Sentiment-/blob/master/Tweets.csv

[3] https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html

[4] https://scikit-learn.org/

[5] https://docs.python.org/3/library/re.html

[6] https://www.nltk.org/