RCC Institute Of Information Technology

Seminar – Team Paper Leading to Project [PGCSE 292]

COVID-19 Sentiment Analysis using Natural Language Processing

Presented By:

Arunava Kumar Chakraborty
Course – MTech (CSE)
Year – 1st ; Sem – 2nd
Roll – MCS2019/001

Introduction

Sentiment Analysis is a text analysis method that detects polarity (e.g. A positive or negative opinion) within text, whether a whole document, paragraph, sentence, or clause.

Sentiment Analysis is a type of Data Mining that measures the inclination of people's opinions through natural language processing (NLP), computational linguistics and text analysis, which are used to extract and analyze subjective information from the web — mostly social media and similar sources.

The analyzed data quantifies the general public's sentiments or reactions toward certain products, people or ideas and reveal the contextual polarity of the information. Sentiment analysis is also known as opinion mining.

Objectives

- **Social Media Monitoring :** In today's day and age, brands of all shapes and sizes have meaningful interactions with customers, leads, and even competition on social networks like Facebook, Twitter, and Instagram. By using sentiment analysis on social media, we can get incredible insights into the quality of conversation that's happening around a brand.
- **Brand Monitoring :** Analyze news articles, blog posts, forum discussions, and other texts on the internet over a period of time to see sentiment of a particular audience. Automatically alert designated team members of online mentions that concern their area of work.
- **Customer Feedback :** Target individuals to improve their service. By automatically running sentiment analysis on incoming surveys, we can detect customers who are 'strongly negative' towards our product or service, so we can respond to them right away.
- **Customer Service :** Sentiment Analysis can be used to automate text classification all incoming customer support queries, route queries to specific team members best suited to respond.
- **Market Research :** Sentiment Analysis empowers all kinds of market research and competitive analysis. It can be used to analyze formal market reports or business journals for long-term, broader trends.

Working Principle

Sentiment analysis is done using algorithms that use **Text Analysis** and natural language processing to classify words as either positive, negative, or neutral. This allows companies to gain an overview of how their customers feel about the brand.

Sentiment Analysis uses various natural language processing (NLP) methods and algorithms.

The main types of algorithms used include:

- **Rule-based** systems that perform sentiment analysis based on a set of manually crafted rules.
- **Automatic** systems that rely on machine learning techniques to learn from data.
- **Hybrid** systems that combine both rule-based and automatic approaches.

Rule based Approach

Usually, a Rule-Based system uses a set of human-crafted rules to help identify subjectivity, polarity, or the subject of an opinion.

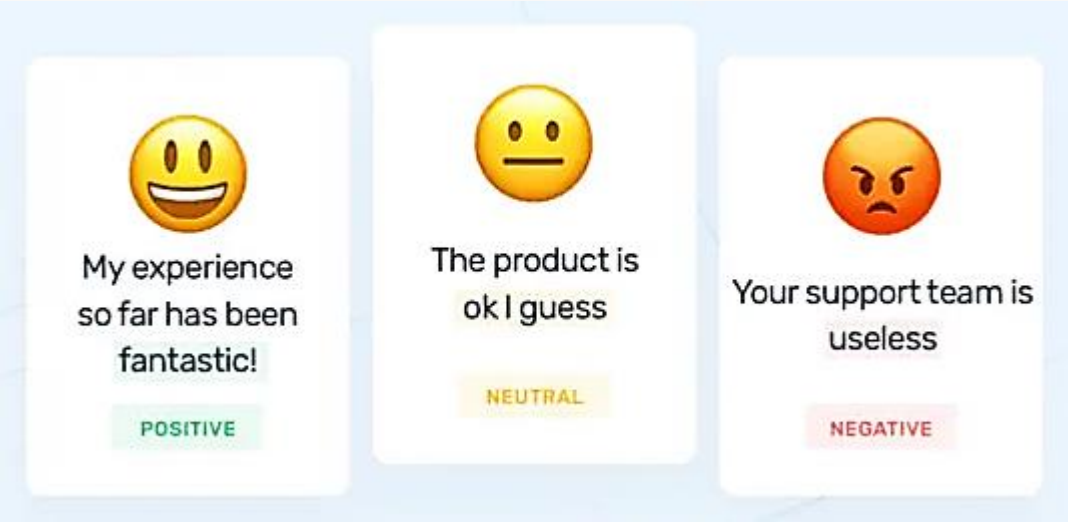
These rules may include various techniques developed in computational linguistics, such as:

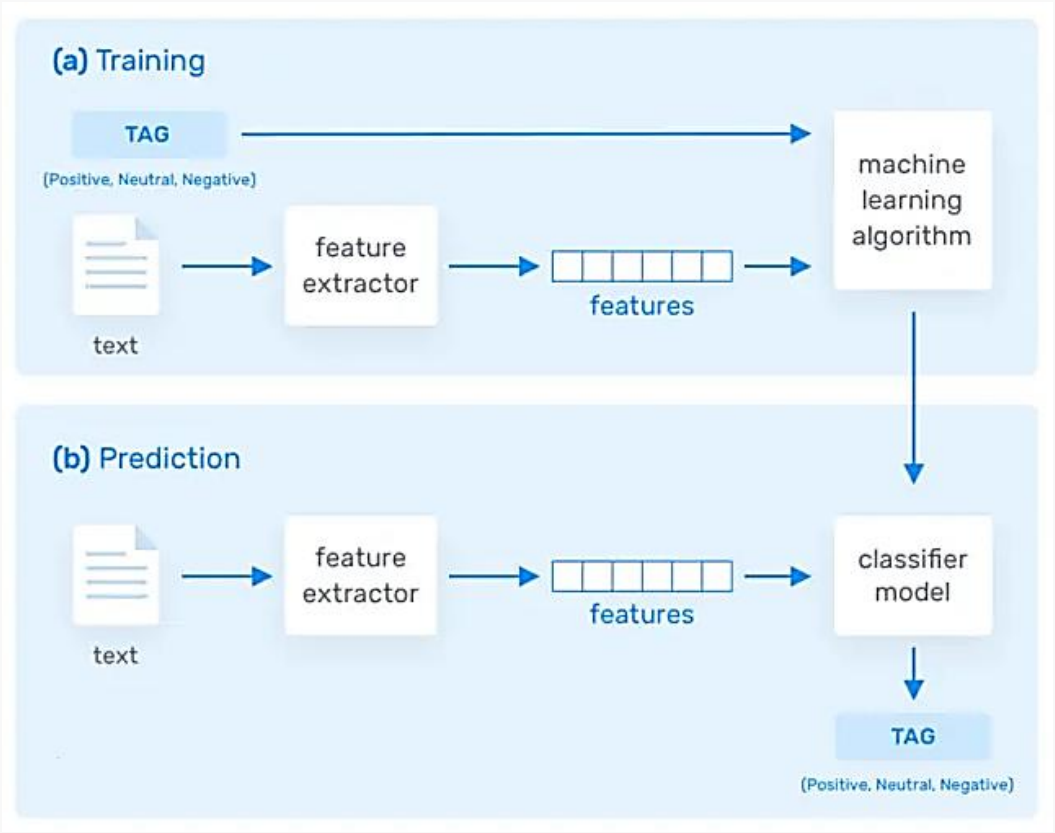
- ***Stemming*, *tokenization*, *part-of-speech tagging* and *parsing*.**
- ***Lexicons (i.e. Lists of words and expressions).***

Rule-Based systems are very naive since they don't take into account how words are combined in a sequence. Of course, more advanced processing techniques can be used, and new rules added to support new expressions and vocabulary.

Automatic Approach

Automatic methods, contrary to rule-based systems, don't rely on manually crafted rules, but on Machine Learning techniques. A Sentiment Analysis task is usually modeled as a ***classification problem***, whereby a classifier is fed a text and returns a category, e.g. **Positive**, **Negative**, or **Neutral**.





Automatic Approach Procedure

- **The Training and Prediction Processes :** The ***feature extractor*** transfers the text input into a ***feature vector***. Pairs of ***feature vectors*** and ***tags*** (e.g. ***Positive***, ***Negative***, or ***Neutral***) are fed into the Machine Learning algorithm to generate a model.

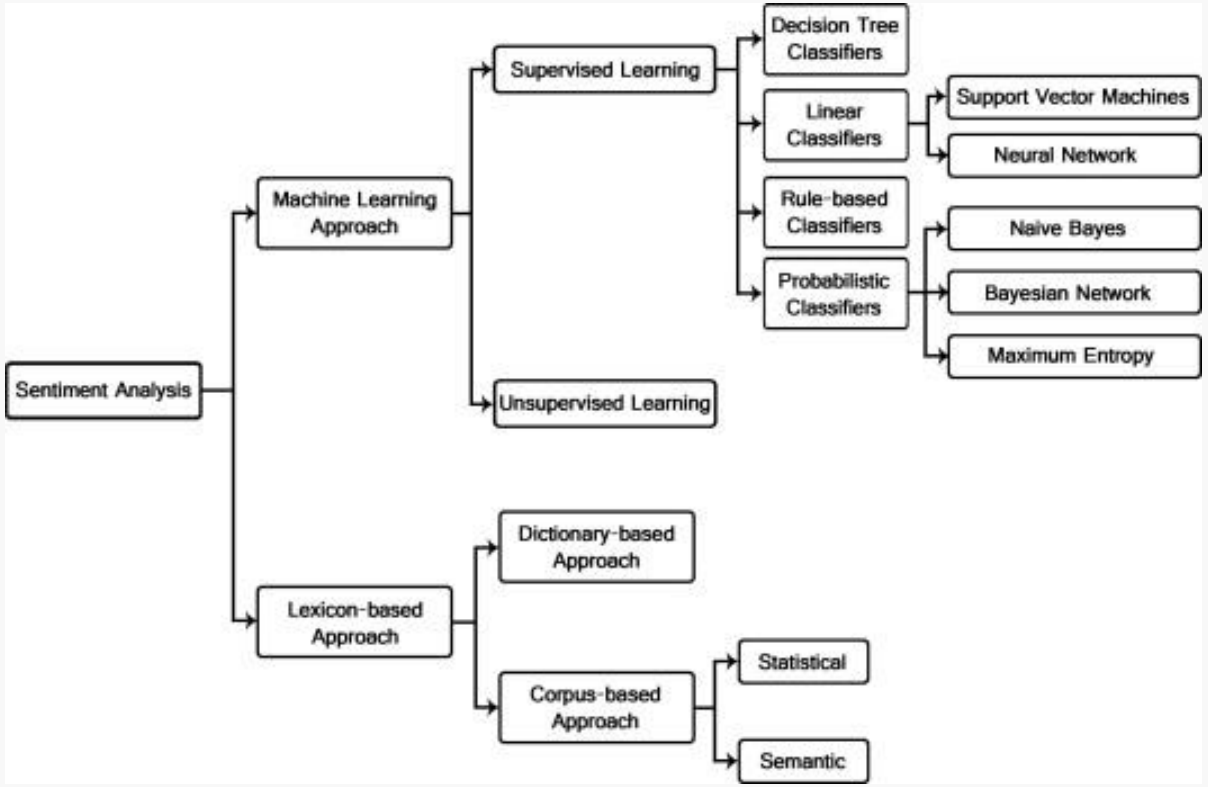
- **Feature Extraction from Text :** ***Feature Extraction*** techniques have been applied based on word embeddings (also known as *word vectors*). This kind of representations makes it possible for words with similar meaning to have a similar representation, which can improve the performance of classifiers.

Automatic Approach Procedure

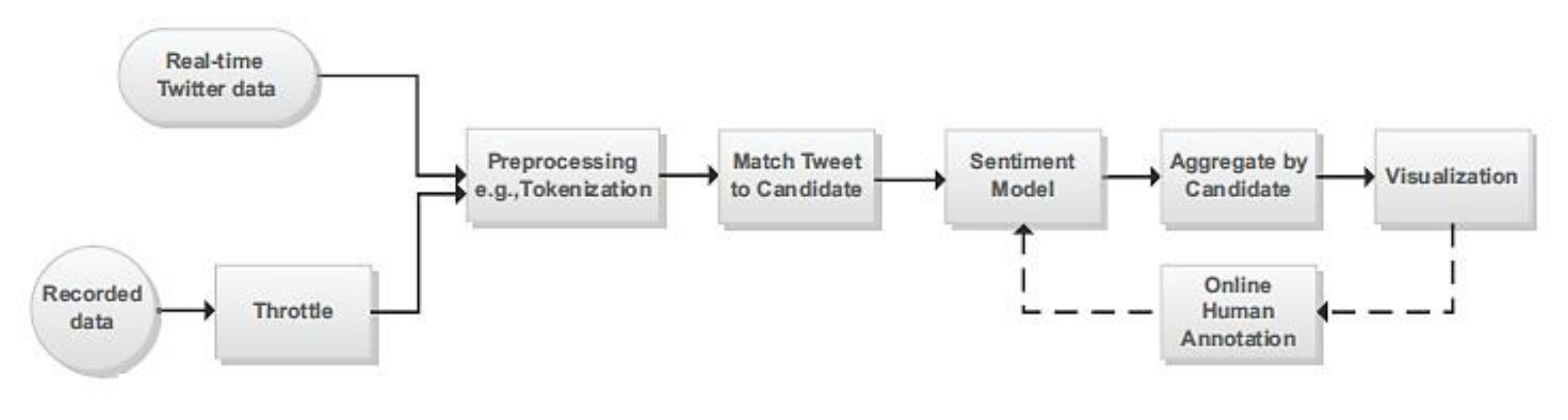
- **Classification Algorithms :** The classification step usually involves a statistical model like Naïve Bayes, logistic regression, support vector machines, or neural networks:
  - **Naïve Baye's :** a family of probabilistic algorithms that uses Bayes theorem to predict the category of a text.
  - **Linear Regression :** a very well-known algorithm in statistics used to predict some value (y) given a set of features (x).
  - **Support Vector Machines :** a non-probabilistic model which uses a representation of text examples as points in a multidimensional space. Examples of different categories (sentiments) are mapped to distinct regions within that space. Then, new texts are assigned a category based on similarities with existing texts and the regions they're mapped to.
  - **Deep Learning:** a diverse set of algorithms that attempt to mimic the human brain, by employing artificial neural networks to process data.

Hybrid Approach

Hybrid Systems combine the desirable elements of rule-based and automatic techniques into one system. One huge benefit of these systems is that results are often more accurate.



System Architecture for real time Processing of Twitter data



Naïve Bayes Algorithm

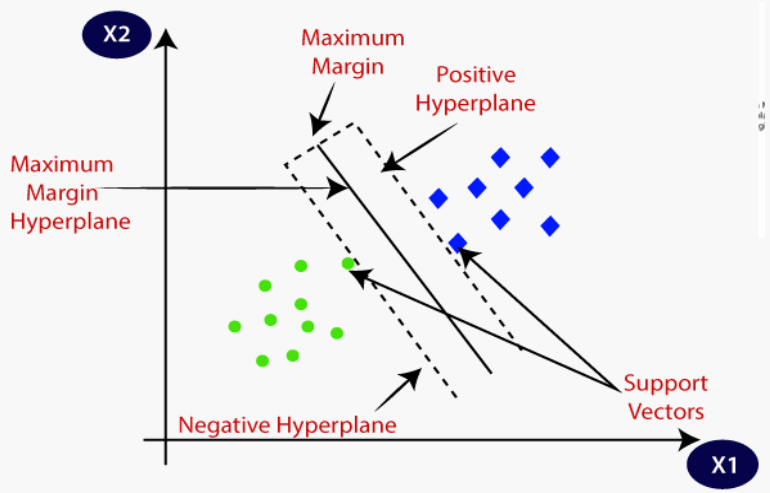
Naïve Bayes is one of the most improved classification (classifier) methods. First in order to perform classification, we must select the features from the data set. All the tweets in the data sets will be processed by the classifiers. A Naïve Bayes classifier assumes that the presence of a particular feature in a class is unrelated to the presence of any other feature. A Naïve Bayes algorithm is very easy to build up and mainly used for a large set of data. It provides a way of calculating $p(c|x)$ from $p(c)$, $p(x)$ and $p(x|c)$. Here $p(c|x)$ is called the posterior probability and it is given by the formula,

$$\boldsymbol{p(c|x) = \frac{p(x|c)\, p(c)}{p(x)}}$$

$P(c|x)$ is the posterior probability of *class* (c, *target*) given *predictor* (x, *attributes*). $P(c)$ is the prior probability of *class*. $P(x|c)$ is the likelihood which is the probability of *predictor* given *class*. $P(x)$ is the prior probability of *predictor*.

Support Vector Machine Algorithm

Support Vector Machine or SVM is one of the most popular supervised learning algorithms, which is used for classification as well as regression problems. The goal of the SVM algorithm is to create the best line or decision boundary that can segregate n-dimensional space into classes so that we can easily put the new data point in the correct category in the future. This best decision boundary is called a hyperplane.



Input:
D: Training dataset of various domains
L: Tier one learning algorithm
M : Tier two learning algorithm
N: Integer specifying the number of various domains

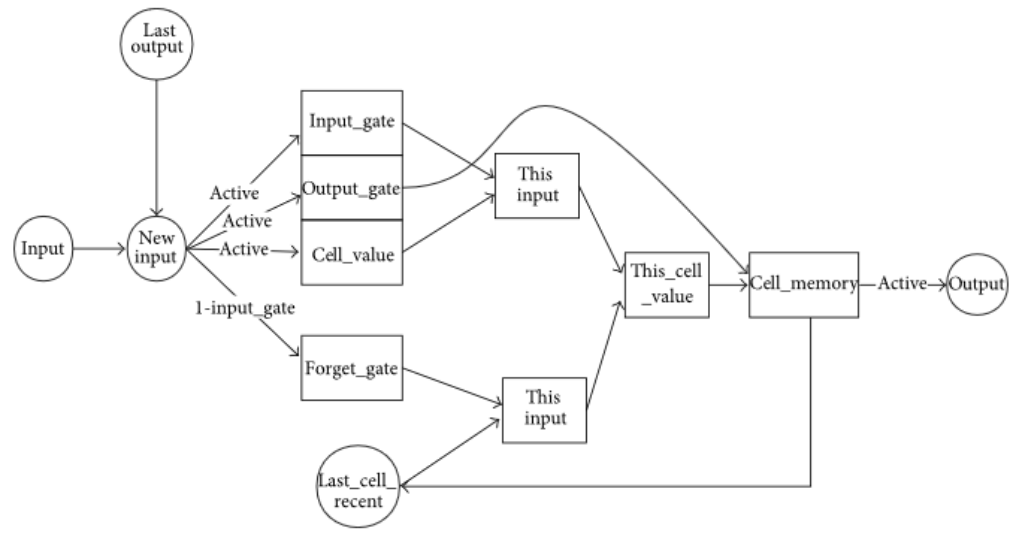
Do n=1 to N
1. Take a D_n of each domain.
2. Call L with D_n and receive the classifier L_n.
3. Call M and combine classifier L_n.

End

SVM Chooses The Extreme Points/Vectors That Help In Creating The Hyperplane. These Extreme Cases Are Called As Support Vectors, And Hence Algorithm Is Termed As Support Vector Machine.

Long Short-Term Memory Algorithm

Long Short-Term Memory (LSTM) Network, a special form of RNN are capable in learning such scenarios. These networks are precisely designed to escape the long term dependency issue of Recurrent Networks.



Require: $input\ Result_{last}\ Cell_{last}\ w\ u\ b\ r$

Ensure: $Cell_{value}\ Result$

while *When having input* **do**

$$temp_{input} = (Result_{last} * u_{input}) + (input * w_{input}) + b$$

$$temp_{output} = (Result_{last} * u_{output}) + (input * w_{output}) + b$$

$$temp_{cell} = (Result_{last} * u_{cell}) + (input * w_{cell}) + b$$

$$input_{gate} = sigmoid(temp_{input})$$

$$output_{gate} = sigmoid(temp_{output})$$

$$forget_{gate} = 1 - input_{gate}$$

$$Cell_{value} = activate(temp_{cell}) * input_{gate} + Cell_{last} * forget_{gate}$$

$$Result = output_{gate} * activate(cell_{value})$$

$$Cell_{value} = (Result, r)$$

end while

LSTMs are good in remembering information for long time. Since more previous information may affect the accuracy of model, LSTMs become a natural choice of use.

Related Work : I

“Sentiment Analysis Of Twitter Data” - *By* Apoorv Agarwal, Boyi Xie, Ilia Vovsha, Owen Rambow, Rebecca Passonneau*; Jan, 2011*

→ **Data Description –** They have 11,875 manually annotated twitter data (tweets) from a commercial source. Each tweet is labelled by a human annotator as **Positive**, **Negative**, **Neutral** or **Junk**.

→ **Resources & Pre-Processing –** They introduced two new resources for pre-processing twitter data: **1)** an emoticon dictionary and **2)** an acronym dictionary.

→ **Prior Polarity Scaling –** They used dictionary of affect in language (DAL) (whissel, 1989) and extend it using wordnet. This dictionary of about 8000 english language words assigns every word a pleasantness score (2 R) between 1 (negative) - 3 (positive).

→ **Design of Tree Kernel –** They have designed a tree representation of tweets to combine many categories of features in one succinct convenient Representation. This tree kernel is an instance of a general class of convolution kernels.

Related Work : I – contd.

“Sentiment Analysis Of Twitter Data” - *By* Apoorv Agarwal, Boyi Xie, Ilia Vovsha, Owen Rambow, Rebecca Passonneau*; Jan, 2011*

→ **Features –** Features can be divided into **three** broad categories: ones that are primarily counts of various features and therefore the value of the feature is a natural number € N. Second, features whose value is a real number € R. These are primarily features that capture the score retrieved from DAL. Thirdly, features whose values are Boolean € B. These are bag of words, presence of exclamation marks and capitalized text.

→ **Experiments & Results –** They presented experiments and results for **two classification tasks**: **1)** positive versus negative and **2)** positive versus negative versus neutral. For the classification task they present **three models** as well as the results for two combination of these models: **1)** Unigram Model (Their Baseline), **2)** Tree Kernel Model, **3)** 100 Senti-Features Model, **4)** Kernel plus Senti-Features, **5)** Unigram plus Senti-Features.

→ **Conclusion –** They presented a comprehensive set of experiments for both these tasks on manually annotated data that is a random sample of stream of tweets and investigated two kinds of models: tree kernel and feature based models and demonstrate that both these models outperform the unigram baseline.

Related Work : II

“Sentiment Analysis On Twitter Data Using Machine Learning Algorithms In Python” - *By S. Siddharth, R. Darsini, Dr. M. Sujithra; Feb,2018*

→ **Keywords –** Machine Learning, Natural Language Processing, Python, Sentimental Analysis.

→ **Data Collection/ Tweet Extraction –** They used the database packages for data (tweet) collection like: **MongoDB**, **PyMongo.**

→ **Pre-Processing –** The data was preprocessed by

1) Converting all uppercase letters to lowercase, **2)** Tokenization, **3)** Removal of non-English words, **4)** Emoticon Replacements, **5)** Removal of stop words.

→ **Feature Extraction –** They extract the aspects from the pre-processed twitter dataset.

1) Features are of three types : **unigram**, **bigram**, **n-gram**, **2)** Adjectives, Adverbs, Verbs and Nouns are good indicators of **Subjectivity** and **Sentiment**. **3)** The presence of a **Negation** usually changes the **Polarity** of the **Sentiment**.

Related Work : II - contd.

“Sentiment Analysis On Twitter Data Using Machine Learning Algorithms In Python” - *By S. Siddharth, R. Darsini, Dr. M. Sujithra; Feb,2018*

→ **Feature Selection –** They are categorized into 4 main types namely,

1) Natural language processing, **2)** Statistical, **3)** Clustering based, **4)** Hybrid Selection

→ **Classification –** They used two types of Machine Learning algorithms. Those are: **1) Naïve Bayes**, **2) Artificial Neural Networks.**

→ **Challenges –** The main Challenges are -

1) Detection of spam and fake reviews, **2)** Limitation of classification filtering, **3)** Asymmetry in availability of opinion mining software, **4)** Incorporation of opinion with implicit and behavior data, **5)** Domain-independence, **6)** Natural language processing overheads.

Related Work : III

“A Deep Learning Inspired Topic-Sentiment Model For Sentiment- Trend Analysis Of GST Tweets In India” - *By Anup Kumar Kolya, Souav Das, Dipankar Das; Jan, 2020*

→ **Keywords –** GST, Sentiment Modelling, Topic Modelling, Deep Learning, Trend Analysis.

→ **Data Collection/ Tweet Extraction –** They have gathered 1,98,964 tweets, or almost 200k unprocessed and raw tweets containing the hash-tags such as **#gst, #gsttax, #gstlaunch, #gstrollout, #gsteffect, #onenationononetax** etc.

→ **Tweet Modelling –**

1) **Topic Modelling –** Here they have considered a parameter **κ** as the keyword of the tweet. Because the keyword makes the most impact determining the relevance of the tweet.

2) **Sentiment Modelling –** The probability of finding the polarity from the remaining text as matching to their topic was determined.

Related Work : III – contd.

“A Deep Learning Inspired Topic-Sentiment Model For Sentiment- Trend Analysis Of GST Tweets In India” - *By Anup Kumar Kolya, Souav Das, Dipankar Das; Jan, 2020*

→ **Pre-Processing –** After pre-processing or cleaning the tweets, they tokenized the tweets into unigrams, bigrams and tri-grams with a frequency of 10,000 words for each type. They removed the stop words from unigrams, but not from bigrams and tri-grams as of keeping the semantics of such phrases intact.

1) Coverage With State-of-the-Art Lexicons, **2)** Sentiment Rating

→ **Sentiment-Trend Modelling –** They have developed **Naïve Bayes Sentiment Analyzer** to assign sentiment scores to the cleaned and preprocessed tweets. Next they developed the **LSTM** model for training and testing of their twitter data and split the dataset into **80:20** for training and testing parts.

1) Popularity-Polarity Modelling, **2)** Error Analysis: Stage 1, **3)** LSTM Modelling, **4)** Error - Analysis: Stage 2, **5)** Trend Analysis.

Related Work : IV

“CoronaTracker: World-wide COVID-19 Outbreak Data Analysis And Prediction” - *By* CoronaTracker Community Research Group*; Mar, 2020*

→ **Keywords –** COVID-19, Data Analysis, Sentiment Analysis, Predictive Modelling, SEIR.

→ **Data Collection –** Data is extracted from verified sources such as **John Hopkins University.** The fetched data will be stored in relational database, **MYSQL**.

→ **Predictive Modelling : SEIR Model – Susceptible-Exposed-Infected-Removed (SEIR)** System That Will Be Used To Describe The Recent Outbreak Of COVID-19 In China. They considered a simple SEIR epidemic model for the simulation of the infectious-disease spread.

→ **Sentiment Analysis –** We use a library called transformers by **huggingface**. The input sentences will be separated by their respective polarity for further analysis like **Topic Modelling** and generating **Word Cloud** for each polarity.

Related Work : IV – contd.

“CoronaTracker: World-wide COVID-19 Outbreak Data Analysis And Prediction” - *By* CoronaTracker Community Research Group*; Mar, 2020*

→ **Findings –**

1) **Current Outbreak Trends :** They have displayed the graphical representation of tends using their CoronaTracker Website.
2) **Predictive Modelling :** They modelled the global trajectory of the infection counts using the SEIR model, 240 days from the start date of 20 January, 2020.
3) **Sentiment Analysis :** They represented the **Word Cloud** for positive and negative sentiments, respectively.

Related Work : V

“Twitter Sentiment Analysis During COVID19 Outbreak” - *By* Dr. Akash D Dubey*; Apr, 2020*

→ **Keywords –** COVID19, Pandemic, Corona Virus, Twitter, Sentiment Analysis.

→ **Data Collection –** 50,000 tweets were collected from each country after every 4 days. For the collection, **RTweet** package in **R** was used. The keywords used for collecting the tweet were **COVID19**, **COVID-19**, **CORONAVIRUS**, **CORONA**, **STAY HOME STAY SAFE** and **covid19pandemic**.

→ **Cleaning & Pre-Processing –** Here the white spaces, punctuation, stop words were removed and the tweets were converted to lower case.

→ **Experiments –** After the data cleaning, the **NRC Emotion Lexicon** was applied with the help of **get_nrc_sentiment** function to analyze the tweets. Once the scoring of the tweets was done on the basis of sentiments and emotions, **Corpus** was created in order to develop the **Word Cloud** for each country.

Related Work : V – contd.

“Twitter Sentiment Analysis During COVID19 Outbreak” - *By* Dr. Akash D Dubey*; Apr, 2020*

→ **Result –** They have discussed the sentiment of tweets of 12 countries. The emotion analysis consist these emotions: **1)** Anger, **2)** Anticipation, **3)** Disgust, **4)** Fear, **5)** Joy, **6)** Sadness, **7)** Surprise, **8)** Trust.

→ **Conclusion –** This research work aimed at analyzing the sentiments and emotions of the people during the **pandemic COVID19**. During the study, it was revealed that countries like Belgium, India and Australia were tweeting about **COVID19** with a positive sentiment, people in China had negative sentiments about the same.

Other Related Works :

VI. "A System For Real-time Twitter Sentiment Analysis Of 2012 U.S. Presidential Election Cycle" - *By Hao Wang, Dogan Can, Abe Kazemzadeh, Francois Bar, Shrikanth Narayanan; Jul, 2012*

→ They Presented a system for real-time twitter sentiment analysis of the **ongoing 2012 U.S. Presidential Election**. To create a **Baseline Sentiment Model**, they used **Amazon Mechanical Turk (AMT)** to get as varied a population of annotators as possible. They used the twitter ***firehose*** and **expert-curated rules** and **keywords** to get a full and accurate picture of the online political landscape.

VII. "A Visual Analytics System For Making Sense Of Real-time Twitter Data" - *By Amir Haghighati, Kamran Sedig; Dec, 2019*

→ They proposed a **visual Analytics System (VAS), VARTTA**. Using a case study, they showed that **VARTTA** can help users survey and make sense of twitter discussions. They demonstrated that **VARTTA** is able to provide users/analysts with **real-time analytics**, along with the possibility of enabling them to offer **custom-made**, **heuristic labeling** or **categorization suggestions for ML errors**.

Other Related Works :

VIII. "End-to-End Sentiment Analysis Of Twitter Data" - *By Apoorv Agarwal, Jasneet Singh Sabharwal; 2012*

→ They Propose an End-to-End Pipeline For Classifying Tweets Into One Of Four Categories: **Objective**, **Neutral**, **Positive**, **Negative**. Traditionally, Objective Category Is defined as Text Segments. They used **Support Vector Machine** with **Linear Classifier** to create the models.

Proposed Work

After analyzing the related works from the previous papers I can conclude that the Sentiment Analysis for **COVID-19 Pandemic** can be a new area of research as there have not sufficient work done on this topic till now.

I am going to develop a Sentiment Analysis model for **COVID-19** using **Natural Language Processing** and **Machine Learning**. The model will mainly focus on the Sentiment Analysis of the people from whole world and will calculate the percentage of their Sentiments(Positive, Negative, Neutral) using **Machine Learning** or **Deep Learning** algorithms.

Finally using **Classification** algorithms like – **Naïve Bayes**, **SVM**, **LSTM** the accuracy percentage, time complexity of the proposed Sentiment Analysis Model can be measured with contrast to the other pre-defined Machine Learning models. After successfully achieving the accuracy level it will also generate the **Word Cloud** for finding the mostly used words from the tweets and plotting the calculated **Sentiment Polarity** through a graph with respect to date and time.

Experiment

As per the proposed work I have tried to develop a simple Sentiment Analysis Application which will perform the following operations:

1) First the application is fetching the tweets on **"COVID-19"** from twitter. I have used the ***tweepy*** API for ***python*** to do so.

I have created a twitter developer account for creating an application to generate the Twitter credentials for the app. These are - **consumer key**, **consumer secret key**, **access token key**, **access secret key.**

2) The fetched tweets are cleaned and preprocessed by using ***NLTK (Natural Language Toolkit)*** package thus the stop words can be removed.

```
import preprocessor as p
from nltk.corpus import stopwords
from nltk.tokenize import word_tokenize
```

```
stop_words = set(stopwords.words('english'))
word_tokens = word_tokenize(tweet)
clean_text = p.clean(data['text'])
```

Experiment

3) Then the cleaned and pre-processed tweets are stored in to a ***SQLITE Database File (.db format).***

```
import sqlite3
conn = sqlite3.connect('twitter.db')
c = conn.cursor()
```

4) The Sentiment for each tweets are calculated using ***vaderSentiment*** tool for ***python*** and store those values in the database.

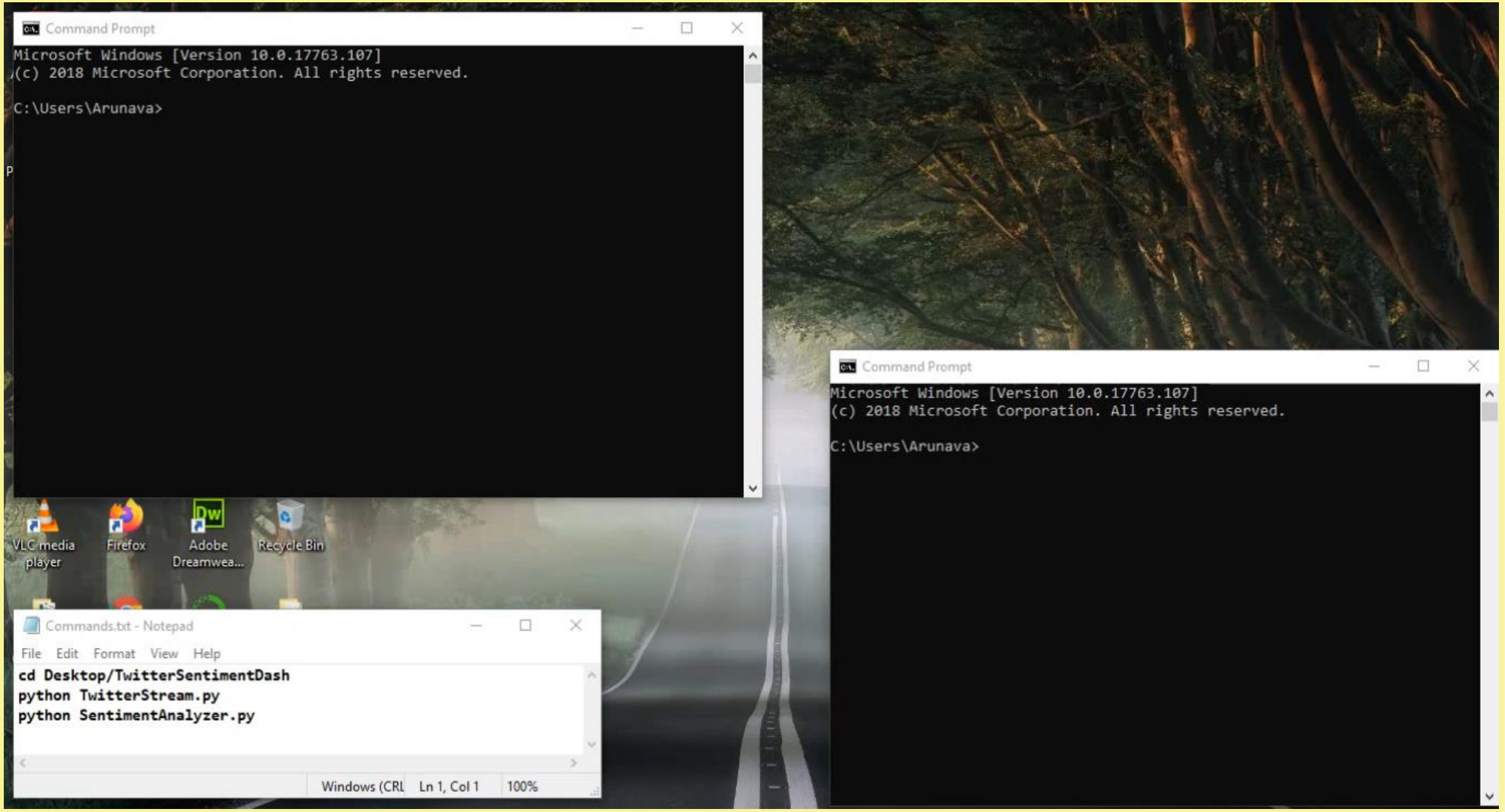
```
from vaderSentiment.vaderSentiment import SentimentIntensityAnalyzer
analyzer = SentimentIntensityAnalyzer()
vs = analyzer.polarity_scores(tweet)
sentiment = vs['compound']
```

Experiment

5) Finally after calculating the sentiment of the tweets the Sentiment Polarities are visualized through graphs using ***dash*** tool for ***python***.

```
import dash
from dash.dependencies import Output, Input
import dash_core_components as dcc
import dash_html_components as html
```

Output


Command Prompt
Microsoft Windows [Version 10.0.17763.107]
(c) 2018 Microsoft Corporation. All rights reserved.
C:\Users\Arunava>
VLC media player
Firefox
Adobe Dreamwea...
Recycle Bin
Commands.txt - Notepad
File Edit Format View Help
cd Desktop/TwitterSentimentDash
python TwitterStream.py
python SentimentAnalyzer.py
Windows (CRL
Ln 1, Col 1
100%
Command Prompt
Microsoft Windows [Version 10.0.17763.107]
(c) 2018 Microsoft Corporation. All rights reserved.
C:\Users\Arunava>

Conclusion & Future Scope

This **Literacy Survey** aimed at analyzing the sentiments and emotions of the people regarding the pandemic COVID-19.

1) In the **Previous Experiment** I have not yet used any **Machine Learning Algorithm** for generating a model to calculate the **Sentiment Polarity** of the tweets. In future I will try to implement this model for further experiment.

2) The **Sentiment** of the **emojis** can be calculated to get more relevant **Sentiment Polarity** of the tweets.

3) Feature Extraction & **Selection** processes can be implemented for gaining more accuracy in the calculation of **Sentiment Analysis**.

4) The **Classification** model can be used to check the accuracy level in contrast of the same of other predefined models.

5) Removing **Natural Language Processing** overheads for faster execution to reduce the time complexity.

Bibliography

[1] “Sentiment Analysis Of Twitter Data” - By Apoorv Agarwal, Boyi Xie, Ilia Vovsha, Owen Rambow, Rebecca Passonneau; Jan, 2011

[2] “Sentiment Analysis On Twitter Data Using Machine Learning Algorithms In Python” - By S. Siddharth, R. Darsini, Dr. M. Sujithra; Feb,2018

[3] “A Deep Learning Inspired Topic-Sentiment Model For Sentiment- Trend Analysis Of GST Tweets In India” - By Anup Kumar Kolya, Souav Das, Dipankar Das; Jan, 2020

[4] “CoronaTracker: World-wide COVID-19 Outbreak Data Analysis And Prediction” - By CoronaTracker Community Research Group; Mar, 2020

[5] “Twitter Sentiment Analysis During COVID19 Outbreak” - By Dr. Akash D Dubey; Apr, 2020

[6] “A System For Real-time Twitter Sentiment Analysis Of 2012 U.S. Presidential Election Cycle” - By Hao Wang, Dogan Can, Abe Kazemzadeh, Francois Bar, Shrikanth Narayanan; Jul, 2012

[7] “A Visual Analytics System For Making Sense Of Real-time Twitter Data” - By Amir Haghighati, Kamran Sedig; Dec, 2019

[8] “End-to-End Sentiment Analysis Of Twitter Data” - By Apoorv Agarwal, Jasneet Singh Sabharwal; 2012

[9] https://towardsdatascience.com

[10] https://ieeexplore.ieee.org/Xplore/home.jsp

[11] https://www.researchgate.net


Thank you!