**\*- important**

**\*\*\_Not so important/optional to include in report**

**Tweepy API:-**

\*Twitter is a popular social network where users share messages called tweets. Twitter allows us to mine the data of any user using Twitter API or Tweepy. The data will be tweets extracted from the user. Tweepy is open-sourced, hosted on [GitHub](https://github.com/tweepy/tweepy) and enables Python to communicate with Twitter platform and use its API.

**\*\***At the time of writing, the current version of tweepy is 1.13. It was released on January 17, and offers various bug fixes and new functionality compared to the previous version. The 2.x version is being developed but it is currently unstable so a huge majority of the users should use the regular version.

Installing tweepy is easy, it can be cloned from the Github repository:

|  |  |
| --- | --- |
| 1  2 | git clone https://github.com/tweepy/tweepy.git  python setup.py install |

Or using easy install:

|  |  |
| --- | --- |
| 1 | pip install tweepy |

Either way provides you with the latest version

**Using tweepy:-**

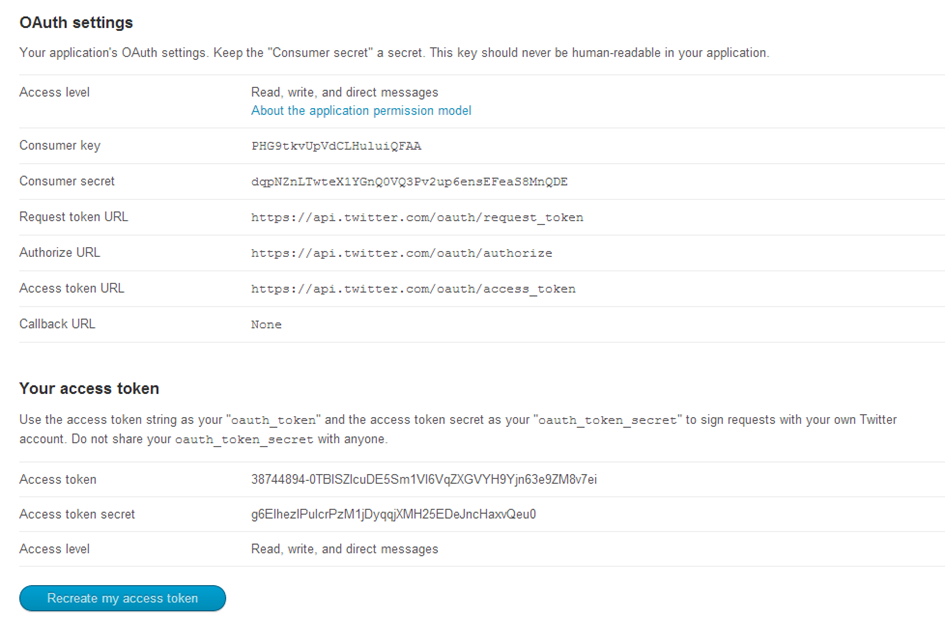
\*Tweepy supports accessing Twitter via Basic Authentication and the newer method, OAuth. Twitter has stopped accepting Basic Authentication so OAuth is now the only way to use the Twitter API.

\*The main difference between Basic and OAuth authentication are the consumer and access keys. With Basic Authentication, it was possible to provide a username and password and access the API, but since 2010 when the Twitter started requiring OAuth, the process is a bit more complicated.

OAuth is a bit more complicated initially than Basic Auth, since it requires more effort, but the benefits it offers are very lucrative:

* Tweets can be customized to have a string which identifies the app which was used.
* It doesn’t reveal user password, making it more secure..

\*After logging in to the portal, and going to "Applications", a new application can be created which will provide the needed data for communicating with Twitter API.



\*This is a screen which has all of the data needed to talk to Twitter network. It is important to note that by default, the app has no access to direct messages, so by going to the settings and changing the appropriate option to “Read, write and direct messages”, you can enable your app to have access to every Twitter feature.

**\*Steps to obtain keys:**   
– Login to twitter developer section  
– Go to “Create an App”  
– Fill the details of the application.  
– Click on Create your Twitter Application  
– Details of your new app will be shown along with consumer key and consumer secret.  
– For access token, click ” Create my access token”. The page will refresh and generate access token.

\*Now in order to authorize our app to access Twitter on our behalf, we need to use the OAuth Interface. Tweepy provides the convenient Cursor interface to iterate through different types of objects. Twitter allows a maximum of 3200 tweets for extraction.

\*Our Code for geeting Sabarmalai tweets:-



Conclusion For Tweepy:-

\*\*To sum up, tweepy is a great open-source library which provides access to the Twitter API for Python. Although the documentation for tweepy is a bit scarce and doesn't have many examples, the fact that it heavily relies on the Twitter API, which has excellent documentation, makes it probably the best Twitter library for Python, especially when considering the Streaming API support, which is where tweepy excels. Other libraries like python-twitter provide many functions too, but the tweepy has most active community and most commits to the code in the last year.

Additional resources used for tweepy can be obtained here:

* [Twitter Developers](https://dev.twitter.com/)
* [Official tweepy documentation](http://pythonhosted.org/tweepy/html/index.html)
* [The tweepy github page](https://github.com/tweepy/tweepy/)
* [mailing list](https://groups.google.com/forum/#!forum/tweepy)
* IRC, Freenode.net #tweepy.

**Existing System:-**

The existing system works only on the dataset which is constrained to a particular topic. The existing systems also do not determine the measure of impact the results determined can have on the particular field taken into consideration and it does not allow retrieval of data based on the query entered by the user i.e. it has constrained scope. In simple words, it works on static data rather than dynamic data. Unsupervised algorithms like Vector Quantization, are used for data compression, pattern recognition, facial and speech recognition, etc and therefore cannot be used in determining sentiment in twitter data. Apriori algorithm fails to handle large datasets and as a result can generate faulty results

**Proposed System:-**

In the proposed system, we will retrieve tweets from twitter using twitter API based on the query. The collected tweets will be subjected to preprocessing. We will then apply the supervised algorithm on the stored data. The supervised algorithm used in our system is Support Vector Machine (SVM). The results of the algorithms i.e. the sentiment will be represented in graphical manner (pie charts/bar charts). The proposed system is more effective than the existing one. This is because we will be able to know how the statistics determined from the representation of the result can have an impact in a particular field.

|  |  |
| --- | --- |
| **Existing System** | **Proposed System** |
| Existing system takes a stored dataset on a particular topic into consideration. | Proposed system will gives you the freedom to choose the data of any topic. |
| Existing system does not provide accurate feature selection(Bag of words) | Proposed system will provide accurate feature selection(TF-IDF) |
| Existing system does not allow the retrieval of data based on the query entered by user. | Proposed system allows retrieval of data based on the query entered by the user. |
| Comes with Two classes(Positive/Negative) | Comes with Three classes(Positive/Negative/Neutral) |
|  |  |

**Future Work:-**

From future perspective, we would like to extend this project by implementing some machine learning algorithms for applications like election results, product ratings, movies' outcomes and running the project on clusters to expand its functionalities. Moreover, we would like to make a web application for users to input keywords and get analyzed results. In this project, we have worked only with unigram models, but we would like to extend it to bigram and further which will increase linkage between the data and provide accurate sentiment analysis results. Computation of overall tweet score can be done for a single keyword which can provide an overall sentiment of public regarding a topic.

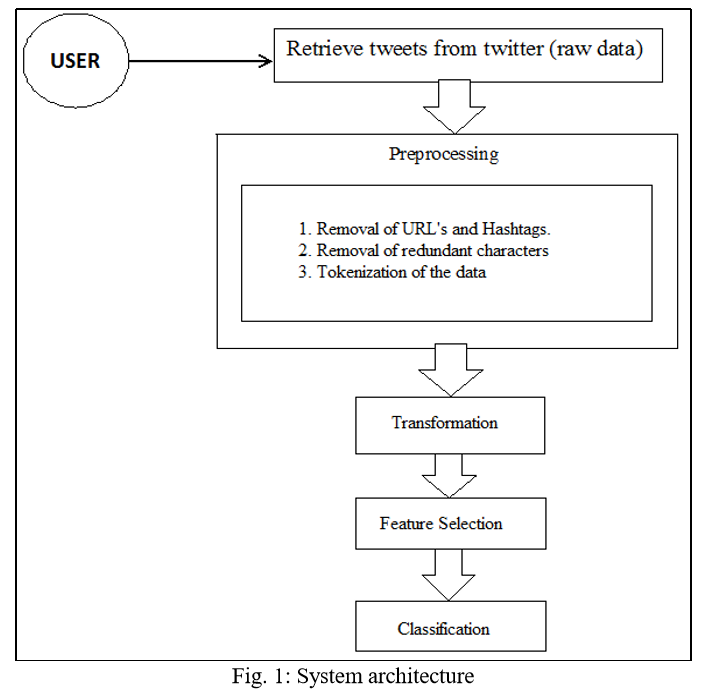
**Conclusion**:-

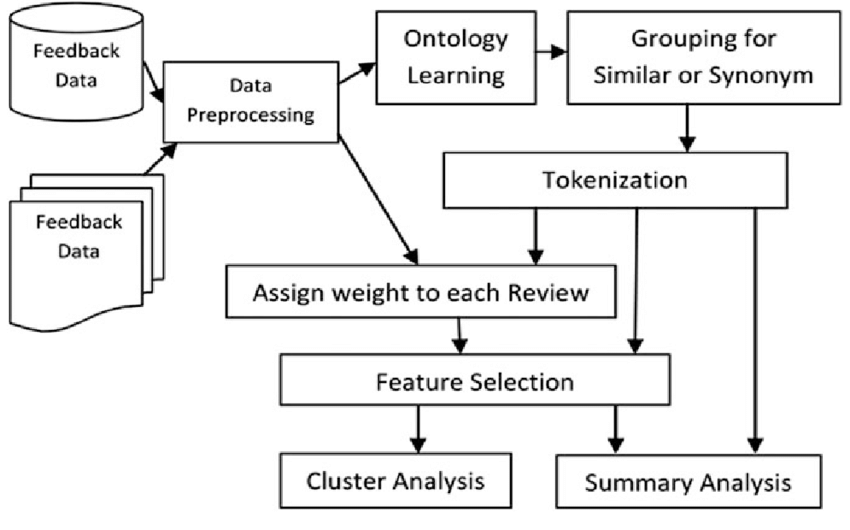
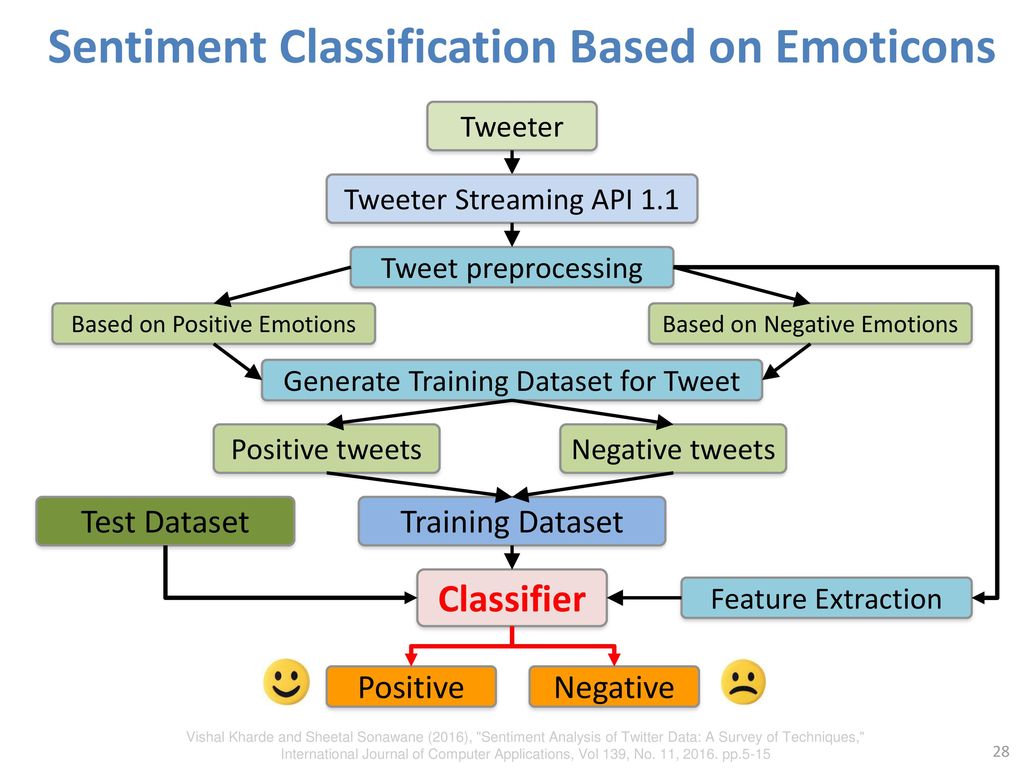
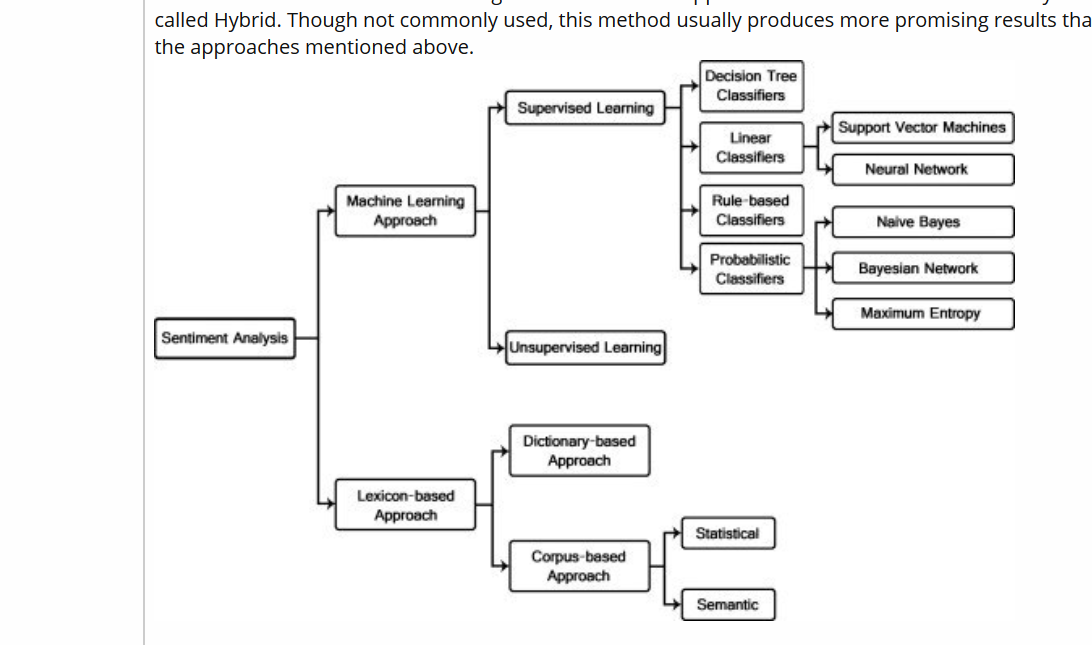
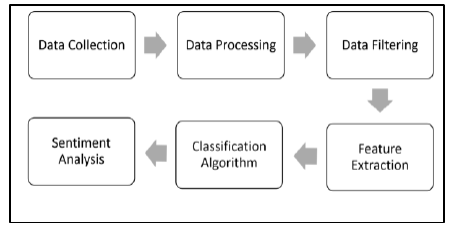
The task of sentiment analysis, especially in the domain of micro-bloging, is still in the developing stage and far from complete. So we propose a couple of ideas which we feel are worth exploring in the future and may result in further improved performance. Twitter sentiment analysis is developed to analyze customers perspectives toward the critical to success in the marketplace. The program is using a machine-based learning approach which is more accurate for analyzing a sentiment; together with natural language processing techniques will be used. We presented results for sentiment analysis on Twitter. We report an overall accuracy for 3-way classification tasks: positive versus negative versus neutral.

**References:-**

* <https://towardsdatascience.com/sentiment-analysis-concept-analysis-and-applications->
* <https://blog.algorithmia.com/introduction-sentiment-analysis/>
* <https://www.researchgate.net/publication/301408174_Twitter_sentiment_analysis>
* P. Lai, “ExtractingStrongSentimentTrendfromTwitter”. Stanford University, 2012
* Dr. Khalid N. Alhayyan & Dr. Imran Ahmad “Discovering and Analyzing Important Real-Time Trends in Noisy Twitter Stream” n.p
* <https://tweepy.readthedocs.io/en/v3.6.0/>
* <https://towardsdatascience.com/the-real-world-as-seen-on-twitter-sentiment-analysis-part-two-3ed2670f927d>
* <https://developer.twitter.com/en/apps>
* https://www.pantechsolutions.net/twitter-sentiment-analysis-using-machine-learning-algorithms-on-python

**Extra Pictures:-**

****

****