

A

PROJECT REPORT ON
Exploratory Data Analysis on ABC News

Submitted in partial fulfillment of the
requirements of the degree of

Bachelor of Engineering

By

CHAKOLA DERECK JOS	118IT1153A
HARISH NATARAJAN	118IT1400A
OMKAR MAHADIK	119IT3251A

Under the guidance of

Prof. Manivannan



Mahatma Gandhi Mission's
College of Engineering & Technology

Department of Information Technology
Mahatma Gandhi Mission's College of Engineering & Technology

Kamothe, Navi Mumbai – 410 209

University of Mumbai

Academic Year: 2021-2022

CERTIFICATE

This is to certify that the project entitled “**Exploratory Data Analysis on ABC News**” is a bonafide work of **Chakola Dereck Jos (118IT1153A), Harish Natarajan (118IT1400A), Mahadik Omkar Uday(119IT3251A)** submitted to the University of Mumbai in partial fulfillment of the requirement for the award of the degree of “**Undergraduate**” in “**Information Technology**”.

(Faculty In charge)
Prof. Manivannan

Table of Contents

CHAPTER 1 : - INTRODUCTION	7
1.1 Introduction	7
1.2 Scope	7
1.3 Project summary and purpose	8
1.4 Overview of the project	8
1.5 Problem definition	9
CHAPTER 2: - TECHNOLOGY AND LITERATURE REVIEW	10
2.1 Brief History of Work Done	10
CHAPTER 3: - SYSTEM REQUIREMENTS STUDY	11
3.1 User Characteristics	12
3.2 SOFTWARE AND HARDWARE REQUIREMENTS:	13
HARDWARE SPECIFICATIONS:	13
CHAPTER 4 :- SYSTEM ANALYSIS	14
4.1 Feasibility Study	14
4.2 Requirement Definition	16
4.3 CHALLENGES IN SENTIMENT ANALYSIS:	22
4.4 APPLICATIONS OF SENTIMENT ANALYSIS:	24
CHAPTER 5:- SYSTEM DESIGN AND ARCHITECTURE	26
5.1 Use Case Diagram	26
5.2 Package Diagram	27
5.3 Sequence Diagram	28
.....	28
CHAPTER 6 :- SYSTEM TESTING	29
Testing Strategies	29
6.1. Unit Testing	29
6.2. Integration Testing	29

6.2.1	Top Down Integration testing.....	29
6.2.2	Bottom-up Integration testing	29
6.3.	System Testing	29
6.4	Performance Testing	30
6.5	Acceptance Testing.....	30
6.5.1	Alpha Testing.....	30
6.5.2	Beta Testing	30
	Testing Methods.....	31
2	Black Box Testing	31
6.6	Verification and Validation:	31
CHAPTER 7 :- IMPLEMENTATION.....		32
7.1.	SNIPPETS	32
7.1.1.	Stopwords	32
7.1.2.	Non-stopword Visualizer	33
7.1.3.	Bi-gram plot	34
7.1.4.	Tri-gram plot.....	35
7.1.5.	LDA plot	36
7.1.6.	WordCloud plot	37
7.1.7.	Affin model plot	38
7.1.8.	Textblob model plot.....	39
7.1.9.	Vader model plot	40
7.1.10.	Model prediction.....	41
CHAPTER 8 :- CONCLUSION AND FUTURE WORK		42
8.1.	Conclusion.....	42
8.2.	Future Work	42
REFERENCES :		44

ACKNOWLEDGEMENT

The success and final outcome of this project required a lot of guidance and assistance from many people and we are extremely fortunate to have got this all along the completion of our project work. Whatever we have done is only due to such guidance and assistance and we would not forget to thank them.

It is matter of great pleasure for us to submit the project report on “Exploratory Data Analysis on ABC News”, as a part of our curriculum.

First and foremost, we would like to thank to our Director Ma’am Dr. Geeta S. Latkar , for giving us an opportunity to do the project work. We would like to thank our H.O.D Prof. Vijay Bhosale and subject in charge Prof. Manivannan, for the valuable guidance and advice. She inspired us greatly to work in this project. Her willingness to motivate us contributed tremendously to our project.

And last but not the least a special thanks goes to my team members, who helped me to assemble the information and gave suggestions to complete our project.

ABSTRACT

Exploratory data analysis (EDA) is an approach using descriptive statistics and graphical tools to better understand data. It is used mainly to maximize insight into a dataset, detect outliers and anomalies, and test underlying assumptions. It is a robust first step before the application of other statistical methods. It is commonly applied in all the fields of data mining, where it is particularly important to study the distribution of data and if relevant to subdivide them (e.g., typology or provenance study). This entry provides a broad insight to what EDA is and how useful it is for visualizing datasets, as illustrated in an example from mining. The project aims to provide a visual analytics of news dataset by means of stopwords plot, non-stopwords plot, Topic modelling i.e. classifying which news belongs to which topic, Word-cloud i.e. visualizing the most frequent words, Also Unsupervised Sentiment analysis model such as Affin Lexicon, Vader Lexicon and Textblob which uses word-list method were used to calculate the polarity and the sentiment of the given news headline as well as the text taken from user. By means of this system understanding the content of news would be much easier and faster

CHAPTER 1 :- INTRODUCTION

1.1 Introduction

Data analysis is the process of applying organized and systematic statistical techniques to describe, recap, check and condense data. It is a multistep process that involves collecting, cleaning, organizing and analyzing. Data mining is like applying techniques to mold data to suit our requirement. Data mining is needed because different sources like social media, transactions, public data, enterprises data etc. generates data of increasing volume, and it is important to handle and analyze such a big data. It won't be wrong to say that social media is something we live by. In the 21st century social media has been the game changer, be it advertising, politics or globalization, it has been estimated that data is increasing faster than before and by the year 2025; about 463 Billion GB/day of additional data will be generated each instant for each person on the earth. More data has been generated in the past two years than ever before in the history of the mankind. It is clear from the fact that the number of internet users are now grown from millions to billions.

Database which is opted for the proposed study is from ABC News. It is now day's very popular news service which provides information on various breaking news and headlines. In this people get to read various news generally less than 140 characters. It is appropriate for analysis as the number of news is large. The objective of the proposed analysis, 'Sentiment Analysis', is the analysis of the enormous amount of data easily available from news and Analyzing the headlines as well as identify the topic which the news belongs to.

Algorithm generates an overall sentiment score from the inputted topic in terms of positive, negative, complex or neutral, further it also works on finding the frequency of the words being used. Word cloud that is a pictorial representation of words based on frequency occurrence of words in the text is also generated. Calculation is actualized utilizing Python attributable to its component rich, thorough and expressive abilities for measurable information.

1.2 Scope

This project will be helpful to the companies, political parties as well as to the common people. It will be helpful to political party for understanding the attitude of people towards their program or facility which the party provides. Similarly, companies also can get review about their new product on newly released hardware or software. Also the movie maker can take review

on the currently running movie. It will also be helpful in analyzing all other fields such as Sports, Weather. By analyzing the news analyzer can get result on what topic the news belongs to, whether it is in favor of the citizens or not.

1.3 Project summary and purpose

This project of analyzing sentiments of news headlines comes under the domain of “*Pattern Classification*” and “*Data Mining*”. Both of these terms are very closely related and intertwined, and they can be formally defined as the process of discovering “useful” patterns in large set of data, either automatically (unsupervised). The project would heavily rely on techniques of “Natural Language Processing” in extracting significant patterns and features from the large data set of tweets and on “*Machine Learning*” techniques for accurately classifying individual un-labelled data samples according to whichever pattern model best describes them. The project also would use analytics technique to visualize which word in the headline contribute more and has more emphasis, what are the

1.4 Overview of the project

This proposal is a web application which is used to analyze the news headlines. We will be performing sentiment analysis in headlines and determine where it is positive, negative or neutral. This web application can be used by any organization office to review their works or by political leaders or by any others company to review about their products or brands. The main feature of our web application is that it helps to determine the opinion about the peoples on products, government work, politics or any other by analyzing the news and classifying the topic in which category it belongs.

The following graphs are presented after analysis :

- Top 10 stop-words bar chart
- Top 10 Non-stop-word bar chart
- Bigram and Trigram Bar Chart
- Word Cloud
- Sentiment Model Comparison (Affin Lexicon vs Textblob vs Vader)
- Topic modelling visualization

1.5 Problem definition

The algorithm proposed works on ABC news headlines, primarily it collects the news headline and then study it with the help of different statistical computing procedures.

The flow of our project is as follow: -

- The data is plotted to see top 10 stop words and Non-stopwords which is let us know the significant words in the headlines
- Bigram and Trigram plots are plotted to see which set of words occurs together.
- Word Cloud is plotted to know the most repeated word in the headlines
- LDA visualization is plotted to know which word contribute to a particular topic.

CHAPTER 2: - TECHNOLOGY AND LITERATURE REVIEW

2.1 Brief History of Work Done

2.1.1. SENTIMENT ANALYSIS

Sentiment analysis is extremely useful in social media monitoring as it allows us to gain an overview of the wider public opinion behind certain topics.

The applications of sentiment analysis are broad and powerful. The ability to extract insights from social data is a practice that is being widely adopted by organizations across the world. Shifts in sentiment on social media have been shown to correlate with shifts in the stock market.

The Obama administration used sentiment analysis to gauge public opinion to policy announcements and campaign messages ahead of 2012 presidential election. Being able to quickly see the sentiment behind everything from forum posts to news articles means being better able to strategize and plan for the future.

It can also be an essential part of your market research and customer service approach. Not only can you see what people think of your own products or services, you can see what they think about your competitors too. The overall customer experience of your users can be revealed quickly with sentiment analysis, but it can get far more granular too.

The ability to quickly understand consumer attitudes and react accordingly is something that Expedia Canada took advantage of when they noticed that there was a steady increase in negative feedback to the music used in one of their television adverts.

2.1.2. WORD CLOUD

Word clouds (also known as text clouds or tag clouds) work in a simple way: the more a specific word appears in a source of textual data (such as a speech, blog post, or database), the bigger and bolder it appears in the word cloud.

A word cloud is a collection, or cluster, of words depicted in different sizes. The bigger and bolder the word appears, the more often it's mentioned within a given text and the more important it is.

Also known as tag clouds or text clouds, these are ideal ways to pull out the most pertinent parts of textual data, from blog posts to databases. They can also help business users compare and contrast two different pieces of text to find the wording similarities between the two

2.1.3. STOPWORDS

The definition of what's a stop word may vary. You may consider a stop word a word that has high frequency on a corpus. Or you can consider every word that's empty of true meaning

given a context.

Words such as articles and some verbs are usually considered stop words because they don't help us to find the context or the true meaning of a sentence. These are words that can be removed without any negative consequences to the final model that you are training.

2.1.4. N-gram

In the fields of computational linguistics and probability, an n-gram is a contiguous sequence of n items from a given sample of text or speech. The items can be phonemes, syllables, letters, words or base pairs according to the application. The n-grams typically are collected from a text or speech corpus. When the items are words, n-grams may also be called shingles

- **Bi-gram:-**

A bigram or di-gram is a sequence of two adjacent elements from a string of tokens, which are typically letters, syllables, or words. A bigram is an n-gram for $n=2$. The frequency distribution of every bigram in a string is commonly used for simple statistical analysis of text in many applications, including in computational linguistics, cryptography, speech recognition, and so on.

- **Tri-gram:-**

Trigrams are a special case of the n-gram, where n is 3. They are often used in natural language processing for performing statistical analysis of texts and in cryptography for control and use of ciphers and codes

2.1.5. LDA

LDA is used to classify text in a document to a particular topic. It builds a topic per document model and words per topic model, modeled as Dirichlet distributions.

- Each document is modeled as a multinomial distribution of topics and each topic is modeled as a multinomial distribution of words.
- LDA assumes that the every chunk of text we feed into it will contain words that are somehow related. Therefore choosing the right corpus of data is crucial.
- It also assumes documents are produced from a mixture of topics. Those topics then generate words based on their probability distribution.

CHAPTER 3: - SYSTEM REQUIREMENTS STUDY

3.1 User Characteristics

Sentiment analysis can be defined as a process that automates mining of attitudes, opinions, views and emotions from text, speech, tweets and database sources through Natural Language Processing (NLP).

Sentiment analysis involves classifying opinions in text into categories like "positive" or "negative" or "neutral". It's also referred as subjectivity analysis, opinion mining, and appraisal extraction.

The words opinion, sentiment, view and belief are used interchangeably but there are differences between them.

- ☐ **Opinion:** A conclusion open to dispute (because different experts have different opinions)
- ☐ **View:** subjective opinion
- ☐ **Belief:** deliberate acceptance and intellectual assent

Sentiment: opinion representing one's feelings

Sentiment Analysis is a term that include many tasks such as sentiment extraction, sentiment classification, subjectivity classification, summarization of opinions or opinion spam detection, among others.

It aims to analyze people's sentiments, attitudes, opinions emotions, etc. towards elements such as, products, individuals, topics ,organizations, and services.

3.2 SOFTWARE AND HARDWARE REQUIREMENTS:

SOFTWARE REQUIREMENTS:-

Operating System: Windows 7/8/8.1/10

Microsoft Word (2016)

Pycharm (Python IDE)

Streamlit (Python Framework)

Jupyter Notebook (IPython IDE)

Notepad (For project requirements file)

HARDWARE SPECIFICATIONS:-

Processor	:	Intel i3 or more
Motherboard	:	Intel® Chipset Motherboard.
Ram	:	4GB or more

CHAPTER 4 :- SYSTEM ANALYSIS

4.1 Feasibility Study:

A feasibility study is a preliminary study which investigates the information of prospective users and determines the resources requirements, costs, benefits and feasibility of proposed system. A feasibility study takes into account various constraints within which the system should be implemented and operated. In this stage, the resource needed for the implementation such as computing equipment, manpower and costs are estimated. The estimated are compared with available resources and a cost benefit analysis of the system is made. The feasibility analysis activity involves the analysis of the problem and collection of all relevant information relating to the project. The main objectives of the feasibility study are to determine whether the project would be feasible in terms of economic feasibility, technical feasibility and operational feasibility and schedule feasibility or not. It is to make sure that the input data which are required for the project are available. Thus we evaluated the feasibility of the system in terms of the following categories:

- ☐ Technical feasibility
- ☐ Operational feasibility
- ☐ Economic feasibility
- ☐ Schedule feasibility

4.1.1. Technical Feasibility

Evaluating the technical feasibility is the trickiest part of a feasibility study. This is because, at the point in time there is no any detailed designed of the system, making it difficult to access issues like performance, costs (on account of the kind of technology to be deployed) etc. A number of issues have to be considered while doing a technical analysis; understand the different technologies involved in the proposed system. Before commencing the project, we have to be very clear about what are the technologies that are to be required for the development of the new system. Is the required technology available? Our system is technically feasible since all the required tools are easily available. PyCharm and Streamlit makes the system more user and developer friendly and although all tools seem to be easily available there are challenges too.

4.1.2. Operational Feasibility

Proposed project is beneficial only if it can be turned into information systems that will meet the operating requirements. Simply stated, this test of feasibility asks if the system will work when it is developed and installed. Are there major barriers to Implementation? The proposed was to make a simplified web application. It is simpler to operate and can be used in any webpages. It is free and not costly to operate.

4.1.3. Economic Feasibility

Economic feasibility attempts to weigh the costs of developing and implementing a new system, against the benefits that would accrue from having the new system in place. This

feasibility study gives the top management the economic justification for the new system. A simple economic analysis which gives the actual comparison of costs and benefits are much more meaningful in this case. In addition, this proves to be useful point of reference to compare actual costs as the project progresses. There could be various types of intangible benefits on account of automation. These could increase improvement in product quality, better decision making, and timeliness of information, expediting activities, improved accuracy of operations, better documentation and record keeping, faster retrieval of information. This is a web based application. Creation of application is not costly.

4.1.4. Schedule Feasibility

A project will fail if it takes too long to be completed before it is useful. Typically, this means estimating how long the system will take to develop, and if it can be completed in a given period of time using some methods like payback period. Schedule feasibility is a measure how reasonable the project timetable is. Given our technical expertise, are the project deadlines reasonable? Some project is initiated with specific deadlines. It is necessary to determine whether the deadlines are mandatory or desirable.

A minor deviation can be encountered in the original schedule decided at the beginning of the project. The application development is feasible in terms of schedule.

4.2 Requirement Definition:

After the extensive analysis of the problems in the system, we are familiarized with the requirement that the current system needs. The requirement that the system needs is categorized into the functional and non-functional requirements. These requirements are listed below:

4.2.1. Functional Requirements

Functional requirement are the functions or features that must be included in any system to satisfy the business needs and be acceptable to the users.

Based on this, the functional requirements that the system must require are as follows:

- System should be able to process new text the user types
- System should be able to analyze headlines and classify them into topics

4.2.2. Non-Functional Requirements

Non-functional requirements is a description of features, characteristics and attribute of the system as well as any constraints that may limit the boundaries of the proposed system.

The non-functional requirements are essentially based on the performance, information, economy, control and security efficiency and services.

Based on these the non-functional requirements are as follows:

- ❖ User friendly
- ❖ -System should provide better accuracy
- ❖ -To perform with efficient throughput and response time

4.2.3. Study of Current System

There are primarily three types of approaches for sentiment classification of opinionated texts:

- ❖ Using a Machine learning based text classifier such as Naive Bayes
- ❖ Using Natural Language Processing
- ❖ Using Word-list sentiment analysis (Un-supervised learning)

We will be using those Un-supervised learning and natural language processing for sentiment analysis of tweets.

4.2.3.1. Word-list Sentiment analysis (Un-supervised Method)

The word-list based text classifiers are a kind of un-supervised machine learning paradigm, where the classifier is already trained on some labeled training data before it can be applied to actual classification task. The training data is usually an extracted portion of the original data hand labelled manually. After suitable training they can be used on the actual test data. The Textblob algorithm and Affin Lexicon are purely word list based trained only on words whereas Vader Lexicon is a kind of model which was also trained to detect complex sentiments as well as emoticons.

VADER LEXICON

The Vader model was trained on complex word list which would output valence score based on emojis, punctuations and word-list to classify whether the sentiment is positive, negative, neutral or complex.

TEXTBLOB

TextBlob is a simple unsupervised algorithm which supports complex analysis and operations on textual data. For lexicon-based approaches, a sentiment is defined by its semantic orientation and the intensity of each word in the sentence. This requires a pre-defined dictionary classifying negative and positive words.

There are various training sets available on Internet such as Movie Reviews data set, twitter dataset, etc. Class can be Positive, negative. For both the classes we need training data sets.

4.2.3.2. PYTHON LANGUAGE

Python is an interpreted high-level general-purpose programming language. Python's design philosophy emphasizes code readability with its notable use of significant indentation. Its language constructs as well as its object-oriented approach aim to help programmers write clear, logical code for small and large-scale projects. Python is dynamically-typed and garbage-collected. It supports multiple programming paradigms, including structured (particularly, procedural), object-oriented and functional programming. Python is often described as a "batteries included" language

due to its comprehensive standard library.

Guido van Rossum began working on Python in the late 1980s, as a successor to the ABC programming language, and first released it in 1991 as Python 0.9.0. Python 2.0 was released in 2000 and introduced new features, such as list comprehensions and a garbage collection system using reference counting and was discontinued with version 2.7.18 in 2020. Python 3.0 was released in 2008 and was a major revision of the language that is not completely backward-compatible and much Python 2 code does not run unmodified on Python 3. Python consistently ranks as one of the most popular programming languages.

4.2.3.3. Features of Python Language

1. Easy to code:

Python is a high-level programming language. Python is very easy to learn the language as compared to other languages like C, C#, Javascript, Java, etc. It is very easy to code in python language and anybody can learn python basics in a few hours or days. It is also a developer-friendly language.

2. Free and Open Source:

Python language is freely available at the official website and you can download it from the given download link below click on the Download Python keyword.

Download Python

Since it is open-source, this means that source code is also available to the public. So you can download it as, use it as well as share it.

3. Object-Oriented Language:

One of the key features of python is Object-Oriented programming. Python supports object-oriented language and concepts of classes, objects encapsulation, etc.

4. GUI Programming Support:

Graphical User interfaces can be made using a module such as PyQt5, PyQt4, wxPython, or Tk in python.

PyQt5 is the most popular option for creating graphical apps with Python.

5. High-Level Language:

Python is a high-level language. When we write programs in python, we do not need to remember the system architecture, nor do we need to manage the memory

4.2.3.4. Streamlit :-

The fastest way to build and share data apps. Streamlit lets you turn data scripts into sharable web apps in minutes, not weeks. It's all Python, open-source, and free! And once you've created an app you can use our free sharing platform to deploy, manage, and share your app with the world.

Accessible app making for everyone (who uses Python). This is the main draw, since it can save time by allowing one to focus on the data science aspect, and also suits those who may not want to learn HTML/CSS. The learning curve is fairly flat.

Cover most common UIs needed in a data app. Plus, they look good! It contains slider, checkbox, radio buttons, a collapsible side bar, progress bar, file upload, etc. Overall these functionalities and the ease of use are impressive. It would be even nicer if can have an information button next to certain components to offer further explanations.

Support multiple interactive visualization libraries. It supports libraries such as plotly, altair, bokeh, Vega-Lite, and pydeck.

4.2.3.5. Pycharm :-

The PyCharm IDE is one of the most popular editors used by professional Python developers and programmers.

The PyCharm IDE is one of the most popular editors used by professional Python developers and programmers. The vast number of PyCharm features doesn't make this IDE difficult to use—just the opposite. Many of the features help make Pycharm a great Python IDE for beginners.

4.2.3.6. Matplotlib:-

Matplotlib is a plotting library for the Python programming language and its numerical mathematics extension NumPy. It provides an object-oriented API for embedding plots into applications using general-purpose GUI toolkits like Tkinter, wxPython, Qt, or GTK

4.2.3.7. TextBlob:-

TextBlob is a python library for Natural Language Processing (NLP).TextBlob actively used Natural Language ToolKit (NLTK) to achieve its tasks. NLTK is a library which gives an easy access to a lot of lexical resources and allows users to work with categorization, classification and many other tasks. TextBlob is a simple library which supports complex analysis and operations on textual data.

For lexicon-based approaches, a sentiment is defined by its semantic orientation and the intensity of each word in the sentence. This requires a pre-defined dictionary classifying negative and positive words. Generally, a text message will be represented by bag of words. After assigning individual scores to all the words, final sentiment is calculated by some pooling operation like taking an average of all the sentiments.

TextBlob returns polarity and subjectivity of a sentence. Polarity lies between [-1,1], -1 defines a negative sentiment and 1 defines a positive sentiment. Negation words reverse the polarity. TextBlob has semantic labels that help with fine-grained analysis. For example — emoticons, exclamation mark, emojis, etc. Subjectivity lies between [0,1]. Subjectivity quantifies the amount of personal opinion and factual information contained in the text. The higher subjectivity means that the text contains personal opinion rather than factual information. TextBlob has one more parameter — intensity. TextBlob calculates subjectivity by looking at the 'intensity'.

4.2.3.8. Affin Lexicon:

Affin Lexicon is a python library for Natural Language Processing (NLP). Affin is a simple library which supports complex analysis and operations on textual data.

For lexicon-based approaches, a sentiment is defined by its semantic orientation and the intensity of each word in the sentence. This requires a pre-defined dictionary classifying negative and positive words. After assigning individual scores to all the words, final sentiment is calculated by some pooling operation like taking an average of all the sentiments.

Affin returns polarity and subjectivity of a sentence. Polarity lies between $[-\infty, +\infty]$, $-\infty$ defines a negative sentiment and $+\infty$ defines a positive sentiment. Negation words reverse the polarity. Affin has semantic labels that help with fine-grained analysis. For example — emoticons, exclamation mark, emojis, etc. Subjectivity lies between $[0,1]$. Subjectivity quantifies the amount of personal opinion and factual information contained in the text. The higher subjectivity means that the text contains personal opinion rather than factual information. TextBlob has one more parameter — intensity. TextBlob calculates subjectivity by looking at the ‘intensity’. Intensity determines if a word modifies the next word.

4.2.3.9. Vader Lexicon:

VADER makes use of certain rules to incorporate the impact of each sub-text on the perceived intensity of sentiment in sentence-level text. These rules are called Heuristics. There are 5 of them.

VADER relies on a dictionary that maps words and other numerous lexical features common to sentiment expression in microblogs.

These features include:

- ❖ A full list of Western-style emoticons (for example - :D and :P)
- ❖ Sentiment-related acronyms (for example - LOL and ROFL)
- ❖ Commonly used slang with sentiment value (for example - Nah and meh)

Five Heuristics are explained below: -

- **Punctuation**, namely the exclamation point (!), increases the magnitude of the intensity without modifying the semantic orientation. For example: “The weather is hot!!!” is more intense than “The weather is hot.”
- **Capitalization**, specifically using ALL-CAPS to emphasize a sentiment-relevant word in the presence of other non-capitalized words, increases the magnitude of the sentiment intensity without affecting the semantic orientation. For example: “The weather is HOT.” conveys more intensity than “The weather is hot.”
- **Degree modifiers** (also called intensifiers, booster words, or degree adverbs) impact sentiment intensity by either increasing or decreasing the intensity. For example: “The weather is extremely hot.” is more intense than “The weather is hot.”, whereas “The weather is slightly hot.” reduces the intensity.

- **Polarity shift due to Conjunctions**, The contrastive conjunction “but” signals a shift in sentiment polarity, with the sentiment of the text following the conjunction being dominant. For example: “The weather is hot, but it is bearable.” has mixed sentiment, with the latter half dictating the overall rating.
- **Catching Polarity Negation**, By examining the contiguous sequence of 3 items preceding a sentiment-laden lexical feature, we catch nearly 90% of cases where negation flips the polarity of the text. For example a negated sentence would be “The weather isn't really that hot.”.

Algorithm :

i. Dictionary generation

The compound score is computed by summing the valence scores of each word in the lexicon, adjusted according to the rules, and then normalized to be between -1 (most extreme negative) and +1 (most extreme positive).

This is the most useful metric if you want a single unidimensional measure of sentiment for a given sentence.

As explained in the paper, researchers used below normalization.

$$x = \frac{x}{\sqrt{x^2 + \alpha}}$$

where x = sum of valence scores of constituent words, and α = Normalization constant (default value is 15)

Testing Goal :-

- Finding the sentiment of given test data file.
- For each document in test set find polarity score and classify Whether sentiment is positive, negative, neutral or complex

4.2.3.10. Natural Language Processing

Natural language processing (NLP) is a field of computer science, artificial intelligence, and linguistics concerned with the interactions between computers and human (natural) languages. This approach utilizes the publicly available library of Opinion Lexicon, which provides a sentiment polarity values for every term occurring in the document. In this lexical resource each term t occurring in Opinion Lexicon is associated to three numerical scores $\text{obj}(t)$, $\text{pos}(t)$ and $\text{neg}(t)$, describing the objective, positive and negative polarities of the term, respectively. These three scores are computed by combining the results produced by eight ternary classifiers. Opinion Lexicon is a large lexical database of English. Nouns, verbs, adjectives and adverbs are grouped into sets of cognitive synonyms, each expressing a distinct concept. Opinion Lexicon is also freely and publicly available for download. Opinion Lexicon's structure makes it a useful tool for computational linguistics and natural language processing. It groups words together based on their meanings. It is nothing but a set of one or more Synonyms. This approach uses Semantics to understand the language.

Major tasks in NLP that helps in extracting sentiment from a sentence:

- ☐ Extracting part of the sentence that reflects the sentiment
- ☐ Understanding the structure of the sentence

Different tools which help process the textual data Basically, Positive and Negative scores got from Opinion Lexicon according to its part-of- speech tag and then by counting the total positive and negative scores we determine the sentiment polarity based on which class (i.e. either positive or negative) has received the highest score.

4.3 CHALLENGES IN SENTIMENT ANALYSIS:

Sentiment Analysis is a very challenging task. Following are some of the challenges faced in Sentiment Analysis of Twitter.

4.3.1. Identifying subjective parts of text:

Subjective parts represent sentiment-bearing content. The same word can be treated as subjective in one case, or an objective in some other. This makes it difficult to identify the subjective portions of text.

Example:

1. The language of the Mr Dennis was very crude.
2. Crude oil is obtained by extraction from the sea beds.

The word „crude“ is used as an opinion in first example, while it is completely objective in the second example.

4.3.2. Domain dependence:

The same sentence or phrase can have different meanings in different domains.

Example:

The word “unpredictable” is positive in the domain of movies, dramas, etc, but if the same word is used in the context of a vehicle's steering, then it has a negative opinion.

4.3.3. Sarcasm Detection:

Sarcastic sentences express negative opinion about a target using positive words in unique way.

Example:

“Nice perfume. You must shower in it.”

The sentence contains only positive words but actually it expresses a negative sentiment.

4.3.4. Thwarted expressions:

There are some sentences in which only some part of text determines the overall polarity of the document.

Example:

“This Movie should be amazing. It sounds like a great plot, the popular actors, and the supporting cast is talented as well. “

In this case, a simple bag-of-words approach will term it as positive sentiment, but the ultimate sentiment is negative.

4.3.5. Explicit Negation of sentiment:

Sentiment can be negated in many ways as opposed to using simple no, not, never, etc. It is difficult to identify such negations.

Example:

“It avoids all suspense and predictability found in Hollywood movies.”

Here the words suspense and predictable bear a negative sentiment, the usage of „avoids“ negates their respective sentiments.

4.3.6. Order dependence:

Discourse Structure analysis is essential for Sentiment Analysis/Opinion Mining.

Example:

A is better than B, conveys the exact opposite opinion from, B is better than A.

4.3.7. Entity Recognition:

There is a need to separate out the text about a specific entity and then analyze sentiment towards it.

Example:

“I hate Microsoft, but I like Linux”.

A simple bag-of-words approach will label it as neutral, however, it carries a specific sentiment for both the entities present in the statement.

4.3.8. Building a classifier for subjective vs. objective tweets.

Current research work focuses mostly on classifying positive vs. negative correctly. There is need to look at classifying tweets with sentiment vs. no sentiment closely.

4.3.9. Handling comparisons.

Bag of words model doesn't handle comparisons very well.

Example:

"IIT's are better than most of the private colleges", the tweet would be considered positive for both IIT's and private colleges using bag of words model because it doesn't take into account the relation towards "better".

4.3.10. Applying sentiment analysis to Facebook messages.

There has been less work on sentiment analysis on Facebook data mainly due to various restrictions by Facebook graph API and security policies in accessing data.

4.3.11. Internationalization

Current Research work focus mainly on English content, but Twitter has many varied users from across.

4.4 APPLICATIONS OF SENTIMENT ANALYSIS:

Sentiment Analysis has many applications in various Fields.

4.4.1. Applications that use Reviews from Websites:

Today Internet has a large collection of reviews and feedbacks on almost everything. This includes product reviews, feedbacks on political issues, comments about services, etc. Thus there is a need for a sentiment analysis system that can extract sentiments about a particular product or services. It will help us to automate in provision of feedback or rating for the given product, item, etc. This would serve the needs of both the users and the vendors.

4.4.2. Applications as a Sub-Component Technology

A sentiment predictor system can be helpful in recommender systems as well. The recommender system will not recommend items that receive a lot of negative feedback or few ratings.

In online communication, we come across abusive language and other negative elements. These can be detected simply by identifying a highly negative sentiment and correspondingly taking action against it.

4.4.3. Applications in Business Intelligence

It has been observed that people nowadays tend to look upon reviews of products which are available online before they buy them. And for many businesses, the online opinion decides the success or failure of their product. Thus, Sentiment Analysis plays an important role in businesses. Businesses also wish to extract sentiment from the online reviews in order to improve their products and in turn their reputation and help in customer satisfaction.

4.4.4. Applications across Domains:

Recent researches in sociology and other fields like medical, sports have also been benefitted by Sentiment Analysis that show trends in human emotions especially on social media.

4.4.5. Applications in Smart Homes:

Smart homes are supposed to be the technology of the future. In future entire homes would be networked and people would be able to control any part of the home using a tablet device. Recently there has been a lot of research going on Internet of Things (IoT). Sentiment Analysis would also find its way in IoT. Like for example, based on the current sentiment or emotion of the user, the home could alter its ambiance to create a soothing and peaceful environment.

Sentiment Analysis can also be used in trend prediction. By tracking public views, important data regarding sales trends and customer satisfaction can be extracted.

CHAPTER 5:- SYSTEM DESIGN AND ARCHITECTURE

5.1 Use Case Diagram

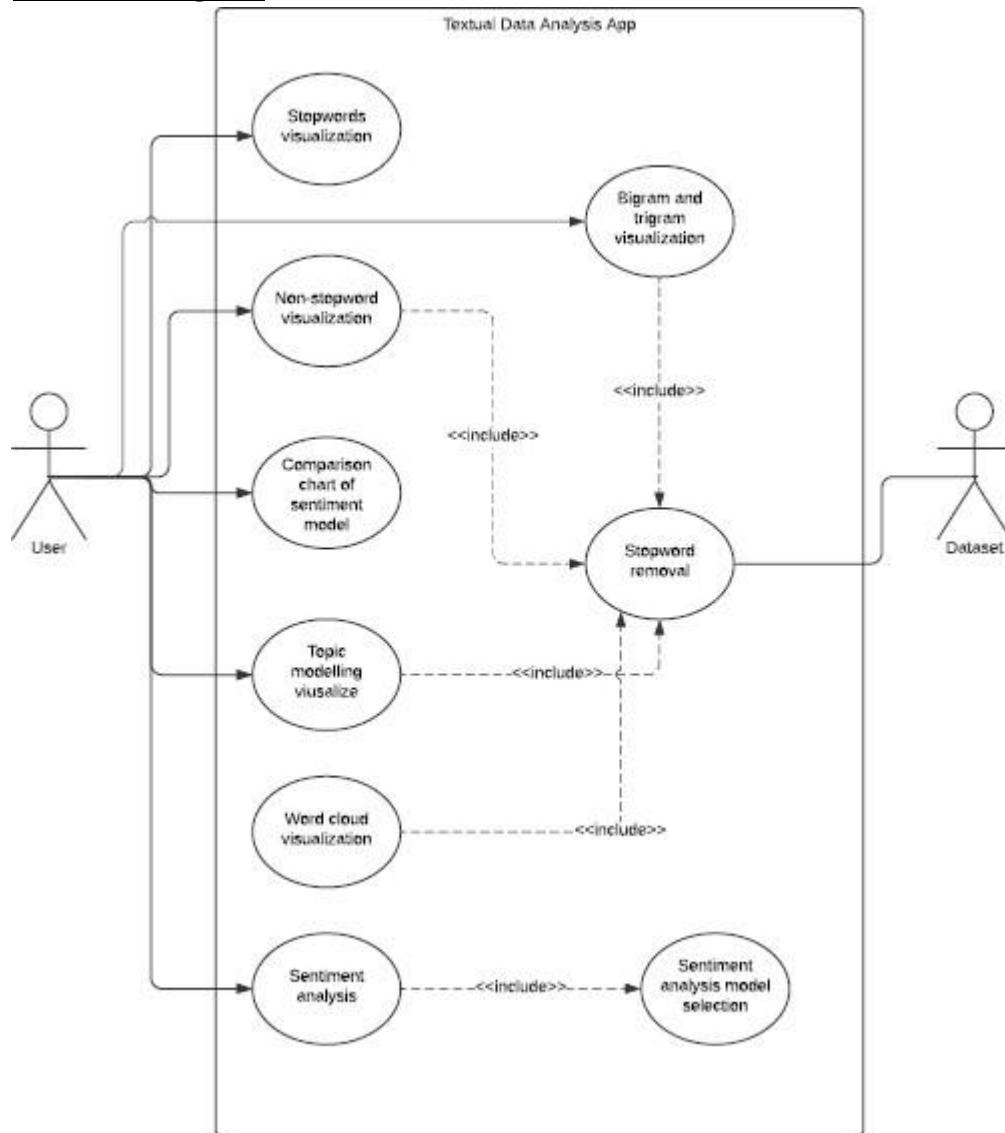


Fig 5.1 Use Case Diagram for Sentiment Analysis using Twitter API

5.2 Package Diagram

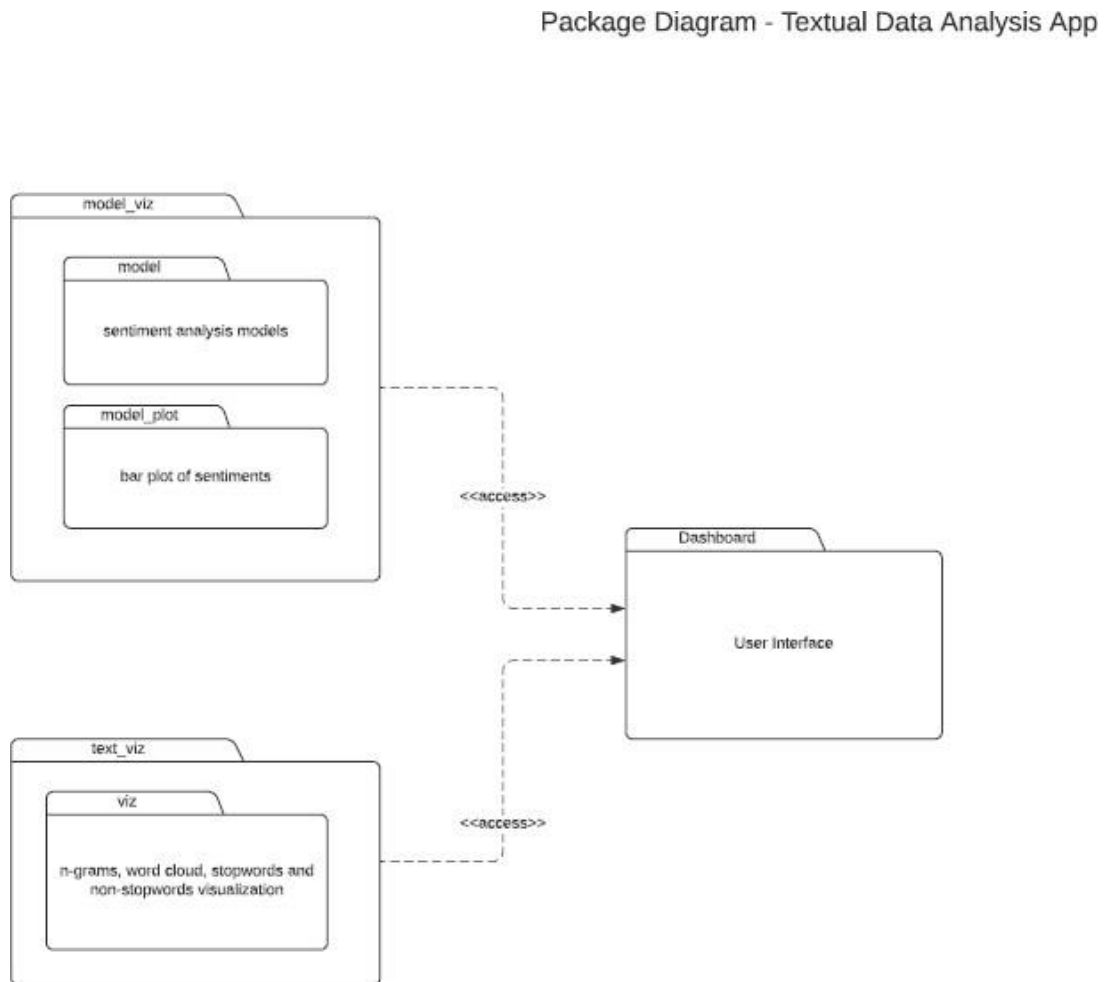


Fig 5.2 Package Diagram for Text Analytics and Sentiment Analysis

5.3 Sequence Diagram

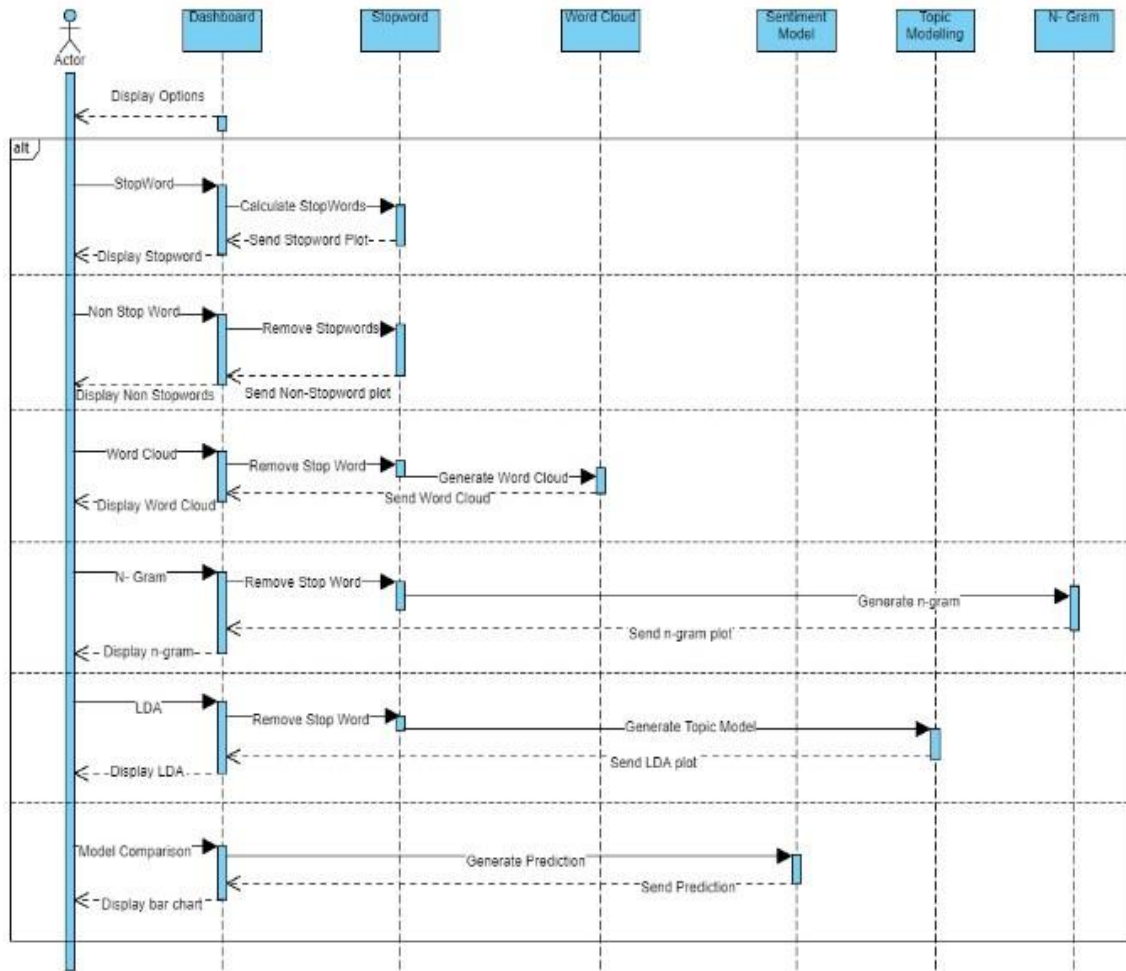


Fig 5.3 Sequence Diagram for Text Analytics and Sentiment Analysis

CHAPTER 6 :- SYSTEM TESTING

Testing is the process of evaluating a system or its component's with the intent to find that whether it satisfies the specified requirements or not .This activity results in the actual, expected and difference between their results i.e testing is executing a system in order to identify any gaps, errors or missing requirements in contrary to the actual desire or requirements.

Testing Strategies

In order to make sure that system does not have any errors, the different levels of testing strategies that are applied at different phases of software development are

6.1. Unit Testing:

Unit testing is performed for testing modules against detailed design. Inputs to the process are usually compiled modules from the coding process. Each modules are assembled into a larger unit during the unit testing process. Testing has been performed on each phase of project design and coding. We carry out the testing of module interface to ensure the proper flow of information into and out of the program unit while testing. We make sure that the temporarily stored data maintains its integrity throughout the algorithm's execution by examining the local data structure. Finally, all error-handling paths are also tested.

6.2. Integration Testing:

The testing of combined parts of an application to determine if they function correctly together is Integration testing .This testing can be done by using two different methods

6.2.1 Top Down Integration testing

In Top-Down integration testing, the highest-level modules are tested first and then progressively lower-level modules are tested.

6.2.2 Bottom-up Integration testing

Testing can be performed starting from smallest and lowest level modules and proceeding one at a time .When bottom level modules are tested attention turns to those on the next level that use the lower level ones they are tested individually and then linked with the previously examined lower level modules. In a comprehensive software development environment, bottom-up testing is usually done first, followed by top-down testing.

6.3. System Testing:

We usually perform system testing to find errors resulting from unanticipated interaction between the sub-system and system components. Software must be tested to detect and rectify all possible errors once the source code is generated before

delivering it to the customers. For finding errors, series of test cases must be developed which ultimately uncover all the possibly existing errors. Different software techniques can be used for this process. These techniques provide systematic guidance for designing test that Exercise the internal logic of the software components, Exercise the input and output domains of a program to uncover errors in program function, behavior and performance. We test the software using two methods: White Box testing: Internal program logic is exercised using this test case design techniques. Black Box testing: Software requirements are exercised using this test case design techniques. Both techniques help in finding maximum number of errors with minimal effort and time.

6.4 Performance Testing:

It is done to test the run-time performance of the software within the context of integrated system. These tests are carried out throughout the testing process. For example, the performance of individual module is accessed during white box testing under unit testing.

6.5 Acceptance Testing

The main purpose of this Testing is to find whether application meets the intended specifications and satisfies the client's requirements. We will follow two different methods in this testing.

6.5.1 Alpha Testing

*This test is the first stage of testing and will be performed amongst the teams. Unit testing, integration testing and system testing when combined are known as alpha testing. During this phase, the following will be tested in the application:

- ☐ Spelling Mistakes.
- ☐ Broken Links.

The Application will be tested on machines with the lowest specification to test loading times and any latency problems.

6.5.2 Beta Testing

In beta testing, a sample of the intended audience tests the application and send their feedback to the project team .Getting the feedback, the project team can fix the problems before releasing the software to the actual users.

Testing Methods:

1 White Box Testing

White box testing is the detailed investigation of internal logic and structure of the Code. To perform white box testing on an application, the tester needs to possess knowledge of the internal working of the code .The tester needs to have a look inside the source code and find out which unit/chunk of the code is behaving inappropriately.

2 Black Box Testing

The technique of testing without having any knowledge of the interior workings of the application is Black Box testing .The tester is oblivious to the system architecture and does not have access to the source code. Typically, when performing a black box test, a tester will interact with the system's user interface by providing inputs and examining outputs without knowing how and where the inputs are worked upon.

6.6 Verification and Validation:

The testing process is a part of broader subject referring to verification and validation. We have to acknowledge the system specifications and try to meet the customer's requirements and for this sole purpose, we have to verify and validate the product to make sure everything is in place. Verification and validation are two different things. One is performed to ensure that the software correctly implements a specific functionality and other is done to ensure if the customer requirements are properly met or not by the end product. Verification is more like 'are we building the product right?' and validation is more like 'are we building the right product?'

CHAPTER 7 :- IMPLEMENTATION

7.1. SNIPPETS

7.1.1. Stopwords

Top 10 stop words

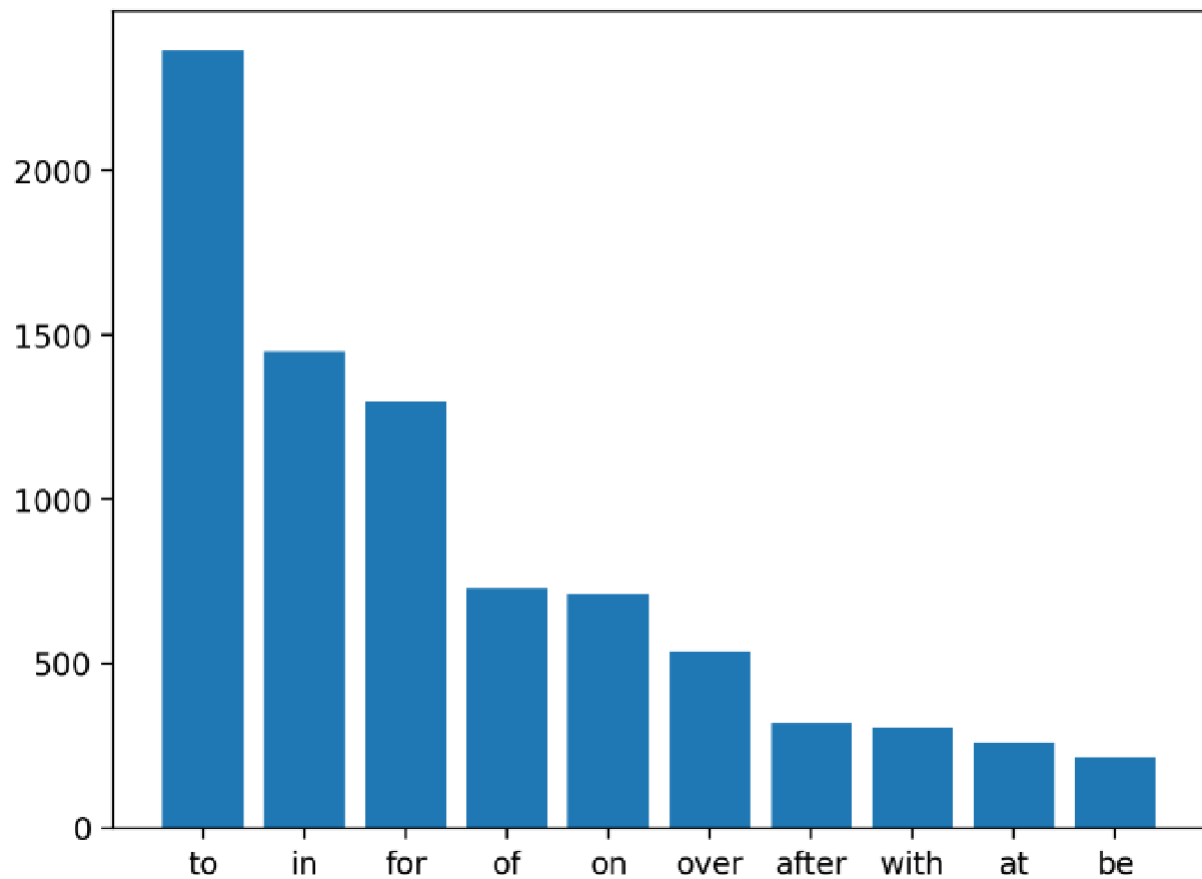


FIG 7.1.1. STOPWORD VISUALIZER

7.1.2. Non-stopword Visualizer

Top 10 non stop words

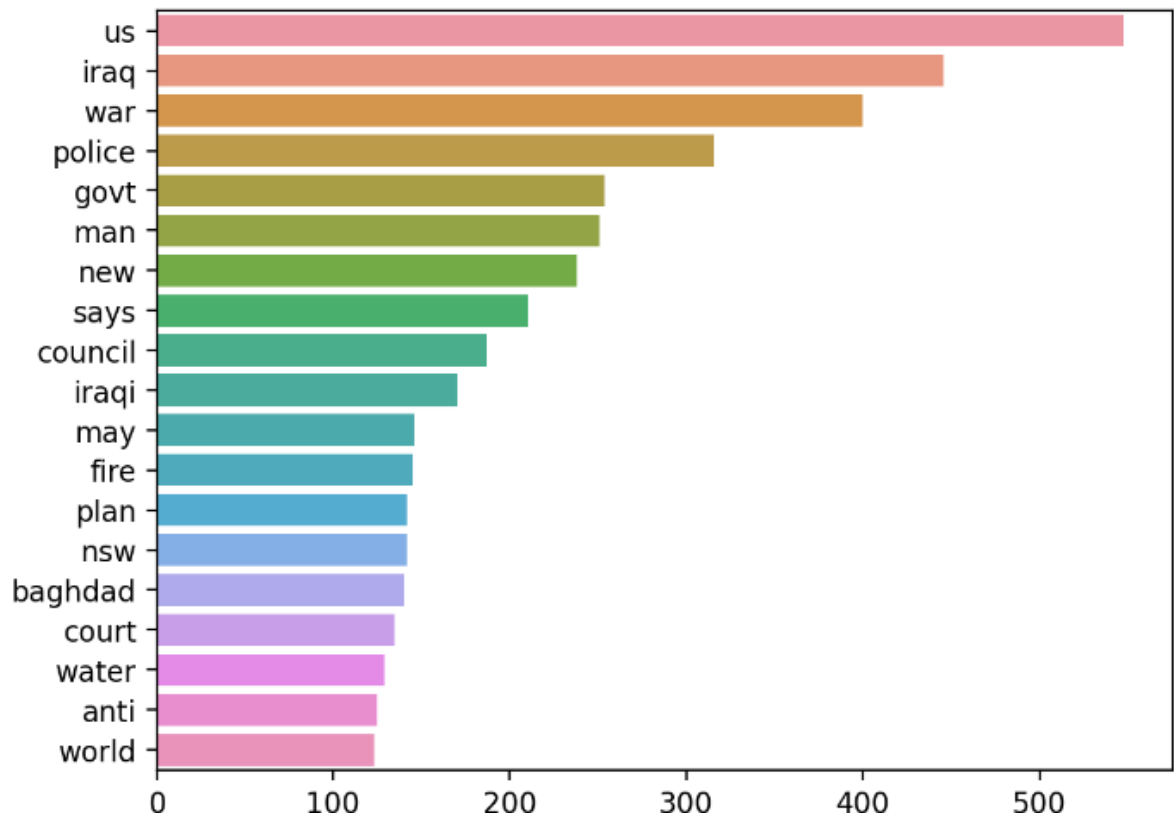


FIG 7.1.2. Non-stopword Visualizer

7.1.3. Bi-gram plot

Bigram Plot

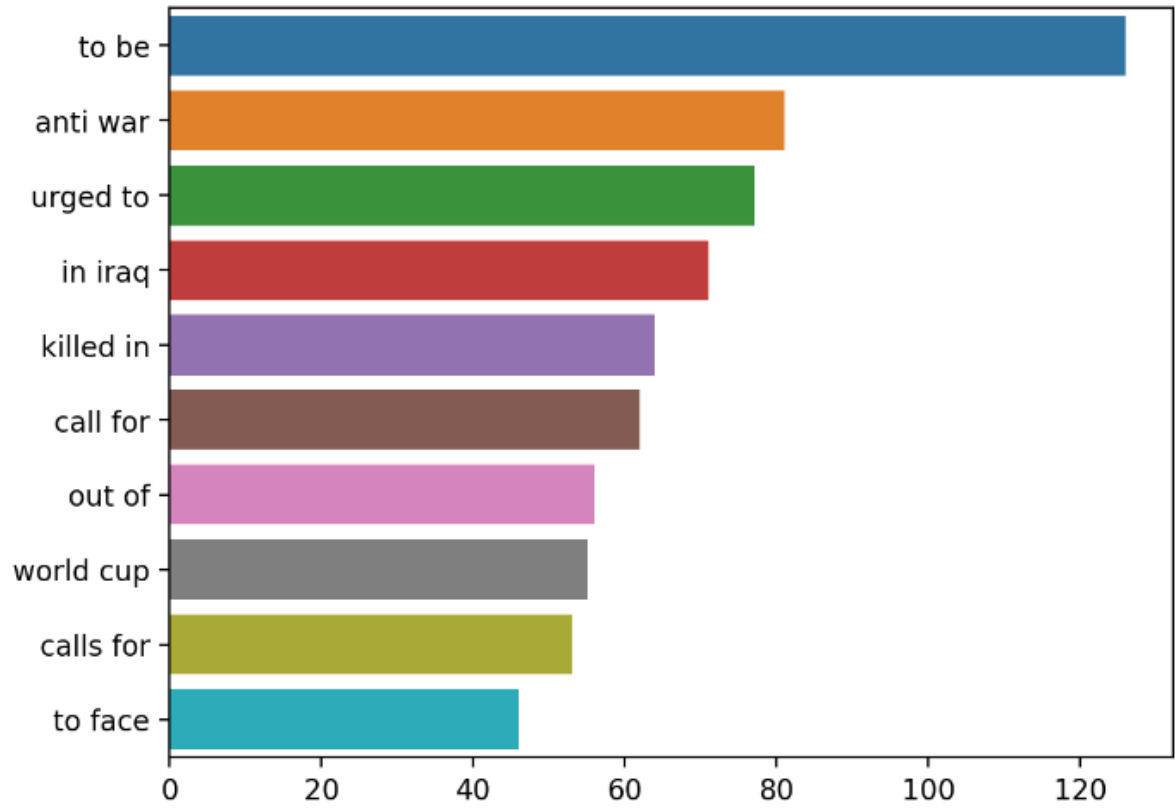


FIG 7.1.3. Bi-gram plot

7.1.4. Tri-gram plot

Trigram Plot

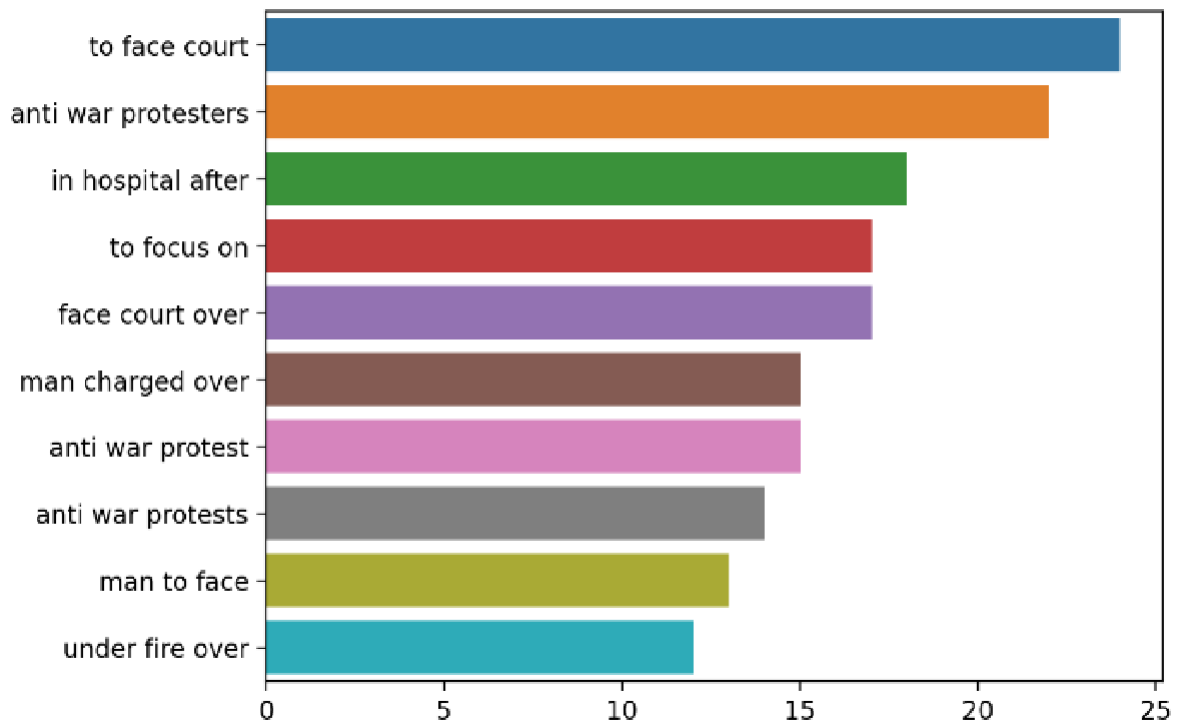


FIG 7.1.4. Tri-gram plot

7.1.5. LDA plot

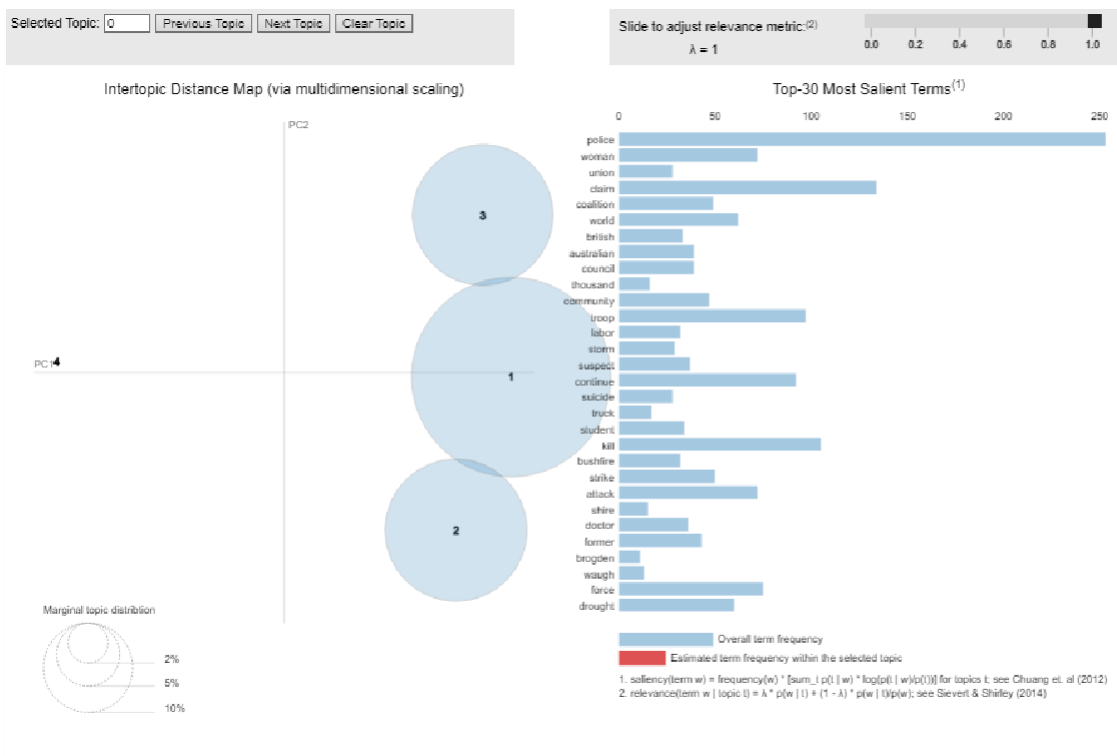


FIG 7.1.5. LDA plot (This plot shows which word contribute to particular topic)

7.1.6. WordCloud plot

Word Cloud

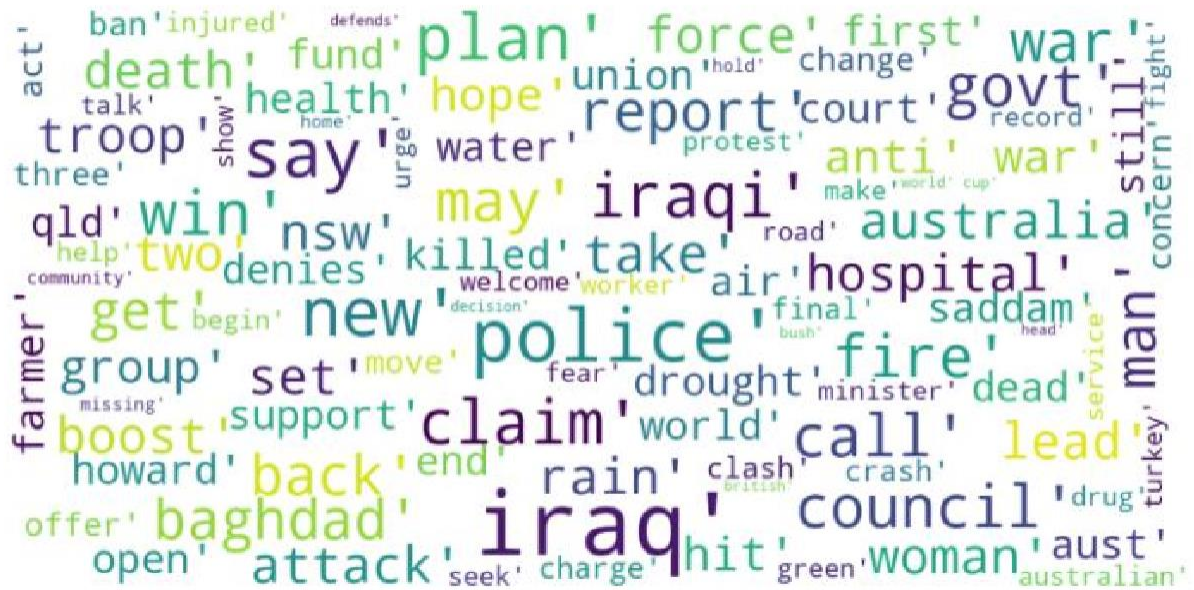


FIG 7.1.6. Word Cloud

7.1.7. Affin model plot

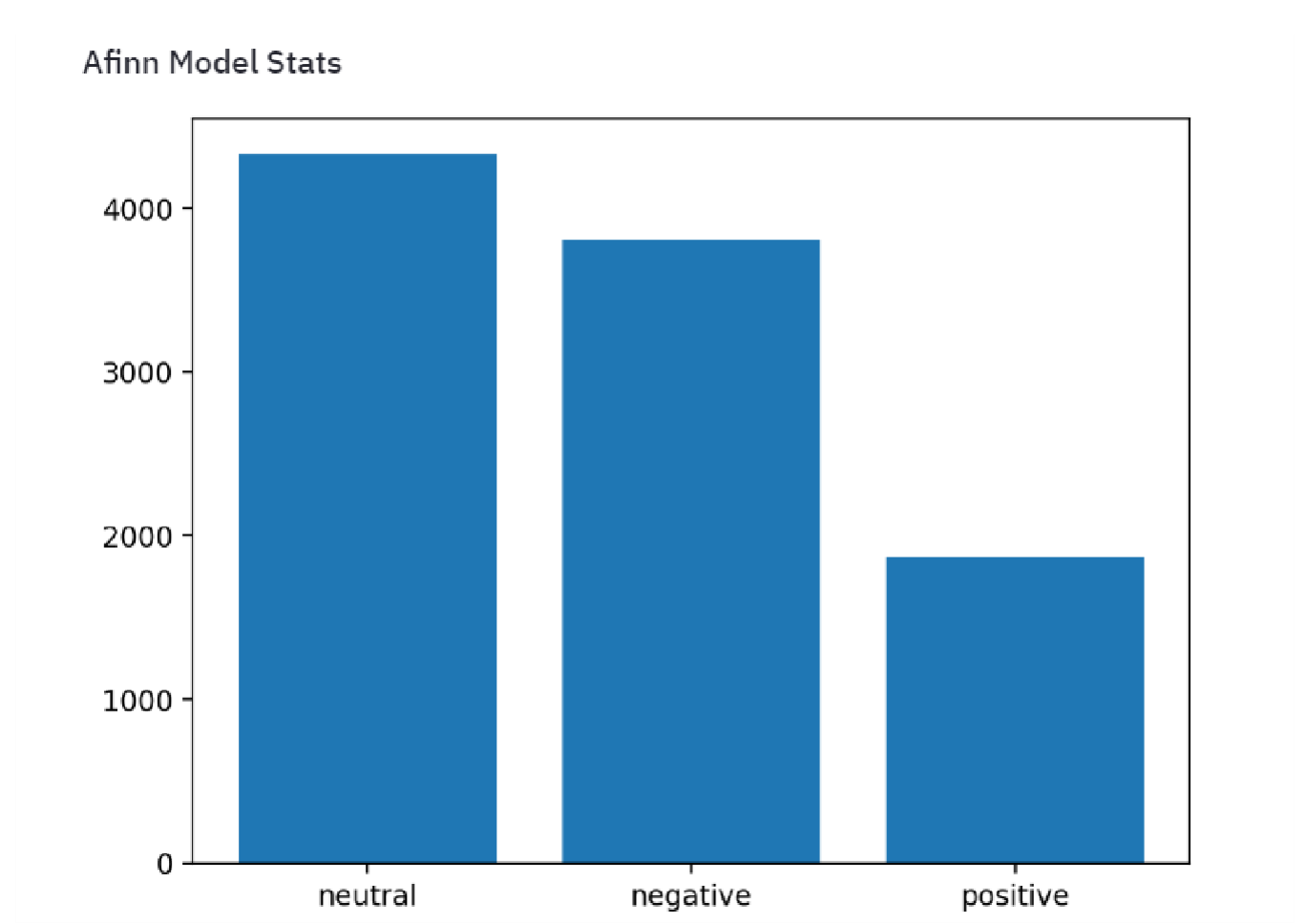


FIG 7.1.7. Comparison of Affin model plot

7.1.8. Textblob model plot

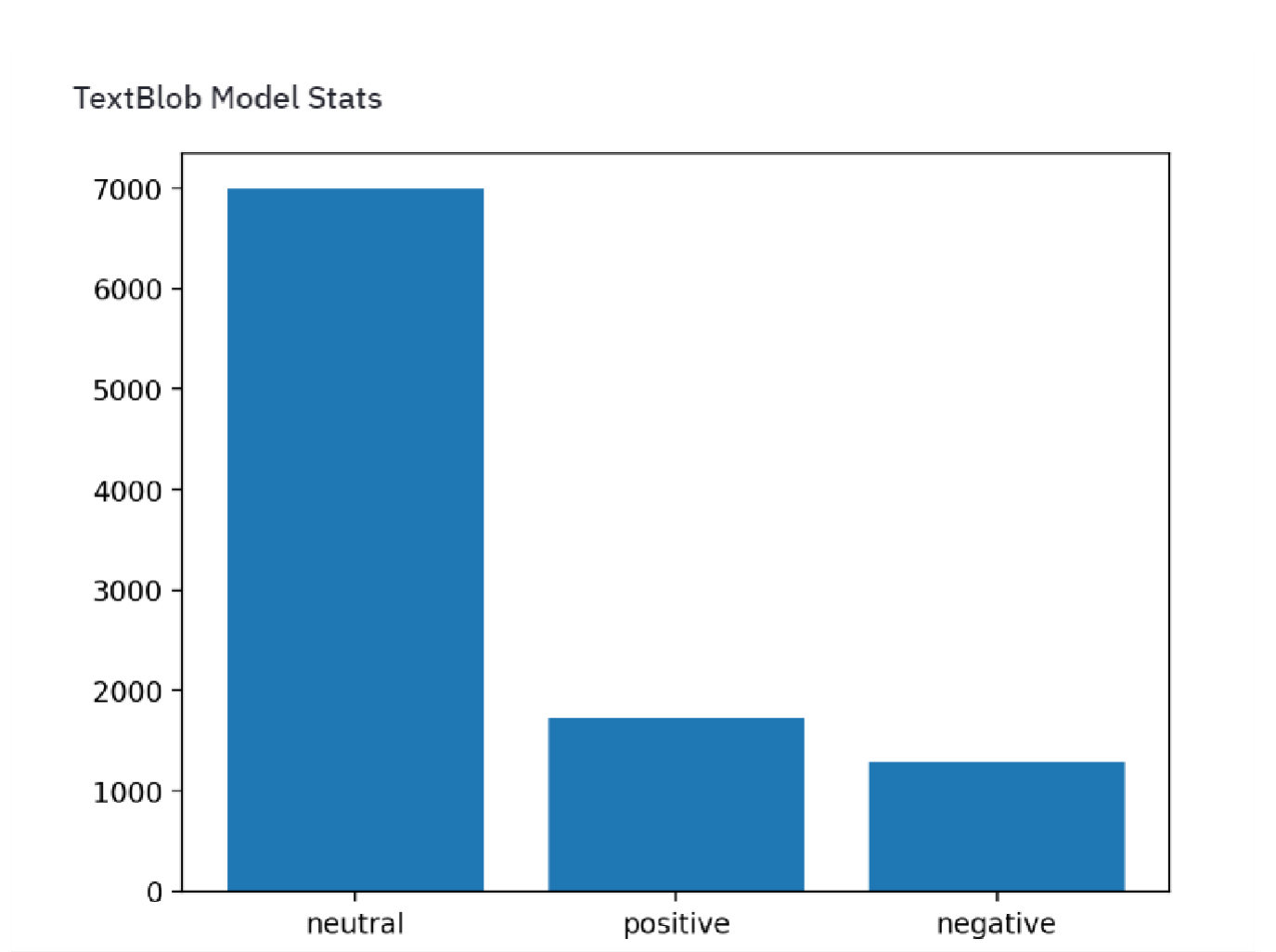


FIG 7.1.8. Textblob comparison plot

7.1.9. Vader model plot

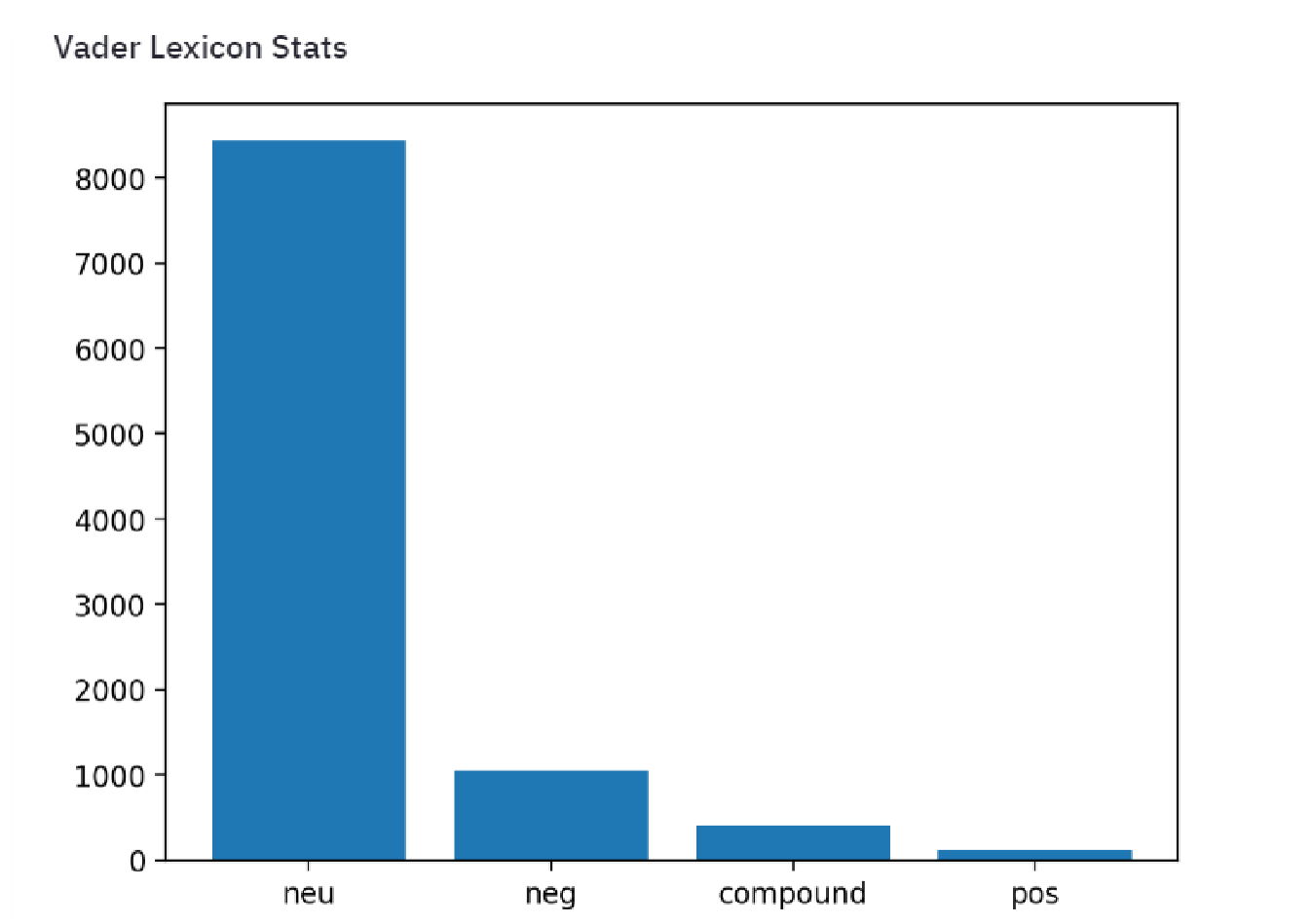


FIG 7.1.9. Vader model comparison plot

7.1.10. Model prediction

Text Analytics

Choose an option

Unsupervised Lexicon Models

Select Model

Choose an option

All Model Comparison

Click here to view model comparison

Sentiment Analysis

Select model for prediction

Afinn Lexicon Model

[Topic Modelling Plot](#)


Sentiment Predictor

Enter Text Here

Sadhguru seeks to unravel karma in new book

Predict

Given Sentiment is Positive



Positive

FIG 7.1.10. Model prediction of Afinn

CHAPTER 8 :- CONCLUSION AND FUTURE WORK

8.1. Conclusion:

The experimental studies performed through the chapters, successfully show that hybridizing the existing natural language analysis and lexical analysis techniques for sentiment classification yield comparatively outperforming accurate results. For all the datasets used, we recorded consistent accuracy of almost 90%.

The first method that we approached for our problem is Word-list based sentiment analyzer. It is mainly based on the dictionary of words with labels such as positive and negative. Testing is straightforward, calculating the sum of scores from the data available. One of the major task is to find the sentiment polarities which is very important in this approach to obtain desired output. In this Unsupervised approach we only considered the words that are available in our dataset and calculated their scores rest of the words are assigned zero scores. We have obtained successful results after applying this approach to our problem.

Clearly from the success of Unsupervised approach is, it can positively be applied over other related sentiment analysis applications like financial sentiment analysis (stock market, opinion mining), customer feedback services, and etc.

8.2. Future Work

Substantial amount of work is left to be carried on, here we provide a beam of light in direction of possible future avenues of research.

- **Interpreting Sarcasm:** The proposed approach is currently incapable of interpreting sarcasm. In general sarcasm is the use of irony to mock or convey contempt, in the context of current work sarcasm transforms the polarity of an apparently positive or negative utterance into its opposite.

This limitation Conclusion and Future Work can be overcome by exhaustive study of fundamentals in "discourse-driven sentiment analysis".

The main goal of this approach is to empirically identify lexical and pragmatic factors that distinguish sarcastic, positive and negative usage of words.

- **Multi-lingual support:** Due to the lack of multi-lingual lexical dictionary, it is current not feasible to develop a multi-language based sentiment analyser.

Further research can be carried out in making the classifiers language independent. The authors have proposed a sentiment analysis system with support vector machines, similar approach can be applied for our system to make it language independent.

- Future research can be done with possible improvement such as more refined data and more accurate algorithm.
- Further the application can be developed into a mobile app

REFERENCES :-

- <https://www.aclweb.org/anthology/C12-2031.pdf>
- <https://towardsdatascience.com/my-absolute-go-to-for-sentiment-analysis-textblob-3ac3a11d524>
- <https://ieeexplore.ieee.org/document/6758829>
- <http://comp.social.gatech.edu/papers/icwsm14.vader.hutto.pdf>