SEMANTIC ANALYSIS OF TWEETS USING LEXICAL AND SEMANTIC CONCEPTS

Identifying the response of the public for an entity

S Sowmiya Sree	Aparna S S	Juanita J
2018103598	2018103008	2018103544

ABSTRACT

The project focuses on analyzing the reach of an entity to the public. Twitter being one of the most preferred microblogging site, is chosen for the analysis. By finding out the overall sentiment from the tweets associated with the concerned entity, it becomes handy to calculate the reach of the entity. Such analysis is very crucial for the present world where business companies are hugely investing and rapidly growing. The tweets are directly fetched from the platform from which the sentiments are fetched and analyzed. This is followed by building a predictive model which predicts the sentiment from a given tweet.

INTRODUCTION

Twitter is one of the micro-blogging sites that have scored its place in the ever increasing social networking epoch. The present scenario of the microblogging sites have made the opinion forming and advancement very efficient. The public opinion formed using tweets is direct. People use emoticons, short hands etc to write the tweets. People tweets can be used to analyze thoroughly and effectively to shape up the opinions, thus the opinion mining. At present, sentiment analysis is one of the most important field in the development of the organization and thus many technologies have been applied to automate the work as in Natural Language Processing(NLP) and Machine learning algorithms(ML).

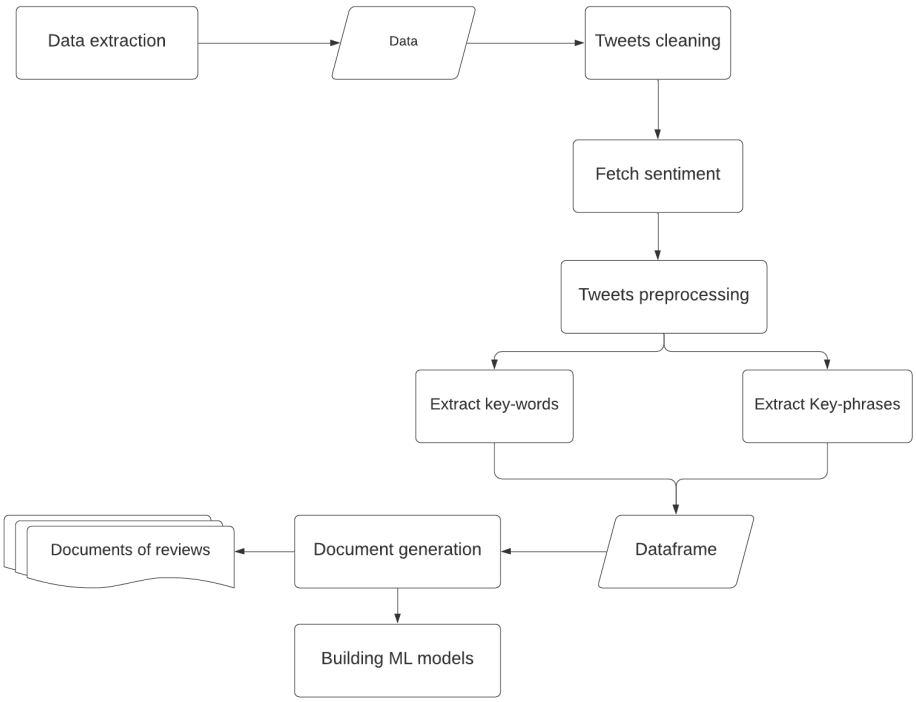
PROBLEM STATEMENT

Until now the work of the sentimental analysis is largely limited review of movies but twitter data analysis is gaining much recognition in the present scenario may be because it provides the latest and the most instant opinion of the public on almost all the latest fields of interest. Twitter sentiment analysis technology provides the methods to survey public emotion about the events or products related to them. Sentiment analysis over Twitter offer organizations a fast and effective way to monitor the publics' feelings towards their brand, business, directors, etc. The project focuses on deriving the sentiments related to a specific entity which implies how the entity has been received by the public.

ENTITY CHOSEN

The entity chosen for the analysis is "***iOS 14" .*** It is the fourteenth and current major release of the iOS mobile operating system developed by Apple Inc. for their iPhone and iPod Touch lines. It was released on September 16, 2020. This is the entity whose reach to the public is going to be analyzed via sentiment analysis of tweets.

ARCHITECTURE



Module 1 - Data extraction
Module 2 - Tweets cleaning
Module 3 - Fetching sentiment
Module 4 - Tweets pre-processing
Module 5 - Document generation
Module 6 - Building ML models

MODULE 1 - Data extraction

The readily available dataset may not provide sufficient information regarding the entity we are interested in. Therefore live tweets are extracted from the Twitter platform via the most popular Application Interface (API) "**Tweepy"**.

Tweepy is an open source Python package that gives you a very convenient way to access the Twitter API with Python. Tweepy includes a set of classes and methods that represent Twitter's models and API endpoints, and it transparently handles various implementation details, such as:

- Data encoding and decoding
- HTTP requests
- Results pagination
- OAuth authentication
- Rate limits

- Streams

The OAuth and API functionalities offered by the tweepy are used for the project which is explained as follows.

OAuth - The Twitter API requires that all requests use OAuth to authenticate. It provides an *OAuthHandler* class that you can use to set the credentials to be used in all API calls. So the required authentication credentials are created to be able to use the API. These credentials are four text strings:

1. Consumer key
2. Consumer secret
3. Access token
4. Access secret

API functionality - The API class has many methods that provide access to Twitter API endpoints. Using these methods, the Twitter API's functionality can be accessed. The *Search* method is used to extract the most recent tweets for a given query.

Using the above functionalities the unity can be provided in the *Query* section along with the maximum limit of the number of tweets to be extracted. 20,000 tweets regarding iOS 14 were extracted for the analysis.

MODULE 2 - Cleaning data

The raw tweets needs some cleansing activity to get rid of the unwanted details for the sentiment extraction. This process is crucial due to the fact that certain redundant details may produce generalized sentiment instead of accurate ones. The following details are removed from the tweets.

1. ***'@names'*** - The mentions of accounts are not so essential for the problem statement
2. ***'http & https'*** - The hyperlinks are unnecessary details
3. ***Empty text*** - As the name suggests, empty texts are of no use to the analysis. Instead they just consume space
4. ***Punctuations, Numbers & Special characters*** - All the numbers, punctuations and special characters have no special purpose in the analysis. It is important to note that 'hashtags' are retained to extract the key information from the tweets.
5. ***Duplicate rows*** - The repetitive tweets are just space consuming and misleading

Language translation

The cleaned tweets are still in different languages since no specification regarding language was given at the time of extraction of tweets. This is to analyze the reviews of the entity from all the people irrespective of language. People might have expressed their approval or disapproval in their native language which shouldn't be left out. It's equally important to translate the tweets to English Since the sentiment analyzer (explained in upcoming sections) captures the sentiment to the best possible accuracy in the universal language.

Google Transalte API - Google Translate API is a simple API that allows you to translate an arbitrary string of text from one language to the other. The API supports two endpoints, “detect” and “translate”. As the name suggests, one is for detecting the language, and the other is for translating from one language to another.

```
tweet = tweets_df['tidy_tweets'][46]
print(tweet)
sid = SentimentIntensityAnalyzer()
polarity_scores = sid.polarity_scores(tweet)
print(polarity_scores)
```

```
iOS 14の「メモ」アプリで4つの新機能を使いこなそう #it #feedly
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0}
```

```
tweet = tweets_df['tidy_tweets'][46]
print(tweet)
translation = translator.translate(tweet)
tweet = translation.text
print(tweet)
sid = SentimentIntensityAnalyzer()
polarity_scores = sid.polarity_scores(tweet)
print(polarity_scores)
```

```
iOS 14の「メモ」アプリで4つの新機能を使いこなそう #it #feedly
Take full advantage of four new features in the iOS 14 Memo app #it #feedly
{'neg': 0.0, 'neu': 0.875, 'pos': 0.125, 'compound': 0.25}
```

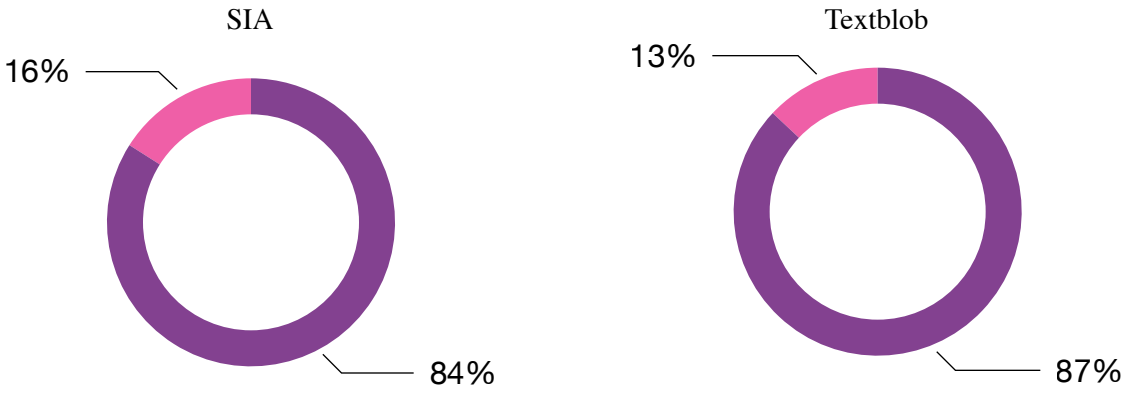
The necessity to translate the tweets from native languages to universal language - English. Such translation returns specific polarity scores

MODULE 3 -Fetching sentiment

The sentiment analyzers can extract accurate sentiments from the cleaned tweets. The sentiments extracted are not only used to analyze the overall response of the public but also to extract the targets for the inputs(tweets) which will be used by the machine learning (ML) models to learn the patterns of positive and negative tweets. Two libraries from the python are used to extract the sentiment which are explained as follows.

VADER’s SentimentIntensityAnalyzer(SIA) - VADER is the library available in the Natural Language ToolKit (NLTK) which is a dedicated tool for processing texts. SentimentIntensityAnalyzer is a specific function provided by the VADER. VADER belongs to a type of sentiment analysis that is based on lexicons of sentiment-related words.

In this approach, each of the words in the lexicon is rated as to whether it is positive or negative, and in many cases, **how** positive or negative. It produces four sentiment metrics from these word ratings which are *Positive, Neutral, Negative* and *Compound*. The first three metrics represent the proportion of the text that falls into those categories. The final metric, the compound score, is the sum of all of the lexicon ratings which have been standardised to range between -1 and 1.



TextBlob - TextBlob is a Python library for processing textual data. It provides a simple API for diving into common natural language processing (NLP) tasks such as part-of-speech tagging, noun phrase extraction, sentiment analysis, classification, translation, and more.

The sentiment object of textblob returns a tuple that contains polarity and subjectivity of the review. Polarity is considered. The value of polarity can be between -1 and 1 where the reviews with negative polarities have negative sentiments while the reviews with positive polarities have positive sentiments.

SIA vs TextBlob

SIA was given higher priority among the two techniques used. Vader Sentiment Analysis works better for with texts from social media and in general as well. When it comes to analyzing comments or text from social media, the sentiment of the sentence changes based on the emoticons. Vader takes this into account along with slang, capitalization etc and hence a better option when it comes to tweets analysis and the associated sentiments. Also, SIA identified more number of negative sentiments than Textblob. This is very important aspect to consider from the viewpoint of class balance which will be dealt in the ML section of the project.

MODULE 4 - Tweets preprocessing

The Key-words and key-phrases are extracted from the cleaned tweets. These are exported to the document for the entity's purposes as well as used in ML models to predict the sentiments of tweets

4. 1 Extraction of key-words

The key-words from the tweets are extracted by following a series of techniques as follows.

- ***Removal of stop-words*** - Stop-words are the English words which does not add much meaning to a sentence. They can safely be ignored without sacrificing the meaning of the sentence. The stop words are removed from the tweets using the 'Stopwords' library available in the NLTK toolkit.

- ***Tokenize*** - The tweets free from stop-words are tokenized (the task of chopping the text up into pieces, called tokens) by a simple python code.

- ***Conversion of tokens to lemma*** - Lemmatization is the process of grouping together the different inflected forms of a word so they can be analysed as a single item. The tokens obtained from the tweets are lemmatized using the the *Wordlemmatizer* library from the *Stem* package available in the NLTK toolkit.
- ***Join all tokens into sentences*** - The lemmatized tokens are joined back to sentences and stored as a separate detail in the dataframe. It is important to note that these sentences may not have proper semantics since its just a combination of all possible keywords from the tweets

Positive words

Negative words

4. 2 Extraction of key-phrases

The key-phrases are extracted from the tweets in two phases explained as follows.

- ***Grammar definition for the parse tree***

```
sentence_re = r'(?:(?:[A-Z])(?:.[A-Z])+.?)|(?:\w+(?:-\w+)*)|(?:\$?\d+(?:.\d+)?%?)|(?:...|)(?:[][.,;"\'?():-_`])'
grammar = r"""
    NBAR:
        {<NN.*|JJ>*<NN.*>}  # Nouns and Adjectives, terminated with Nouns

    NP:
        {<NBAR>}
        {<NBAR><IN><NBAR>}  # Above, connected with in/of/etc...
"""
chunker = nltk.RegexpParser(grammar)
```

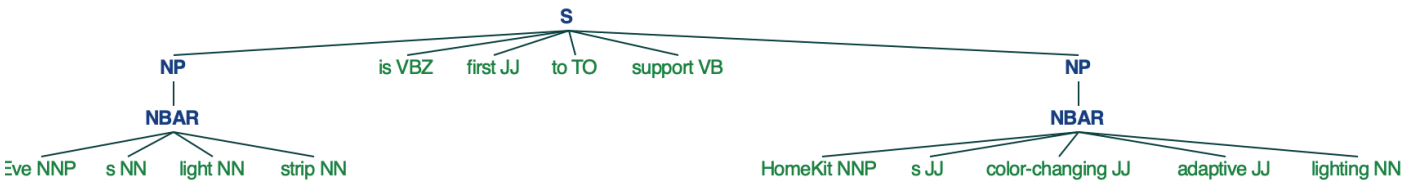
The regular expression to define the pattern of valid tokens.
The grammar for the parser on which the token and associated POS is fit and key-phrases are extracted.

A parse tree is constructed for every tweet from which the key-phrases are extracted.

1. *Building tokenizer* - Instead of simply tokenizing, a pattern has been pre-defined by means of a regular expression. The strings are tokenized in such a way that they match these patterns by the 'regexp_tokenize' function from NLTK

2. *Obtaining the POS (Parts of Speech)* - It has been proved that using POS to derive the key-phrases which are used to fit the predictive models improve their performance to a great extent. The 'pos_tag' function in the 'tag' library of NLTK returns the tokens with their POS(Parts Of Speech) as pairs.

3. *Fitting tokens in RegexpParser* **-** The parser is defined by a grammar in which the nouns and adjectives are terminated with nouns. The derived POS are combined with the token and fitted into the parser as <token, POS> pair.



Sample parser of a tweet constructed using the grammar mentioned above

4. *Obtaining key-phrases* - The parser is then passed into a helper class function elaborated as follows

- ***The helper class***

The helper class is simply a python class which encompasses the following functions to extract the key-phrases from the parse tree.

1. *Initialization of lemmatized and stemmer* - Initializes the the lemmatizer and stemmer from the NLTK. PorterStemmer class chops off the 'es' from the word. On the other hand, WordNetLemmatizer class finds a valid word.

2. *Extract leaves* - Splits the tree into subtrees and yields the leaves of the parse tree which are the NounPhrases (NP).

3. *Normalize* - The extracted words are normalized to lowercase and stems and are lemmatized.

4. *Acceptable word* - Checks conditions for acceptable word: length, stopword. The length of the word can be increased when considering larger phrases .

5. *Get terms* - Makes use of the above functions to yield terms which form the key-phrases.

MODULE 5 - Document generation

Having processed the tweets and extracted the sentiments, the tweets are now segregated into positive and negative reviews. The segregated tweets along with the key-phrases and key-words are incorporated into documents with 'csv' extension. The purpose of generating files is to attain the purpose of the project. The analysis is performed to identify the reach of the entity to the public. So the entity must get to know the reach to improvise its services. The 'positive_reviews.csv' helps the entity to further concentrate on sections that the public enjoys and 'negative_reviews.csv' is very crucial for the entity to immediately work on the troublesome aspects of the entity.

tweets_en	sentistrength
iOS 14 I feel that the battery is exhausted.	{'neg': 0.263, 'neu': 0.737, 'pos': 0.0, 'comp...
iPad, it should be able to watch 4K on YouTube...	{'neg': 0.0, 'neu': 0.901, 'pos': 0.099, 'comp...
I was updated to ios 14... but I can't decorat...	{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound...
how come people display the menu on ios 14 is ...	{'neg': 0.0, 'neu': 0.775, 'pos': 0.225, 'comp...
I wonder if iOS 14.2 will come out 10 or earli...	{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound...
...	...
We're glad you reached out. Earlier today, we ...	{'neg': 0.0, 'neu': 0.784, 'pos': 0.216, 'comp...
Vill be broadcast on SBS Power FM 'Cultwo Show...	{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound...
This issue has been fixed in iOS 14.2C and sho...	{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound...
I actually got this message long time ago in t...	{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound...
theres also a translate app built into ios 14😝	{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound...

Positive_reviews.csv

tweets_en	sentistrength
Safari on iOS 14: How to translate websites in...	{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound...
How to Fix iOS 14 & iPadOS 14 Wi-Fi Problems	{'neg': 0.231, 'neu': 0.769, 'pos': 0.0, 'comp...
❌ Transport for Wales update: 13:16 Maesteg to...	{'neg': 0.105, 'neu': 0.895, 'pos': 0.0, 'comp...
Suchitra is the iOS 14.1 of Vanitha. A dangero...	{'neg': 0.129, 'neu': 0.871, 'pos': 0.0, 'comp...
How to Fix iOS 14 & iPadOS 14 Wi-Fi Problems	{'neg': 0.231, 'neu': 0.769, 'pos': 0.0, 'comp...
...	...
Apple Seeds New iOS 14.2 Versions Which Stops ...	{'neg': 0.118, 'neu': 0.882, 'pos': 0.0, 'comp...
Appleeleases iOS 14.2 GM Fixing Annoying Bug	{'neg': 0.31, 'neu': 0.69, 'pos': 0.0, 'compou...
fuck beta iOS 14 like why has it taken the ent...	{'neg': 0.159, 'neu': 0.727, 'pos': 0.114, 'co...
Woke up to iOS 14 update. This is mad cool. ❄️	{'neg': 0.221, 'neu': 0.621, 'pos': 0.159, 'co...
Imma get the ios 14 later- Me still scared ;-;"	{'neg': 0.244, 'neu': 0.756, 'pos': 0.0, 'comp...

Negative_reviews.csv

MODULE 6 - Building predictive models

The machine learning models are built using two approaches namely ***Key-words*** and ***Key-phrases***. After evaluating the performance of the models with the two inputs, the one that produces best result is finalized. The tasks involved in building the classifier model are as follows .

1. ***Data preprocessing -*** The retrieved data so far contains the raw tweets, translated tweets, sentiment, strength of sentiment, key-phrases and key-words. The target variable I.e, Sentiment is converted into numerical values ('pos' - 1 and 'neg' - 0) for the ease of computation .

tweets_en	absolute_tidy_tweets	key_phrases	sentiment	sentistrength
iOS 14 I feel that the battery is exhausted.	iOS I feel battery exhausted	[]	pos	{'neg': 0.263, 'neu': 0.737, 'pos': 0.0, 'comp...
iPad, it should be able to watch 4K on YouTube...	iPad able watch K YouTube long time good long ...	[ipad, 4k, youtube, long time, long time, 4k]	pos	{'neg': 0.0, 'neu': 0.901, 'pos': 0.099, 'comp...
I was updated to ios 14... but I can't decorat...	I updated io I cant decorate	[]	pos	{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound...
how come people display the menu on ios 14 is ...	come people display menu io good	[]	pos	{'neg': 0.0, 'neu': 0.775, 'pos': 0.225, 'comp...
I wonder if iOS 14.2 will come out 10 or earli...	I wonder iOS come earlier	[]	pos	{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound...
I see ... Wai recently upgraded to iOS 14, and...	I see Wai recently upgraded iOS added feature ...	[wai, ' t need]	pos	{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound...
#Flossy 😍🤟🏻 Great theme for iOS 14 #WidgetTool...	Great theme iOS BG	[😍🤟🏻 great theme, ios 14 bg]	pos	{'neg': 0.0, 'neu': 0.661, 'pos': 0.339, 'comp...
APPLE MUSIC 🔥 (support ios 14) 3 bulan : 25.00...	APPLE MUSIC support io bulan io rb Garansi Akt...	[apple music 🔥 , garansi | aktivasi, perpanjang...	pos	{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound...
Safari on iOS 14: How to translate websites in...	Safari iOS How translate website German	[safari]	neg	{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound...
How to Fix iOS 14 & iPadOS 14 Wi-Fi Problems	How Fix iOS amp iPadOS WiFi Problems	[ipados 14 wi-fi problems]	neg	{'neg': 0.231, 'neu': 0.769, 'pos': 0.0, 'comp...

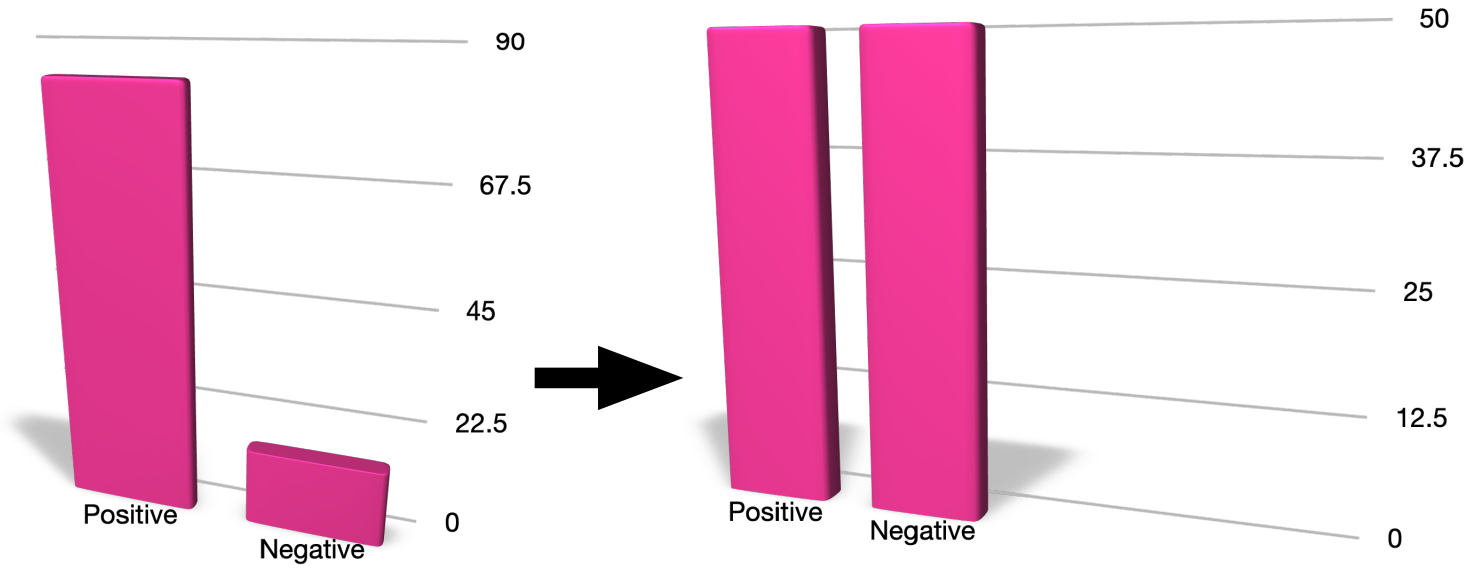
Data extracted from tweets

2. ***Feature extraction -*** Both the inputs chosen (key-words and key-phrases) are in texual format which the predictive model cannot process. Hence these texts are converted to sparse matrix containing 'float' values which store the elements in compressed sparse row format. Two techniques used for the purpose are *Bag Of Words (BOW)* and *Term Frequency - Inverse Document Frequency (TF - IDF)*.

Bag of Words (BOW) - It's a collection of words to represent a sentence with word count and mostly disregarding the order in which they appear.

Term Frequency - Inverse Document Frequency (TF - IDF) - A technique to quantify a word in documents, the weight of each word is computed which signifies the importance of the word in the document.

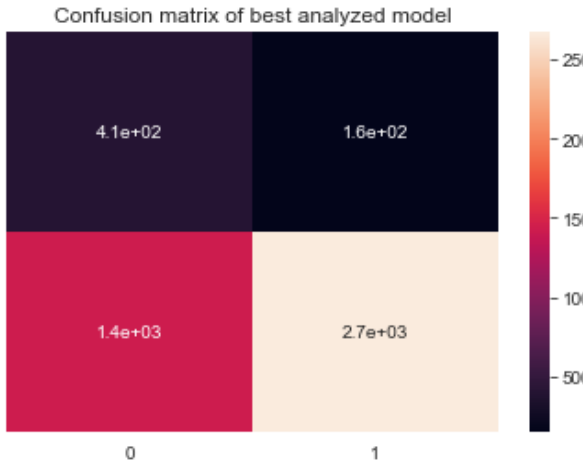
3. ***Class balance -*** The target classes which are 1 (positive sentiment) and 0 (negative sentiment) are highly imbalanced. Such huge imbalance would produce results in the favor of the majority class (positive class) considering the minority class (negative sentiment) as noise. This may produce poor results or lead to overfitting. To avoid such issues, the classes are balanced by samping methods. The sampling methods either increase the count of minority class or decrease the count of majority class or do both in equal proportions. The ***combined sampling technique*** provided by the SMOTETomek function from the combine_sample library was chosen to balance the classes.



Training the classifier - The data is split into
train and test sets for training and evaluation purposes. The train data is fit into a base classifier model. Through this the model learns the patterns of positive and negative tweets with which it can predict tweets in future. The classifier model chosen is the ***Support Vector Machine (SVM).***

A support vector machine (**SVM**) is a supervised machine learning model that uses classification algorithms for two-group classification problems. The algorithm creates an optimal hyperplane which separates the data into classes. It is highly preferred by many as it produces significant accuracy with less computation power. It transforms the data in lower dimensions to higher dimensions are employs separate hyperplanes to segregate the classes along with the maximum marginhyperplane.

4. ***Evaluating the classifier*** - The trained model is evaluated with the test set and the results were most accurate in the model which had key-phrases as the input and that with the one which extracted features using TF-IDF technique. It had the maximum overall accuracy of 88% (cumulative accuracy of metrics like Precision, Recall and F1-score on both the classes.)



The confusion matrix plots the

True Positives (TP) - Bottom right
False Positives (FP) - Bottom left
True Negatives (TN) - Top left
False Negatives (FN) - Top right

The scale on the right helps to identify the intensity of correctness and mistake.

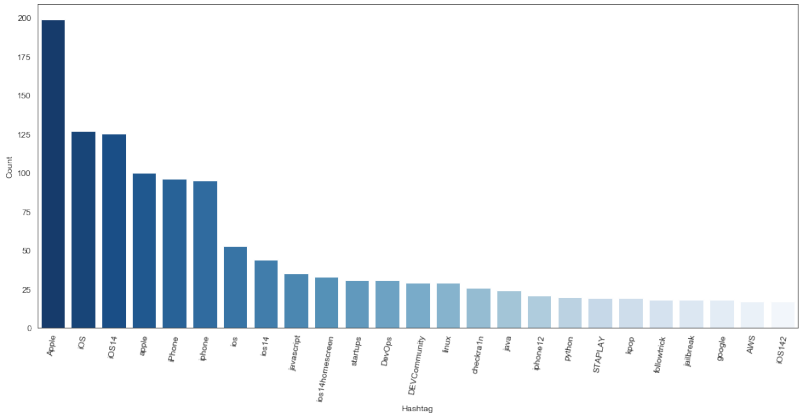
It can be observed that fairly better TP and TN counts are achieved

RESEARCH AGENDA

The analysis can be improvised further by extracting all the tweets regarding the entity instead of specifying a maximum limit. Further new methodologies can be employed to properly count the senti-strength of the emojis in the tweets (Although the employed sentiments analyzer takes them into account). Strong classifier models or even deep learning techniques can be employed to build more accurate predictive models.

INFERENCES AND CONCLUSION

The tweets concerned with the entity have been extracted legally from reliable social media and processed to suite the upcoming processing tasks. Further the sentiments of the tweets are fetched to best possible precision. The overall reaction of the public for the entity chosen (iOS 14) is ***Positive*** though there are some serious issues to be addressed. The two documents generated can be used to focus on the dissatisfied area. Further the best concluded predictive models designed is expected to predict the tweets with a decent accuracy of 88%. It is noticed that, despite the broad analysis, there is still a need for an in-depth investigation of specific directions. Therefore, a research agenda is provided, illustrating potential future work demands.



The most commonly used hashtags in the tweets

REFERENCES

1. **Alleviating data sparsity for twitter sentiment analysis** by ***Saif, Hassan; He, Yulan and Alani, Harith*** published in *2012*.

2. **Real Time Sentiment Analysis of Political Twitter Data Using Machine Learning Approach** by ***Joylin Priya Pinto*** and ***Vijaya Murari T*** published in *2019*