Sentiment And Behavior Analysis Based On Social Media

Chanpreet Singh
Information technology
(Data Science)
Carleton University, Ottawa, ON, Canada

Prof Dr Omair Shafiq
Carleton School of Information Technology
Faculty of Engineering and Design
Carleton University, Ottawa, ON, Canada

I. ABSTRACT

Social Media networking growing on enormous rate creating lots and lots of data on cloud that arises a lot of problems in dealing with the storage and categorical management of hidden information which could be utilized by substitution of data into meaningful sources. Such Meaningful sources would enable to learn about current trends as well as helps to analyze latest records for the human behavior and their sentiments towards any product, current topic or any issue as well as a person, gadget or anything on social media. The Main aim of this project is to target a certain set of information provided on a real time dataset or a previously collected dataset from twitter or any other social media to be analyzed in such a way that the algorithm accelerates towards providing a feedback about the behavior of people on a certain topic on social media in terms of the positivity and the negativity of the topic selected. For this research we focused on Airline Sector which is a very busy organization under the field of travelling playing a crucial role in high income profits. On the other hand, this sector has not been much studied on the field of data science to understand the advances as well as improvements to be done with the help of social media using Artificial neural networks and other classifiers.

II. INTRODUCTION

The Burning question of today's social media marketing as well as corporate industries related with Online shopping and Businesses Worldwide revolves around how people would react to an updated situation. This situation could be a product, an update version of a software, a new user interface environment or anything that has to deal with social media or anything that is available online publicly where people can provide their opinions. These opinions are mostly recorded on various types of platforms under which they have different forms and situations on which the information gets collected. These platforms include social media such as YouTube, Facebook, twitter, BlogSpot, tumbler, Pinterest, WordPress, and many other sources where people can interact within each other or any publicly available feed. To add to it, the Sources under these platforms depend as media such as photos, audio or video, text or an advertisement etc. where information gathered is in the form of comments, likes as well as reactions in terms of emoticons and public pole or a measure scale ranging from low to high. The whole lot of this information is generating every single second. So, the main aim is to control or simply these opinions into something useful and easily readable form such as a positive feedback, a negative feedback or a neutral feedback from public posts in order to come across the best output for future. For instance, a company launched a new speaker making 1000 units and planned to make 5000 if the first launch of units gets successful. So, the company sells the first 1000 units and tells the users to activate their warranty by providing their valuable feedback after using the product within 30 days. After this whole process is done and the information gets collected in terms of sentences which are broken into words and from their database of positive and negative words each feedback gets matched of having more positive words or negative. If the company gets more positive words so that means their launch is successful and if not, then they have to make some upgrades in the speaker to prelaunch as a new version. This was on a smaller scale of unit but imagine on a larger scale and on any other product or let's say any current topic on internet such as a topic related to politics that whether a politician of a certain party will win or not. For this we have two different topics, which includes selection of a politician's popularity

on social media and on the contrast the selection of the political party which is more in liking by the public opinion. Here the theory involves a concept of "Bag of words" this process involves an online available or a local unit host data which includes positive words folder as well as a negative words folder on a larger scale.

Moving further the both queries will run on the algorithm for selecting comments on any social media, lets say for this case we can select twitter, Twitter is most popular because tweets are public and easily available with full access and freedom to analyze under developer mode facility by twitter API. The next step involves reading the information on related articles or topics which are relevant to the same political party and issue with politics or any other topics such as hash tags or meta tags from other sources sharing information via twitter and getting pre processed by cleaning and matching from the bag of words concept. The last stage will leave with the results for showing some number of readable tweets, the positive and the negative ratio of both the queries in each case sub divided into two parts. Ultimately giving a prediction idea that this political party is mostly liked and on the other hand a politician with the most positive feedback have more popularity and positive feedback giving more chance of win for some accuracy of the algorithm. According to most of the research papers which will be further described in this paper there is no such algorithm which can say that the following event is a true event making it like an accuracy above 90% Which makes data scientists to work harder and this is what that makes behavior and sentimental analysis so important making lifestyle much more easier for almost all the industries to make decisions which will take act upon the future in a positive manner.

III. LITERATURE REVIEW

To begin with, there are many research papers related to social media out of which the most important ones which have a deep impact on sentimental analysis dealing with the user behaviors and reactions on content through social media will be discussed further in this research paper. Marcelo Maia et al [1] in a research paper conducted a study on YouTube dataset which explores the behavior sentiments of an individual through the types of content searches, views , shared on social media and repeated as well as based on other behavior factors as upload select videos to watch , choose friends to share and connect, comment, like , dislike or rank favorite content, subscribe to channels , blogs and users and do many other interactions [1] decides the next video to be shown and the types of advertisements to be played and helps to learn various other concepts of user's utility behavior as well as the attributes and the meta-tags which analyze the data by artificial intelligence to match the sudden actions by different types of users. In a research by Eugene Agichtein et al [2] a study explains that getting feedback from people about making a priority for marking a benchmark of certain websites helps a lot in behavior analysis through the reactions as well as emotions connected by people on output results. Sometimes, these points are not being exposed in a written manner which makes it harder to be known by the business or the media marketing industries which lacks the fact that there is a need of improvement in certain part of a particular website. Therefore, the conclude that web search algorithms need improvements in clicking, browsing and text search query for competitive web search filtering to provide them a particular rank by using the feedback in a disciplined manner which is shown in their research helping improve accuracy. Moving Further, there is another interesting paper by Jaimie Y. Park ∗et el [3] which focus on a study for how an individual acts when searching for an image on internet and what factors determine the behavior of the output when the results are shown and how they manage to figure out using the filters to get their desired searched image. They also conducted an informative survey study which explains what, how and why people search for specific images. From this we can get an idea of how queries could manipulate a specific search for images. For instance, if a person wants an orange with a knife is different from a person searching a knife with orange and using some other filters. So even if we change any small detail from the filtrations the whole query goes on some other tangent because the query will focus on the first word and check all the metatags for that word and hence getting it to be prioritized than the second word. In another research paper

by Wesley G. Siqueira and Laercio A. Baldochi [4] Changed the way an analyst would see the use of logs generated by user on web pages. They proposed an object modeling based document under tree leveling structure[4] to understand the concept of how important the tree structure can guide about the insights generated by the graphical plotting of the user reacting up on how an individual uses web pages and all this information could be distinguished by separating different elements on unique web logs reading with the use of graphs by tree structure generated by several applications which are working based upon web. A paper in 2016 by Guimei Liu et el [5] used the concept for user behavior study to unfold the facts for E commerce, to predict a user with a repetitive action for being a returning as well as a similar type of shopping habit consistency in buying and selecting items bringing profits to organizations. This paper helps a lot to understand the power of promoting a product because in the end of every segment every business wants to cut the promotion costs and increase the number of returning customers which are loyal that ultimately acts as return on their investment in business or product sales. Now this paper has a solution to this condition by making a model which extracts each and every connection between the features to be selected as well as the profile category such as users, brands, category, items, merchants and their interactions [5]. This method creates a web of co-relational links within the profiles making it to look like a three dimensional structure where all the layers are inter-dependent upon each other describing a pattern that how a user select the product and what else to recommend to the user will be decided by the various analytical models which learns and trains under the conditions for prediction tasks in future. In a paper published in 2016 by Xianfen Xie and Binhui Wang [6] they elaborated the concept more briefly for the recommendation of web pages by studying user behavior and interlined relation of the topics to be selected by the method of twofold clustering [6]. In this paper they improved the copying of information to show the fresh content distinguished from the previous available content by making the classes in form of popular topics , recently released topics as well as the history and the new clusters inter-relationship to precise the knowledge which the user will get in future with the help of two-step clustering [6] because it will automate the system to make the distribution of collection data points into a space which is 2D in nature and clearly separates the historical data which the user has already viewed to neglect information from the web pages which are not given much statistics by the user views as well as a lot of other factors. On the other hand another interesting paper by jiahui liu et el [7] tells about how people behavior gets notified and could be calculated towards predicting the next news article information to be provided or pop up advertisements as well as related topics to be targeted to the user while learning from its previous statistics over the reactions received on the past searches. For example if a user searched for car news or any engines of cars and prices for a week ultimately within the next week automatically next time the user logins the browser , it will show them the topics related to cars first without even entering any context on the same pace of keeping it representation as a news, advertisement or any information update which is co-relational to cars. The researchers used Bayesian model [7] for this prediction technique which as a result increases the onclick rating as well as the user insights, time spend hours within more reach to other users. For additional topics to be covered such as involving the negativity as well as the positivity of the content available on the internet a paper published in 2018 on " Community interaction and conflict on the web " [8] by Srijan Kumar et el [8] comes with a very informative and neat representation of understanding that how this process actually works from distinguishing between various topics on the internet as well as the pros and cons with insights of negative, neutral and conflicting content in the sub topics of each type respectively [8]. The results show that the lesser the number of users belonging to a sub division of smaller community under any topics related to different fields on the internet are highly creating conflicts with their peer impact over the communities which has higher number of users [8]. These conflicts arise the decrease in number of users for a centralized channel or their activities or usage get manipulated by such negative comments or responses targeted by very

small number as 1 % of the total users spreading spam content or fake views or feedback making a larger impact. Thus a model called Long short-term memory (LSTM) [8] which is an artificial recurrent neural network (RNN) architecture was used to early alert the defenders for making a straight contact with the attackers [8] and block content resolving problems for future and bringing a positive feedback environment in the field of digital media network on internet using these frameworks for safe interaction between several communities [8]. A paper by Xiaohui Xie Et el [9] has a very specific contribution on a topic that image search is been a vital resource for users to interact more of a real time behavior and giving feedback on what they see and how they behave according to it when they search something and see the results. This paper helps to understand beyond this limit for "why people search for images "[9] and how they react according to the output studied by their feedback sentiment analysis of behavior when they have several sessions of searching logs [9]. They group different sectors for users who want to search for just learning, finding something or as well as for creativity purposes to entertain [9]. The factors are altered to reform upon a prediction model for how much time user spends , hover , scrolling as well as clicking or selecting [9] which for future use helps the next scrolling images to be automatically be ready in fitting to that context learning by previous factors according to different users [9]. Social media influencers are a burning question of the time because these users are those who can create a positive as well as negative hype on any topic due to majority of people following them through social media and believe them as their role models such as actors, writers, singers, motivational bloggers etc. Similar to this topic a paper by Zeynep Zengin Alp Et el [10] shows that there are number of social media platforms out of which twitter is the most friendly because of being an open data base to analyze as well as with more public users as compared to other platforms where privacy is the major concern to extract information[10]. This paper helps to extract information related to media uploaded by users at one place to transfer and diffuse completely all over by manipulating the target marketing using technique of PPR (Personalized page rank)[10] which will ultimately help to gather tweets as well as media available by expert influencers on any specific topic already available free of cost just in need to be spread virally all over helping to save factors such as cost, time, promotional advertisement as well as transport, communication , portfolios costings and planned media marketing using professional models etc. [10]. Another Research paper which helps to explore and gain more knowledge upon the topic for learning sentiment analysis is covered under various categories of selection of an article, journal, a webpage or any information provided to study and research for that topic. To add to it, this research paper includes a detailed review compared model for 52 articles from 2010 to 2016 [11] which are sub divided into platforms such as techniques used, actions of study performed under analytics on different data sets , the topic revolving around different types of studies and extractions from different data available on social media platforms, objectives as well as conclusions to be improved for future researchers interested in selecting of a topic and field is clearly given more scope by this research helping to gain on sub categories of data scientists to be finding out the tendency to up and downs, inconsistency and preprocessing to read and write while making it consistent for ease of use, behavior, trend setter as well as the knowledge upon which one could classify this type of immense information into different metrics in marketing [11]. A recent research published in 2018 by T. Stefano Et el [12] guides a way to a field which has pros and cons over the topic "Personality change"[12] over which conclusions have been drawn under behavior of individual in an organization matrix content [12].This article helps to study that behavior change is important sometimes to grow , learn and boost creativity as well as improving consistent growth by learning under attributes, character as well as idiosyncrasy manner[12]. To add to it, Commitments, demands, obligation as well as self-build up helps to establish studies which optimizes the approach based upon the results on all these factors upon individual users in an organization has been represented in this research paper [12]. A very important journal research published in June 2018 by Jieun Shin Et el [13] has a very crucial role in contribution to this research paper which directly imposes on usage of how and what type of rumors[13] can influence an impact for creating a false information which can lead to future disasters in politics of a country [13]. The liability or tendency to change of a false statement which gets spread and restated a thousand times by the

users who can create a viral thread of the same information to create a hype will ultimately can cause a good decision to be look like a bad one and hence resulting in a wrong heavily influenced culture where any powerful person can spread a rumor by costing money and promotions against defamation of any other political party through the power of public database management systems lacking word to vector detection by using artificial neural networking to delete or spam as well as review or consent the information related to any person live or dead or anything related towards that topic. This will help to stop making any mindset towards the community which will have their rights to elect any political parties ruling over them for the next years. Thus, this research has used several methodologies learning the patterns, the context, messages as well as the resources used for the original source of the rumors [13] turning to be shown as news [13]. Ultimately this technique will help the future researchers to control and diffuse the real or true information not being get dominated or manipulated by the false or fake news using time series frame study and releasing tension revolving around the targeting zone in advance prediction [13].Within the increase of websites as well as the enormous amount of data available to users today is generating to double its extent within every second of download , upload , browsing or cache generating web logs. These factors help to determine the type of search behavior a user has to go through to find a topic or product or any information to their interest. Such a research has been shown by Xipei Luo Et el[14] In a paper studying "User behavior analysis based on user interest by web log mining"[14].This research paper contributes in the field of web browsing analysis to study a model of 5 stages [14] which clearly elaborates the state of mind of a person to be judged by the futuristic algorithmic performance to introduce a new topic automatically similar or recommended to the user by study of their previous pattern of testing, training and learning about the whole data preprocessing [14]. Their research includes a sign of recognition of patterns using "Web Log Mining [14]" which are similar or consistent during a small and a large session of browsing which is then compared under the clustering[14] as well as statistical methodologies[14] which results a meaningful pattern which is similar not so important for the user to save[17] and then learning behaviour of the user logs for during the long and the short terms for future prediction of topic, product, subject of study , or any item listed to be adjusted while scrolling, clicking or advertisements to be shown just changing the metatags according to the user interest to increase more engaging time [14]. Opinion mining[19] referred to as Analysis to study Sentiment behaviour of people using internet under any field or trending contents by social media refers to the identity, recognition, abstraction as well as eradication using processing of natural language, text and word to vector analysis, and biometrics, as well as computational linguistic context or syntax [16] can be further studied under a research done by Michela Del Vicario Et el [15] on the basis of which they study the behaviour of user information on Facebook pages anticipated into different communities [15]. The research shows facts that whenever we are dissolving an information to internet it will ultimately get a mixed or biased output whenever it revolves from other person's or influencers opinion to end resulting in manipulating public visual data consumed worldwide [15]. The division into two sharply contrasting groups or sets of opinions or beliefs can be directly seen under the researchers where 1 million users[15] data was studied for patterns under sub division of internal gesture or dynamics[15].Ultimately they travel the information directly for the true and original sources of topic and it's inner informational context and gets it revolved around sub groups and various different communities and influencers[15] and gets the desired output when compared to the original the true information and the returning or the extracted information which gets manipulated again and again on a large scale of argument and users opinion as well as readers mindset[15]. On the other hand, to add to this research contribution a study conducted to analyse user behaviours under the advance research in cyber systems by Melissa Turcotte Et el [17] helps to build a model prevents mislead or false usage of user credentials as well as personal , private information used by attackers on internet[17].This research works under a very step by step technique which helps to study the logs generated by a user in various sessions[17] ,study the data gap to be considered in judging what's important and

automatically filtering the future actions to be taken or alert the user[17] , then they helps to

make a equation and study the co-relation between the server , client and the type of event[17] to be taken on a time frame and learning patterns from the possible outcomes[17] which are true and then detecting that any other attacker is to be alerted if the event of the user does not matches the previously trained pattern [17].Ultimately the model starts to utilize the whole framework to provide the suitable recommendations which are similar to the users behaviour of sessions determined and predicted by the technique involving the information which is covariate[17] to the context of the particular new session in which the log is to be matched before it gets created for inclusion in future learning model[17].

IV. COMPARATIVE ANALYSIS

A Comparison between previous related works as well as the technologies used to get a layout framework of what and how the problem statement will be designed through this table.

Refe-rence's:	TOPIC	METHODOLOGY AND TECHNIQUE OR ALGORITHM	DATASET	RESULTS
[1]	Identifying User Behavior in Online Social Networks∗[1]	K-Means Clustering[1]	YouTube[1]	User profiles using social features helping in identifying dominant user's behavior[1]
[2]	Web Search Ranking[2]	IMPLICIT USER FEEDBACK MODEL [2]	3,000 queries and 12 million user interactions[2]	Ranking Methods Compared Using various evaluation metrices[2]
[43]	Airline Sentiment Analysis[43]	K-Means algorithm, Latent Semantic Analysis (LSA)[43]	Twitter [43]	individual airline analysis, Big five personality traits
[44]	The Case of Airline Quality Rating[44]	Naïve Bayes Algorithm[44]	Twitter [44]	polarity classification accuracy of 86.4%[44].
[45]	An emotional polarity analysis of consumers' airline service tweets [45]	Lexicon Analysis [45]	Sample dataset of 2105 tweets [45]	Sentiment Scores and Stream Graphs used to show comparison of airlines[45].

Refe-rence's:	TOPIC	METHODOLOGY AND TECHNIQUE OR ALGORITHM	DATASET	RESULTS
[46]	Tweeting the friendly skies: Investigating information exchange among Twitter users about airlines[46]	qualitative content analysis[46]	8,978 user tweets and 260 airline tweets[46]	Information exchange among airline users[46]
[47]	Gaining customer knowledge in low cost airlines through text mining[47]	Spherical K-Means (SK-Means), K-means, naÏve Algorithm[47]	About 10,895 tweets (data collected for two and a half months) are analysed[47]	Customers sentiments and opinions towards low-cost carriers by utilizing social media information.
[48]	Sentimental Analysis for Airline Twitter data[48]	Naïve Bayes algorithm[48]	1295 tweets[48]	categorizing them in neutral, negative and positive sentiments[48]
[49]	Sentiment Classification System of Twitter Data for US Airline Service Analysis[49]	Decision Tree, Random Forest, SVM, K-Nearest Neighbors, Logistic Regression, Gaussian Naïve Bayes and AdaBoost.[49]	14640 tweets from twitter US dataset [49][50]	Decision Tree 63%, Random Forest 85.6%, SVM 81.2%, Gaussian Naïve Bayes 64.2%, AdaBoost 84.5%, Logistic Regression 81%, KNN 59%.[49]
[51]	An Ensemble Sentiment Classification System of Twitter Data for Airline Services Analysis[51]	Naive Bayes, SVM, Bayesian Network, C4.5 Decision Tree and Random Forest[51]	107866 tweets and 12,864 tweets cross-validation data[51]	Lexicon-based 60.5%, Naïve Bayesian 82.4%, Bayesian Network 82.2%, SVM 77.8% , C4.5 Decision Tree 83.7%, Random Forest 83.5%, Ensemble 84.2%.[51]
[52]	An approach to sentiment analysis – the case of airline quality rating[52].	Lexicon, Sentiment Detection, Naïve Bayes algorithm, CTM with the VEM algorithm[52]	Real Time Tweet Extraction[52] Tested on AQR calculation per 1000 tweets[52]	Airline Quality Rating (AQR), Naïve Bayesian 86.4%, and Sentiment Topic Recognition Results[52]
[53]	Uncovering customer service experiences with Twitter: the case of airline industry [53]	lexicon approach and vector-space model [53]	67,953 publicly shared tweets [53]	Calculated areas under customer satisfaction, dissatisfaction as well as delight [53].

Table 1.1

V. PROBLEM IDENTIFICATION

When starting with this research, the first concept come to appear was the understanding of the concept of digital media working worldwide with connection to people and their interactions within any topic related to social media terming under a concept of Netnography [28][29]. These topics include anything related to any subject matter, an organization or it's services, any product or place, any place or industry, a business etc. and so on. Moving towards the major issue which is to be dealt with involves things which need to be get controlled in a certain manner that they don't have any negative or false impact on the people retrieving the information available on internet these days. All this is because of the immense amount of database formation occurring on a daily basis and feedback or posts related to any subject is produced during any interaction of user publicly releasing opinion on social media. Such a great source of this type of publicly posted available database system is Twitter [34]. Now, we know that thousands of issues could be solved with the help of data science working under the field of sentiment analysis as discussed under the literature review section of this paper But there are some hidden problems and issues also which are remain untouched under some models or we can say that very less or few people have discussed and produced some useful research In some cases as provided under the section in comparative analysis[35].

Such a problem is related with the topic "Airline Sentiment Analysis" [50]. Now, what is airline sentiment analysis. A large amount of datasets are being created daily and filled up within various types of information but imagine the source of development of these datasets is the real users who are actually creating blogs, tweets, posts, contents, postings, images, videos or ,any other media or anything which can be uploaded via the utility of internet or social media platforms on the world wide web. This information is then used by various other businesses, enterprise , as well as organizations or multinational IT companies to develop a technique that uses a model to convert that information into useful and meaningful datasets which are then studied by data scientists to gather a pattern on which they perform statistical predictive calculations to predict or detect a problem or statistics[41][39]. This results in improvements in the social, economic as well as a digital marketing growth of any industry [38]. For example Amazon feedback and reviews helps you to select a better product for the next time automatically when you submit a comment because the metatags in those comments are studied according to the pattern and the list of products in the shopping cart by the user as well as the hover time and click with the previously watched products to decide what type of product advertisements will be shown to the user using artificial neural networking which balances the quality of products studied from the feedback of the customer. Suppose a user likes a cotton quality T-shirt from brand A and comments Bad on a T-shirt from brand B then after sometime when he refreshes it's next search or purchase something new automatically the user will be displayed products similar to brand A and other same type brands and least of brand B will be removed automatically. This whole process comes under the study of user behavior data analysis on sentiments which is already discussed under the literature review. And now moving further with the solution of Airline sentiment analysis which comes under the one of the most important topics in terms of money because airlines these days are getting expensive due to improvements in quality of services and on the other hand, there are some airlines which are improving costs to more economic dealing within providing more customer service at less costs but sacrificing their quality. So how do airlines actually get to know whether the quality standards on the practical scale in the real world matches the idea of what the company or owners of any airlines wanted to be. Here's now a solution which defines this problem known as "Airline Sentiment Analysis "because All this happens when the data from a lot of sources is collected and analyzed on the basis of feedbacks from users in the real world posting or sharing their experiences on social media without being in a biased environment.

This means that a feedback form filled just at the time of landing or while sitting in the plane is different from the feedback which the airline will send on the email after a week or by post or any other method online social media survey. To add to it, this study comes in control when the

problem is dealt by using a major source of collecting the best or the most popular publicly available data in social media list referred as Twitter Datasets [50]. Airline sentiment analysis (ASA), classifies an airlines social/public image based on social media datasets that identifies which airlines are performing optimally in domains such as Car services, Parking, Restaurants, Complementary stay lounge, as well as Sub division service categories of a airline support system which includes the following factors as Comfort, Luggage, Time discipline management or the Punctuality, Orientation Or Managerial Ground Service, Food, Employees or the airline staff, Entertainment Facilities such as Duty-Free Shopping , Arcades, Massage Therapy and Bars other recreational activities like Internet, Work space offices for officials etc. and so on[36][26].

The whole Experience from the ground to the Flight Experience, This research investigated passenger's customer journey experience from when they book their ticket and leave their home, until arriving at their destination, and the whole flight experience using data collected on twitter dataset provided by Kaggle[50] based on their behavior information on certain topics involving Negative tweets, Positive Tweets, as well as Neutral Information using a novel framework of statistical algorithmic models to calculate the behavior of an airline depending upon various factors using certain features provided under the dataset. It also shows which airlines are worst off and in between. Twitter is the best source for this type of study because as mentioned earlier in the introduction it's public and easy to fetch using a twitter developer account and then converting into a useful set of data by using word to vector processing and bag of words approach which we will study in the solution part.

GOAL: Develop a tool or a model which helps to classify a relation between a tweet and the context to be conveyed as a digital information on social media to be detected as Positive, Neutral or Negative.

Research Questions

Q1. How can we extract unrevealed facts from the information statistics on insights studied from a user or a customer 's satisfaction or dissatisfaction upon the services by airlines in form of feeds to be processed from twitter?

Q2. How can we study or catch the crucial as well as significant concerns in a customer's or user's opinion in context to the airline service experiences from twitter which is being verified at the same time to be acting as an astute source of detection tool for this type of problem?

Q3. Which classifier is the best and what types of feeds from twitter would be appropriate to choose from which a model can be highly efficient to gather, detect and predict the best type of informative awareness in terms of airline experience as per the user's perspective?

Q4. How can the datasets be converted into useful information that can be utilized by airlines and in what ways the airlines will take that information into real time practical application in action?

Q5. Difference between the implementation of information transformation taking place between the microblogs from social media in engaging as well as employing by the users in contrast with the airlines in context of exchanging data/information?

Q6. What type of categories can be discovered from the types of contexts between the response feedback or exchange of information between a user and the airline communication under social media feedback management system?

VI. PROPOSED SOLUTION

The process of investigating a string to be categorized under being negative, positive and neutral is being referred to as a term called opinion mining where opinion is classified under various categories which in this case is inclined towards sentiments as well as behavioral changes of people interacting under a topic on social media, news, blogs or other places[19]. To add to it, Mining is termed here as extracting the inter-connection between the value points in conversion on a dataset platform may be as a csv file or any other type which is further utilized by the classifiers to be preprocessed under certain algorithmic statistical models for getting a useful solution to a problem or predicting a future

outcome for a conditional as well as an unconditional scenario. Thus, a solution for the association rule mining as well as clustering is being set up for the research to reveal the internal truth and facts helping retrieval of some hidden information from the insights. This will help the portrayal of the brand qualitative as well as the quantitative concept to be easier to understand and more interactive with users. On the other hand, this research will help both the users/client as well as the business/industry which relate directly or indirectly with the airlines. For example If there are 50 people who posted on social media about 3 airlines which have same route out of which Airline 1st can be booked by agent A or B but airline 2nd was booked by agent C only and Airline 3rd was private to direct booking by the company online or agent C. Now Case First, after a week of flights being travelled and bookings are carried off there are 40 feeds out of which 30 were about the airline 1st by agent A to be bad quality at high cost but 10 feeds at same time for agent B books flight 1st show low cost with some changes at good quality. Here the solution for first case that next time people will get bookings recommended from agent B and not agent A and also on the same side results from agent B will help the airline 1st to get this feedback manually to improve in context to customer satisfaction in case of agent A. Case Second, the flight directly booked from agent C under case of airline 3rd gets 10 feeds saying 8 out of them were negative towards direct booking costs and airline 2nd gets 2 feeds which were negative towards agent c booking costs. So we compare it on a small scale cluster zone then agent C is not good at booking airline 2nd but proved good results under airline 3rd. So here in this small-scale example for just 50 feeds, imagine there are thousands of hundreds of people travelling daily and posting blogs everywhere on the internet and a simple sorting and clustering with a concept of pre-processing will change the whole traditional way of feedback and response with improvement in time.

A simple solution to solve this problem starts with conversion of information retrieval. In this case we will use twitter to extract dataset because being a publicly available source of data it is not enclosed in boundaries of simple questionnaires from airlines which are preset and not being proved helpful in taking out as much more as the natural language conversation done between users on social media can come out within it.

As an example, so for this we had a tweet for having 8 words out of which it could have emoticons, symbols, words, numbers and other characters etc. We break down the words into vector to transform the words into understanding of a concept known as bag of words over which Let's say the 9 words out of which 2 are emoticons in a sentence. So we are only left with 7 words, now to match a word to be a grammatical word, a noun, a place, name or any other thing which is not related to an emotion, feeling, behavior trait ,a word which could be classified as negative or positive will be discarded when shifted in word2vec on python classifier getting epochs compared to that sentence from the bags of words on cloud which are divided into negative set of array having all sorts of negative words and similarly another set in case of positive words. Then after the matching is calculated for a sentence Let's say "good is right <3 where negative is bad :) ". Solution of this sentence will be good, is, right, where, negative, is, bad gets breakdown without the emoticons as well as the connections of "is" and "where" which gets compared on each word one by one separated by the comma giving good, right as positive as +1 ,+1 and negative, bad as -1,-1 which will be together added and comes out to be zero and the sentiment analysis for this tweet will be called as a "Neutral".

Now for this research paper with the same concept on a larger picture and a big scale industry like airlines a perfect dataset needs to be extracted as well as a process which will be helpful to make us understand the roadmap of solving this problem will be discussed further as solution in this flow diagram.

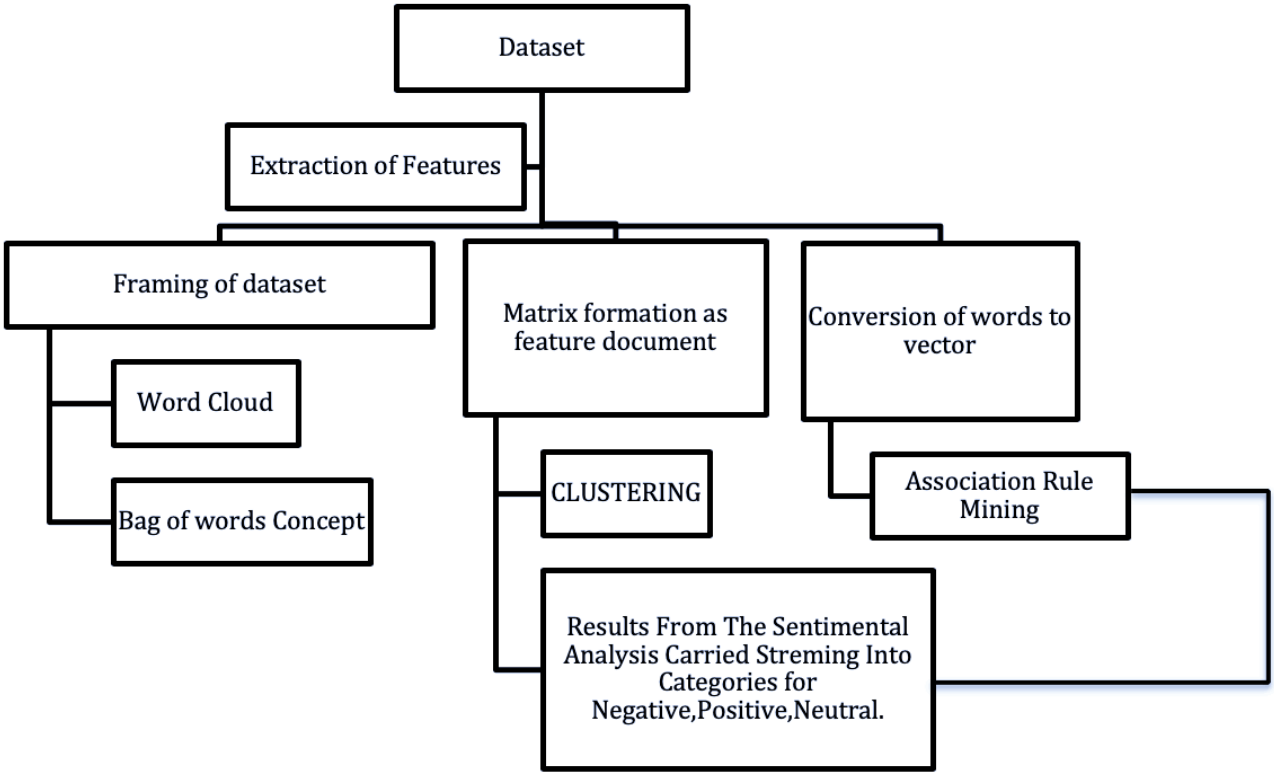

Table 2.1

To Begin with, the agenda of sentiment analysis on airline tweets based upon interactions on social media is carried by following the research roadmap as following:

- Datasets (what type of dataset, how and what process carried the data collection or what was the characteristic properties of the dataset if it was publicly available) [55].
- Pre-Processing (This stage involves all the steps carried under processing to Refine the Dataset as NLD (Natural Language Data) procedures) [55].
- Classifications Based upon Rules (This stage involves a probability check that can sentiments be classified if only with the use of some basic rules without performing the machine learning classifiers) [55].
- Binary Classifications under Machine Learning (This stage involves the implementation on sentiments be classified as positive and negative using machine learning algorithms) [55].
- Multi-class Classifications under Machine Learning (Comparing the various modules used for classifying the sentiments under prediction to be a neutral, a negative or a positive context vector and then selecting the most highly efficient classifier for testing on another data under any conditions to be unsupervised or supervised for further evaluations) [55].
- Evaluation and Results (representing the demographic insights via plots and graphs for better outline prospectus) [55].

VII. Methodology

The goal of this project was to develop a tool that will classify a tweet regarding airline as being positive, neutral or negative where in this research another case is also involved which only performs the sentiment analysis for 2 cases that is positive and negative essentially to classify the sentiment behind a tweet. To practice this research in action a twitter dataset from Kaggle known as "tweets" [50] was downloaded as a csv file. This dataset involves more than 14000 tweets which were set according to a conversion from word to vector as category which classifies a sentence to be positive, negative or neutral. The text in each of the tweet was converted into features and several different classifiers were trained on this dataset which we will discuss as we move further. These classifiers form an ensemble model which was eventually used to classify the sentiments in the tweet [37].

Hardware and Software Requirements

The Hardware Requirements for this project were as following:
Hardware Overview:

- Model Name: MacBook Pro 15 (2012)
- Processor Name: Intel Core i7
- Processor Speed: 2.3 GHz
- Total Number of Cores: 4
- L2 Cache (per Core): 256 KB
- L3 Cache: 6 MB
- Hyper-Threading Technology: Enabled
- Memory: 4 GB 1600 MHz DDR3

The software requirement involves installation of Python 3.6 or latest and under utility of python in any user interface context several modules need to be carried as following:

- Scikitlearn
- Pandas
- NLTK
- Pickle
- String
- Numpy
- Wordcloud
- Keras
- Spacy
- Seaborn

This research is unique for making a better understanding of dataset and how it has been formed and also to apply the 2 sub routines as 2 use cases under which we follow a 3 x 3 class for negative, positive and neutral and on the other hand a 2 x 2 class for only the positive and negative sentiment classifiers. To begin with, the dataset involves tweets which involves details from 6 airlines namely Southwest, Delta, United, American, U.S Airways, Virgin America [50].The dataset involves problem statements from which some Research questions were formed on the basis of which a categorical issue was occurring under which it becomes very hard to classify the tweets to be labeled as dealing to some particular field such as marketing tweet related to airline or a tweet which is just used for seeking or answering some information to a user as well as from a airline to the user as it's response/feedback. Some Issues which contains the following tags were declared under a sub topic directly related to the type of context in the tweet such as Lost of the luggage, Customer Services, Late or a Cancelled Flight, some unanswered issues, Bad experiences in flight, Issues under booking of flight and others. To solve all these measures and using the top extensive features to make a co-relational blueprint of what and how the process of analysis will be gone through here's a brief description of models to be discussed further and models which are utilized for this research.

In Research Paper [54] By Saqib Iqbal et el there is a very brief description of learning about various types of technologies as well as methods to be taken care of while implementing by understanding their pros and cons with a clear vision of practical implementation by the table 1 in the research paper [54].this helps us to gain a vision of what manipulation should be done to alter the dataset by making a complexity to be n log n which makes less the loops within less time to commute helping performance as well as

accuracy of the results. Moving further for the solution there are various models under the field of data analytics under which this research comes under the category for all the protocols which dealt with Sentiment Analysis as well as natural Language Processing [51][45][49].

Now, Sentiment Analysis Methods comprises of Lexicon-based methods as well as Hybrid methods as well as machine learning based methods. Under Lexicon based analysis methods Dictionary-based methods as well as corpus-based methods are included which further gets classified as statistical or semantic reformations. On the other hand, the hybrid methods get breakdown under Machine learning or lexicon-based and both together in one reconstruction which further becomes a vector version method termed as csk. In addition to it, the machine learning methods are categorized under three forms of learning such as unsupervised, supervised as well as semi-supervised. In this research we use supervised learning model by using various types of classifiers out of which the most vital are Naïve-Bayes, Neural Networks, Gaussian-Bayesian network, Support Vector Machine, Decision Tree, SGD Classifier, Random Forest, K-Nearest Neighbors Classifier, SVC, Logistic Regression as well as Perceptron Classifier[49][51][37][29].

To begin with the solution, we first need to gain the basic understanding of the dataset where first step involves the Tokenization of context in all the tweets extracted and performing a conversion into a regular expression which is readable as a validate dataset entry for the next part which makes the regular expressions to go on a vector conversion but before that goes through a series of loops formation which have use cases removing stop words involving any grammatical English content which does not directly or indirectly help in conveying for a sentiment in that sentence. Last but not the least all the words are converted under a condition formed for word2vec profile model where matrix of readable digital sense information is received by the algorithmic classifier to perform certain statistical analysis on the Vector acknowledgement as output. Such a type of Regular expression involves certain type of cleaning processes before making the csv file to be analyzed which includes the following:

- Converting the lower or the upper cases on a single platform.
- Removing all the URLs which does not produce a productive layout on conveying the sentiment of that particular tweet.
- Getting rid of the hashtags by converting them into simplified words.
- Replacing all the extra spaces or loops which makes no sense to be counter vectorized taking space conveying no meaning.
- Removing the Identity '@' symbol in front of all the usernames so as it gets counted getting rid of all the symbols used to tag an ID on twitter.
- Taking the '&' symbol to be substituted by the real word 'and' getting counted as an addition to the extension of a sentence which may help in knowing better about the nature of the sentence to be negative or positive.

This was a whole concept behind the tweets [50] dataset provided by Kaggle [50] publicly available. Now we begin with the utility of actions to be performed while learning side by side from comparison in both cases such as the 3x3 classifier case under which we will discuss about the sentiments as negative, positive as well as neutral category and the 2x2 category which involves classifying tweets as negative or positive prediction value using similar models.

The Framework models for Sentiment Analysis classifiers and Algorithms used are explained as follows:

1. Naïve-Bayes,
2. Neural Networks
3. G-Bayesian network,
4. Support Vector Machine,
5. Decision Tree,
6. SGD Classifier,
7. Random Forest,
8. K-Nearest Neighbors Classifier,
9. SVC,
10. Logistic Regression,
11. Perceptron Classifier.

Understanding the concept of classifiers deeply which are most important for this project research with the help of following definitions:

1. Logistic Regression: Logistic Regression is the classification algorithm that tells whether the particular value belongs to a particular class. It does not directly predict the outcome, rather, it interprets the numerical value of probability, how strongly any value is associated with the particular class. It uses the sigmoid function for the prediction of probability [55].

$F(X)= e^{m+nX} / 1 + e^{m+nX}$, where X is the data point and f(X) returns the probability of the data value belonging to a particular class [55].

2. Decision Tree Classifier: It is another supervised learning method used to predict qualitative outcomes. It predicts whether the particular value belongs to a particular class using the tree approach. It considers single attribute at the root, other attributes as internal nodes and decision as leaf node. It uses the concept of entropy, information gain and Gini index for selecting the attribute for the root node [56][57].

For every feature, the entropy is calculated by the sum of the positive instances (pyes)
predicted and the negative instances predicted(pno)[56][57].

3. SGDC Classifier: It uses stochastic gradient descent for building linear models. It requires hyperparameter tuning for its functioning. Gradient Descent is an optimization algorithm in which weights are re-updated incrementally and it forms a curved or bell-shaped curve. The difference between stochastic gradient descent and simple gradient descent is in the course of weights updation. Weights are not accumulated in SGDC, but they are updated after training each sample [58][59].

The weight updation is carried by using the concept:
New weight = old weight + change in weight (where change in weight = target weight - output weight).[59][58]

4.Random Forest Classifier: This algorithm works on the approach of building multiple trees and a group of trees is known as forest. The trees are generated using a rule-based approach similar to the approach of decision trees. Trees are split into nodes using the best-split method. It stores the predicted outcome from the testing features and calculates the votes for each prediction. It uses the highest voted predictions for its consideration. The importance of each feature is calculated using
Importance = Sum of splits of the node ‘k’/ Sum of the total number of nodes and then this value is normalized [60][61].

5.K-neighbours Classifier: This algorithm is a supervised learning algorithm that works for classification problems using the concept of neighboring. The values are classified into groups and values in each group are called neighbors of each other. The determination of the number of neighbors is done using the distance as parameter and distance can be Euclidean distance, Chebyshev, cosine, Jaccard, etc. [62][63]
These distances are sorted in ascending order and then sorting which gives minimum loss is taken into consideration for selection of the number of neighbors. The concept for the same is 1/K (Sum of all the predictions of particular instance) [62][63].

6.Perceptron: Perceptron is the Neural Networks approach, that accepts input, performs calculations on it and give the results. It is also the linear binary classifier. The input can be a single input in the vector form, or it comprises of the input layer. It performs some transformation on the values, and it has the activation function which works when the calculated value crosses the limit of the threshold. For each time weight updation, all the weights corresponding to their inputs are multiplied and then added [64][65].

7. SVC: Support Vector Machine is another classification algorithm that is used to classify values into categories using the concept of hyperplanes. Data values are divided into different hyperplanes and the best hyperplane which fits the values is taken into consideration. Decision boundary is built within the hyperplanes to classify the values [66].

8. Adaboost: It is the supervised learning algorithm that works on the collection of weak

classifiers. In this, weak learners made to form a group to give a strong decision. Weak learners may have slight confidence above 50% for the Error at particular level = (Correct value-Predicted value)/Predicted value which defines the misclassification rate. And this value is modified to use the weights of inputs and changed to Error at particular level = Sum (weights*prediction error)/sum(weights) [67][68]

9. Gaussian Naive Bayes: It is the classification algorithm that considers that data is generated from the gaussian. It predicts the probability of the data value belonging to a particular class. The prior probability of the class is multiplied by the likelihood of the event, which is called posterior probability, and then the result will be divided by the evidence. The representation of the joint probability will be: Prob (Yz,K(i,0)),..... K(i,j)) = Prob(K(i,0)),.....K(i,j)|Yz) [69]

10. Artificial Neural Network: The most important as well as the unique portion of this research is utilizing one of the highly efficient classifiers which helps to make an algorithm works just as fast as possible like a neuron receptor works making this model acting just like a working of a human brain. This Model learns the pattern just like the way our brain neurons work making a highly complex web trap like paths which transfers signals connecting the whole network making possible for this extensive multiple folded layered optics to work and imitate with use of epochs in an algorithm helping a model to run several times and learning as well as predicting the way it works from its behaviour as well as other features also not just simply depending upon the attributes like other algorithms. ANN backgrounds the field for Artificial Intelligence helping various advances in the field of robotic technology, neuroscience as well as statistics and many other fields [70].

Input Cross Multilevel neurons give signals to receptors multiplying on a larger scale into the hidden level of model like a black box and then gives an output after receiving the similar types of data from those same input neurons and getting transformed by the black box imitating like a Human brain reactive series of work using those nodes manipulating the biased weights in that function which contains the input to counter balance the error ratio[70].

value belonging to a particular class [67][68]. The concept for the error and prediction can be defined as:

Moving further for the research questions to be solved with the use of various methods the most important part is to visualize a problem on a large scale and on every aspect because sometimes the minimal problems are the ones which cannot be modified because there is no such awareness that the datasets could be moulded and utilized in any other way.

Just for example a comparison between a pie chart with colours, a pie chart with shapes and numbers and a pie chart with different colours and shapes and percentages if gets compared on all these scales on all the 3 different types of pie plots. The results will be very interesting because all of them would raise different opinions to be studied by different type of mindsets as well as different organizations will be in need to study the graphs as per their demand and utility.

Now the research questions 4th and 5th will be solved with the use of graphs and plots whereas the 3rd will be solved with the computation as well as complexity scores on the basis of precision, recall, specificity, sensitivity and other scales. The whole concept for the proposed solution is discussed further.

Table 3.1 showing dataset description briefly to better understand the data and it's attributes with feature type [50][49] Provided by Kaggle under the description of dataset in reference [50].

S.No	Name of attribute	Data Type	Data Size
1.	Tweet_id	Ordinal attribute denoting the id of the tweet	18-digit number
2.	airline_sentiment	Nominal attribute denoting the sentiment of the customer	String values 'Positive', 'Negative', 'Neutral'
3.	airline_sentiment_confidence	Ordinal value depicting the confidence of sentiment	Numeric between 0 and 1
4.	negative_reason	Ordinal attribute denoting the reason for negative sentiment	20 % Customer Service Issue, 11% late flight, 8% Can't tell, 6% canceled flight and 55% other values
5.	negativereason_confidence	Ordinal attribute denoting the confidence in the reason of negative sentiment	A value between 0 and 1
6.	airline	Nominal attribute denoting the name of the airline	Virgin America, Delta, Southwest American, US Airways, United
7.	airline_sentiment_gold	Nominal attribute depicting the sentiment	Positive, Negative, Neutral, Nan
8.	name	Ordinal attribute depicting the name	7701 unique values
9.	negativereason_gold	Nominal attribute depicting the sentiment	Positive, Negative, Neutral, Nan
10.	retweet_count	Ordinal attribute denoting the number of retweets	Contains 0 Numeric Value
11.	text	Ordinal attribute depicting the text of the tweet	String with 14427 unique values
12.	tweet_coord	Ordinal attribute denoting the coordinates of the tweet	Two numeric values
13.	tweet_created	Ordinal attribute depicting the date and time at which tweet is created	Dates ranging between 17 Feb,15 to 25 Feb,15
14.	tweet_location	Ordinal attribute depicting the location of the tweet	3082 Unique string values
15.	user_timezone	Ordinal attribute depicting the time zone of the user	86 Unique String values

Loading Libraries

IMPORTING THE LIBRARIES

```
1. import seaborn as sns
2. import io
3. from sklearn.ensemble import
   RandomForestClassifier, AdaBoostClassifier,
   GradientBoostingClassifier
4. from wordcloud import WordCloud,
   STOPWORDS , ImageColorGenerator
5. import datetime as dt
6. import string
7. from collections import OrderedDict
8. import unicodedata
9. from sklearn.metrics import accuracy_score
10. from sklearn.feature_extraction.text import
    CountVectorizer
11. from sklearn.feature_extraction.text import
    TfidfTransformer
12. import matplotlib.pyplot as plt
13. import seaborn as sns
14. %matplotlib inline
15. from sklearn.linear_model import
    SGDClassifier
16. plt.style.use('ggplot')
17. import spacy
```

LOADING THE PACKAGES

```
1. from sklearn.model_selection import
   train_test_split
2. import matplotlib.pyplot as plt
3. from sklearn.model_selection import KFold
4. from itertools import tee
5. from sklearn.metrics import confusion_matrix,
   accuracy_score
6. from sklearn.neighbors import
   KNeighborsClassifier
7. from sklearn.neighbors import
   KNeighborsClassifier
8. from sklearn.ensemble import
   RandomForestClassifier
9. from sklearn.linear_model import Perceptron
10. from sklearn.tree import DecisionTreeClassifier
11. from sklearn.svm import SVC, LinearSVC,
    NuSVC
12. from sklearn.linear_model import
    LogisticRegression
13. import numpy as np
```

LOADING KERAS PACKAGES FOR ANN

#Using Tensor Flow Backend

```
1. from keras.models import Sequential,
   load_model
2. from keras.layers import Dense, LSTM,
   Embedding, Dropout
3. from keras.preprocessing.text import Tokenizer
4. from keras.preprocessing.sequence import
   pad_sequences
5. from bs4 import BeautifulSoup
6. from gensim import corpora, models, similarities
7. from sklearn.model_selection import KFold
8. from itertools import tee
9. from sklearn.metrics import confusion_matrix,
   accuracy_score
10. import re
11. import pandas as pd
12. import nltk
13. import os
```

LOADING TIME COMPUTATION PACKAGES

```
1. import time
2. from nltk.tokenize.toktok import
   ToktokTokenizer #importing tiktok
3. tokenizers = ToktokTokenizer() #initializing
   tokenizer
4. stopwords_list =
   nltk.corpus.stopwords.words('english')
   #importing stopwords
```

Data Loading

LOADING THE DATASET INPUT

```
#Importing the Dataset

1. twitter_data = pd.read_csv('Tweets.csv')
   #reading file containing tweets
2. twitter_data_copy = twitter_data #making copy
   for usage
3. twitter_data_visualization =
   pd.read_csv('Tweets.csv') #making copy for
   visualization
4. twitter_data_copy =
   twitter_data_copy.text.dropna() #removing null
   values
5. twitter_data_copy =
   twitter_data_copy.reset_index(drop=True)
   #resetting index
6. twitter_data.head()
```

OUTPUT FOR LOADING DATASET

```
# data.dtypes
1. tweet_id                    =  int64
2. airline_sentiment           =  object
3. airline_sentiment_confidence =  float64
4. negativereason              =  object
5. negativereason_confidence   =  float64
6. airline                     =  object
7. airline_sentiment_gold.     =  object
8. name                        =  object
9. negativereason_gold         =  object
10. retweet_count              =  int64
11. text                       =  object
```

```
12. tweet_coord          =  object
13. tweet_created        =  object
14. tweet_location       =  object
15. user_timezone        =  object

# { dtype: object }
```

```
# data.head()
```

tweet_id	570301031407624196
airline_sentiment	negative
airline_sentiment_confidence	1.0000
negativereason	Bad Flight
negativereason_confidence	0.7033
airline	Virgin America
airline_sentiment_gold	NaN
name	jnardino
negativereason_gold	NaN
retweet_count	**0**
text	@VirginAmerica and it's a really big bad thing about it
tweet_coord	NaN
tweet_created	2015-02-24 11:14:45 -0800
tweet_location	NaN
user_timezone	Pacific Time (US & Canada)

Data Visualization

```
1. df_text =
   pd.DataFrame(twitter_data_visualization['text']) #extracting column of
   text
```

Word Cloud

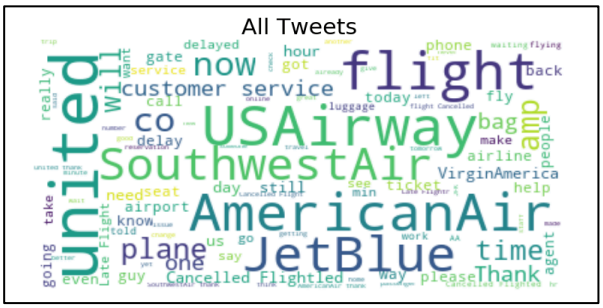

Fig 4.1 Word cloud for all tweets

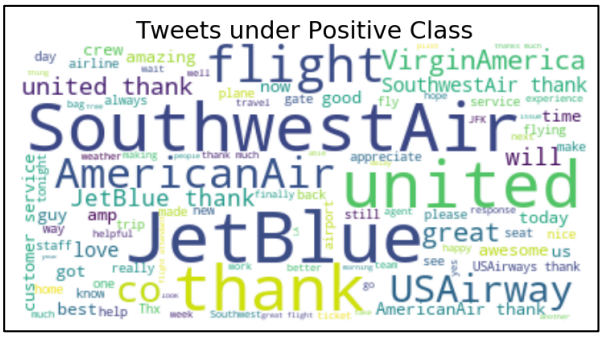

Fig 4.2 Word cloud for Positive tweets

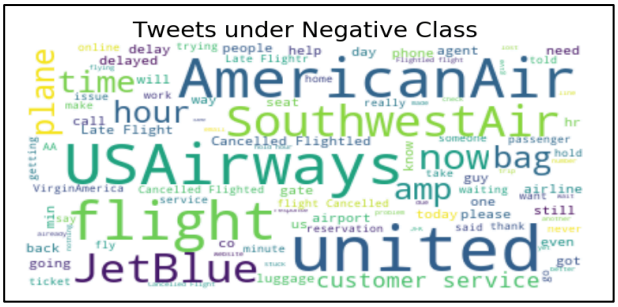

Fig 4.3 Word Cloud for negative tweets

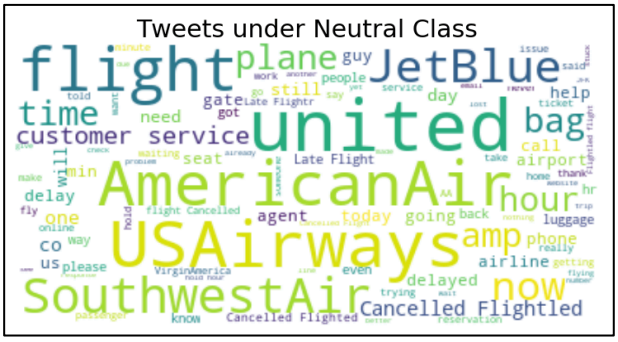

Fig 4.4 Word cloud for neutral tweets

Airlines in the dataset

```
In [79]:  1  tweet_created_for_visualization =
          2  for dd in twitter_data_visualizati
          3      tweet_created_for_visualizatio

In [80]:  1  twitter_data_visualization.airline

Out[80]: Virgin America      504
         Delta              2222
         Southwest          2420
         American           2759
         US Airways         2913
         United             3822
         Name: airline, dtype: int64

In [81]:  1  pd.Series(twitter_data_visualizati
          2                fontsize=10,label=True
```

Fig 5.1

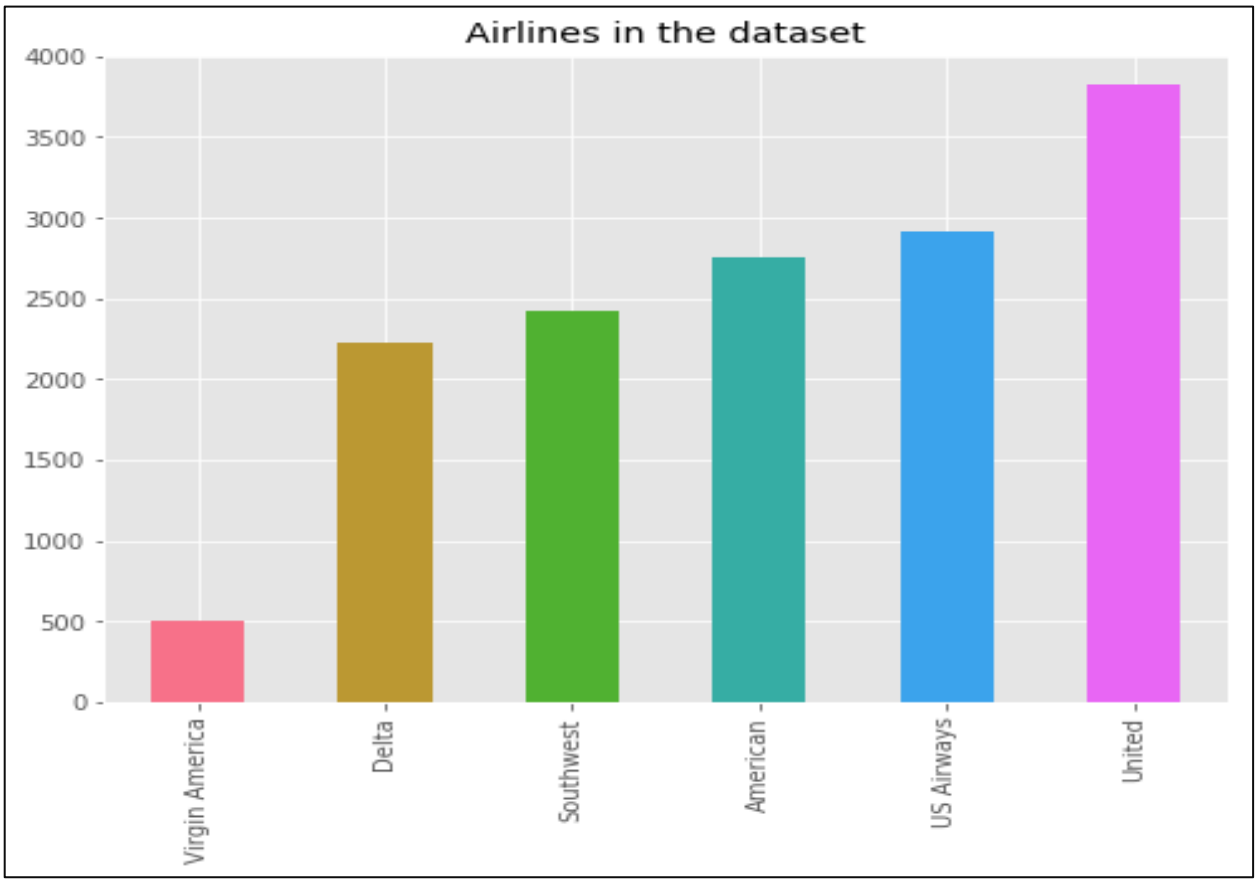

Fig 5.2

Sentiments Count

```
pd.Series(twitter_data_visualization["airline_sentiment"]).value_counts(ascending=True) #counting sentiments
```

```
positive     2363
neutral      3099
negative     9178
Name: airline_sentiment, dtype: int64
```

```
lors_filling=sns.color_palette("bright", 10)
tal_count=pd.Series(twitter_data_visualization["airline_sentiment"]).value_counts(ascending=True).plot(
  kind = "bar",color=colors_filling,figsize=(8,6),rot=1, title = "Total_Airline_Sentiment") #visualizing sentiments
```

Fig 5.3

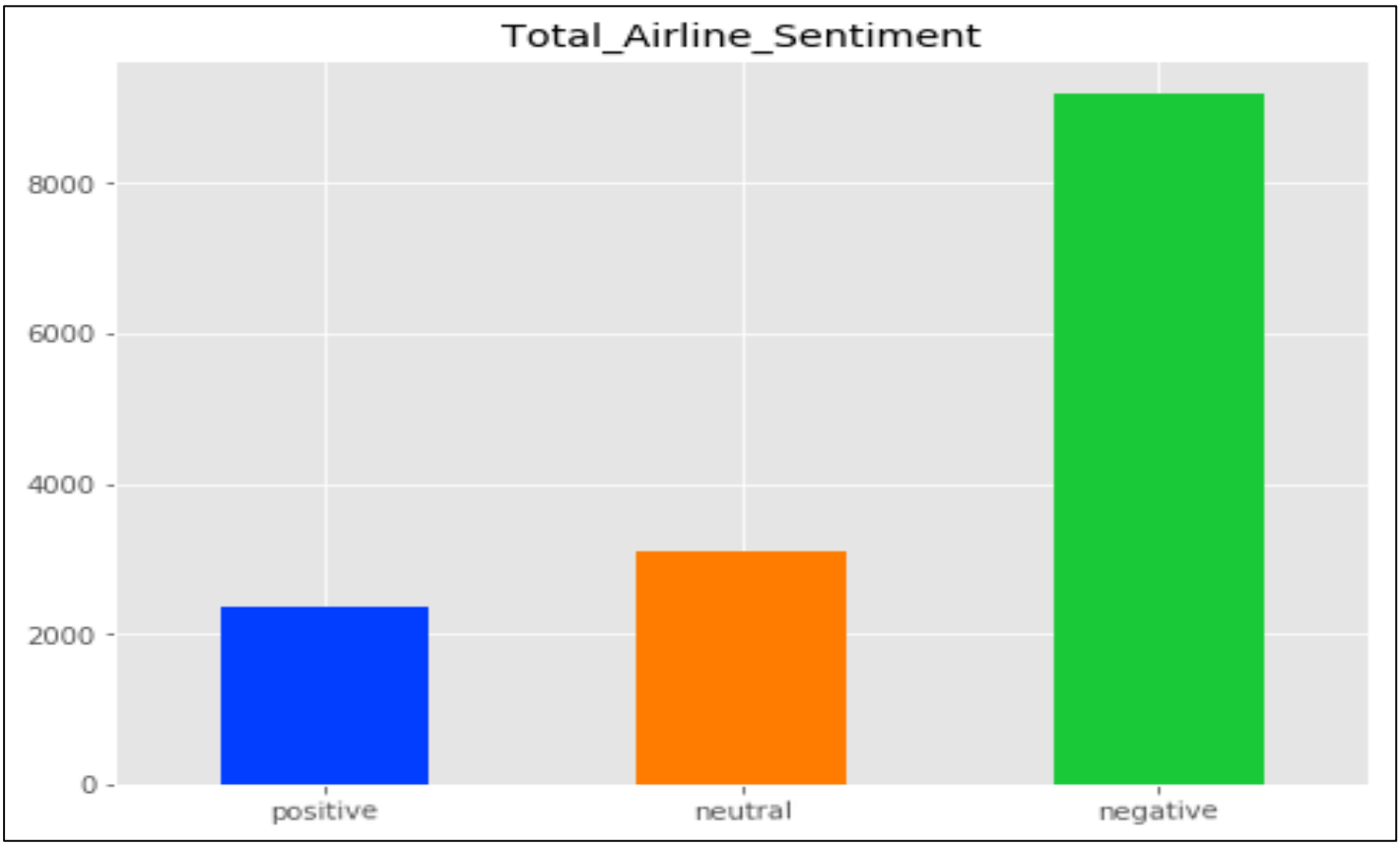

Fig 5.4

The Distribution of sentiments with all the attributes in ratio for each other to be categorized on the basis of sum of the whole 3 types of sentiments on a percentile scale:

1. **#Via Variable: (Airline_sentiment)**
2. **Neutral: 21.17%**
3. **Positive: 16.14%**
4. **Negative: 62.69%**

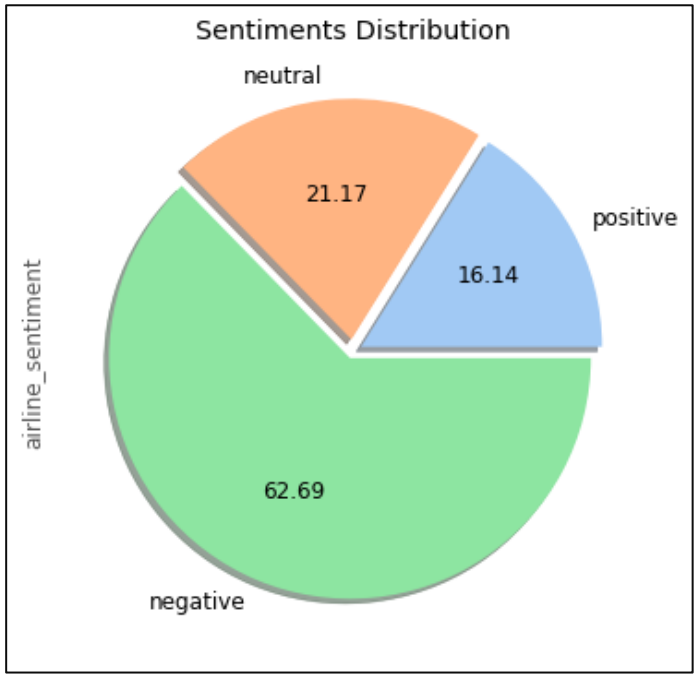

Fig 5.5

Understanding of the sentiments category distributon by reasons :

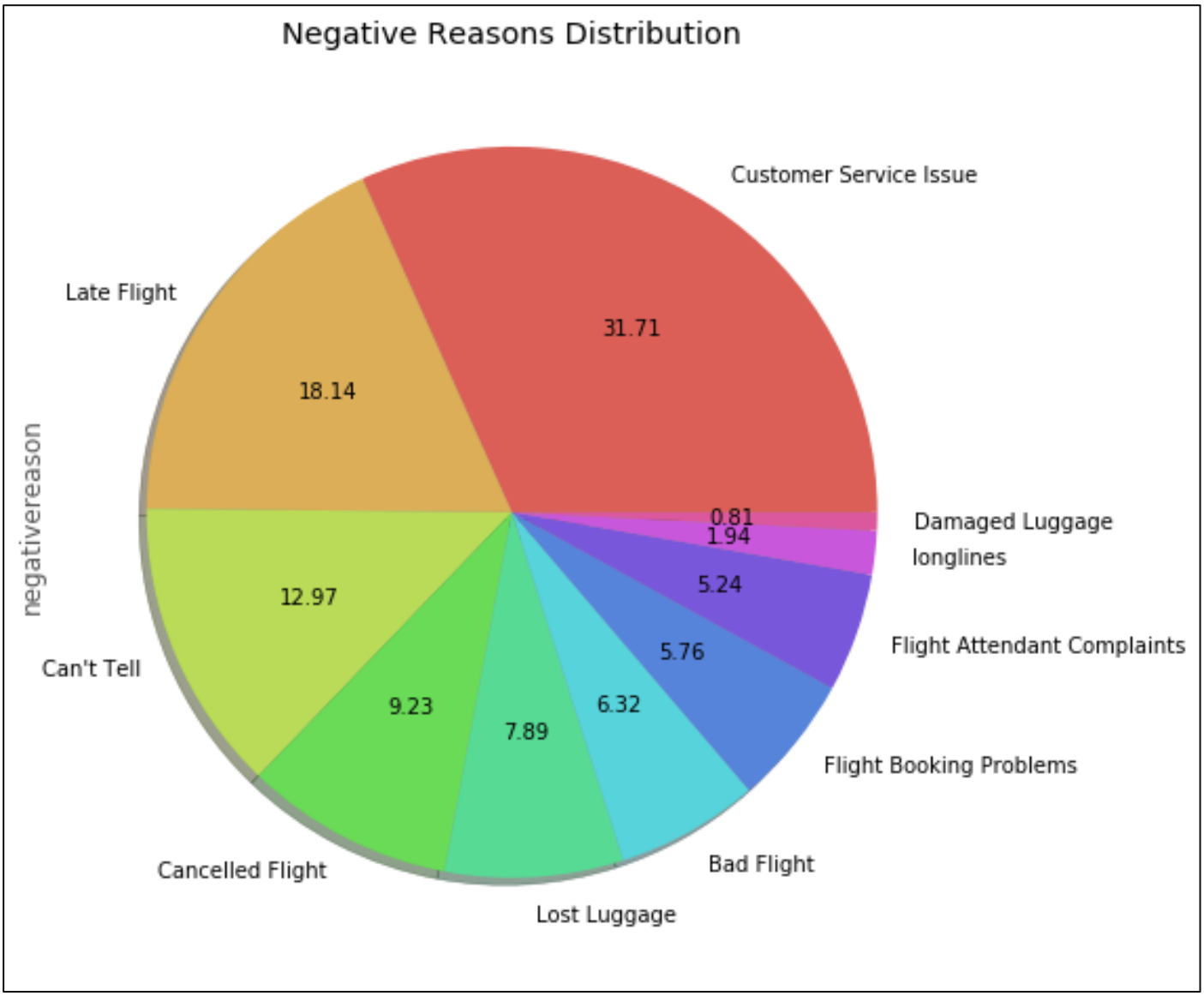

Fig 5.6

Different Timezones of User

```
1. pd.Series (twitter_data_visualization #
   ["user_timezone"]).value_counts().head(8).plot(kind"bar",color=sns.color_palette("husl")
2. ,figsize=(8,6),title = "Timezones of User").

3. #visualizing timezones of user.
```

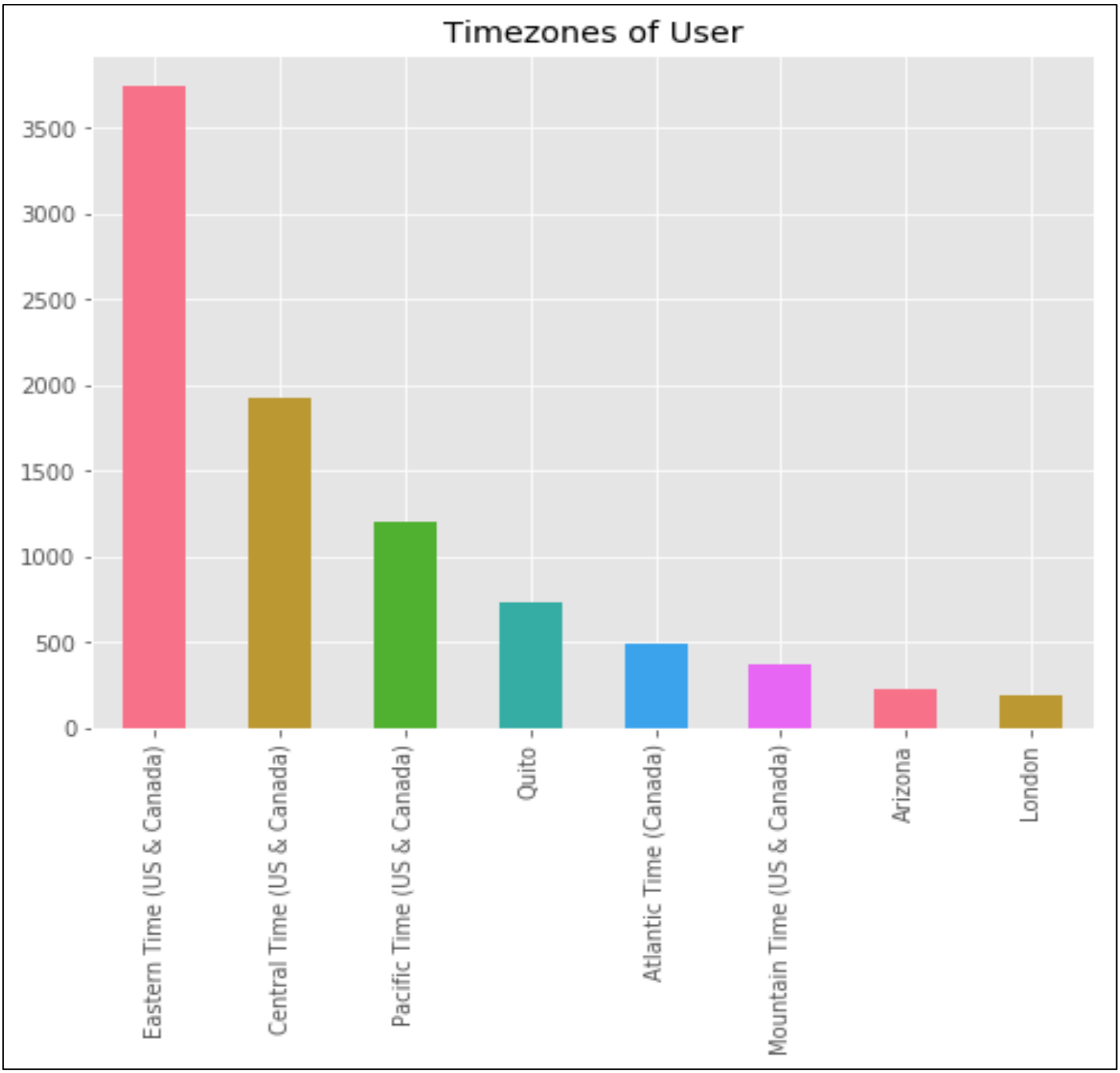

Fig 5.7

airline airline_sentiment	American	Delta	Southwest	US Airways	United	Virgin America
negative	1960	955	1186	2263	2633	181
neutral	463	723	664	381	697	171
positive	336	544	570	269	492	152

Fig 5.8

SENTIMENTS ACCORDING TO AIRLINES

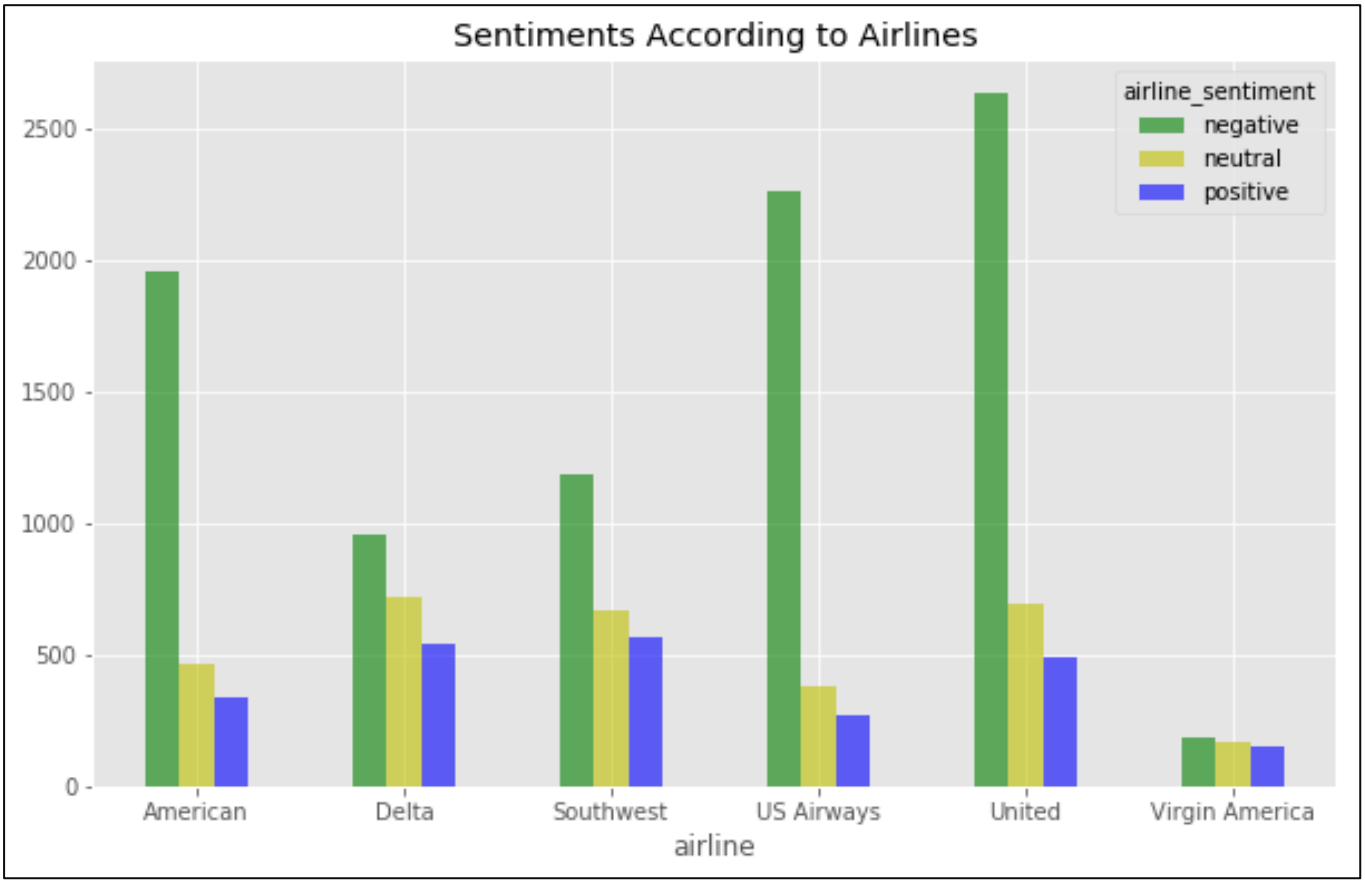

Fig 5.9

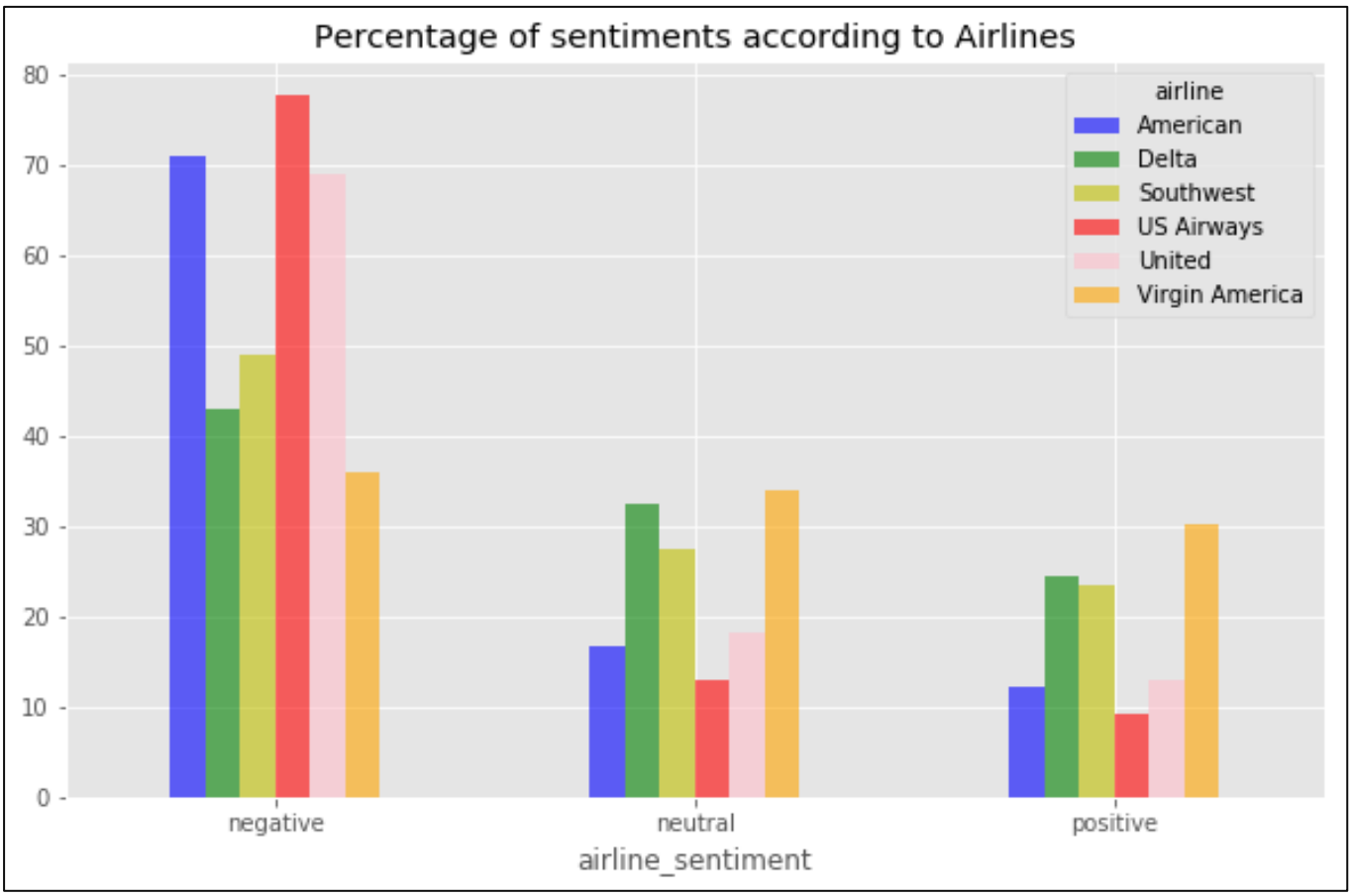

Fig 5.10

SENTIMENTS DISTRIBUTED BY DATE

Represented on the basis of events to be scored at a level showing a number of tweets within a distinguishable manner to their sentiment character as negative, positive or neutral.

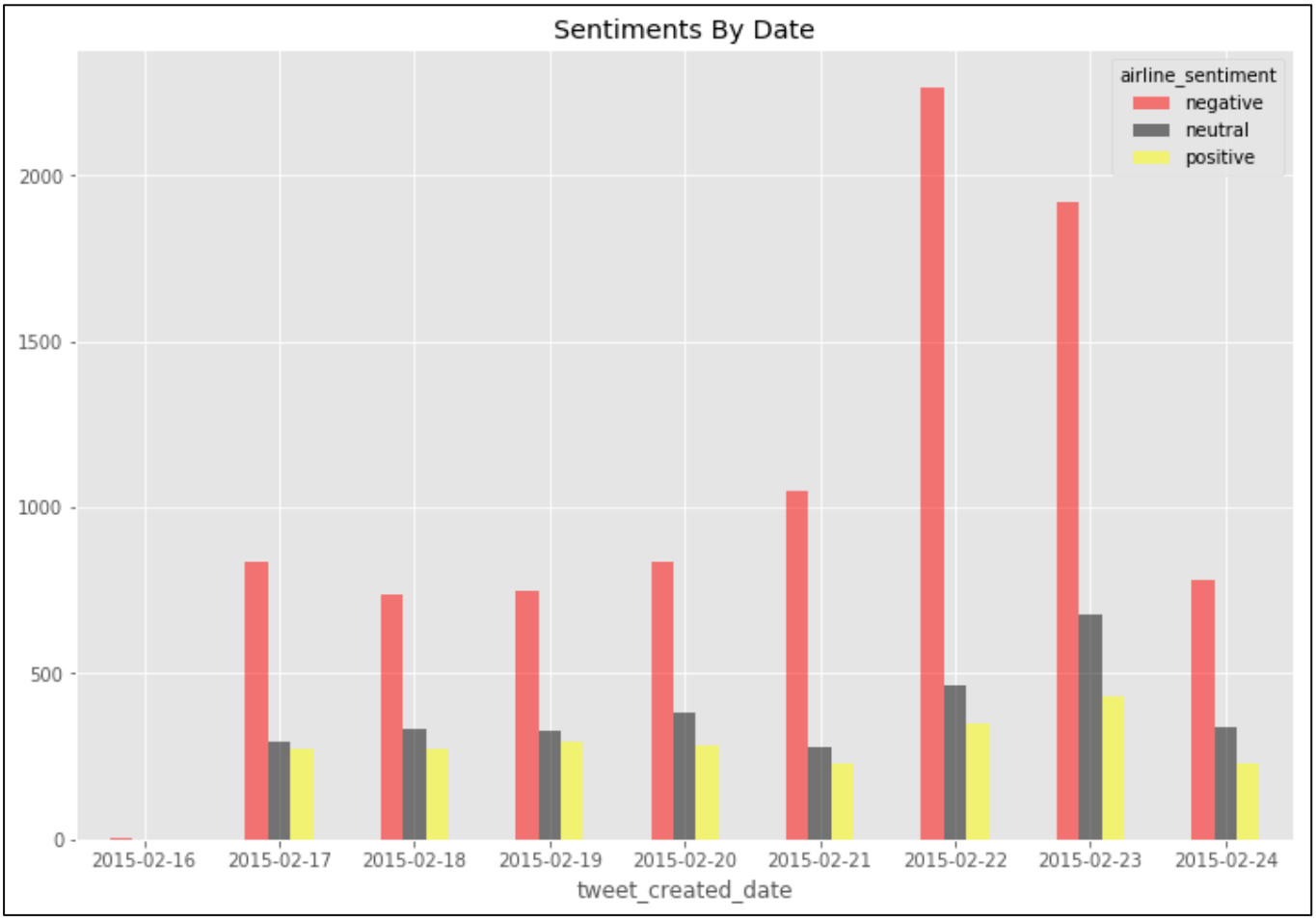

Fig 6.1

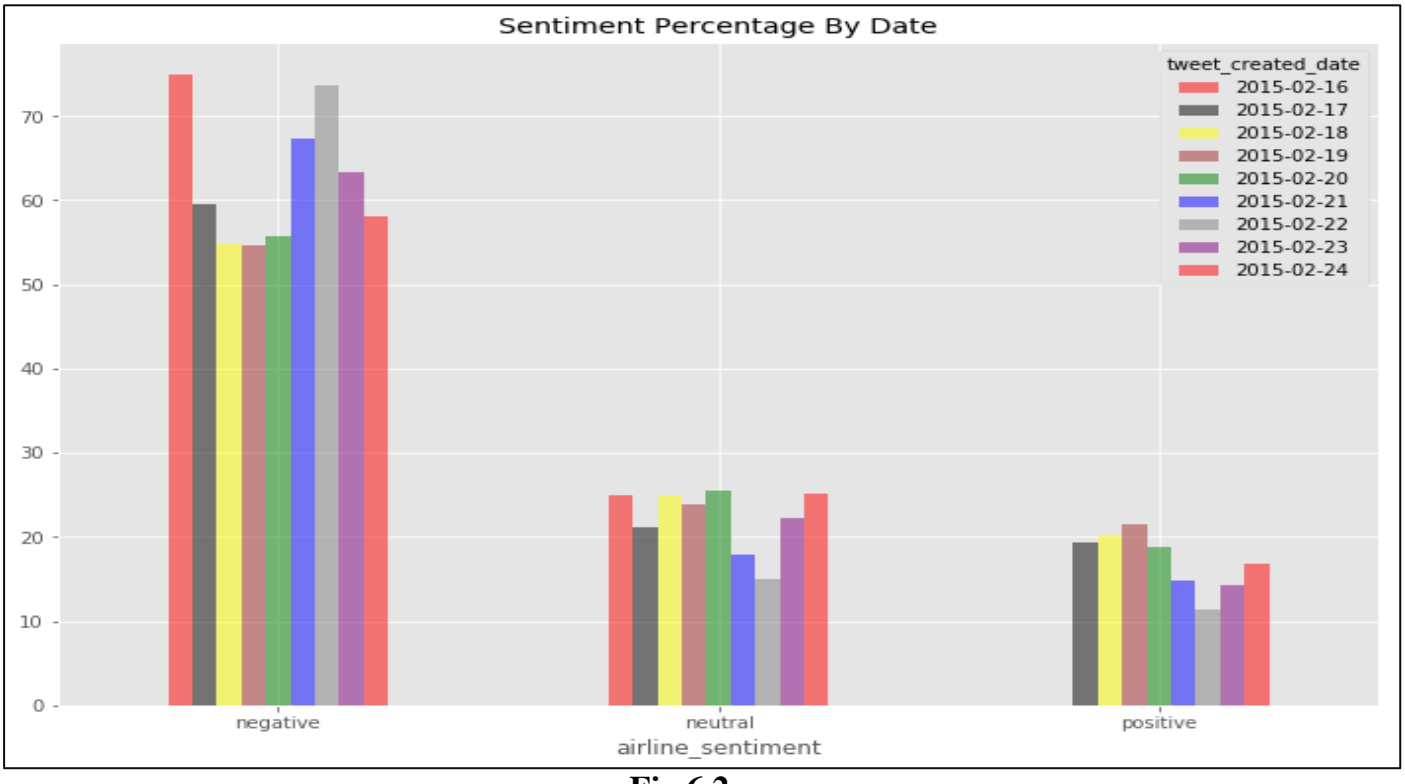

Fig 6.2

Fig 6.3

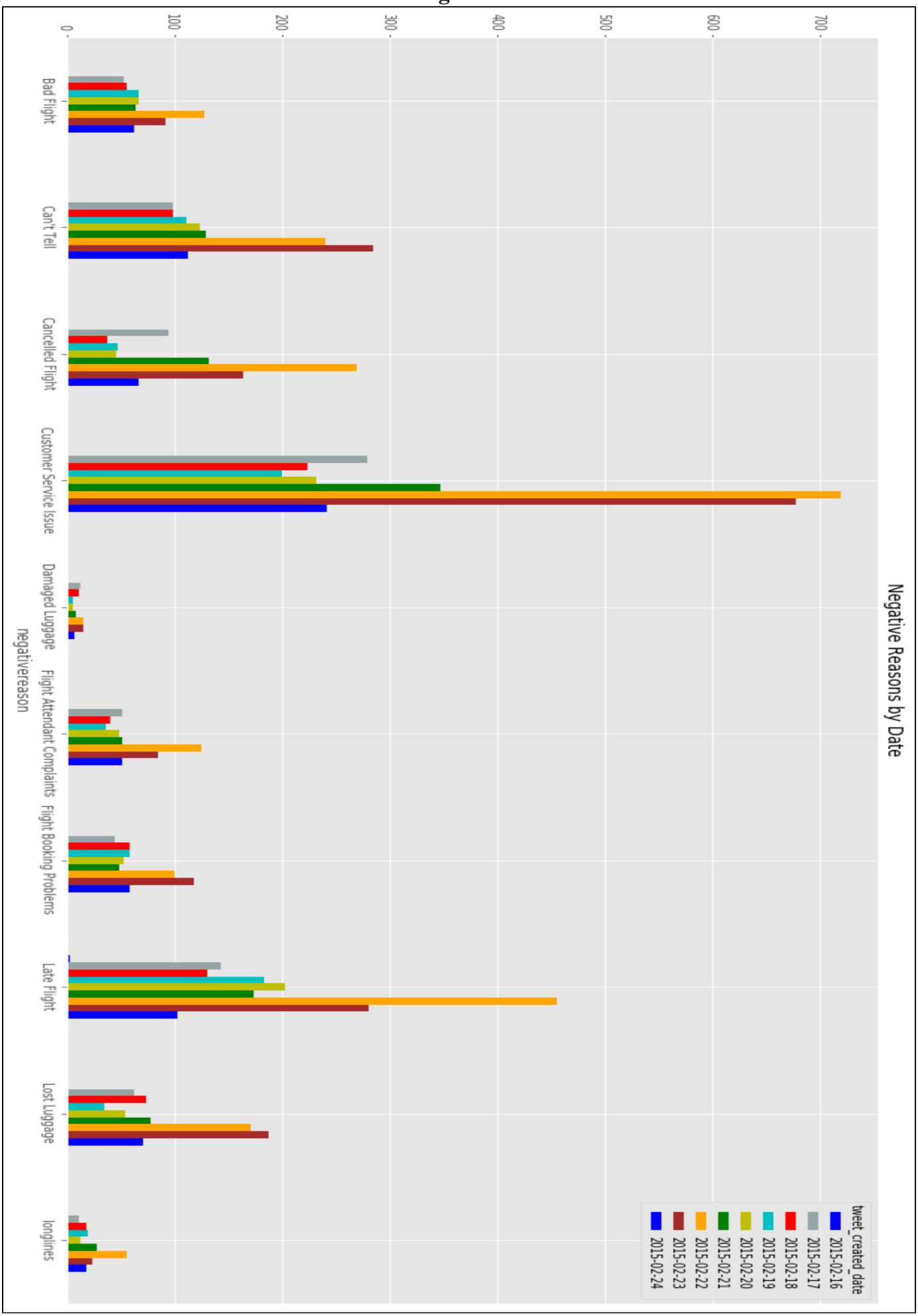
Negative Reasons by Date
tweet_created_date
2015-02-16
2015-02-17
2015-02-18
2015-02-19
2015-02-20
2015-02-21
2015-02-22
2015-02-23
2015-02-24
0
100
200
300
400
500
600
700
Bad Flight
Can't Tell
Cancelled Flight
Customer Service Issue
Damaged Luggage
Flight Attendant Complaints
Flight Booking Problems
Late Flight
Lost Luggage
longlines
negativereason

Fig 6.3 Represents Negative reasons which were gathered from the analysis by categorizing them on the basis of date

Negative Reasons By Time

```
hour_value_counts = twitter_data_visualization.groupby(['tweet_created_hour']).negativereason.value_counts() #numbe
hour_value_counts.unstack()#unstacking values for output
```

negativereason	Bad Flight	Can't Tell	Cancelled Flight	Customer Service Issue	Damaged Luggage	Flight Attendant Complaints	Flight Booking Problems	Late Flight	Lost Luggage	longlines
tweet_created_hour										
0	10.0	14.0	11.0	27.0	NaN	4.0	2.0	13.0	15.0	1.0
1	3.0	15.0	3.0	29.0	3.0	3.0	3.0	7.0	13.0	1.0
2	3.0	11.0	7.0	46.0	NaN	11.0	3.0	11.0	14.0	3.0
3	10.0	17.0	17.0	51.0	1.0	3.0	5.0	21.0	17.0	4.0
4	15.0	33.0	20.0	74.0	2.0	18.0	13.0	38.0	21.0	2.0
5	20.0	32.0	33.0	108.0	2.0	16.0	18.0	61.0	20.0	3.0

Fig 6.4

Plotting Graph for Negative Reasons By Time

1. **hour_value_counts_graph = hour_value_counts.unstack().plot(kind='line',marker='o',linestyle='dashed',markersize=10,color=['b','#95a5a6','r','c','y','g','m','w','#e74c3c','#34495e'],figsize=(12, 12),rot=0,title="Negative Reasons by Time")#visualizing number of negative reasons by time**
2. **hour_value_counts_graph.set_xlabel("Time")**
3. **hour_value_counts_graph.set_ylabel("Negative Reasons")**

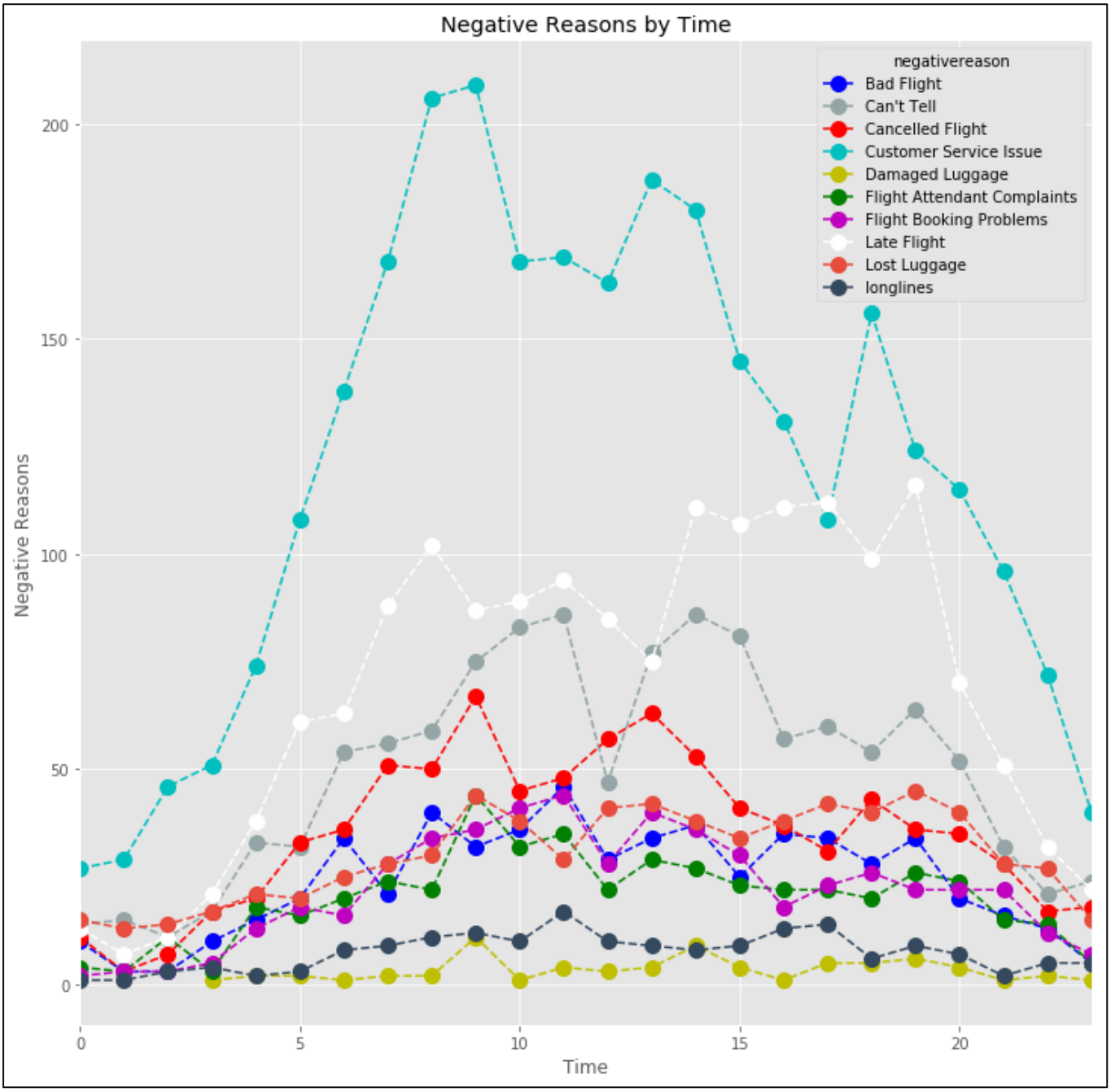

Fig 6.5

Data Cleaning

```
text_correction_list = {"ain't": "am not", "what's": "what is","what've": "what have","when's": "when is","when've"
  "would've": "would have","aren't": "are not","can't": "cannot","can't've": "cannot have","'cause": "because", "co
  "there'd've": "there would have","there's": "there is","they'd": "they would","they'd've": "they would have","the
  "they'll've": "they will have","they're": "they are","they've": "they have","to've": "to have","wasn't": "was not
  "shouldn't": "should not","shouldn't've": "should not have","so've": "so have", "so's": "so is","that'd": "that w
  "we'll": "we will", "we'll've": "we will have","we're": "we are","we've": "we have","weren't": "were not","what'l
  "what'll've": "what will have", "what're": "what are","wouldn't": "would not","wouldn't've": "would not have","y'
  "y'all'd": "you all would","y'all'd've": "you all would have","y'all're": "you all are","y'all've": "you all have
}#list of abbreviations
compile_correct_words = re.compile('(%s)' % '|'.join(text_correction_list.keys()))##compiling abbreviations
```

Fig 6.6

As discussed earlier, the data cleaning plays a crucial role in improving the testing as well as training of a statistical algorithmic performance which in this case is done for various topics such as abbreviations , joint connectors, symbols spaces and so on grammatical issues which were not required under the process of compiling the correction of words on the training portion of the research for example “We’LL” is changed to We will, further examples are like removing the url’s as well as extra spaces columns, hashtags and so on shown below:

```
data_preprocessing(sentiment_text): #function for preprocessing
expression_for_link    = re.compile('((https?):((//)|(\\\\))+([\w\d:#@%/;$()~_?\+-=\\\.&](#!)?)*)', re.DOTALL) #re
processed_link         = re.findall(expression_for_link, sentiment_text) #finding above symbols in text
for link in processed_link: #parsing each found symbol
    sentiment_text = sentiment_text.replace(link[0], ', ') #replacing symbol with space
sentiment_text= remove_punctuation_marks(sentiment_text) #calling function remove_punctuation_marks
sentiment_text= sentiment_text.replace('RT', '') #replacing RT with space
def matching(match): #function for matching
    return text_correction_list[match.group(0)]#returning list for match
sentiment_text = compile_correct_words.sub(matching, sentiment_text.lower())#converting text to lowercase
extra_digits = r'[^a-zA-z0-9\s]'#checking extra digits
sentiment_text = re.sub(extra_digits, '', sentiment_text) #remove extra digits
sentiment_text = unicodedata.normalize('NFKD', sentiment_text).encode('ascii', 'ignore').decode('utf-8', 'ignore')
return sentiment_text
```

Fig 6.7

1. def data_preprocessing(sentiment_text):
2. #function for preprocessing
3. expression_for_link = re.compile('((https?):((//)|(\\\\))+([\w\d:#@%/;$()~_?\+-=\\\.&](#!)?)*)', re.DOTALL)
4. #removing links
5. processed_link = re.findall(expression_for_link, sentiment_text)
6. #finding above symbols in text
7. for link in processed_link:
8. #parsing each found symbol
9. sentiment_text = sentiment_text.replace(link[0], ', ')
10. #replacing symbol with space
11. sentiment_text= remove_punctuation_marks(sentiment_text)
12. #calling function remove_punctuation_marks
13. sentiment_text= sentiment_text.replace('RT', '')
14. #replacing RT with space
15. def matching(match):
16. #function for matching
17. return text_correction_list[match.group(0)]
18. #returning list for match
19. sentiment_text = compile_correct_words.sub(matching, sentiment_text.lower())
20. #converting text to lowercase
21. extra_digits = r'[^a-zA-z0-9\s]'
22. #checking extra digits
23. sentiment_text = re.sub(extra_digits, '', sentiment_text)
24. #remove extra digits
25. sentiment_text = unicodedata.normalize('NFKD', sentiment_text).encode('ascii', 'ignore').decode('utf-8', 'ignore')
26. # normalizing sentiment text
27. return sentiment_text

```
def remove_punctuation_marks(sentiment_text):at_the_rate_hashes = ['@','#']
# initializing array of symbols
for p in  string.punctuation:
#checking in punctuations
if p not in at_the_rate_hashes : sentiment_text = sentiment_text.replace(p,' ')
#replacing punctuations with space
processed_text =  for pt in sentiment_text.split():
#splitting text
pt = pt.strip()
if pt: if pt[0] not in at_the_rate_hashes:
# checking and removing symbols
# processed_text.append(#return ' '.join(processed_text)
```

Data Preprocessing

```
#def converting_text_to_vector():
c_v = CountVectorizer()

#positive_and_negative_tweets = twitter_data[twitter_data.airline_sentiment != 'neutral']
X_vector = c_v.fit_transform(positive_and_negative_tweets['cleaned_tweets'])

#tfidf_transformer = TfidfTransformer(norm='l2')

#Assemble the new variables in such a way that the return function x and y processed the whole counter vector status onto the further classifiers for initialising on a digital counter scale.
(Using variable function)X = tfidf_transformer And fit_transform(X_vector)

y = positive_and_negative_tweets['airline_sentiment']
return X,y
```

Storing the processed tweets converted from text to vector :

```
twitter_data['cleaned_tweets'] = storing_processed_tweets
positive_and_negative_tweets = twitter_data[twitter_data.airline_sentiment != 'neutral']
X,y=converting_text_to_vector()
```

Applying Models with only positive and negative sentiments.

1. K Nearest neighbors

```
1. def knn_working(X,y):
2. #Defining the function
3. kfolds = KFold using the model and operate to
   select slpit number (n_splits=5)
4. train_test_data =# used for training and then k
   folds is applied  kfolds.split(X)
5. accuracy_list=[] #accuracy function starts here
   then the computation time is calculated with use of
   next time function series as shown in results by the
   table of evaluation.
6. starting_timing = time.time()
7. print('Running for KNN only for positive and
   negative sentiments for 5 folds using K-fold')
8. knn_= KNeighborsClassifier(n_neighbors=11)
9. #fitting and prediction stages
10. calculate_knn=knn_.fit(x_d, y_d) . y_predictions =
    calculate_knn.predict(X[td[1]])
11. #then the predicted accuracy with use of the
    confusion matrix table function shows the 2 x 2
    matrix as shown further in results.
12. accuracy_list_positive_and_negative
    ['knn']=accuracy_list knn_working(X,y)
13. #results are displayed
```

2. Decision Tree

```
#Defining the model
1. def decision_tree_working(X,y): #kfold splits
2. kfolds = KFold(n_splits=5)
3. train_test_data = kfolds.split(X)
4. accuracy_list=[] #accuracy listing on each run
iteration
5. starting_timing = time.time() #time ccompute
function
6  print('Running for Decision Tree only for positive
   and negative sentiments for 5 folds using K-fold')
7  sum_of_accuracy = 0 #PRINT the results of mean
   accuracy
8  sum_f1_score = 0
9  train_test_data, train_test_data_cp =
   tee(train_test_data) #data train and test function
10 for td in train_test_data:
11 y = np.array(y) #running the array
12 x_d = X[td[0],:]
13 y_d = ((y))[td[0]]
14 dt_= DecisionTreeClassifier()
15 calculate_dt_=dt_.fit(x_d, y_d) #confusion matrix
   function is applied on the model after this and
   predicted results are shwn in the further result
   section
16 y_predictions = calculate_dt_.predict(X[td[1]])
```

3. Stochastic Gradient Descent Algorithm

```
1. # define the function #def sgdc_working(X,y):
2. kfolds = KFold(n_splits=5) #kfold splits
3. train_test_data = kfolds.split(X) #training the
   model
4. accuracy_list=[]
5. starting_timing = time.time()
6. print('Running for SGDC Classifier only for
   positive and negative sentiments for 5 folds using
   K-fold')
7. sum_of_accuracy = 0
8. sum_f1_score = 0
9. train_test_data, train_test_data_cp =
   tee(train_test_data) #blackboc model run under the
   projection which provides the output to be
   produced with use of  confusion matrix shown in
   solutions and results passed by evaluation phase
10. for td in train_test_data:
11. y = np.array(y) #array is running to frame sgdc
    function
12. x_d = X[td[0],:]
13. y_d = ((y))[td[0]]
14. sgdc_= SGDClassifier(loss = 'log')
15. calculate_sgdc_= sgdc_.fit(x_d, y_d)
16. y_predictions = calculate_sgdc_.predict(X[td[1]])
17. accuracy_list_positive_and_negative['sgdc']=accur
    acy_list
18. accuracy_mean_list_positive_and_negative.append
    ((sum_of_accuracy)/5)
19. sgdc_working(X,y) #print accuracy results
```

4. Random Forest Classifier

```
1. # function used def random_forest_working(X,y):
2. kfolds = KFold(n_splits=5) #method by k folds
   applied
3. train_test_data = kfolds.split(X)
4. accuracy_list=[] #accuracy time compute for
   function to print output
5. starting_timing = time.time()
6. print('running for Random Forest Classifier only
   for positive and negative sentiment for 5 folds
   using K-fold')
7. sum_of_accuracy = 0
8. sum_f1_score = 0 # training the model
9. train_test_data, train_test_data_cp =
   tee(train_test_data)
10. for td in train_test_data:# testing and array function
    insights running the function RFC.
11. y = np.array(y)
12. x_d = X[td[0],:]
13. y_d = ((y))[td[0]]
14. rfc_= RandomForestClassifier()
15. calculate_rfc_= rfc_.fit(x_d, y_d)
16. y_predictions = calculate_rfc_.predict(X[td[1]])
17. accuracy_list_positive_and_negative['rfc']=accurac
    y_list #print output Accuracy.in next step
18. accuracy_mean_list_positive_and_negative.append
    ((sum_of_accuracy)/5)
19. #run : random_forest_working(X,y)
```

5. Perceptron Algorithm

```
1. #def perceptron_working(X,y):
2. # split ( k fold method )kfolds = KFold(n_splits=5)
3. train_test_data = kfolds.split(X)
4. accuracy_list=[]
5. starting_timing = time.time()
6. print('Running for Perceptron only for positive and
   negative sentiments for 5 folds using K-fold')
7. sum_of_accuracy = 0
8. sum_f1_score = 0
9. # running function train_test_data,
   train_test_data_cp = tee(train_test_data)
10. for td in train_test_data:
11. y = np.array(y)
12. x_d = X[td[0],:]
13. y_d = ((y))[td[0]]
14. perceptron_= Perceptron(tol=1e-3,
    random_state=0)
15. calculate_perceptron_= perceptron_.fit(x_d, y_d)
16. y_predictions =
    calculate_perceptron_.predict(X[td[1]])
17. accuracy_list_positive_and_negative['perceptron']=
    accuracy_list
18. accuracy_mean_list_positive_and_negative.append
    ((sum_of_accuracy)/5)
19. perceptron_working(X,y)
```

6. Support Vector Classifier Algorithm

```
1. ddef svc_classifier(X,y):
2. ( #runnig the kfolds preset fitting )
3. print('Running for Support Vector Classifier only
   for positive and negative sentiments for 5 folds
   using K-fold')
4. sum_of_accuracy = 0
5. sum_f1_score = 0
6. train_test_data, train_test_data_cp =
   tee(train_test_data)
7. for td in train_test_data:
8. y = np.array(y)
9. x_d = X[td[0],:]
10. y_d = ((y))[td[0]]
11. svc_= SVC(kernel="rbf", C=100., gamma=0.01,
    probability=True, degree=3)
12. calculate_svc_= svc_.fit(x_d, y_d)
13. y_predictions = calculate_svc_.predict(X[td[1]])

14. accuracy_list_positive_and_negative['svc']=accura
    cy_list
15. accuracy_mean_list_positive_and_negative.append
    ((sum_of_accuracy)/5)
16. svc_classifier(X,y)
```

7. Logistic Regression

```
1. def logistic_regression_working(X,y):
2. kfolds = KFold(n_splits=5)
3. train_test_data = kfolds.split(X)
4. accuracy_list=[]
5. starting_timing = time.time()
6. print('Running for Logistic Regression only for
   positive and negative sentiments for 5 folds using
   K-fold')
7. sum_of_accuracy = 0
8. sum_f1_score = 0
9. train_test_data, train_test_data_cp =
   tee(train_test_data)
10. for td in train_test_data:
11. y = np.array(y)
12. x_d = X[td[0],:]
13. y_d = ((y))[td[0]]
14. lgr_=
    LogisticRegression(C=100,solver='liblinear',max_i
    ter=300)
15. calculate_lgr_= lgr_.fit(x_d, y_d)
16. y_predictions = calculate_lgr_.predict(X[td[1]])
17. accuracy_list_positive_and_negative['lgst']=accura
    cy_list
18. accuracy_mean_list_positive_and_negative.append
    ((sum_of_accuracy)/5)
19. logistic_regression_working(X,y)
```

8. Adaboost Classifier

```
1. def ada_boost_working(X,y):
2. kfolds = KFold(n_splits=5)
3. train_test_data = kfolds.split(X)
4. accuracy_list=[]
5. starting_timing = time.time()
6. print('Running for Adaboost only for positive and
   negative sentiments for 5 folds using K-fold')
7. sum_of_accuracy = 0
8. sum_f1_score = 0
9. train_test_data, train_test_data_cp =
   tee(train_test_data)
10. for td in train_test_data:
11. y = np.array(y)
12. x_d = X[td[0],:]
13. y_d = ((y))[td[0]]
14. adb_= AdaBoostClassifier()
15. calculate_adb_= adb_.fit(x_d, y_d)
16. y_predictions = calculate_adb_.predict(X[td[1]])
17. accuracy_list_positive_and_negative['adb']=accura
    cy_list
18. accuracy_mean_list_positive_and_negative.append
    ((sum_of_accuracy)/5)
19. ada_boost_working(X,y)
```

Comparison of Accuracy of different Models during 5 Folds

```
accuracy_df=pd.DataFrame.from_dict(accuracy_list_positive_a
accuracy_df_transpose=accuracy_df.transpose()
accuracy_df_transpose['Mean Accuracy']=accuracy_mean_list_p
accuracy_df_transpose
```

	0	1	2	3	4	Mean Accuracy
knn	0.886531	0.875650	0.834055	0.922877	0.922877	0.888398
decision_tree	0.854916	0.843154	0.804593	0.885182	0.878683	0.853305
sgdc	0.909918	0.876083	0.838821	0.944541	0.938908	0.901654
rfc	0.877003	0.834922	0.802860	0.925043	0.915945	0.871155
perceptron	0.886098	0.868718	0.858319	0.927210	0.923744	0.892817
svc	0.912516	0.895581	0.878250	0.944974	0.941075	0.914479
lgst	0.906886	0.893847	0.873917	0.933709	0.934575	0.908587
adb	0.895193	0.865251	0.832322	0.934142	0.920711	0.889524

Fig 7.1

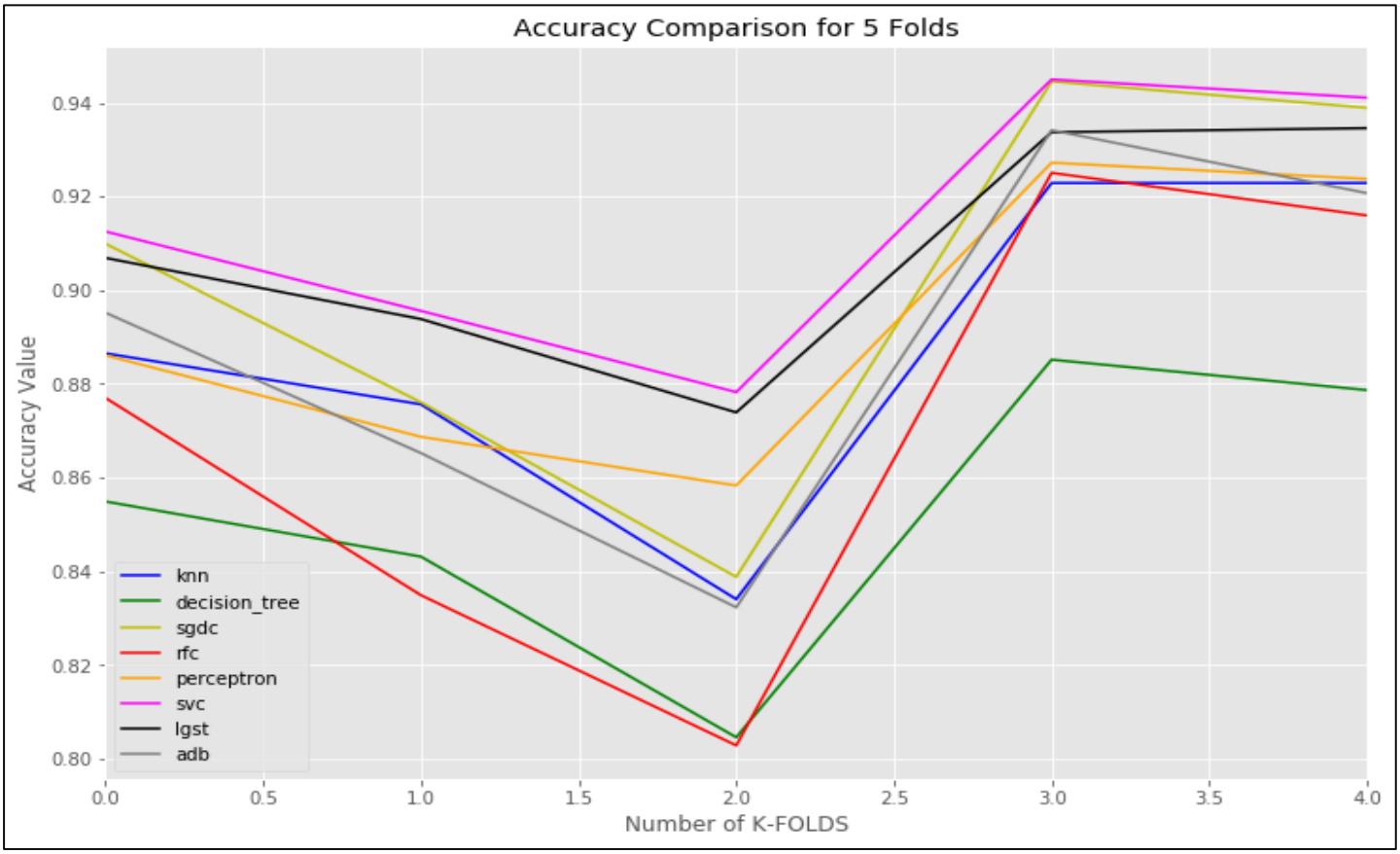

Fig 7.2

Comparison of Accuracy of different Models during 5 Folds for all the classifiers under prediction of positive and negative tweets shown by fig 7.2 and fig 7.1

Applying Models with all three Sentiments

1. Artificial Neural Networks

#transforming text to lowercase

```
1. twitter_data['text'].apply(lambda x: x.lower())
   #transforming text to lowercase
2. twitter_data['text'] =
   twitter_data['text'].apply(lambda x: re.sub('[^a-
   zA-z0-9\s]', '', x))
```

```
3. calculate_token = Tokenizer(num_words=5000,
   split=" ")
4. calculate_token.fit_on_texts(twitter_data['text'].
   values)

5. X_data =
   calculate_token.texts_to_sequences(twitter_data
   ['text'].values)
6. X_data = pad_sequences(X_data)
```

```
7. model_for_ann = Sequential()
8. model_for_ann.add(Embedding(6000, 256,
   input_length=X_data.shape[1]))
9. model_for_ann.add(Dropout(0.3))
10. model_for_ann.add(LSTM(256,
    return_sequences=True, dropout=0.3,
    recurrent_dropout=0.2))
11. model_for_ann.add(LSTM(256, dropout=0.3,
    recurrent_dropout=0.2))
12. model_for_ann.add(Dense(3,
    activation='softmax'))
```

```
13. model_for_ann.compile(loss='categorical_crosse
    ntropy', optimizer='adam',
    metrics=['accuracy'])
14. model_for_ann.summary()
```

```
15. get_predictions =
    pd.get_dummies(twitter_data['airline_sentimen
    t']).values
```

```
1. X_train_data, X_test_data, y_train_data,
   y_test_data = train_test_split(X_data,
   get_predictions, test_size=0.2, random_state=0)
```

```
2. starting_timing = time.time()
```

```
3. model_for_ann.fit(X_train_data, y_train_data,
   #epochs=8, #batch_size=32, #verbose=1)
4. print("completed in = ",time.time() -
   starting_timing)
```

```
5. predictions_for_ann =
   model_for_ann.predict(X_test_data)
```

```
6. positive_count, neutral_count, negative_count =
   0, 0, 0
7. real_positive, real_neutral, real_negative = 0, 0,
   0
8. for i, predictions in
   enumerate(predictions_for_ann):
9. if np.argmax(predictions)==2:
10. positive_count += 1
11. elif np.argmax(predictions)==1:
12. neutral_count += 1
13. else:
14. negative_count += 1

15. if np.argmax(y_test_data[i])==2:
16. real_positive += 1
17. elif np.argmax(y_test_data[i])==1:
18. real_neutral += 1
19. else:
20. real_negative +=1

21. print('Positive predictions are :', positive_count)
22. print('Neutral predictions are: ', neutral_count)
23. print('Negative predictions are: ',
    negative_count)
24. print('Real positive are: ', real_positive)
25. print('Real neutral are: ', real_neutral)
26. print('Real negative: ', real_negative)
```

Positive predictions are : 456
Neutral predictions are: 572
Negative predictions are: 1900
Real positive are: 444
Real neutral are: 614
Real negative: 1870

As defined earlier act as a reference for the research question third, Ann plays a very crucial role in this research just by performing the recurrent connections to the nodes we have used eight epochs where batch size has been lifted to 32 with a verbose of 1 in context to running for the prediction model performing and showing the concurrent statistical insights of the computational ability of the model which further gets transferred with the use of variables parsed for complexity as well as time on a big O notation with O Log(np) as Log (2p) where p is the number of hidden neurons solved as 2/3 of the number of features with the addition of the output on that amout which ultimately gives the big o complexity as O log(5.3) conveyed as OLog(n) format which makes us easier to understand that only one simple fold fr condition is applied as compared to multithreaded for loop complexities for helding it act fast towards the mean epoch accuracy and time response.

```
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
embedding_1 (Embedding)      (None, 33, 256)           1536000
_________________________________________________________________
dropout_1 (Dropout)          (None, 33, 256)           0
_________________________________________________________________
lstm_1 (LSTM)                (None, 33, 256)           525312
_________________________________________________________________
lstm_2 (LSTM)                (None, 256)               525312
_________________________________________________________________
dense_1 (Dense)              (None, 3)                 771
=================================================================
Total params: 2,587,395
Trainable params: 2,587,395
Non-trainable params: 0
```

Fig 7.3

```
Epoch 1/8
11712/11712 [==============================] - 81s 7ms/step - loss: 0.654
9 - acc: 0.7336
Epoch 2/8
11712/11712 [==============================] - 79s 7ms/step - loss: 0.440
4 - acc: 0.8311
Epoch 3/8
11712/11712 [==============================] - 78s 7ms/step - loss: 0.360
0 - acc: 0.8632
Epoch 4/8
11712/11712 [==============================] - 78s 7ms/step - loss: 0.300
6 - acc: 0.8885
Epoch 5/8
11712/11712 [==============================] - 78s 7ms/step - loss: 0.260
2 - acc: 0.9054
Epoch 6/8
11712/11712 [==============================] - 82s 7ms/step - loss: 0.209
0 - acc: 0.9233
Epoch 7/8
11712/11712 [==============================] - 81s 7ms/step - loss: 0.182
2 - acc: 0.9337
Epoch 8/8
11712/11712 [==============================] - 90s 8ms/step - loss: 0.160
1 - acc: 0.9421
completed in =  648.8068859577179
```

Fig 7.4

Other Models using all three sentiments

Data Preprocessing for three sentiments

```
all_three_sentiments_data=twitter_data
```

```
training_sample,testing_sample =
train_test_split(twitter_data,test_size=0.2,rando
m_state=42)
```

```
c_v_all = CountVectorizer()
X_vector_all =
c_v_all.fit_transform(all_three_sentiments_data['
cleaned_tweets'])

ttfidf_transformer =
TfidfTransformer(norm='l2')
X_all =
ttfidf_transformer.fit_transform(X_vector_all)
y_all =
all_three_sentiments_data['airline_sentiment']
```

```
X_vector_all_copy =
c_v_all.fit_transform(all_three_sentiments_data['
text'])
X_all_copy =
ttfidf_transformer.fit_transform(X_vector_all_co
py)
y_all_copy =
all_three_sentiments_data['airline_sentiment']
```

2. KNN Algorithm

```
def knn_working_all_three(X,y):
kfolds = KFold(n_splits=5)
train_test_data_x = kfolds.split(X)
train_test_data, train_test_data_cp =
tee(train_test_data_x)
accuracy_list=[]
print('Running for KNN for all three sentiment
for 5 folds using K-fold')
for td in train_test_data_x:
yyy = np.array(y)
x_d = X[td[0],:]
y_d = ((yyy))[td[0]]
knn_= KNeighborsClassifier(n_neighbors=12)
calculate_knn=knn_.fit(x_d, y_d)
yy_predictions =
calculate_knn.predict(X[td[1]])
accuracy =
accuracy_score(yy_predictions,testing_sample['
airline_sentiment'])
accuracy_list.append(accuracy)
accuracy =
accuracy_score(yy_predictions,testing_sample['
airline_sentiment'])
returned_val= c
print("Mean Accuracy for KNN for all three
Sentiments for 5 folds is ",sum(accuracy_list)/5)
accuracy_list_all_three_sentiments['knn']=accu
racy_list

knn_working_all_three(X_all_copy,y_all_copy)
```

```
Running for KNN for all three sentiment for 5 folds
using K-fold
completed in =  9.11896800994873
Confusion Matrix is
 [[1522  450  362]
 [ 168   68   48]
 [ 199   62   49]]
Mean Accuracy for KNN for all three Sentiments for 5
folds is  0.5346311475409836
```

3. Decision Tree

```
def decision_tree_all_three(X,y):
kfolds = KFold(n_splits=5)
train_test_data_x = kfolds.split(X)
train_test_data, train_test_data_cp =
tee(train_test_data_x)
accuracy_list=[]
print('Running for Decision Tree for all three
sentiment for 5 folds using K-fold')
for td in train_test_data_x:
yyy = np.array(y)
x_d = X[td[0],:]
y_d = ((yyy))[td[0]]
ddtt_= DecisionTreeClassifier()
calculate_ddtt=ddtt_.fit(x_d, y_d)
yy_predictions =
calculate_ddtt.predict(X[td[1]])
accuracy =
accuracy_score(yy_predictions,testing_sample['
airline_sentiment'])
accuracy_list.append(accuracy)
accuracy =
accuracy_score(yy_predictions,testing_sample['
airline_sentiment'])
returned_val=
print("Mean Accuracy for Decision Tree for all
three Sentiments for 5 folds is
",sum(accuracy_list)/5)
decision_tree_all_three(X_all,y_all)
```

```
Running for Decision Tree for all three sentiment for 5
folds using K-fold
completed in =  12.396363019943237
Confusion Matrix is
 [[1262  384  300]
 [ 360  115   96]
 [ 267   81   63]]
```

Mean Accuracy for Decision Tree for all three Sentiments for 5 folds is 0.4816939890710382

4. Stochastic Gradient Descent Algorithm

```
1.  def sgdc_all_three(X,y):
2.  kfolds = KFold(n_splits=5)
3.  train_test_data_x = kfolds.split(X)
4.  train_test_data, train_test_data_cp =
    tee(train_test_data_x)
5.  accuracy_list=[]
6.  starting_timing = time.time()
7.  print('Running for SGDC for all three
    sentiment for 5 folds using K-fold')
8.  for td in train_test_data_x:
9.  yyy = np.array(y)
10. x_d = X[td[0],:]
11. y_d = ((yyy))[td[0]]
12. sggdc_= SGDClassifier(loss = 'log')
13. calculate_sggdc=sggdc_.fit(x_d, y_d)
14. yy_predictions =
    calculate_sggdc.predict(X[td[1]])
15. accuracy =
    accuracy_score(yy_predictions,testing_sample['
    airline_sentiment'])
16. accuracy_list.append(accuracy)
17. accuracy =
    accuracy_score(yy_predictions,testing_sample['
    airline_sentiment'])
18. returned_val=
    confusion_matrix(yy_predictions,testing_sampl
    e['airline_sentiment'])
19. print("completed in = ",time.time() -
    starting_timing)
20. print("Confusion Matrix is \n",returned_val)
21. print("Mean Accuracy for SGDC for all three
    Sentiments for 5 folds is ",sum(accuracy_list)/5)
22. accuracy_list_all_three_sentiments['sgdc']=accu
    racy_list
23. accuracy_mean_list_all_three.append(sum(accu
    racy_list)/5)
24. sgdc_all_three(X_all_copy,y_all_copy)
```

Running for SGDC for all three sentiment for 5 folds using K-fold completed in = 0.39816713333129883
Confusion Matrix is
[[1629 498 397]
[112 35 32]
[148 47 30]]
Mean Accuracy for SGDC for all three Sentiments for 5 folds is 0.5458333333333333

5. Random Forest Classifier

```
1.  def random_forest_all_three(X,y):
2.  kfolds = KFold(n_splits=5)
3.  train_test_data_x = kfolds.split(X)
4.  train_test_data, train_test_data_cp =
    tee(train_test_data_x)
5.  accuracy_list=[]
6.  starting_timing = time.time()
7.  print('Running for Random Forest for all three
    sentiment for 5 folds using K-fold')
8.  for td in train_test_data_x:
9.  yyy = np.array(y)
10. x_d = X[td[0],:]
11. y_d = ((yyy))[td[0]]
12. rffc_= RandomForestClassifier()
13. calculate_rffc=rffc_.fit(x_d, y_d)
14. yy_predictions = calculate_rffc.predict(X[td[1]])
15. accuracy =
    accuracy_score(yy_predictions,testing_sample['
    airline_sentiment'])
16. accuracy_list.append(accuracy)
17. accuracy =
    accuracy_score(yy_predictions,testing_sample['
    airline_sentiment'])
18. returned_val=
    confusion_matrix(yy_predictions,testing_sampl
    e['airline_sentiment'])
19. print("completed in = ",time.time() -
    starting_timing)
20. print("Confusion Matrix is \n",returned_val)
21. print("Mean Accuracy for Random Forest for
    all three Sentiments for 5 folds is
    ",sum(accuracy_list)/5)
22. accuracy_list_all_three_sentiments['rfc']=accur
    acy_list
23. accuracy_mean_list_all_three.append(sum(accu
    racy_list)/5)
24. random_forest_all_three(X_all_copy,y_all_copy
    )
```

Running for Random Forest for all three sentiment for 5 folds using K-fold
completed in = 13.194983959197998
Confusion Matrix is
[[1594 492 383]
[158 49 48]
[137 39 28]]
Mean Accuracy for Random Forest for all three Sentiments for 5 folds is 0.555396174863388

6. Perceptron

```
1.  def perceptron_all_three(X,y):
2.  kfolds = KFold(n_splits=5)
3.  train_test_data_x = kfolds.split(X)
4.  train_test_data, train_test_data_cp =
    tee(train_test_data_x)
5.  accuracy_list=[]
6.  starting_timing = time.time()
7.  print('Running for Perceptron for all three
    sentiment for 5 folds using K-fold')
8.  for td in train_test_data_x:
9.  yyy = np.array(y)
10. x_d = X[td[0],:]
11. y_d = ((yyy))[td[0]]
12. ppdd_= Perceptron(tol=1e-3, random_state=0)
13. calculate_ppdd=ppdd_.fit(x_d, y_d)
14. yy_predictions =
    calculate_ppdd.predict(X[td[1]])
```

```
accuracy = accuracy_score(yy_predictions,testing_sample['airline_sentiment'])
accuracy_list.append(accuracy)
accuracy = accuracy_score(yy_predictions,testing_sample['airline_sentiment'])
returned_val= confusion_matrix(yy_predictions,testing_sample['airline_sentiment'])
print("completed in = ",time.time() - starting_timing)
print("Confusion Matrix is \n",returned_val)
print("Mean Accuracy for Perceptron for all three Sentiments for 5 folds is ",sum(accuracy_list)/5)
```

```
Running for Perceptron for all three sentiment for 5 folds using K-fold
completed in =  0.5071799755096436
Confusion Matrix is
 [[1507  437  367]
 [ 151   53   32]
 [ 231   90   60]]
Mean Accuracy for Perceptron for all three Sentiments for 5 folds is  0.4937841530054644
```

7. Support Vector Classifier

```
def svcc_all_three(X,y):
kfolds = KFold(n_splits=5)
train_test_data_x = kfolds.split(X)
train_test_data, train_test_data_cp = tee(train_test_data_x)
accuracy_list=[]
starting_timing = time.time()
print('Running for SVC for all three sentiment for 5 folds using K-fold')
for td in train_test_data_x:
yyy = np.array(y)
x_d = X[td[0],:]
y_d = ((yyy))[td[0]]
svcc_dd_=  SVC(kernel="rbf", C=100., gamma=0.01, probability=False, degree=3)
calculate_svcc_dd=svcc_dd_.fit(x_d, y_d)
yy_predictions = calculate_svcc_dd.predict(X[td[1]])
print("completed in = ",time.time() - starting_timing)
print("Confusion Matrix is \n",returned_val)
print("Mean Accuracy for SVC for all three Sentiments for 5 folds is ",sum(accuracy_list)/5)
accuracy_list_all_three_sentiments['svc']=accuracy_list
accuracy_mean_list_all_three.append(sum(accuracy_list)/5)
svcc_all_three(X_all_copy,y_all_copy)
```

```
Running for SVC for all three sentiment for 5 folds using K-fold
completed in =  87.24719595909119
Confusion Matrix is
 [[1523  443  373]
 [ 153   52   39]
 [ 213   85   47]]
Mean Accuracy for SVC for all three Sentiments for 5 folds is  0.5040300546448088
```

8. Logistic Regression

```
def lgst_all_three(X,y):
kfolds = KFold(n_splits=5)
train_test_data_x = kfolds.split(X)
train_test_data, train_test_data_cp = tee(train_test_data_x)
accuracy_list=[]
starting_timing = time.time()
print('Running for Logistic Regression for all three sentiment for 5 folds using K-fold')
for td in train_test_data_x:
yyy = np.array(y)
x_d = X[td[0],:]
y_d = ((yyy))[td[0]]
lgdd_= LogisticRegression(C=100,solver='liblinear',max_iter=300)
calculate_lg_dd=lgdd_.fit(x_d, y_d)
yy_predictions = confusion_matrix(yy_predictions,testing_sample['airline_sentiment'])
print("completed in = ",time.time() - starting_timing)
print("Confusion Matrix is \n",returned_val)
print("Mean Accuracy for Logistic Regression for all three Sentiments for 5 folds is ",(sum(accuracy_list)/5))
accuracy_list_all_three_sentiments['lgst']=accuracy_list
accuracy_mean_list_all_three.append(sum(accuracy_list)/5)
lgst_all_three(X_all_copy,y_all_copy)
```

```
Running for Logistic Regression for all three sentiment for 5 folds using K-fold
completed in =  1.9272210597991943
Confusion Matrix is
 [[1520  435  366]
 [ 152   58   41]
 [ 217   87   52]]
Mean Accuracy for Logistic Regression for all three Sentiments for 5 folds is  0.5010928961748634
```

9. Adaboost Classifier

```
def adaboost_all_three(X,y):
kfolds = KFold(n_splits=5)
train_test_data_x = kfolds.split(X)
```

```
4.  train_test_data, train_test_data_cp =
    tee(train_test_data_x)
5.  accuracy_list=[]
6.  starting_timing = time.time()
7.  print('Running for Adaboost Classifier for all
    three sentiment for 5 folds using K-fold')
8.  for td in train_test_data_x:
9.  yyy = np.array(y)
10. x_d = X[td[0],:]
11. y_d = ((yyy))[td[0]]
12. adb_=  AdaBoostClassifier()
13. calculate_adb=adb_.fit(x_d, y_d)
14. yy_predictions = calculate_adb.predict(X[td[1]])
15. # accuracy is calculated
16. #accuracy_list_all_three_sentiments['adb']=acc
    uracy_list
17. adaboost_all_three(X_all_copy,y_all_copy)
```

```
Running for Adaboost Classifier for all three sentiment
for 5 folds using K-fold
completed in =  76.03218197822571
Confusion Matrix is
 [[1154  350  290]
 [ 472  137  120]
 [ 263   93   49]]
Mean Accuracy for Adaboost for all three Sentiments
for 5 folds is  0.49678961748633876
```

Comparison of Accuracies for 5 folds using all three sentiments

```
1.  accuracy_df_all_three=pd.DataFrame.from_dic
    t(accuracy_list_all_three_sentiments)
2.  accuracy_df_all_three_transpose=accuracy_df_
    all_three.transpose()
3.  accuracy_df_all_three_transpose['Mean
    Accuracy']=accuracy_mean_list_all_three
4.  accuracy_df_all_three_transpose
```

```
5.  accuracy_line_graph_all_three =
    accuracy_df_all_three.plot(kind='line',figsize=(
    12,
    8),rot=0,color=['b','g','y','r','orange','magenta','
    black','grey'],title="Accuracy Comparison for 5
    Folds for all three sentiments")
6.  accuracy_line_graph.set_xlabel("Number of K-
    FOLDS")
7.  accuracy_line_graph.set_ylabel("Accuracy
    Value")
```

	0	1	2	3	4	Mean Accuracy
knn	0.527322	0.520150	0.502391	0.563525	0.559768	0.534631
dt	0.465505	0.487363	0.439891	0.523907	0.491803	0.481694
sgdc	0.552254	0.524249	0.483607	0.590505	0.578552	0.545833
rfc	0.562842	0.544399	0.505464	0.593579	0.570697	0.555396
perceptron	0.482923	0.460724	0.421790	0.550205	0.553279	0.493784
svc	0.502732	0.485314	0.418374	0.559768	0.553962	0.504030
lgst	0.498634	0.473361	0.423497	0.553279	0.556694	0.501093
adb	0.537910	0.487705	0.504098	0.496585	0.457650	0.496790

Fig 7.5

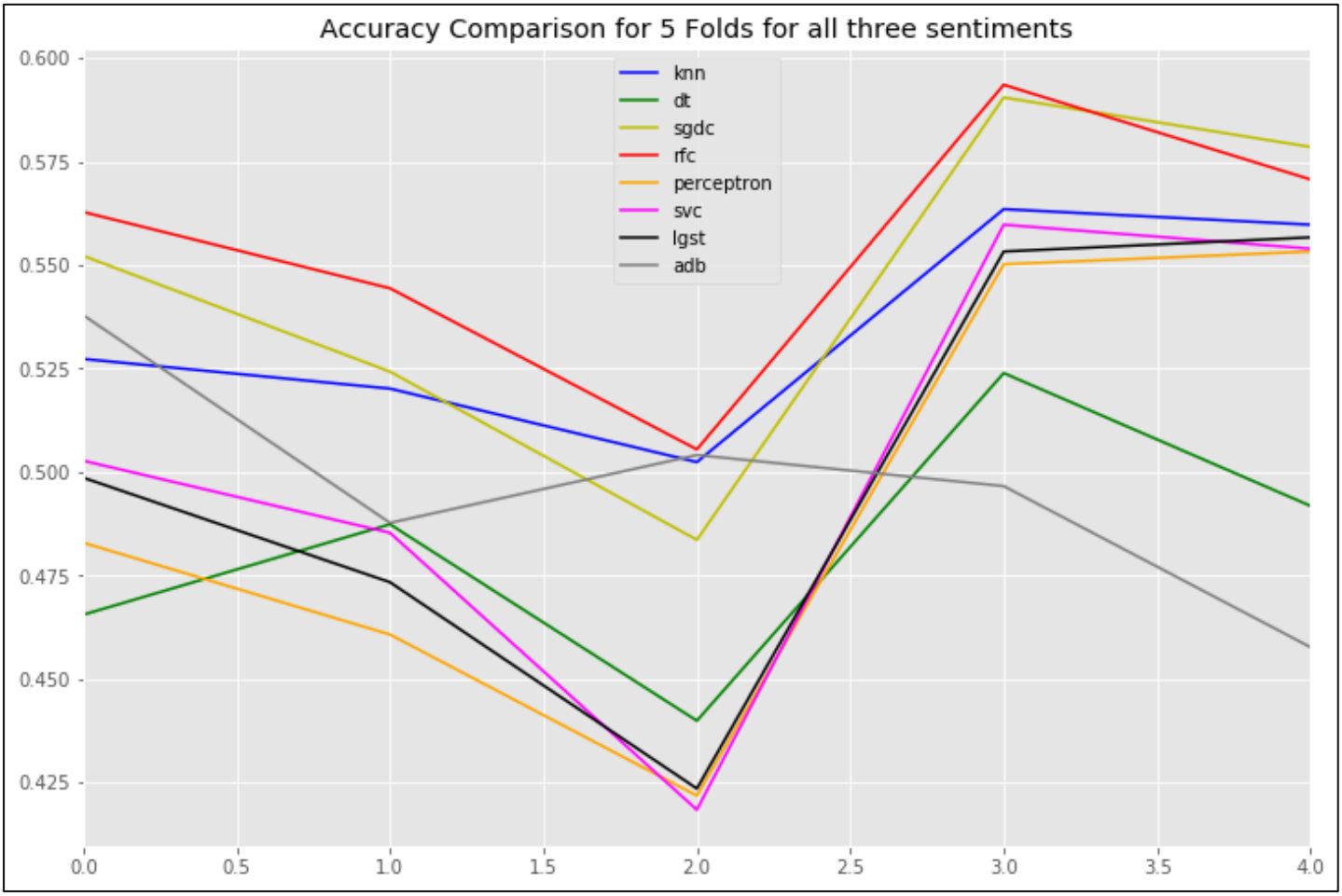

Fig 7.6

VIII. EVALUATION AND SCORES FOR EACH MODEL

Identifying and understanding the confusion matrix briefly for positive and negative sentiment prediction of scores with the use of results from confusion matrix from all classifier algorithms on various evaluation basis and formulas as following:

1. Decision Tree Classifier
Confusion Matrix is

	True Positive	True Negative
Predicted Positive	**[1821**	**151]**
Predicted Negative	**[108**	**228]**

TP = 1821 , FN = 151 , FP = 108 and TN = 228

Measure	Value	Derivation
Sensitivity	0.9440	TPR = TP / (TP + FN)
Specificity	0.6016	SPC = TN / (FP + TN)
Precision	0.9234	PPV = TP / (TP + FP)
Negative Predictive Value	0.6786	NPV = TN / (TN + FN)
False Positive Rate	0.3984	FPR = FP / (FP + TN)
False Discovery Rate	0.0766	FDR = FP / (FP + TP)
False Negative Rate	0.0560	FNR = FN / (FN + TP)
F1 Score	0.9336	F1 = 2TP / (2TP + FP + FN)
MCC	0.5731	Using Matthews Correlation Coefficient

2. SGD Classifier
Confusion Matrix is

	True Positive	True Negative
Predicted Positive	**[1958**	**14]**
Predicted Negative	**[155**	**181]**

TP = 1958 , FN = 14 , FP = 155 and TN = 181

Measure	Value	Derivation
Sensitivity	0.9266	TPR = TP / (TP + FN)
Specificity	0.9282	SPC = TN / (FP + TN)
Precision	0.9929	PPV = TP / (TP + FP)
Negative Predictive Value	0. 5387	NPV = TN / (TN + FN)
False Positive Rate	0.0718	FPR = FP / (FP + TN)
False Discovery Rate	0.0071	FDR = FP / (FP + TP)
False Negative Rate	0.0734	FNR = FN / (FN + TP)
F1 Score	0.9586	F1 = 2TP / (2TP + FP + FN)
MCC	0.6741	Using Matthews Correlation Coefficient

3. Random Forest Classifier
Confusion Matrix is

	True Positive	True Negative
Predicted Positive	**[1946**	**26]**
Predicted Negative	**[175**	**161]**

TP = 1946 , FN = 26 , FP = 175 and TN = 161

Measure	Value	Derivation
Sensitivity	0.9175	TPR = TP / (TP + FN)
Specificity	0.8610	SPC = TN / (FP + TN)
Precision	0.9868	PPV = TP / (TP + FP)
Negative Predictive Value	0.4792	NPV = TN / (TN + FN)
False Positive Rate	0.1390	FPR = FP / (FP + TN)
False Discovery Rate	0.0132	FDR = FP / (FP + TP)
False Negative Rate	0.0825	FNR = FN / (FN + TP)
F1 Score	0.9509	F1 = 2TP / (2TP + FP + FN)
MCC	0.6023	Using Matthews Correlation Coefficient

4. K Neighbors Classifier
Confusion Matrix is

	True Positive	True Negative
Predicted Positive	**[1905**	**67]**
Predicted Negative	**[103**	**233]**

TP = 1905 , FN = 67 , FP = 103 and TN = 233

Measure	Value	Derivation
Sensitivity	0.9487	TPR = TP / (TP + FN)
Specificity	0.7767	SPC = TN / (FP + TN)
Precision	0.9660	PPV = TP / (TP + FP)
Negative Predictive Value	0.6935	NPV = TN / (TN + FN)
False Positive Rate	0.2233	FPR = FP / (FP + TN)
False Discovery Rate	0.0340	FDR = FP / (FP + TP)
False Negative Rate	0.0513	FNR = FN / (FN + TP)
F1 Score	0.9573	F1 = 2TP / (2TP + FP + FN)
MCC	0.6916	Using Matthews Correlation Coefficient

5 Perceptron Classifier
Confusion Matrix is

	True Positive	True Negative
Predicted Positive	**[1888**	**84]**
Predicted Negative	**[81**	**255]**

TP = 1888 , FN = 84 , FP = 81 and TN = 255

Measure	Value	Derivation
Sensitivity	0.9589	TPR = TP / (TP + FN)
Specificity	0.7522	SPC = TN / (FP + TN)
Precision	0.9574	PPV = TP / (TP + FP)
Negative Predictive Value	0.7589	NPV = TN / (TN + FN)
False Positive Rate	0.2478	FPR = FP / (FP + TN)
False Discovery Rate	0.0426	FDR = FP / (FP + TP)
False Negative Rate	0.0411	FNR = FN / (FN + TP)
F1 Score	0.9581	F1 = 2TP / (2TP + FP + FN)
MCC	0.7137	Using Matthews Correlation Coefficient

6. SVC Classifier
Confusion Matrix is

	True Positive	True Negative
Predicted Positive	**[1922**	**50]**
Predicted Negative	**[92**	**244]**

TP = 1922 , FN = 50 , FP = 92 and TN = 244

Measure	Value	Derivation
Sensitivity	0.9543	TPR = TP / (TP + FN)
Specificity	0.8299	SPC = TN / (FP + TN)
Precision	0.9746	PPV = TP / (TP + FP)
Negative Predictive Value	0.7262	NPV = TN / (TN + FN)
False Positive Rate	0.1701	FPR = FP / (FP + TN)
False Discovery Rate	0.0254	FDR = FP / (FP + TP)
False Negative Rate	0.0457	FNR = FN / (FN + TP)
F1 Score	0.9644	F1 = 2TP / (2TP + FP + FN)
MCC	0.7414	Using Matthews Correlation Coefficient

7. Logistic Regression Classifier
Confusion Matrix is

	True Positive	True Negative
Predicted Positive	**[1904**	**68]**
Predicted Negative	**[83**	**253]**

TP = 1904 , FN = 68 , FP = 83 and TN = 253

Measure	Value	Derivation
Sensitivity	0.9582	TPR = TP / (TP + FN)
Specificity	0.7882	SPC = TN / (FP + TN)
Precision	0.9655	PPV = TP / (TP + FP)
Negative Predictive Value	0.7530	NPV = TN / (TN + FN)
False Positive Rate	0.2118	FPR = FP / (FP + TN)
False Discovery Rate	0.0345	FDR = FP / (FP + TP)
False Negative Rate	0.0418	FNR = FN / (FN + TP)
F1 Score	0.9614	F1 = 2TP / (2TP + FP + FN)
MCC	0.7321	Using Matthews Correlation Coefficient

8. Adaboost Classifier
Confusion Matrix is

	True Positive	**True Negative**
Predicted Positive	**[1907**	**65]**
Predicted Negative	**[118**	**218]**

TP = 1907 , FN = 65 , FP = 118 and TN = 218

Measure	Value	Derivation
Sensitivity	0.9417	TPR = TP / (TP + FN)
Specificity	0.7703	SPC = TN / (FP + TN)
Precision	0.9670	PPV = TP / (TP + FP)
Negative Predictive Value	0.6488	NPV = TN / (TN + FN)
False Positive Rate	0.2297	FPR = FP / (FP + TN)
False Discovery Rate	0.0330	FDR = FP / (FP + TP)
False Negative Rate	0.0583	FNR = FN / (FN + TP)
F1 Score	0.9542	F1 = 2TP / (2TP + FP + FN)
MCC	0.6622	Using Matthews Correlation Coefficient

RUNTIME AND BIG O NOTATION COMPLEXITY ANALYSIS FOR EACH ALGORITHM FOR POSITIVE AND NEGATIVE SENTIMENTS

Model	Individual Execution Time	Big O Notation
SVC	119.4 sec	O (n_{SV} p)
KNN	5.710 sec	O(np)
SGDC	0.2866sec	O(pN_{Trees})
PERCEPTRON	0.2179sec	O(n^2)
RANDOM FOREST	6.19 sec	O(p log n)
DECISION TREES	11.175 sec	O(p)
ADABOOST	34..2 sec	O(p log n)
LOGISTIC REGRESSION	0.5001 sec	O(p)

COMPARISON OF ACCURACY FOR THE SENTIMENT ANALYSIS FROM PREVIOUS RESEARCH PAPERS FOR POSITIVE AND NEGATIVE SENTIMENTS.

Model	Previous Accuracy	Verified New Results
SVC	81.2%[49]	91.44%
KNN	59%[49]	88.83%
SGDC	-	90.16%
PERCEPTRON	-	89.28%
RANDOM FOREST	85.6%[49]	87.11%
DECISION TREES	63%[49]	85.33%
ADABOOST	84.5%[49]	88.95%
LOGISTIC REGRESSION	81%[49]	90.85%

RUNTIME AND BIG O NOTATION COMPLEXITY ANALYSIS FOR EACH ALGORITHM FOR POSITIVE, NEGATIVE AND NEUTRAL SENTIMENTS

Model	Individual Execution Time	Big O Notation
SVC	87.24 sec	O (n_{SV} p)
KNN	9.11 sec	O(np)
SGDC	0.398 sec	O(pN_{Trees})
PERCEPTRON	0.507 sec	O(n^2)
RANDOM FOREST	13.19 sec	O(n^2)
DECISION TREES	12.39 sec	O(p)
ADABOOST	76.03 sec	O(p log n)
LOGISTIC REGRESSION	1.92 sec	O(logn^2)
ANN	648.80 sec	OLog(n)

COMPARISON OF ACCURACY FOR THE SENTIMENT ANALYSIS FROM TWO SENTIEMNTS AS NEGATIVE AND POSITIVE IN A SIMILAR CLASSIFIER FOR THE CHANGE IN STATISTICS FOR POSITIVE, NEGATIVE AND NEUTRAL SENTIMENTS.

Model	Accuracy for 3 sentiments (positive, negative & neutral)	Accuracy for 2 sentiments (positive & negative)
SVC	50.40%	91.44%
KNN	53.46%	88.83%
SGDC	54.58%	90.16%
PERCEPTRON	49.37%	89.28%
RANDOM FOREST	55.53%	87.11%
DECISION TREES	48.16%	85.33%
ADABOOST	49.67%	88.95%
LOGISTIC REGRESSION	50.10%	90.85%

In this Research, a very unique concept for involving the recurrent neural networks for combination to performance on both the concepts where time and resources are important for the future of this research which could be practically be relied on a larger scale involves a status for great success where the algorithm runs on a 648.80 sec of time computation and proves a 94.21% of accuracy which was not utilized by the implementation of Ann in previous research papers which are published or proved to do so.

Best Classifier	Accuracy Score	Time Taken
ANN	94.21%	648.80 sec

UNDERSTANDING AND REVEALING CONCEPT BEHIND THE RESEARCH QUESTIONS SOLVED USING PROPOSED METHODS

REAL TIME USEAGE OF RESULTS:

Overall the whole solutions for these research questions can be seen through the visualization of the dataset also. But for a brief reasoning and a deep understanding of how and why the concept arises can be seen via examples for each of the following R.Q's:

Solving R.Q.1 The proposed solution provides a layout architecture for all the models as well as the data analytics to be performed upon the dataset which gives us an opportunity to go behind the case study for what made the user to create that blog or tweet post for which if we study the dataset for a single tweet for example "@XYZFLIGHT took less time which is good , food was bad , seating was uncomfortable" . Now after performing this tweet from our proposed solution the results will be SENTIMENT PREDICTED : Negative , REASON : BAD FLIGHT , because the addition as well as the subtraction of whole sum of the negative and the positive words is negative (which is more in count) form of sentiment shown by results and on the other hand bad flight gives an idea of both that the service was bad and term "flight" gives the hint for something could be food, seating, service or any other thing. So, with the help of certain types of tweets an individual can study a customer's satisfaction or dissatisfaction level upon airlines services.

Solving R.Q.2 The precision of each model with the help of several problems predicting sentiment

to be a negative or a positive is visually represented by the Bar-graph for negative reasons by time. This is because “Time” plays a crucial factor in understanding the reasons for what ratio the things have not been changed or any sector that needs more improvement on a particular factor influencing an attribute for which the same reasons turn out to be having a higher value each time. To add to it, The scale for the same tweets gets verified because the dataset involves time based tweets for which imagine a same type of flight for example have a tweet in negative sentiment for luggage problems. Now imagine after every week the same dataset is again collected for the next 5 weeks each time the tweets are similar to the first week problem. This situation leads an increment in the problem for luggage after each renewal verifying that the problem has not been solved and needs attention in that sector for luggage may be lost luggage, Broken Luggage, Wrong Address Delivered Luggage or any other service related will be exposed with the help of this model.

Solving R.Q3 As shown and discussed earlier, the ANN classifier act as the best model. To add to it, when a classifier shows a relatively better accuracy as the other models the datasets can be relied to be classified and analyzed statistically towards them which helps the airlines to understand and believe the model building trust on technology as well as in real time and practical implementation proves to solve various problems helping informative awareness in terms of airline experience as per the user’s perspective.

Solving R.Q4 While conversion of the datasets from word into vector formation the figure 5.6, fig 5.9 and fig 5.10 shows how data can be used for the purpose of learning the topics on which airlines has to work and on which part of the problem is to be more focused by any specific type of airlines or on the other hand the datasets can also be used for a specific type of problem to learn from it generally occurred in every airlines by the common roles using the graphs and plots to better visualize and find the common areas in each airlines and relation between the problems.

Solving R.Q5 The figure 6.7 and the data processing stage shows the proper utility for the implementation of information transformation taking place between the micro-blogs from social media which acts as the tweets or microblogs to regenerate into a useful and meaningful data that helps to convey a message or information helpful to the airline industry and the services related to it.

R.Q.6 There exists various types of datasets and information available for the study of sentimental analysis on airlines but there is a gap between a function which helps to sort that certain types of datasets will be processed to follow a high level structure in order to classify themselves into some categories such as Airline response to user or vice versa, User reply to another user in context of issue or any conversation related to airline industry, Marketing industry tweets related to airlines such as food, banners, advertisements and so on, a customer finding resources or information for community reporting or support feedback to users in updating the portal of airline schedules as well as any topic for Grievances act as some major roles of categories that can be discovered from the types of contexts between the response feedback or exchange of information between a user and the airline communication under social media feedback management system.

IX. DISCUSSION AND IMPLICATIONS OF RESULTS

The Airline Industry is one of the most developing and growing economic sectors for the fast transportation means as well as comfort in business making the most efforts to make a mark up to perfection. The results for this research helps to solve similar but the most effective and meaningful manners by data analytic methodologies to solve the problem of maintaining and keep up to the mark pace of all the zones in context for the agents, customers, the services as well as any departments that are co-relational to the airline at an ease of access.

NOVELTY OF THIS RESEARCH

For all the tests as well as sentiment analysis performed on the basis of any social media. No one has done a published research using ANN (Artificial Neural Networks) creating a recurrent layers for using k folds and fitting methods under which ANN is verified and applied proving the statistics to be the best classifier and also secondly to add to it, ANN helps us to show and proof a verified accuracy by functional time

computation of 648 seconds approx. representing an 8th epoch for batch 32 accuracy to be more than 94.21% which has been tested and computed with the confusion matrix produced shown in screenshot fig 7.4.

PRACTICALITY AND REAL TIME USEAGE

This research has been solving a 6 solution problems with the use of research questions that helps to understand the real utility of this research which can be straight forward use on the basis of twitter dataset and any other data set model which has been verified and proved to show such accuracies by applied confusion matrix functions in the program file shown in the evaluation section Ultimately making this research new and original to help and guide a new way of representation of data with the use of word cloud which roughly guides how the raw data can be used in such a meaningful manner and can be classified in real time by the Airlines to understand better by visualization graphic plots to exactly see which sector needs the most attention making it very easy o understand by any technical or non – technical individual.

The fig 5.6 is a very important part of the research questions to know the hidden portions which in further gets related on the basis of time, date as well as airlines to co-relate that which airlines show the most negative and positive and also helps to visualize that if it's a negative area or positive area for the percentage of each airline in fig 5 .9 and fig 5.10 making it accessible and better presentation for more convincingly communication with the help of proved results that predicts the models and shows truly that this research is the best version of the previous classifiers.

Moreover, SGDC, PERCEPTRON and ANN are the new approaches proved to solve the similar problem with advanced and unique methodology gaining a high accuracy where of all the sentiment analysis with twitter on airline industry been done so far this research achieves the highest accuracy using ANN.

To add to it, No other research has done a 2 by 2 and additional 3 x 3 sentiment analysis compared which makes us understand why the neutral section of sentiment is also important. This research has both case scenarios where we are finding a 2x2 matrix for positive as well as Negative sentiment which has been done in previous researches but with low accuracy and complex methods whereas this research also includes a verified 3x3 matrix showing all the negative , positive as well as neutral sentiment prediction and the comparison table also for both the confusion matrix achieved accuracies ultimately providing solutions to all the research questions arising so far and still helping to be get better in future research.

FUTURE WORK

For the future work, the more the database , the higher the accuracy as well as efficiency of the model can be achieved for which this research can be modified by creating an all way platform where not only twitter but other platforms will be used for getting extracting the already collected information into csv datasets from Facebook, Instagram, google feedback etc. Also this can be done with a real time extraction of putting the access token on the developer modes of twitter and Facebook which helps to collect the real time data but it sometimes has its own pros and cons such as more time consumption as well as more data cleaning requirement and pre-processing before the creation of dataset into a csv version of information extracted. We can also develop a different neural network using a better or slightly advanced chain version of batch number and with new verbose amount unit or the number of epochs also this can be help us to visualize the real time screening on the airports for a practical implementation of this research where the users can actually see the services and sections already maintained by Airlines using some colours such as shown in the negative sentiments reasons which will break down the whole concept into a zone of being done and already under progress with comparisons of the things that can be shown red or any other colour that has been still not working good. All this can be achieved through a better connection of SQL database management systems where data analytics can be applied to google surveys in real time and these classifiers can be modified to use exactly at the same placements where they will be helpful for the government, airline industry and people related services to it and most importantly building a trust amongst the customers feedback and

services getting improved in real time making it a more efficient, reliable, real time system just by implementation of this research as a base unit for a more effective future of Airline Industry growth.

REFERENCES

1. Marcelo Maia, Jussara Almeida, and Virgílio Almeida. 2008. Identifying user behavior in online social networks. In Proceedings of the 1st Workshop on Social Network Systems (SocialNets '08). ACM, New York, NY, USA, 1-6. DOI=http://dx.doi.org/10.1145/1435497.1435498.
2. Eugene Agichtein, Eric Brill, and Susan Dumais. 2006. Improving web search ranking by incorporating user behavior information. In Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval (SIGIR '06). ACM, New York, NY, USA, 19-26. DOI=http://dx.doi.org/10.1145/1148170.1148177.
3. Jaimie Y. Park, Neil O'Hare, Rossano Schifanella, Alejandro Jaimes, and Chin-Wan Chung. 2015. A Large-Scale Study of User Image Search Behavior on the Web. In Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems (CHI '15). ACM, New York, NY, USA, 985-994. DOI: https://doi.org/10.1145/2702123.2702527 .
4. A Siqueira W.G., Baldochi L.A. (2018) Leveraging Analysis of User Behavior from Web Usage Extraction over DOM-tree Structure. In: Mikkonen T., Klamma R., Hernández J. (eds) Web Engineering. ICWE 2018. Lecture Notes in Computer Science, vol 10845. Springer, Cham DOI:https://doi.org/10.1007/978-3319916620_14.
5. Liu, G., Nguyen, T.T., Zhao, G., Zha, W., Yang, J., Cao, J., Wu, M., Zhao, P., Chen, W.: Repeat buyer prediction for e-commerce. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD 2016, pp. 155–164. ACM, New York (2016)
6. Xie, X. & Wang, B. "Web page recommendation via twofold clustering: considering user behavior and topic relation" ; Neural Computing & Applications" (2016) 29: 235. https://doi.org/10.1007/s00521-016-2444-z.
7. Jiahui Liu, Peter Dolan, and Elin Rønby Pedersen. 2010. Personalized news recommendation based on click behavior. In Proceedings of the 15th international conference on Intelligent user interfaces (IUI '10). ACM, New York, NY, USA, 31-40. DOI: https://doi.org/10.1145/1719970.1719976 .
8. Srijan Kumar, William L. Hamilton, Jure Leskovec, and Dan Jurafsky. 2018. Community Interaction and Conflict on the Web. In Proceedings of the 2018 World Wide Web Conference (WWW '18). International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, Switzerland, 933-943. DOI: https://doi.org/10.1145/3178876.3186141 .
9. Xiaohui Xie, Yiqun Liu, Maarten de Rijke, Jiyin He, Min Zhang, and Shaoping Ma. 2018. Why People Search for Images using Web Search Engines. In Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining (WSDM '18). ACM, New York, NY, USA, 655-663. DOI: https://doi.org/10.1145/3159652.3159686 .
10. Zeynep Zengin Alp, Şule Gündüz Öğüdücü,"Identifying topical influencers on twitter based on user behavior and network topology",Knowledge-Based Systems,Volume 141,2018,Pages 211-221,ISSN 0950-7051,https://doi.org/10.1016/j.knosys.2017.11.021.
11. Nikolaos Misirlis, Maro Vlachopoulou,"Social media metrics and analytics in marketing – S3M: A mapping literature review",International Journal of Information Management,Volume 38, Issue 1,2018, Pages 270-276,ISSN 0268-4012, https://doi.org/10.1016/j.ijinfomgt.2017.10.005.
12. Stefano Tasselli, Martin Kilduff and Blaine Landis," Personality Change: Implications for Organizational Behavior", Academy of Management Annals Vol.12, No. 2, Published Online:9/Jul/2018, https://doi.org/10.5465/annals.2016.0008.
13. Jieun Shin, Lian Jian, Kevin Driscoll, François Bar,"The diffusion of misinformation on social media: Temporal pattern, message, and source, Computers in Human Behavior",Volume 83,2018,Pages 278-287,ISSN 0747-5632, https://doi.org/10.1016/j.chb.2018.02.008.
14. X. Luo, J. Wang, Q. Shen, J. Wang and Q. Qi, "User behavior analysis based on user

interest by web log mining," 2017 27th International Telecommunication Networks and Applications Conference (ITNAC), Melbourne, VIC, 2017, pp. 1-5. DOI: 10.1109/ATNAC.2017.8215435.

15. Michela Del Vicario, Fabiana Zollo, Guido Caldarelli, Antonio Scala, Walter Quattrociocchi,"Mapping social dynamics on Facebook: The Brexit debate",Social Networks,Volume 50,2017,Pages 6-16,ISSN 0378-8733, https://doi.org/10.1016/j.socnet.2017.02.002.
16. "Sentiment Analysis " Wikipedia web page URL: https://en.wikipedia.org/wiki/Sentiment_analysis
17. Turcotte, Melissa Sanna Passino, Francesco Moore, Juston Shane Heard, and Nicholas,"User Behaviour Analytics" : Los Alamos National Laboratory-UR-19-22194:, 2019-03-12, Url:https://permalink.lanl.gov/object/tr?what=info:lanl-repo/lareport/LA-UR-19-22194.
18. Kietzmann, J. H. (2011). Social media? Get serious! Understanding the functional building blocks of social media. Business horizons , 54- 3, 241-251.
19. Mishra, N., & C.K.Jha. (2012). Classification of Opinion Mining techniques. International Journal Of Computer Applications, 0975-8887.
20. Jaganadh, G. (2012). Opinion mining and Sentiment analysis CSI Communication.
21. Liu, B. (2011). Opinion Mining and Sentiment Analysis. AAAI, San Francisco.
22. Bo Pang and Lillian Lee. 2008. Opinion Mining and Sentiment Analysis. Found. Trends Inf. Retr. 2, 1-2 (January 2008), 1-135. DOI=http://dx.doi.org/10.1561/1500000011.
23. Cha, M., Haddadi, H., Benevenuto, F., & Gummadi, K. P. (2010). Measuring User Influence in Twitter: The Million Follower Fallacy. ICWSM, 10, 10-17.
24. Goyal, A., Bonchi, F., & and Lakshmanan, L. V. (2010.). Learning Influence Probabilities In Social Networks. ACM WSDM.
25. Gaspar, R. S. (2014). Tweeting during food crises: A psychosocial analysis of threat coping expressions in Spain, during the 2011 European EHEC outbreak. International Journal of HumanComputer Studies, 72.2 , 239-254.
26. Sheth, A. P. (2010). Understanding events through analysis of social media.
27. Mike Thelwall, Kevan Buckley, and Georgios Paltoglou. 2011. Sentiment in Twitter events. J. Am. Soc. Inf. Sci. Technol. 62, 2 (February 2011), 406-418. DOI=http://dx.doi.org/10.1002/asi.21462.
28. Milstein, S. B. (2008). Twitter and the micro-messaging revolution: Communication, connections, and immediacy. 140 characters at a time. O'Reilly Media, Incorporated,.
29. Federico Neri, Carlo Aliprandi, Federico Capeci, Montserrat Cuadros, and Tomas By. 2012. Sentiment Analysis on Social Media. In Proceedings of the 2012 International Conference on Advances in Social Networks Analysis and Mining (ASONAM 2012) (ASONAM '12). IEEE Computer Society, Washington, DC, USA, 919-926. DOI=http://dx.doi.org/10.1109/ASONAM.2012.
30. Pang, B. a. (2008). Opinion mining and sentiment analysis. Foundations and trends in information retrieval, 2(1-2), 1-135. [
31. R. A. S. C. Jayasanka, M. D. (2014). Sentiment Analysis for Social Media. URL: http://dl.lib.mrt.ac.lk/handle/123/11050.
32. Smeureanu, I. a. (2012). Applying Supervised Opinion Mining Techniques on Online User Reviews. Informatica Economică, 16-2.
33. Wang, H. a. (2008)). A knowledge management approach to data mining process for business intelligence. Industrial Management & Data Systems, 108-5, 622-634.
34. Hassan, A. V. (2010). What's with the attitude?: identifying sentences with attitude in online discussions. In Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing, (pp. 1245-1255).
35. Singh, P. K. (2014). Methodological study of opinion mining and sentiment analysis techniques. International Journal on Soft Computing, 5(1), 11
36. Anindya Ghose and Panagiotis G. Ipeirotis. 2007. Designing novel review ranking systems: predicting the usefulness and impact of reviews. In Proceedings of the ninth international conference on Electronic commerce (ICEC '07). ACM, New York, NY, USA, 303-310. DOI: https://doi.org/10.1145/1282100.1282158.
37. Segerberg, A. a. (2011). Social media and the organization of collective action: Using Twitter to explore the ecologies of two climate change protests. The Communication Review , 14-3, 197-215.
38. Weng, J. E.-P. (2010). Twitterrank: finding topic-sensitive influential twitterers. In

Proceedings of the third ACM international conference on Web search and data mining,, 261-270.
39. Osimo, D. a. (2012). Research challenge on opinion mining and sentiment analysis. Universite de Paris-Sud, Laboratoire LIMSI-CNRS, Bâtiment , 508.
40. Ellonen, H.-K. A. (2010). The effect of website usage and virtual community participation on brand relationships. International Journal of Internet Marketing and Advertising, 56-1, 85-105.
41. Laine, M. a. (2010). Monitoring social media: tools, characteristics and implications. Springer Berlin Heidelberg., 51- 2, 193–198.
42. Graffigna, G., & Riva, G. (2015). Social Media Monitoring and understanding: an integrated mixed methods approach for the analysis of social media. International Journal of Web Based Communities, 11.1, 57-72. International Journal of Computer Science & Engineering Survey (IJCSES) Vol.6, No.5, October 2015 27
43. Priyanga Gunarathne, Huaxia Rui and Abraham Seidmann, "when social media delivers customer service:differential customer treatment in the airline industry" Url: https://www.misq.org/skin/frontend/default/misq/pdf/appendices/2018/V42I2Appendices/06_14290_RA_GunarathneAppendices.pdf.
44. Adeborna, Esi and Siau, Keng, "AN APPROACH TO SENTIMENT ANALYSIS –THE CASE OF AIRLINE QUALITY RATING" (2014). PACIS 2014 Proceedings. Paper 363. http://aisel.aisnet.org/pacis2014/363.
45. Mostafa, M.M. Soc. Netw. Anal. Min. (2013) 3: 635. https://doi.org/10.1007/s13278-013-0111-2.
46. Dharmavaram Sreenivasan, N., Sian Lee, C. and Hoe-Lian Goh, D. (2012), "Tweeting the friendly skies", Program: electronic library and information systems, Vol. 46 No. 1, pp. 21-42. https://doi.org/10.1108/00330331211204548.
47. Yee Liau, B. and Pei Tan, P. (2014), "Gaining customer knowledge in low cost airlines through text mining", Industrial Management & Data Systems, Vol. 114 No. 9, pp. 1344-1359. https://doi.org/10.1108/IMDS-07-2014-0225.
48. Deb Dutta Das et al 2017 IOP Conf. Ser.: Mater. Sci. Eng. 263 042067 , URL: https://iopscience.iop.org/article/10.1088/1757-899X/263/4/042067/pdf.
49. A. Rane and A. Kumar, "Sentiment Classification System of Twitter Data for US Airline Service Analysis," 2018 IEEE 42nd Annual Computer Software and Applications Conference (COMPSAC), Tokyo, 2018, pp. 769-773., doi: 10.1109/COMPSAC.2018.00114.
50. Twitter US Airline Sentiment, Kaggle Dataset URL: https://www.kaggle.com/crowdflower/twitter-airline-sentiment.
51. Y. Wan and Q. Gao, "An Ensemble Sentiment Classification System of Twitter Data for Airline Services Analysis," 2015 IEEE International Conference on Data Mining Workshop (ICDMW), Atlantic City, NJ, 2015, pp. 1318-1325. doi: 10.1109/ICDMW.2015.7
52. Adeborna, Esi and Siau, Keng, "AN APPROACH TO SENTIMENT ANALYSIS –THE CASE OF AIRLINE QUALITY RATING" (2014). PACIS 2014 Proceedings. Paper 363. http://aisel.aisnet.org/pacis2014/363.
53. A Simone Guercini, P., Misopoulos, F., Mitic, M., Kapoulas, A. and Karapiperis, C. (2014), "Uncovering customer service experiences with Twitter: the case of airline industry", Management Decision, Vol. 52 No. 4, pp. 705-723. https://doi.org/10.1108/MD-03-2012-0235
54. Iqbal, S., Zulqurnain, A., Wani, Y., & Hussain, K. (2016). The survey of sentiment and opinion mining for behavior analysis of social media. ArXiv, abs/1610.06085.,URL: https://arxiv.org/abs/1610.06085.
55. Trevor Hastie, T. Robert., F. Jerome, "The Elements of Statistical Learning" URL: https://web.stanford.edu/~hastie/Papers/ESLII.pdf.
56. Rahul Saxena, "HOW DECISION TREE ALGORITHM WORKS" Web page URL https://dataaspirant.com/2017/01/30/how-decision-tree-algorithm-works/.
57. Madhu Sanjeevi ," Decision Trees Algorithms", Web page URL :" https://medium.com/deep-math-machine-learning-ai/chapter-4-decision-trees-algorithms-b93975f7a1f1".
58. Quora Wbsite Article URL : https://www.quora.com/Whats-the-difference-between-gradient-descent-and-stochastic-gradient-descent.
59. Vinay Patlolla, "How to make SGD Classifier perform as well as Logistic Regression using parfit",URL: https://towardsdatascience.com/how-to-

make-sgd-classifier-perform-as-well-as-logistic-regression-using-parfit-cc10bca2d3c4.

60. Stacey Ronaghan," The Mathematics of Decision Trees, Random Forest and Feature Importance in Scikit-learn and Spark" ,URL: https://medium.com/@srnghn/the-mathematics-of-decision-trees-random-forest-and-feature-importance-in-scikit-learn-and-spark-f2861df67e3.
61. Saimadhu Polamuri, "HOW THE RANDOM FOREST ALGORITHM WORKS IN MACHINE LEARNING",URL:" https://dataaspirant.com/2017/05/22/random-forest-algorithm-machine-learing/.
62. Statsoft.com Web Page Url : http://www.statsoft.com/Textbook/k-Nearest-Neighbors.
63. Tavish Srivastava, "Introduction to k-Nearest Neighbors: A powerful Machine Learning Algorithm (with implementation in Python & R)", URL: https://www.analyticsvidhya.com/blog/2018/03/introduction-k-neighbours-algorithm-clustering/.
64. Sagar sharma," What the Hell is Perceptron?",URL : https://towardsdatascience.com/what-the-hell-is-perceptron-626217814f53.
65. Akshay Chandra Lagandula," Perceptron Learning Algorithm: A Graphical Explanation Of Why It Works" ,URL: https://towardsdatascience.com/perceptron-learning-algorithm-d5db0deab975.
66. Avinash Navlani, July 12th, 2018," Support Vector Machines with Scikit-learn",URL: https://www.datacamp.com/community/tutorials/svm-classification-scikit-learn-python.
67. Haebichan Jung, "Adaboost for Dummies: Breaking Down the Math (and its Equations) into Simple Terms",URL: https://towardsdatascience.com/adaboost-for-dummies-breaking-down-the-math-and-its-equations-into-simple-terms-87f439757dcf.
68. Jason Brownlee," Boosting and AdaBoost for Machine Learning",URL:" https://machinelearningmastery.com/boosting-and-adaboost-for-machine-learning/".
69. Dr. GP Pulipaka, "Applying Gaussian Naïve Bayes Classifier in Python: Part One" ,URL " https://medium.com/@gp_pulipaka/applying-gaussian-na%C3%AFve-bayes-classifier-in-python-part-one-9f82aa8d9ec4".
70. Tavish Srivastava, "How does Artificial Neural Network (ANN) algorithm work? Simplified!" URL: https://www.analyticsvidhya.com/blog/2014/10/ann-work-simplified/.