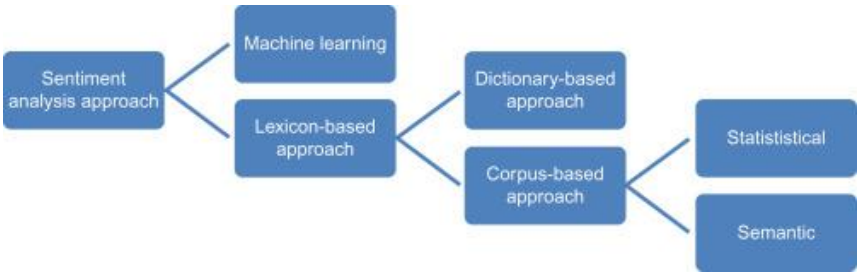
1. INTRODUCTION

Sentiment Analysis

Sentiment analysis (SA) or Opinion Mining (OM) is helpful in solving a vast range of problems that are interest of human and computer interaction experts and specialists. It is also an area of interest for psychologists, linguists ,advertisers , producers , consumers , and politicians. Since the dawn of data , its analysis and the huge volume in which it is generated on daily basis, it has been of utmost importance that this data be analysed for varied uses that humans need to fulfil there tasks. It is used for analysing sentiment of text passed as a parameter to the analysing function which is mainly based on either the machine learning approach or the lexicon based approach. The Machine learning based approach uses an algorithm and builds a model by the method of feature selection i.e. by learning from labelled data sets[1]. Some of the popular and most frequently used methods are Naïve Bayes classifiers, Support Vector Machine(SVM) and Random Forest. The algorithms which are mentioned have the ability to learn every kind of feature for classification by using the criteria of optimization but the problem with ML based algorithms is that the sentiment based classifiers are trained on already pre-feeded labelled data in a particular domain and these then does not work in different domains so to overcome this problem lexicon based methods are preferred over ML based algorithms[2]. The approach based on Lexicon provides us with the information that which words and phrases are positive and negative. So Lexicon based approach is nothing but a list consisting of lexical features which are commonly labelled, based on the linguistic orientation , and it is either positive or negative. Experts and linguists , for proceeding with lexicon based approach have to initially create a sentiment lexicon by the systematic arrangement and collection of sentiment list of words which is based on corpus based approaches and dictionary based approach. After the list of sentiment words is compiled , polarity of the given input , based on the positive and negative indicators is determined which are now known based on the given lexicon. The dictionary based approach is achieved by taking a few examples initially and after that any online dictionary can be used for the expansion of sentiment lexicon by including synonyms and antonyms of those words. This dictionary can be filtered by human inspection[2]. The corpus based approach is used to find the sentimental inclination of

context based specific words. The following two methods are used under corpus based approach:-

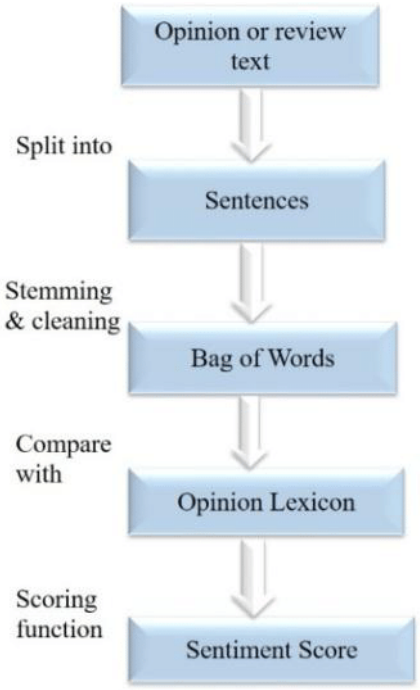
A) Statistical based approach – If some combination of alphabets comes up with an erratic behaviour but in a beneficial manner , then the overall polarity of the sentence is considered to be positive but if the combination shows a behaviour which is pessimistic in nature then the overall polarity of the sentence changes and tends to have a negative inclination. But if the prevalence of the optimistic and pessimistic text is equal in many ways then the overall polarity tends to have a neutral inclination[12]
B) Semantic based approach – Under this method , values are assigned to words based on opinion or sentiment and also to the words which are closer to the words to which values are being assigned. This is done by finding synonyms and antonyms to words to which values are assigned[12]



1. 1 Figure showing the break-down of lexicon based approach

Under the lexicon approach , some of the lexicons such as GI (General Inquirer) and LIWC (Linguistic Inquiry and Word Count) classifies the words into the categories of "POSITIVE" and "NEGATIVE" according to the way of there connotation orientation. LIWC has a pool of 4,500 words which have been classified into 76 categories and of them also, approximately 905 words are such which are classified in two categories , particularly related to sentiment analysis .LIWC is a well tabulated and validated lexicon created by more than a decade of hard-work by experts , specialists , linguists and psychologists. Although LIWC has a vast use in defining and redefining words and there sentiment but it lacks a very significant feature which is a very crucial aspect of sentiment analysis when talking in terms of social media such as Twitter i.e. LIWC does not take into account the acronyms, emoticons , initialisms and slangs[2].

There are other lexicons also such as SentiWordNet , SenticNet and ANEW (Affective Norms for English Words). These type of lexicons associate with the valence scoring of text for analysing the sentiment intensity. The SentiWordNet consists of more than one lakh synsets and these are labelled with sentiment scores categorised as positive , negative and objective. Although it is not as good resource as WordNet still it has varied uses. Lexicon based approach has the biggest advantage that it has its domain independence and above all it can be very easily improved and included in any application [2].



1. 2 Figure showing different levels of lexicon-based analysis

Web Scraping

In our everyday lives , data is generated in humongous amounts and we need to filter , define and refine this data as per our needs and uses. But there is a problem , that this data is scattered over this infinite pool called WWW (World Wide Web)[3] , hence the data filtration and refining becomes a problem. Hence we need a tool or maybe a method which let's us collect this data and the information related to that data at one place , which then could be used for further analysis. Web scraping or extraction or harvesting is a technique which is used to pull data from the "World Wide Web" and then use it or store as per the requirement. Generally, data on the web is scrapped by using the Hyper text transfer protocol (HTTP) or using a web browser. That can be

accomplished by a user(manually) or by the use of a web crawler. Because humongous amounts of data is continuously being generated on the web , web scraping is a very dynamic and powerful tool for collection of data Today's scenario is that , web scrapper tools have been modified. From the current human – aided procedures to being developed into fully automated systems which are capable of converting entire website into a dataset , web scrapping tools and methods have changed a lot with time. Scraping data from the web has two sequential steps i.e. first is the acquiring of web based resources and second is the extraction of desired information from the data that is acquired[3]. First a HTTP request is formulated which can get or acquire the data and information from the selected website. The request which will be sent to the website can be modified in either a URL form which contains a GET query or it could be a HTTP message containing a POST query. As and when the request is received and compiled by the website , the resources which were requested from that website are retrieved and are finally sent back to the program which sent for to get the resources. The data which is received can be in any format be it in HTML[3] (from which web pages are built) , maybe the data-feeds which are in XML or JSON type format or the received resources may be in multimedia type format such as pictures , audio , videos etc. This data (be it in any form) is first downloaded , after that process of retrieving useful data is initiated by parsing data and information received and during this parsing the data is simultaneously re-formatted and organised in a more presentable format or it can be more conveniently said as structured form of data, is made available. For web scraping , two modules are very much needed , one module performs the [3]composing of the HTTP requests like the selenium while the other module is needed for the extraction and parsing of information from the basic HTML code using PyQuery or BeautifulSoup. The main use of UrlLib2 is that it defines a standardised set of functions for redirections , cookies and authentication whereas selenium is a wrapper binding the web browser such as Google Chrome or Mozilla Firefox , thereby enabling the programmer to automate the work of browsing a website by coding. As far as data extraction is concerned , Beautiful-Soup is made in such a way that it can scrape HTML and XML document content and provides easy-to-be-used python based functions for the purpose of navigation , searching and modification of a parse tree, [3]which is basically a toolkit used for breaking down a HTML file and extracting information by using lxml or html5lib. The programming of Beautiful Soup or the structure of this library is such that it can by itself detect the encoded parsing under the processing and

automatically convert the parsed code into client readable format. In the same way, PyQuery has a set of jQuery functions for the parsement of xml documents. But it is still unable to compete with Beautiful Soup because PyQuery only has a supporting hand for lxml so that xml documents can be processed at a faster rate[3].

1.1 BACKGROUND

Vader

Sentiment analysis describes the definition, description, and evaluation of Valence Aware Dictionary for sEntiment Reasoning which is also known as VADER. Sentiment analysis based on Vader is the reform in the identification and designation of the opinions which are expressed by people on social media so that people's intention can be determined towards any particular event occurred in past , feedback related to a particular product, recommendations given about something or analyse data for some specific use[1]. The user's intention may be one of the main factors in determining what sentiment will get embedded in the article or review being posted by the user. Vader uses a collection and combination of the evaluative and quantifiable functions to produce and then validate a standard which shines in those names when sentiment lexicons are discussed about and above all this lexicon is very specifically tuned to microblogging type context. Next comes the compilation of evaluative and quantifiable lexical features with a special emphasis on five generalised rules which will encircle the grammar based and syntax based assumptions and conventions which human beings emphasize upon when expressing the sentiment intensity[2]. The inclusion of these heuristics will on a definite basis improve the accuracy of sentimental analysis driver engine spanning across many domain contexts , be it social media analysis , reviewing the news articles , movie related reviews and product based information and reviews). Well , to a greater interest and when experimented upon this Vader lexicon , it gave remarkable results and performed very well in the domain of social media and the analysis of posts by people. To a greater surprise , it is also showed that individual human raters having a correlation coefficient (r = 0.888) was very much close to the correlation coefficient described by Vader(r=0.881) , this analysis is based on the same ground basis which is the average group mean derived from no less than 20 human raters , performing the analysis of each and every tweet. And if we proceed with further analysis , the classification frequency of human raters is 0.84 whereas for Vader it is

0.96 , this means that Vader is able to correctly classify tweets into optimistic (positive) , pessimistic(negative) and neutral (positive and negative sentiment intensity is equal). Vader lexicon features has its own advantages and it also retains the features of more common lexicons like LIWC , it is much more big in nature but can be simply inspected , can be understood very well and above all it can be applied in a much more faster way with an easy extension. Vader sentiment based lexicon shines among the lexicons because it has a standardised quality and has been verified and validated by human beings. Vader is distinguished from other lexicons , especially LIWC because it is more sensitised to expressions which are used in day-to-day conversations or tweets(in the case of twitter) , in general to social media and it can be easily extended to other domains as well[2].

BeautifulSoup

Web scraping or extraction or harvesting is a technique which is used to pull data from the "World Wide Web" and then use it or store as per the requirement. Whenever we are discussing about web scraping, web pages is the first thing that comes to our mind because they are storage houses of large amounts of data[4]. The process by which we pull the information out of the web pages by using a web scraper is called web extraction. Nowadays we see every type of data being displayed on web pages and so it is obvious that when this data will be scraped from the web pages then every data type be it multimedia such as video , images , text , time or titles etc , everything will be scraped.[4] A website is a collection of many web pages and it is based on the data which the owner wants to display on the website. Also, there are two types of web pages , one which is static while the other one which is dynamic. Static web scraping of data or web scraping of data from static web pages is very easy because here we have to send an HTTP request to the server side and the response which we will get, that will be the same page for which the request was sent for and this does not require any extra or more of a complicated process. But if we talk about dynamic web pages and scraping data from them , it is a very complicated process as the dynamic pages on the server side are based on server side scripting languages like Js, ASP, PHP. Hence it would be more time taking to load and moreover it is possible that instead of getting a web page for the request that was made , we get a Js code as a response from the side of server. So it can be said that the web scraping that we do , is done by sending an HTTP request to the website and then we get the page source which is then converted to a soup object

and then we can get the required elements from it. But if the website was Js rendered which means Js will provide elements which are on the website , then simple sending the HTTP request will not get us the required data[4]. So in the case of Dynamic type web scraping , it is required for the website to get loaded fully , after which the page source is fetched and therefore all the data is then loaded first and then scripting takes place. This way dynamic or the continuously changing data can also be scraped from the web pages. The library used for parsing the HTML docs and defining the tags of HTML is BeautifulSoup. For parsing , it first creates a parse tree for the already parsed pages and then data is extracted through HTML. It receives the required data from HTML , XML and the other known markup languages. Say for example we came across a webpage that has all the relevant data related to our area of interest and requirement , and we want to scrape that data. For scraping that data , we would be needing BeautifulSoup for fetching the required data from that webpage and it also removes the HTML tags and the other saved information[5].

1.2 MOTIVATION

Sentiment Analysis

Sentimental analysis is a vast field which needs exploration and discovery and this helps us determine what in true sense the user wants to convey through his words and what is the underlying emotion the user wishes to express through his oral or written form. In the past people used to express themselves through letters , oral form of communication etc which were usually long and the underlying sentiments were very much visible to the eye but today the length of expression has been substantially reduced and hence it is very difficult to understand and determine the sentiment of the expression by the person. Generally people choose that medium to express themselves , which is open , transparent and people get a feeling that they are a part of a large community where there are other people also who have had those type of experiences. Moreover we human beings are creatures who like to express ourselves and therefore our voice and opinions regarding any specific issue matters a lot. Our interaction with people on that level will be helpful in the promotion of a better world. So , to understand people's emotions better and to determine the underlying sentiments in the expressions and posts by the people on various social media handles was my motivation to do sentiment analysis.

Web Scraping

In the world that we live in today , information is the most significant factor in the determination and retrieval of results which are carried out by further analysis. But to get this information we need , data , which are basically raw facts and figures, from various sources. This data is not available to everyone or everyone are not able to access this data and henceforth this led to the birth of web scraping. Scraping data from the web has entirely modified the way we use to retrieve data from web because now data retrieval has become very much simple and due this web scraping , many businesses have profited and got a boost in there sales as the collection of leads was now possible. This process of collecting and gathering unstructured form of data is very interesting and if used carefully , can turn profitable for many types of businesses , has a great potential to enhance scientific researches and can prove informative for personnel usage. The business of advertisement relies on the direct advertisement methods which are allocated through many pages for the current service to understand the actual context , data scraping and web wrapping are used in combination with content inspection tools for the context based analysis of the page. Also when data sets are provided by researchers and analysts , they are made available to the public, but the way they are made public is different. A structured API is used for providing the dataset but this dataset is only accessible using forms and HTML which calls for the use of web wrapping , which basically converts the content into a relational form so that it can be processed as structured data. Web scraping has also increased the personnel use of those tools which are required to combine components and page into a single collection of web pages.

1.3 SCOPE

This project covers the domain of web scraping and sentiment analysis in the context of news. News is scraped from web based on the input keyword by user and top ten most recent news are then listed on the website with all there information such as when the news was uploaded , what is the media , the description of the news , the title of the news and the link to directly redirect to the news web-page. Input keyword could be any term related to some event happened in past or something that is going to happen. News will be scraped and it is not based on the fact that it is only limited to any specific

region or geographical area rather it will fetch the news directly based on input term or the most near searched and the matched results. The purpose is to get the latest and updated news to people 24x7 which will be free from any kind of political bias. The limitation is that as live scraping is done so therefore it takes a bit longer to search and get the results and print on website

In the case of sentiment analysis , here user will have to input the term and then that term will be searched in the database , if it is there then directly data will be taken from database and if not then tweets related to that term will be scraped from twitter and saved in the database along with there positive , negative , weaklyPositive , weaklynegative and neutral tweet count. This categorisation of tweets into respective categories is based on the sentiment that each tweet is assigned by the sentiment lexicon. The purpose of this analysis is that , whenever any person wants to tweet something regarding any topic on twitter then he will have the idea regarding that topic , that what is the on-going public sentiment for that topic and then whatever he wants to tweet about can again be analysed using the sentiment lexicon and then that tweet could also be formatted by the user if that tweet gives a sentiment or conveys something that ought not to be conveyed. The limitation is that if the term is not present in the database then it takes a bit longer to scrape tweets from twitter and perform there analysis.

1.4 OBJECTIVES

The objectives is to develop a interactive website which can aggregate information regarding a particular topic and if the user wants to make a public comment regarding that issue , the second objective would be to make the user aware regarding that issue and the ongoing people's sentiments regarding it.

1.5 TIMELINE

Table 1.1 shows the timeline of the project

Sr.No	Week	Activity
1	30/05/2022 to 05/06/2022	1) Learnt the basics of python and Django
2	06/06/2022 to 12/06/2022	2) Learnt and implemented web scraping

		3) Learnt MySQL connector to fetch and retrieve data from database
3	13/06/2022 to 19/06/2022	4) Learnt and implemented sentiment analysis using Vader and by using twitter API tweets are scrapped
4	20/06/2022 to 26/06/2022	5) Debugging of code and start the designing of website
5	27/06/2022 to 03/07/2022	6) Finished the website designing
6	04/07/2022 to 10/07/2022	7) Connected backend and frontend using Django and added some more features to website

1.6 PROJECT REPORT OUTLINE

In Chapter 2, Literature review is about the the working of beautiful soup , what is a DOM structure and the different functions which beautifulSoup uses to look for the labels in the Dom structure. Also , in the case of sentiment analysis , method to scrape tweets are discussed and then for the sentiment analysis NLTK package is discussed and the pre-processing of tweets are discussed about. Further the working of Vader is discussed.

In Chapter 3, methodology is discussed about how this project is made and what are the various python library requirements for the project and further the explanation of various snippets of code.

In Chapter 4, Results and Discussions have the code outputs with the explanation of each output. Also the various advantages and dis-advantages are listed regarding the project.

In Chapter 5, Conclusion and future work has been described.

In Chapter 6, References have been listed.

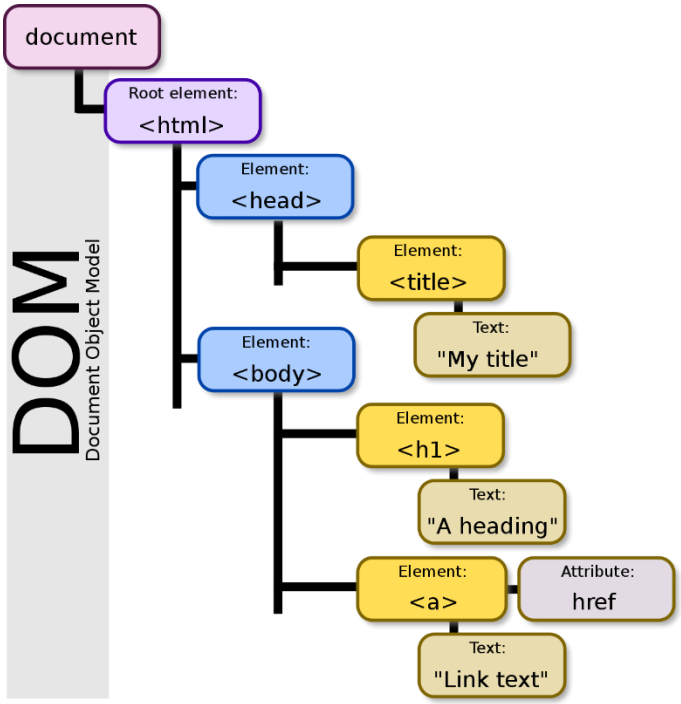
2. LITERATURE REVIEW

2.1 Working of beautiful Soup

This has been used for scraping data from web . It is a python based collection of data stored in the form of a database and it has a base of HTML or XML computational based engine which is used for pulling out , analysing and modifying information which is present in the DOM (it is basically a cross platform and linguistic independent base which sees HTML and XML documents in the form of a tree type structure and each node of the tree is a object which represents specified part of the whole document) tree of web pages. It can send an unparalleled series of unique and concise DOM based visitor interfaces which directly or indirectly helps developers build a system which could be basis for the other models and moreover it has a very special characteristic i.e., across the platform , it can be operated upon, indicating that it is flexible[3].

Clients of Beautiful Soup can, as indicated by their requirements, introduce explicit HTML/XML examination motors (e.g., xml, html5lib, and so forth.). Taking lxml for instance, clients can instate Beautiful Soup with the accompanying call: Beautiful Soup (markup,"lxml").Subsequent to instating, Beautiful Soup gets all the relating DOM tree structures for HTML records. Then utilizing an inherent series of associated works that relate with DOM API, it visits, gets, and alters the property estimations or sets of the assigned hubs of the DOM tree. Beautiful Soup has not just has an underlying series of DOM API, yet in addition many recently added new interfaces that standard DOM APIs don't have, settling on decisions more straightforward for engineers and developers[4].

Beautiful Soup will break down any archive passed to it, including a wide range of HTML/XML reports and typical records. In any case, just with HTML and XML records that meet specifications with Beautiful Soup create a comparing DOM tree and complete a progression of API calls to get the assigned information. Involving HTML for instance, Beautiful Soup first purposes a HTML analytics engine to change over the HTML record into a DOM tree, and afterward the client can utilize Beautiful Soup's provided request and altering capabilities to work on a DOM tree [11].



2. 1 Structure of a DOM being represented in the form of a tree [11]

In most of request interfaces provided by Beautiful Soup (e.g., findPrevious, findPreviousSibling,findParend, findParents, and so forth), all will call the two capabilities _findOne and _findAll, or, in other words that all requests occur inside these two capabilities. At the point when Beautiful Soup characterizes these capabilities, Python's stretching boundary list innovation can complete different DOM API capabilities with an alternate number of boundaries and values to call _findOne and _findAll, consequently accomplishing two distinct objectives. For example, _findOne can look through assigned tag or the assigned substance of a tag, while _findAll can look through all labels or content that meet its predefined conditions inside the entire DOM tree[3].

2.2 API based Data Extraction

A feature of object-oriented programming languages called an API enables programmers to create software for a specific application using a reference programme library. The operating system or application programme of a device specifies the API, via which a requester—another device or a client user—can submit requests and wait for results. APIs provide communication between various software applications and access to their services. To add new features or functionalities to the existing methods , developers might create new classes or expand existing ones . An endpoint, a

component that watches for requests, is how a client API is accessed made from the communication's client side to the server side over HTTP, anticipating a return of a response. [13]

2.3 Natural Language Tool-Kit

This NLTK is a naturalised language based processing platform for python which has been developed in the combination of linguistics involving computations in the year 2001 at the Pennsylvania university . It gives at hand an interface which has 50 corpora and the lexicon based resources which is based on SentiWordNet which has a collection of string and text computing libraries for the lemmatization , classification and tokenization. In NLTK , the score based on sentiments is calculated from sentiWordNet which has a table format of polarity score based on each synset in the WordNet categorising it into positive and negative with the values ranging from 0.0 to 1.0 , in every case the final sum is 1.0 [2].

The synsets included in WordNet , each of them is very much uniquely distinguished because of the use of the POS and ID pairs[2].

Table 2. 1 SentiWord Net values [2]

POS	ID	Positive Score	Negative Score	Synset Term
A	00071142	0.5	0.5	Impressed
A	00070111	0	0	Enhansive
A	00065064	0.625	0	Good
A	00061664	0.5	0	Neat

Pre-Processing [8]

While doing analysis on tweets , they can be pre-processed so that unnecessary data could be filtered out:

1) Removal of Re-tweets: Re-tweets can be removed from the tweet.
2) Case sensitive analysis: If we don’t want to use a case sensitive analysis then we can convert upper case words lower case.
3) Removal of Stop words: All those words which are not carrying any sentiment and also not effect the meaning of the sentence can be removed.

4) Stemming of words :For filtration of sentences, words can be reduced to there roots and thereby reducing the complexity of sentence in analysis.
5) Special character and numbers don't express any sentiments so they can be removed
6) Slangs and abbreviations can be expanded so that they can be analysed
7) Sometimes wrong spellings don't convey any meaning and sentiments and hence can be discarded
8) Words can be POS tagged and hence they can be classified under the headings nouns , adjectives and verbs and hence it involves efficient implementation of analysis
9) Removal of username and URL's can be beneficial as they don't carry any sentimental value

Working of Vader

Vader is a type of sources package which comes under NLTK.VADER is a tool used to do the sentiment analysis whose approach is lexicon and rule based that is used in the expressions expressed on social media. Vader uses a method of scoring which is applied to results after analysing the [2]sentiment of the tweet (in the case of scrapping tweet from twitter) and as just said we must get the tweet in order to analyse the sentiment of tweet , so for this purpose we use TweePy API , from where we can extract twitter data and by using this we have the real time access to publicly available raw tweets. Vader includes within it a sentiment lexicon and together with the inclusion of some syntactic rules , the sentiment analysis of text by Vader could be improved. Vader cause of origin is the abbreviations , emojis and slangs used in posts on social media by people and these are like a shorter way of expressing yourself on social media. The syntactic rules used are[2]:

A) Emotional intensifying marks: The interrogation and exclamation marks lead to increment and decrement of the sentiment intensity and thus influencing the positive and negative polarity of the text. Eg "This is great !!!" will have a greater positive score as compared to "This is great"[9]

B) The play of capital text: The intensity of a word which is capitalised in nature and when other words are not capitalised gives a upper hand in intensity to that word when the analysis is performed .Eg " This is GREAT" will have a greater positive score as compared to "This is great"[9].

C) Negatory marks: If a word such as "great" gets preceded by a word such as "not" then this could just revert the ongoing positive polarity of the whole sentence and give a negative sentiment. For eg " The environment is not good here" will turn the sentiment of the score as negative [9].

D) Words which gives a boost: There are some words such as "very" and "extremely" which intensify the underlying emotion of the text and this leads to increment in the intensity of the word. Eg " The environment is good here" will have a less score as compared to "The environment is very good here" [9].

E) The contrast creating words such as "but" which can shift the signals of the whole sentence and change the polarity of the text with the sentiment of the text following the contrasting word being the dominating one. Eg " The environment here is good but it was better in Shimla" has a mixed type of sentiment but the later half of the sentence is more overpowering [9].

Specifically , in case of Vader lexicon when a sentence or text is passed to the sentient intensity analyser , then the analyser pulls out those words which carry a sentimental value to them and the influence or intensity they have caused to the sentence.[2] This lexicon ranges the polarity from -4 to +4 , where -4 has the maximum negative intensity and +4 has the maximum positive intensity, but when final scoring of the sentence is done then the scores are normalised between -1 and +1 and this standard of normalisation has given this method another name as compound scores. The compound based score gives us the nature of sentence , talks about its intensity and reviews the polarity. So to give the overall intensity of the sentence , all the polarity scores of sentiment words are added and the normalised scores are thus received:[10]

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

So here x is defined as the sum of polarity scores and alpha is the constant whose numerical value is generally taken as 15. Example – " The environment here is good and food is nice". Both the words here good and nice have a polarity as 1.9 and 1.8 which on [6]addition gives the value as 3.7 and to get the compound score of the following sentence we do the normalisation and get the answer as 0.6907 , which is between -1 and +1.Now the categorisation of tweets into various categories happens based on the value of compound score, if the value of compound score is between -0.5 and 0.5 then the tweet is considered as neutral in sentiment and if the value of compound score is greater than 0.5 then it is considered as positive and if the value of the compound score is less than -0.5 then the sentiment of the tweet is considered to be negative[10].

3. METHODOLOGY

3.1 Creation of a virtual environment

Production of a different Virtual Environment for the project is fundamental as it permits the program to introduce and utilize the Python libraries without making any struggles with other programs utilizing similar libraries. It is expected to make the Virtual Environment in a similar organizer as of the whole Project.

3.2 Installation of python libraries

The python libraries must be installed by using the Windows PowerShell or by using the IDE terminal (PyCharm). “pip” command is used to install various libraries.

3.3 Code editor used for programming

VS code is used as the code editor for programming. This code editor is very robust and is a good choice for big projects , the only limitation is that it is slow to start up.

3.4 Commands used :

1. *python virtualenv env*

   The above command will create a virtual environment

2. *activate.bat*

   The above command will be used for activating virtual environment

3. *python-admin startproject hello*

   The above command will create a project in the vs folder named “hello” and will create many files mainly with “.py” extension and these files are the required files during the making of a Django project.

4. *python manage.py runserver*

   The above command will start the server and a link will be provided to us in the terminal , and on-clicking on that link , we will be redirected to a web page telling us that server is started successfully.

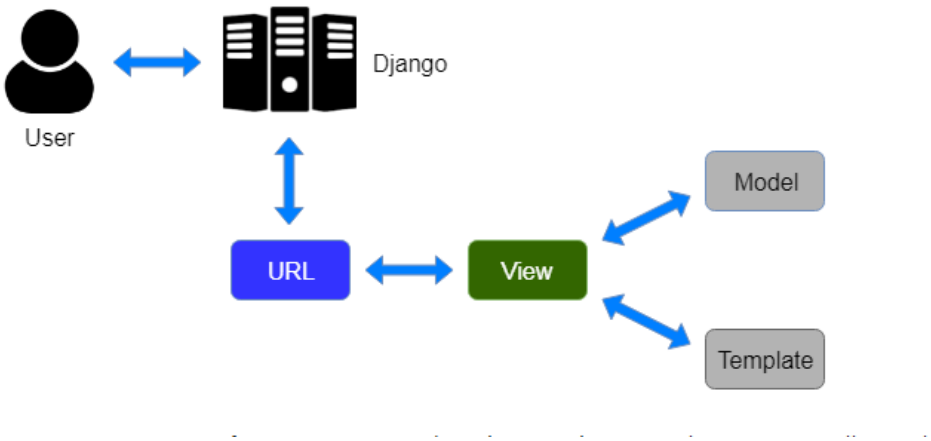
5. *python manage.py startapp mynewapp*

The above command is used to make an app ; basically a project contains apps and this command will lead to the creation of an app and contains ".py" files which are required when we make changes in the functionality of the app.

3.6 Django MVT Architecture

The Model View Template architecture is a thing arrangement plan. It is a gathering of three gigantic parts Model ,View and Template. The Model assists with managing information base. It is an information access layer which handles the information.

The Template is a show layer which handles User Interface part totally. The View is utilized to execute the business thinking and mark of connection with a model to pass on information and renders an association.



3. 1 Diagram showing the flow of control in Django MVT architecture

3.7 Modules and Functions Used:

1) For Installation: "pip install tweepy"

   import tweepy

   This module is used for accessing the twitter API and scraping tweets from twitter .

2) For Installation : "pip install vaderSentiment"

   from vaderSentiment.vaderSentiment import SentimentIntensityAnalyser

This module is used for analysing the sentiment of the tweet passed as a parameter to the sentiment intensity analyser.

3) For Installation : “pip install datetime”
   from datetime import timedelta
   from datetime import date
   This module is used for getting the current date and saving it with the tweets in the database and timedelta has been used for adding some days to the current date for checking the date of tweet and for how long the tweet has been in the database.

4) For installation: “pip install mysqlconnector”
   import mysql.connector as sqltor
   This module has been used to connect my python code with the sql database and to pass strings as sql commands to mysql and fetch the required records and this connector will be referred to as sqltor.

5) For Installation: “pip install GoogleNews”
   from GoogleNews import GoogleNews
   This module has been used to get the news along with all the required data of the news.

6) from django.urls import path,include
   This module has been used so that when a request comes , then it could be redirected to the specified app , so basically path function has the parameters for the request and for that request to which app , which has been included in the path function it should be redirected to.

7) From “app” import views
   This tells us that from our app folder we are trying to import views which are basically the main part of MVT architecture and decides how will the data be visible to us.

8) from django.shortcuts import render

The Django module is a hub of functions which are commonly used in the context of view type classes.Here the render collects the given template and combines it with the dictionary and finally returns a Httpresponse (object) along with that rendered text.

3.8 Working

News Aggregation

```
from GoogleNews import GoogleNews

def Scrapper(str1):
        gn= GoogleNews()
        gn.search(str1)

        result=gn.results()
        return result
```

Figure 3. 2 Code snippet of Web scrapping of news

Figure 3.2 displays the code snippet for web scrapping of news , here first Googlenews class is being imported from GoogleNews python library and then within the scrapper function , an object of class has been created and then the search keyword inputted by the user is passed to the search function. This function searches for the given inputted keyword and then the results function is called which gives title , link , description , media and date time for the given news and then this result is returned to the calling function.

Sentiment Analysis

API User Verification

```
#reading the data from text file
my_access_keys= open('C:\\Users\\User\\Desktop\\Folder\\college\\Sem 5\\NTCC in house practical training\\

#Authentication data
my_API_key=my_access_keys[0]
my_API_key_secret=my_access_keys[1]
my_access_token=my_access_keys[2]
my_access_token_secret=my_access_keys[3]

#Twitter authentication handler code
auth= tweepy.OAuthHandler(consumer_key=my_API_key,consumer_secret=my_API_key_secret)
auth.set_access_token(my_access_token,my_access_token_secret)
api= tweepy.API(auth)
```

Figure 3. 3 Authentication Handler

Figure 3.3 shows the authentication handling. Here the access keys are stored in another text file which are accessed using the file read operation. In the file pointer value of every key is being saved as list and then every key is being allotted its value. After that using the tweepy authentication handler , all four keys are passed as parameters to the functions which basically verifies the authenticity of the user.

Analysis

```
#Analysing the tweets
for tweet in tweets:
    final_text= tweet.full_text.replace('RT','').replace('"','')
    analysis = sid.polarity_scores(final_text)
    count=count+1
    compoundsen=analysis['compound']
    if compoundsen<=-0.5000 and compoundsen>=-1.0000:
        negativesen=negativesen+1
    if compoundsen>-0.5000 and compoundsen<0.0000:
        weaknegative=weaknegative+1
    if compoundsen==0.0000:
        neutralsen=neutralsen+1
    if (compoundsen>0.0000 and compoundsen<0.5000):
        weakpositive=weakpositive+1
    if compoundsen>=0.5000 and compoundsen<=1.000:
        positivesen=positivesen+1

    Insert_object ="""Insert into tweets values(\"{}\",'{}',{},{},{},{},{},'{}',{})""".format(final_text,Ite
    cursor.execute(Insert_object)
    mycon.commit()
```

Figure 3. 4 showing the analysing and categorisation of tweets

Figure 3.4 displays that initially each tweet is scraped using twitter API and then all those scraped Tweets are stored in tweet which is basically the variable which contains the all the scraped tweets. Each tweet undergoes required pre- processing .i.e. if a tweet has a heading as RT then to clear it , replace function has been used which replaces the heading "RT" with white spaces. The importance of doing this is that , now every tweet which has the starting as "RT" will be blank thereafter and hence one word less for analysis. Similarly, all the double quotes are also replaced from tweet. Further now sentiment intensity analyser has been used which is basically the object of vader lexicon

for calculating the sentiment of tweet and hence this tweet is passed as a parameter to the polarity_scores method which basically returns the sentiment in a dictionary form: { 'pos' : 0.6575 , 'neg' : -0.6363 , "neu" : 0.6273 , "compound" : 0.7373 }. We will be utilizing the 'compound' element of the dictionary to classify our tweet into the respective categories.Each category has its individual counter and every time the respective counters are increased by one , whenever the compound score falls in one of the categories. Now this data of incremented counter with the respective tweet is added to the database, along with the tweet. For this I have used the sql connector for storing the data of the tweets in the database.

```
def Delete_the_tweets(Iterm,Leftover_tweets):

    #reading the data from text file
    my_access_keys= open('C:\\Users\\User\\Desktop\\Folder\\college\\Sem 5\\NTCC in hou

    #Authentication data
    my_API_key=my_access_keys[0]
    my_API_key_secret=my_access_keys[1]
    my_access_token=my_access_keys[2]
    my_access_token_secret=my_access_keys[3]

    #Twitter authentication handler code
    auth= tweepy.OAuthHandler(consumer_key=my_API_key,consumer_secret=my_API_key_secret
    auth.set_access_token(my_access_token,my_access_token_secret)
    api= tweepy.API(auth)

    Deletion_object="delete from tweets where Keyword='{}' Limit {}".format(Iterm,Leftc
    cursor.execute(Deletion_object)
    return
```

Figure 3. 5 showing the deletion of tweets

Figure 3.5 shows the deletion of tweets, first the authentication takes place and then the tweets get deleted. Here deletion of tweets have been made because when the same keyword was searched again after the time period of seven days , then if it happens that regarding that same keyword 300 latest new tweets are not found , then to maintain the number of tweets in the database as 300 only , remaining number of tweets are deleted from the database and now all the new tweets found will be stored in the database with the remaining one's giving the total count as 300.

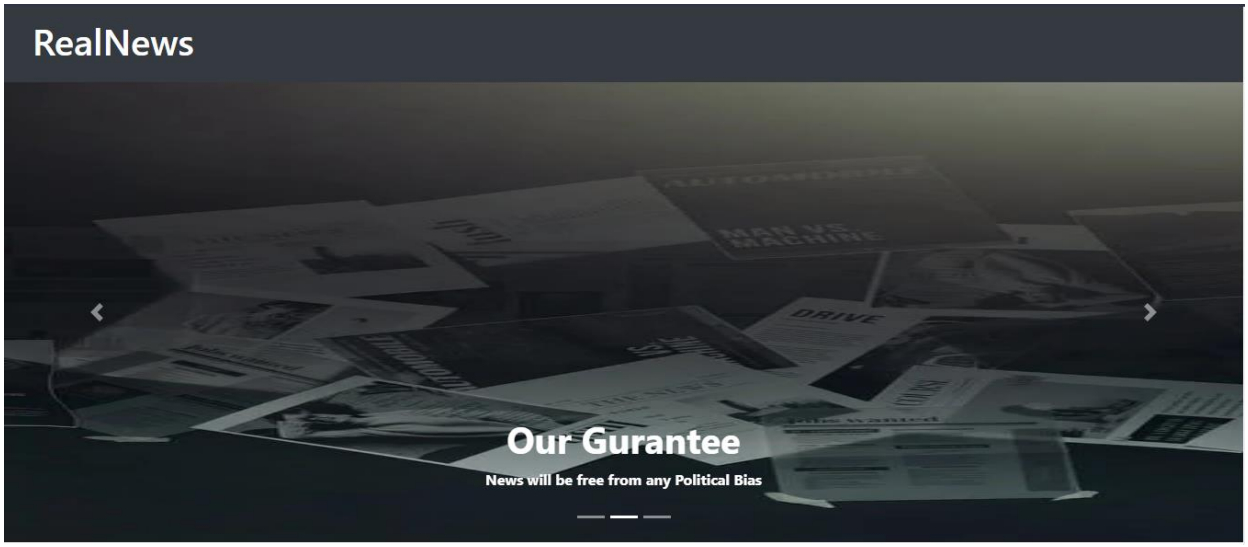
```
def New_tweet_count(tweet_scanning_amt,str1):
        #Checking if authentication of user is successful or not
        #reading the data from text file
        my_access_keys= open('C:\\Users\\User\\Desktop\\Folder\\college\\Sem 5\\NTCC in house practical training\\Web

        #Authentication data
        my_API_key=my_access_keys[0]
        my_API_key_secret=my_access_keys[1]
        my_access_token=my_access_keys[2]
        my_access_token_secret=my_access_keys[3]

        #Twitter authentication handler code
        auth= tweepy.OAuthHandler(consumer_key=my_API_key,consumer_secret=my_API_key_secret)
        auth.set_access_token(my_access_token,my_access_token_secret)
        api= tweepy.API(auth)

        #passing the parameters to cursor method
        count=0
        tweets= tweepy.Cursor(api.search_tweets,q=str1,tweet_mode="extended",lang='en').items(tweet_scanning_amt)
        for tweet in tweets:
            count=count+1
        return count
```

Figure 3. 6 showing the count of new tweets

Figure 3.6 shows the new tweet count function where first the authentication of user takes place and then we search and count for the new tweets to check whether there are 300 new tweets or not. For searching, tweepy library's cursor method is used which has parameter as api.search_tweet which searches for tweets on twitter , second parameter is the keyword regarding which search has to be made , third parameter is the tweet_mode which is set as extended which means full text tweet will be scraped and nothing will be truncated and all those tweets having language as English. The items parameter sets the limit for the number of tweets to be scraped or analysed. The results are stored in tweets variable and then iterations are made, with the increment of counter at each step.

4. RESULTS DISCUSSION

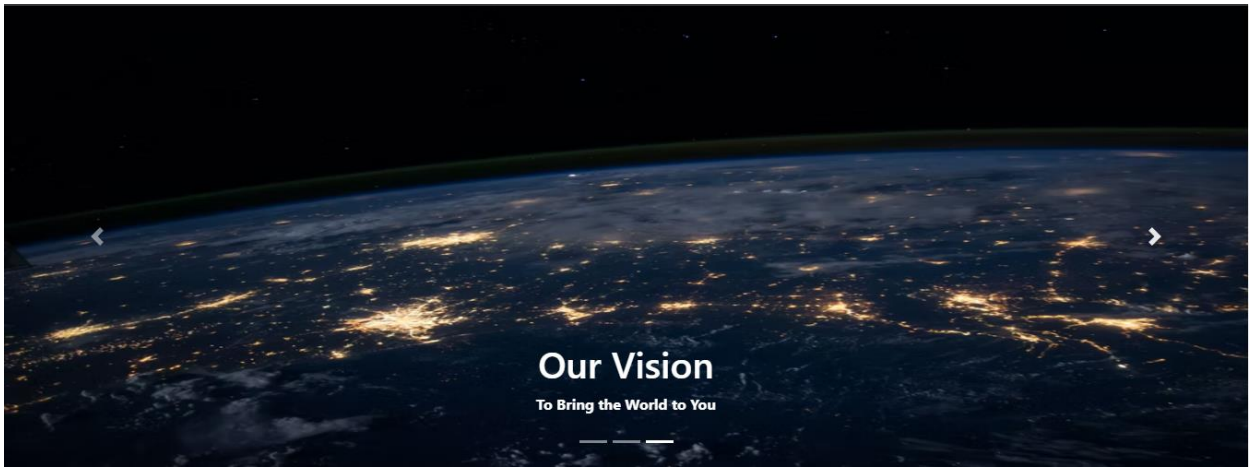
4.1 View of Website



4. 1 Screenshot showing Heading and sliders in the first view of the website

Figure 4.1 displays the first view of the of the website where the Name of the website is being displayed as “Real News” and for this heading tag has been used. Now below this is the slider which has been added , there are 3 sliders and the three slides have different pictures and different text is being displayed on each of them. The other two slides are as follows :-

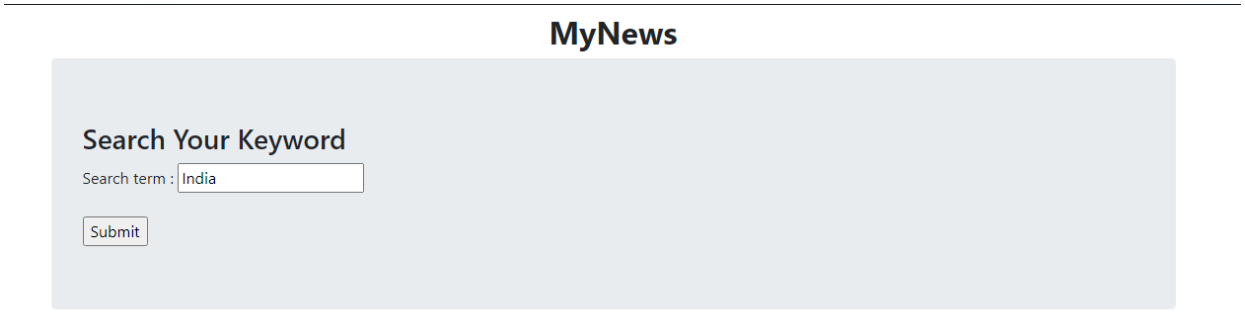
4. 2 Screenshot showing the second image of slider with the corresponding message



4. 3 Screenshot showing the third image of the slider with the corresponding message

Figure 4.2 and Figure 4.3 are the other two sliders and for inserting pictures “img” tag has been used and link of each picture has been provided in the src attribute , and also the dimensions are adjusted as per the requirements. The text inserted on each slide is added with the help of style heading tag and font settings have been edited by using the style attribute , the colour and size has been changed. The bold tag has been used for making the heading in each slide as bold.

4.2 News Aggregator



4. 4 Screenshot showing search bar to search the term for which the news needs to be scraped for

Figure 4.4 shows that container is separately enclosed under a div tag and contained within the class attribute and now the <h2> has been used for displaying the heading as “MyNews” and the user has to enter the keyword of his choice in search bar and then hit enter or click submit button. Here to take input , form tag has been used which returns the input keyword to the program at the backend. Now the heading “ Search Your Keyword” , search bar and submit button are all inside the jumbotron component of Bootstrap.

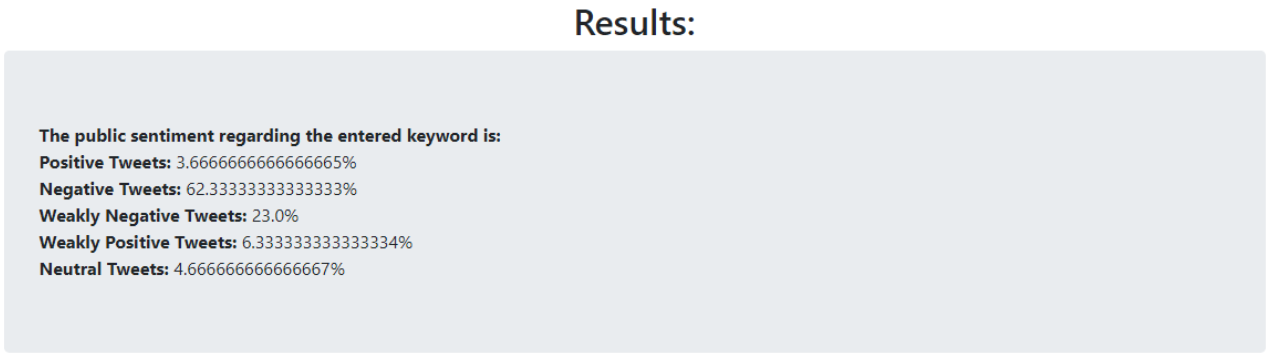
4. 5 Screenshot showing the view of the scraped news along with the data with that news i.e title , media, description , link , date and date-time

Figure 4.5 shows that when the keyword was searched for, then that keyword was passed as a input to the program at the backend and the output of the program was passed as a input to the website and displayed. Whenever news will be scraped , along with that all the data related to that news such as title , Media , date-time , Description , Link and Duration(When was it published on the site from where it is scraped). All these top 10 news are displayed in a separate jumbotron and an internal scroll bar is added for scrolling down to every news.

4.3 Sentiment Analysis

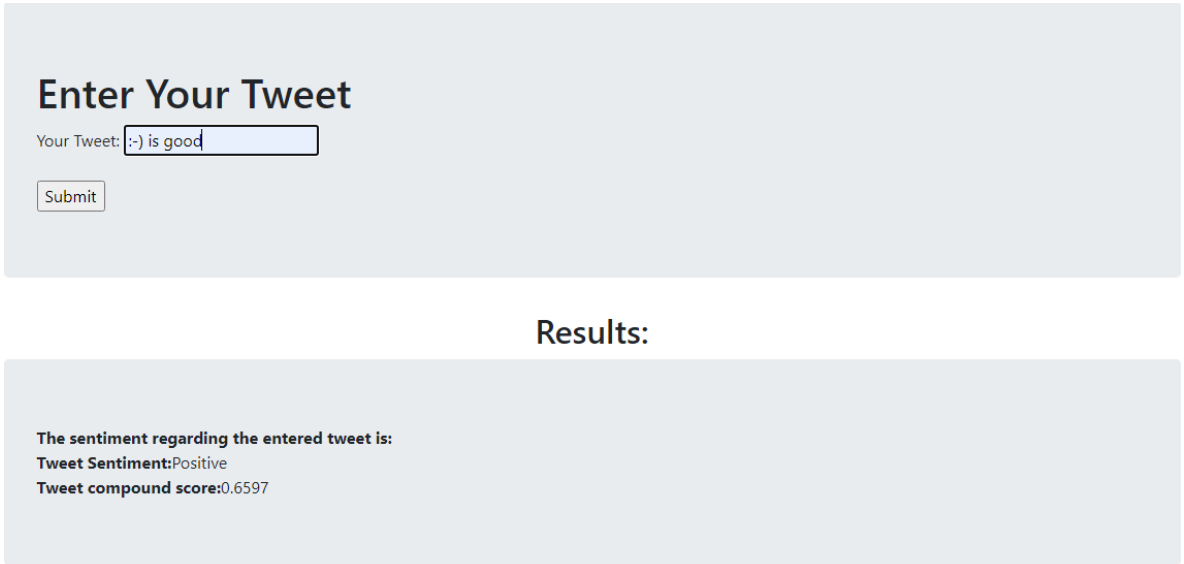
4. 6 Screenshot showing search bar for the topic we need to do the sentiment analysis for

Figure 4.6 shows that heading “Sentimental analysis” is added with the help of heading tag and within the jumbotron heading “Search your keyword” has been added , user can enter the keyword in the search bar and then click on the submit button. The input term is passed as a input to the program at the backend and the results are fetched and displayed on the website.

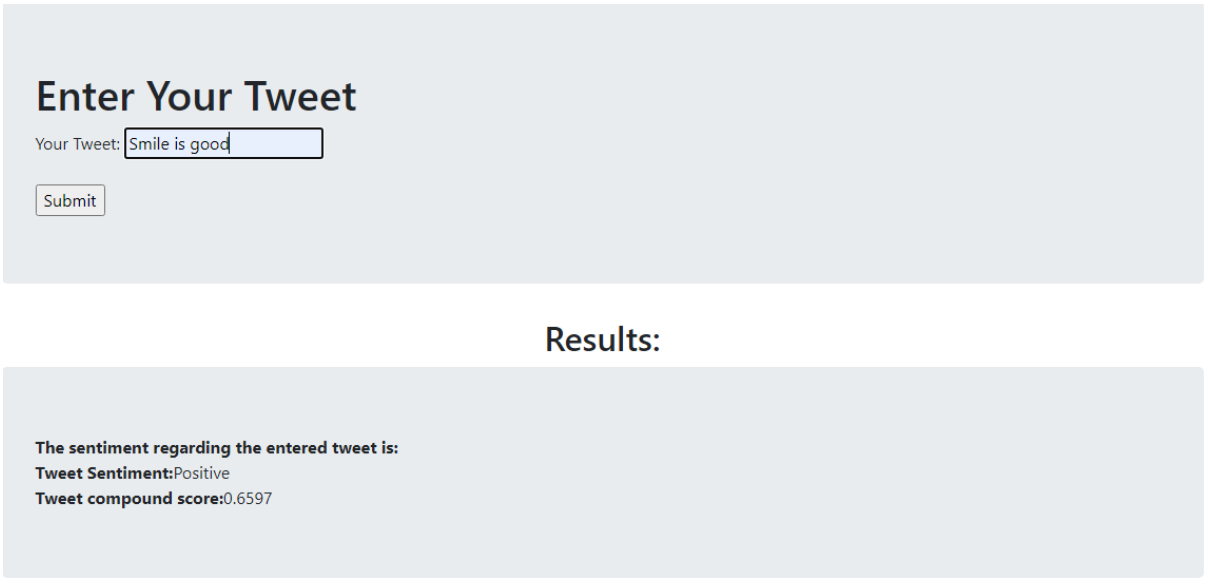


4. 7 Screenshot showing the categorisation of the tweets into various categories

Figure 4.7 shows that the “Results:” heading is again inserted using the heading tag and with a limit of around 300 tweets , analysis of each tweet is performed and with the classification of each tweet into respective categories , we get the results. The respective categories and the above written statement in the screenshot are enclosed within a jumbotron.



4. 8 Screenshot showing the input tweet and the resultant sentiment and compound score of the tweet



4. 9 Screenshot showing the input tweet and the resultant sentiment and compound score of the tweet; the difference is that the inputted tweet has only text but still the score and sentiment coincides with the above score and sentiment when emoticon was used

Figure 4.8 and Figure 4.9 shows the analysis of the tweets which the user wishes to post on twitter and in the above screenshots , in one of the screenshots simple statement is inputted “Smile is good” while in the other screenshot statement with the emoticon has been used “ :-) is good” . Vader gives the same polarity score to both the statements and hence Vader proves that it is an ideal case to be used for sentiment analysis in the context of social media.

4.4 Advantages:

1) News is live scraped so there is no delay of news.
2) It is flexible in nature in the sense that more news i.e. greater than 10 could be scraped off , region could be specified , language can be added as a parameter and by clicking on links directly the user will be redirected to the news web-page.
3) User now could be made aware of the on-going public sentiment regarding any specific topic.
4) User can know the sentiment and message his tweet will convey to others on social media.
5) It will help prevent rumours and the spreading of false information among people
6) It will help an individual realize whether and what to comment about socially sensitive topics

4.5 Dis-advantages

1) It takes a bit long to scrape news.
2) It takes long to scrape tweets , if the entered keyword is not previously present in the database.
3) Sometimes the length of tweet is too long to get saved in the database and hence this gives an error
4) The authentication handler function didn’t work and because of that each function which scrapes tweets using twitter API needs authentication , the code of that function had to be repeated.
5) The current program is not able to scrape the date at which the tweet was posted on twitter.
6) In the case of web scraping , images are not scraped with the required news and sometimes date time parameter returns none.

5. CONCLUSION AND FUTURE WORK

As the internet has grown exponentially and markets are , in today's world totally data dependent, it is now a necessity to have access to the updated and latest data be it in any field. Data is the main decision making factor and the cause that any business can run profitably or not. In the era of today , every person can find the growth and emergence of a new factor every year and web scraping is no different from it. This factor has its foundation built on the basis of structured and unstructured data. In the case of news also, people can save so much of there time by using news aggregators instead of looking individual news sites on the web. News aggregators collect all the data related to a particular news item at one place and thus help save a lot of time for the users.

Twitter sentimental analysis falls under the headline of mining of text and opinions and here focus is on the analysis of tweets and to present the results under different categories and sub-categories. These results can help to analyse and forecast customer behaviour towards any product and his relationship with the producer , analysis of comments on various social media handles or helping a political party decide which way to pave for diverting the views of public based on the on-going public sentiment. The full process of analysis involves collecting data , sentiment analysis and then its classification. To some extent it also involves pre-processing of tweets but in the context of Vader which is built especially for social media handles where people mostly are interested in using short forms , slangs and emojis have a very intense influence on the sentiment intensity of the overall tweet and if pre-processing done then all of these will be removed off from the sentence maybe due to non-understandability of the short forms or short forms and slangs being seen as spelling errors. With the changes in this field and the on-going development , slowly but surely models of analysis will come which will have high accuracies.

Future Work

Both web scraping and sentiment analysis have immense scope in the future. Web scraping can be more refined and can target specific information which the user has demanded and moreover the current flaws within the existing modules must also be resolved to make them more efficient. To improve web scraping specific functions can be added to scrape images and date-time which is not appearing in some instances must be visible in every instance then. Secondly , the sentiment analysis part can analyse the sentiment of the text, and Vader , which is a lexicon based analysis tool specially made

for analysing the sentiments of posts by people on social media has a very high level of accuracy and is at par with human raters for analysis. Here the time which is required to scrape new tweets can be improved and for authentication handling a specific function must be made which will reduce code repetition and save time while debugging.

REFERENCES

[1] C. Hutto and E. Gilbert, “VADER: A Parsimonious Rule-Based Model for Sentiment Analysis of Social Media Text”, *ICWSM*, vol. 8, no. 1, pp. 216-225, May 2014.

[2] V. Bonta, N. Kumaresh, and N. Janardhan, “A Comprehensive Study on Lexicon Based Approaches for Sentiment Analysis,” Asian Journal of Computer Science and Technology, vol. 8, no. S2, pp. 1–6, Mar. 2019, doi: 10.51983/ajcst-2019.8.s2.2037.

[3] B. Zhao, “Web Scraping,” Encyclopedia of Big Data, pp. 1–3, 2017, doi: 10.1007/978-3-319-32001-4_483-1.

[4] Deepak Kumar Mahto and Lisha Singh, “A dive into Web Scraper world”, Institute of Electrical and Electronics Engineers (IEEE), 2016

[5] “Web Scraping: Applications and Scraping Tools,” International Journal of Advanced Trends in Computer Science and Engineering, vol. 9, no. 5, pp. 8202–8206, Oct. 2020, doi: 10.30534/ijatcse/2020/185952020.

[6] S. M.K., “Social Media Sentiment Analysis for Opinion Mining,” International Journal of Psychosocial Rehabilitation, vol. 24, no. 5, pp. 3672–3679, Apr. 2020, doi: 10.37200/ijpr/v24i5/pr202075.

[7] R. E. Mitchell, Web scraping with Python : collecting more data from the modern web. Sebastopol, Ca: O’reilly Media, 2018.

[8] B. Chauhan Vipul Kumar, G. Ashish, and Amita, “ISSN: 2454-132X Impact factor: 4.295 Twitter Sentiment Analysis Using Vader.” [Online]. Available: https://www.ijariit.com/manuscripts/v4i1/V4I1-1307.

[9] S. Rao, N. Monica, P. Nikhila, T. Tejasri, and B. Maram, “Positivity Calculation using Vader Sentiment Analyser.” Accessed: Jul. 18, 2022. [Online]. Available: http://ijeais.org/wp-content/uploads/2020/3/IJAER200303.pdf

[10] TY - JOUR AU - Adarsh, R. AU - Patil, A. AU - Rayar, S. AU - Veena, K.M. PY - 2019/03/01 SP - 540 EP - 543 T1 - Comparison of VADER and LSTM for sentiment analysis VL - 7 JO - International Journal of Recent Technology and Engineering ER

[11] “Document Object Model,” Wikipedia, Oct. 09, 2020. https://en.wikipedia.org/wiki/Document_Object_Model

[12]“Lexicon-Based Approach - an overview | ScienceDirect Topics,” www.sciencedirect.com. https://www.sciencedirect.com/topics/computer-science/lexicon-based-approach

[13] I. Dongo, Y. Cadinale, A. Aguilera, F. Martínez, Y. Quintero, and S. Barrios, “Web Scraping versus Twitter API,” Proceedings of the 22nd International Conference on Information Integration and Web-based Applications & Services, Nov. 2020, doi: 10.1145/3428757.3429104.