MAULANA AZAD
NATIONAL INSTITUTE OF TECHNOLOGY
BHOPAL INDIA, 462003

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Time Series of Sentiment Analysis

Minor Project Report

Semester

Submitted by:

Pranjal Varshney	171112249
Vedant Mathe	171112231
Gagandeep Singh	171112204
Rahul Pandey	171112301

Under the Guidance of
Dr. SK Saritha
DEPARTMENT OF COMPUTER SCIENCE
AND ENGINEERING
Session: 2019-20

MAULANA AZAD
NATIONAL INSTITUTE OF TECHNOLOGY
BHOPAL INDIA, 462003

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

CERTIFICATE

This is to certify that the project report carried out on "**Time Series of Sentiment Analysis**" by the 3rd year students:

Pranjal Varshney	171112249
Vedant Mathe	171112231
Gagandeep Singh	171112204
Rahul Pandey	171112301

Have successfully completed their project in partial fulfilment of their Degree in Bachelor of Technology in Computer Science and Engineering.

Dr. S.K.Saritha
(Minor Project Mentor)

TABLE OF CONTENTS

DECLARATION

We, hereby declare that the following report which is being presented in the Minor Project Documentation Entitled as "**Time Series using Sentiment Analysis**" is an authentic documentation of our own original work and to best of our knowledge. The following project and its report, in part or whole, has not been presented or submitted by us for any purpose in any other institute or organization. Any contribution made to the research by others, with whom we have worked at Maulana Azad National Institute of Technology, Bhopal or elsewhere, is explicitly acknowledged in the report.

Pranjal Varshney	171112249
Vedant Mathe	171112231
Gagandeep Singh	171112204
Rahul Pandey	171112301

ACKNOWLEDGEMENT

With due respect, we express our deep sense of gratitude to our respected guide and coordinator Dr. Dhirendra Pratap Singh, for his valuable help and guidance. We are thankful for the encouragement that he has given us in completing this project successfully.

It is imperative for us to mention the fact that the report of minor project could not have been accomplished without the periodic suggestions and advice of our project guide Dr.S.K.Saritha and project coordinators Dr. Dhirendra Pratap Singh and Dr. Jay Trilok Chaudhary.

We are also grateful to our respected director Dr. N. S. Raghuwanshi for permitting us to utilize all the necessary facilities of the college.

We are also thankful to all the other faculty, staff members and laboratory attendants of our department for their kind cooperation and help. Last but certainly not the least; we would like to express our deep appreciation towards our family members and batch mates for providing the much needed support and encouragement.

ABSTRACT

In recent years, Social media have emerged as popular platforms for people to share their thoughts and opinions on all kind of topics. Tracking opinion over time is a powerful tool that can be used for sentiment prediction or to detect the possible reasons of a sentiment change. Understanding topic and sentiment evolution allows enterprises or government to capture negative sentiment and act promptly.More and more people express their opinions on social media such as Facebook and Twitter. Predictive analysis on social media time-series allows the stake-holders to leverage this immediate, accessible and vast reachable communication channel to react and proact against the public opinion. In particular, understanding and predicting the sentiment change of the public opinions will allow business and government agencies to react against negative sentiment and design strategies such as dispelling rumors and post balanced messages to revert the public opinion.We model the collective sentiment change without delving into micro analysis of individual tweets or users and their corresponding low level network structures.Finally, we study the usability of outliers detection and different measures such as sentiment velocity and acceleration on the task of sentiment tracking.

In this Project, we explore conventional time series analysis methods and their applicability on various topics and sentiment trend analysis. We use various dataset for predicting sentiments over time.

1.Introduction

Recent years have seen the rapid growth of social media platforms that enable people to express their thoughts and perceptions on the web and share them with other users. Many people write their opinion about products, movies, people or events on microblogs, blogs, forums or review sites. The so-called User Generated Content (UGC) is a good source of user opinions and mining it can be very useful for a wide variety of applications that require understanding public opinion about a concept. For example, enterprises can capture negative or positive opinions of customers about products or about competitors and improve the quality of their services or products accordingly. It is also very important for government to understand the public opinion regarding different social issues and act promptly.

A time series is a sequence of numerical data points in successive order. In investing, a time series tracks the movement of the chosen data points, such as a security's price, over a specified period of time with data points recorded at regular intervals.

1.1 What Is a Time Series?

A time series can be taken on any variable that changes over time. In investing, it is common to use a time series to track the price of a security over time. This can be tracked over the short term, such as the price of a security on the hour over the course of a business day, or the long term, such as the price of a security at close on the last day of every month over the course of five years.

Time Series Analysis

Time series analysis can be useful to see how a given asset, security, or economic variable changes over time. It can also be used to examine how the changes associated with the chosen data point compare to shifts in other variables over the same time period.

For example, suppose you wanted to analyze a time series of daily closing stock prices for a given stock over a period of one year. You would obtain a list of all the closing prices for the stock from each day for the past year and list them in chronological order. This would be a one-year daily closing price time series for the stock.

Delving a bit deeper, you might be interested to know whether the stock's time

series shows any seasonality to determine if it goes through peaks and troughs at regular times each year. Analysis in this area would require taking the observed prices and correlating them to a chosen season. This can include traditional calendar seasons, such as summer and winter, or retail seasons, such as holiday seasons.

Alternatively, you can record a stock's share price changes as it relates to an economic variable, such as the unemployment rate. By correlating the data points with information relating to the selected economic variable, you can observe patterns in situations exhibiting dependency between the data points and the chosen variable.

Time Series Forecasting

Time series forecasting uses information regarding historical values and associated patterns to predict future activity. Most often, this relates to trend analysis, cyclical fluctuation analysis, and issues of seasonality. As with all forecasting methods, success is not guaranteed.

1.2 What Is a Sentiment Analysis?

Sentiment Analysis is the process of determining whether a piece of writing is positive, negative or neutral. A sentiment analysis system for text analysis combines natural language processing (NLP) and machine learning techniques to assign weighted sentiment scores to the entities, topics, themes and categories within a sentence or phrase.

Sentiment analysis helps data analysts within large enterprises gauge public opinion, conduct nuanced market research, monitor brand and product reputation, and understand customer experiences. In addition, data analytics companies often integrate third-party sentiment analysis APIs into their own customer experience management, social media monitoring, or workforce analytics platform, in order to deliver useful insights to their own customers.

Sentiment analysis focuses on developing methods that can classify a text as expressing positive or negative sentiment . However, public opinion towards a specific topic changes over time. Time series models seem to be an appropriate tool for sentiment tracking. Leveraging time series analysis on public opinions can be useful to understand data and identify patterns, trends and seasonality. In addition, time series is helpful in identifying outliers which are likely to be related with events that caused a sentiment change. Finally, they can be useful for sentiment prediction and intervention analysis that capture future sentiment and

detect the effect of a single event on the public opinion respectively. The development of models that focus on sentiment dynamics has recently attracted much research interest.The aim of this preliminary study is to explore the usability of conventional time series methods on sentiment tracking and how they can be leveraged to address other challenging tasks such as detection of reasons of sentiment change. We first explore and decompose the data we collected from Twitter about different topics with the aim to identify patterns, trends and seasonality. In addition, we try to detect the outliers and investigate their usability in detecting important events or reasons of change. This is very important as it is possible to use the data that are around those peaks to automatically detect the sentiment change reasons.

2. Literature review and survey

2.1 Time Series Forecasting using a hybrid ARIMA and Neural Network Model by G.Peter Zhang

Autoregressive integrated moving average (ARIMA) is one of the popular linear models in time series forecasting during the past three decades. Recent research activities in forecasting with artificial neural networks (ANNs) suggest that ANNs can be a promising alternative to the traditional linear methods. ARIMA models and ANNs are often compared with mixed conclusions in terms of the superiority in forecasting performance. In this paper, a hybrid methodology that combines both ARIMA and ANN models is proposed to take advantage of the unique strength of ARIMA and ANN models in linear and nonlinear modeling. Experimental results with real data sets indicate that the combined model can be an effective way to improve forecasting accuracy achieved by either of the models used separately.

2.2 Time+User Dual Attention Based Sentiment Prediction for Multiple Social Network Texts With Time Series by(Lei Li ; Yabin Wu ; Yuwei Zhang ; Tianyuan Zhao)

Sentiment analysis of social network texts can effectively reflect the development and changes of public opinions. At the same time, prediction and judgment of public opinion development can also play a key role in assisting decision-making and effective management. Therefore, sentiment analysis for hot events in online social media texts and judgment of public opinion development have become popular topics in recent years. At present, research on textual sentiment analysis is mainly aimed at a single text, and there is little-integrated analysis of multi-user and multi-document in unit time for time series. Moreover, most of the existing methods are focused on the information mined from the text itself, while the feature of identity differences and time sequence of different users and texts on social platforms are rarely studied. Hence, this paper works on the public opinion texts about some specific events on social network platforms and combines the textual information with sentiment time series to achieve multi-document sentiment prediction. Considering the related features of different social user identities and time series, we propose and implement an effective time+user dual attention mechanism model to analyze and predict the textual information of public opinion. The effectiveness of the proposed model is then verified through experiments on real data from a popular Chinese microblog platform called Sina Weibo.

3. Gaps identified

Using RNN-
Both ARIMA and ANN models have achieved successes in their own linear or nonlinear domains. However, none of them is a universal model that is suitable for all circumstances. The approximation of ARIMA models to complex nonlinear problems may not be adequate. On the other hand, using ANNs to model linear problems have yielded mixed results. For example, using simulated data, when there are outliers or multicollinearity in the data, neural networks can significantly outperform linear regression models. The performance of ANNs for linear regression problems depends on the sample size and noise level. Hence, it is not wise to apply ANNs blindly to any type of data. Since it is difficult to completely know the characteristics of the data in a real problem, RNN that has both linear and nonlinear modeling capabilities can be a good strategy for practical use. By using RNN, different aspects of the underlying patterns may be captured.

In theory, RNNs are absolutely capable of handling such "long-term dependencies." A human could carefully pick parameters for them to solve toy problems of this form. Sadly, in practice, RNNs don't seem to be able to learn them. The problem was solved by LSTM.

Using LSTM-
Long Short-Term Memory (LSTM) networks are a modified version of recurrent neural networks, which makes it easier to remember past data in memory. The vanishing gradient problem of RNN is resolved here. LSTM is well-suited to classify, process and predict time series given time lags of unknown duration. It trains the model by using back-propagation.

We are using RNN with LSTM for predicting sentiment over time as efficient approach for . It fills the gaps made by researchers who used ANN and ARIMA models

4.Proposed work and methodology-

Proposed work-

Data Preprocessing-

- Data is preprocessed for semantic analysis before creating the RNN model and LSTM architecture.
- For the preprocessing of data, Load the dataset obtained into the system and visualize it using python libraries like matplotlib,pandas etc (identifying the important parameters which are dependent variables in the dataset).
- Next we convert all the reviews into lowercase to make them similar in pattern.
- After this, Data Cleaning is necessary for efficient working of the model, For Data Cleaning, Remove all the punctuation from the reviews as they are not relevant while predicting the sentiments, Remove all the un-useful waste data before actual work start .
- At last,Create a list of reviews using python library (pandas) as dependent variables.

Tokenization-

- Tokenization is performed before sentiment analysis for classification of the data over different sentiments.
- For this,Create a vocabulary to integer mapping dictionary to tokenize each words.
- Encode the words so that the model can easily recognize it as a variable or a parameter.
- Analyze the length of the reviews.
- Now to make the dataset efficient we need to remove the outliners present in it. So we find all the reviews which are extremely long or extremely short and remove them to maintain consistency in the dataset.
- Now the main task is to pad/truncate the reviews in remaining data.

Sentiment Analysis-

- We perform Sentiment Analysis by our tokenized bag of words using python libraries (NLTK / TEXTBLOB) to process the textual data.
- Sentiment Analysis is performed for classification of data over the different sentiments before analyzing it on the time series.
- We create a sparse matrix of the words for sentiment Analysis.
- After this we need to apply time series on the reviews which are now in the

form of sparse matrix as a model of bag of words.

Splitting-

- Next we split the dataset into 3 parts:

1. Training set
2. Validation set
3. Testing set

- With the help of K-Fold validation we split the dataset into training and testing set using various combinations to make our prediction model more efficient and robust.
- The data is then divided into various batches as we are applying batch processing in our model.

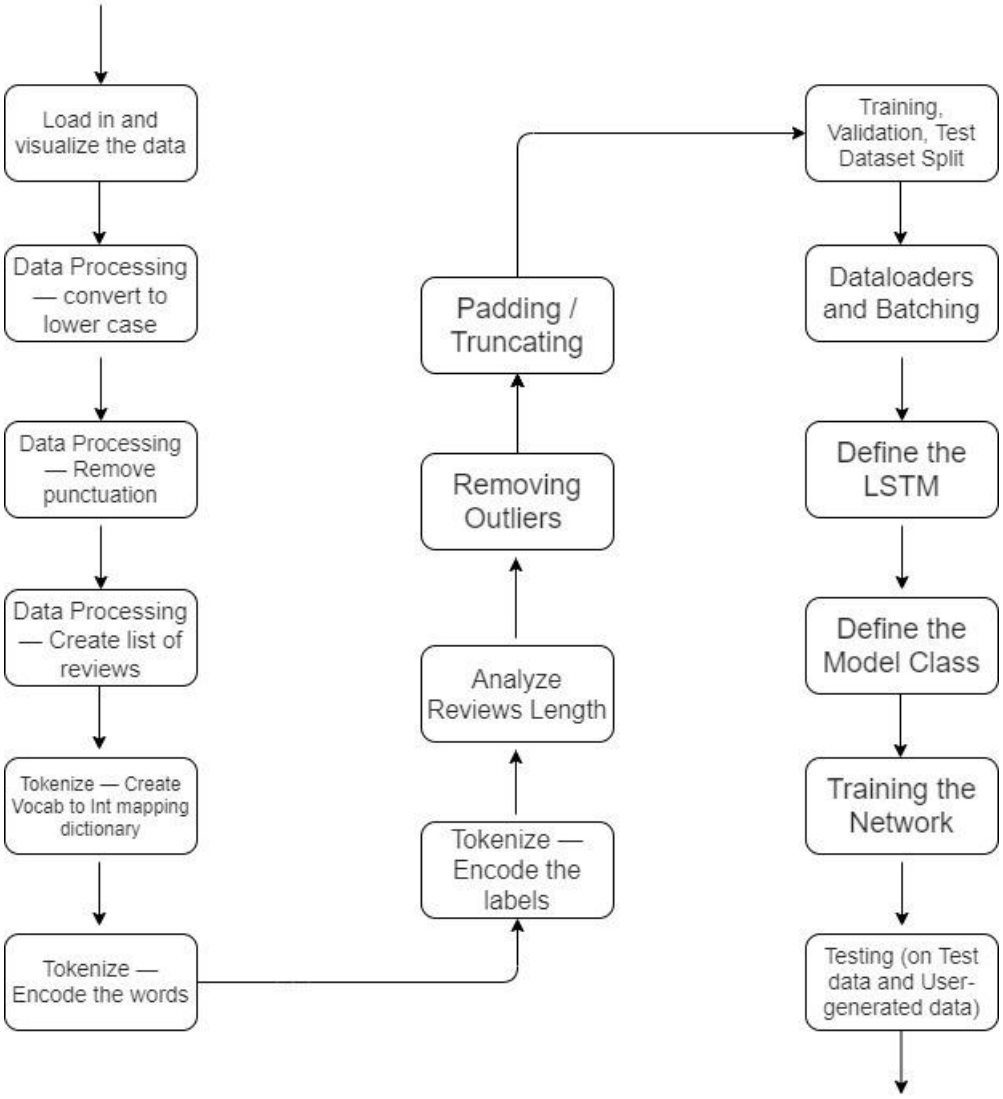
Implementing RNN with LSTM Architecture-

- After this we create a classifier of recurrent neural network using deep learning techniques for creating time series of the data obtained from sentiment analysis and we then apply the LSTM architecture over it to remove the problem of long distance dependencies ,exploding gradient and vanishing gradient.

- Now we create a neural network of multiple layers (number of layers will be determined during testing period of project for performance evaluation). Then the model is trained for suitable number of epochs.

Training and Testing-

- Finally the model created is tested for the test data as well as for the user generated data and evaluated for the efficiency.



Flow chart for the proposed work

Methodology-

Using Sentiment Analysis-

Sentiment analysis is the automated process that uses AI to identify positive, negative and neutral opinions from text. Sentiment analysis is widely used for getting insights from social media comments, survey responses, and product reviews, and making data-driven decisions.

There are many methods and algorithms to implement sentiment analysis systems, which can be classified as:

- **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.

In our project, we are using **Hybrid** approach for sentiment Analysis.

Rule-based Approaches

Usually, rule-based approaches define a set of rules in some kind of scripting language that identify subjectivity, polarity, or the subject of an opinion.

The rules may use a variety of inputs, such as the following:

- Classic NLP techniques like stemming, tokenization, part of speech tagging and parsing.
- Other resources, such as lexicons (i.e. lists of words and expressions).

A basic example of a rule-based implementation would be the following:

1. Define two lists of polarized words (e.g. negative words such as bad, worst, ugly, etc and positive words such as good, best, beautiful, etc).
2. Given a text:
   i. Count the number of positive words that appear in the text.
   ii. Count the number of negative words that appear in the text.
3. If the number of positive word appearances is greater than the number of negative word appearances return a positive sentiment, conversely, return a negative sentiment. Otherwise, return neutral.

This system is very naïve since it doesn't take into account how words are combined in a sequence.

Automatic Approaches

Automatic methods, contrary to rule-based systems, don't rely on manually crafted rules, but on machine learning techniques. The sentiment analysis task is usually modeled as a classification problem where a classifier is fed with a text and returns the corresponding category, e.g. positive, negative, or neutral (in case polarity analysis is being performed).

Said machine learning classifier can usually be implemented with the following :

Classification Algorithms

The classification step usually involves a statistical model like Naïve Bayes, Logistic Regression, Support Vector Machines, or Neural Networks:

- Naïve Bayes: a family of probabilistic algorithms that uses Bayes's 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. These examples are mapped so that the examples of the different categories (sentiments) belong to distinct regions of that space.. Then, new texts are mapped onto that same space and predicted to belong to a category based on which region they fall into.
- Deep Learning: a diverse set of algorithms that attempts to imitate how the human brain works by employing artificial neural networks to process data.

Hybrid Approaches

The concept of hybrid methods is very intuitive: just combine the best of both worlds, the rule-based and the automatic ones. Usually, by combining both approaches, the methods can improve accuracy and precision.

Using RNN-

As well known, RNN mainly deals with the processing of sequence data, such as text, speech, and climate. This type of data exists in an orderly relationship with each other; each piece of data is associated with the previous piece. For example, in a text, a word in a sentence is related to its preceding word. In the climate data, the temperature of a day is related to the temperature of previous day.

Basic RNN has multiple neuron-node to form a network. Each node (neuron) has a time-varying real-valued activation. Each connection has a real-valued weight, which can be modifible in each case. In the architecture, the output of the neuron at time t-1 will be passed to the input at time t and add the data of itself at time t to generate the output at time t. Recurrently exploiting the neuron node to cascade multiple neuron elements to form a RNN.

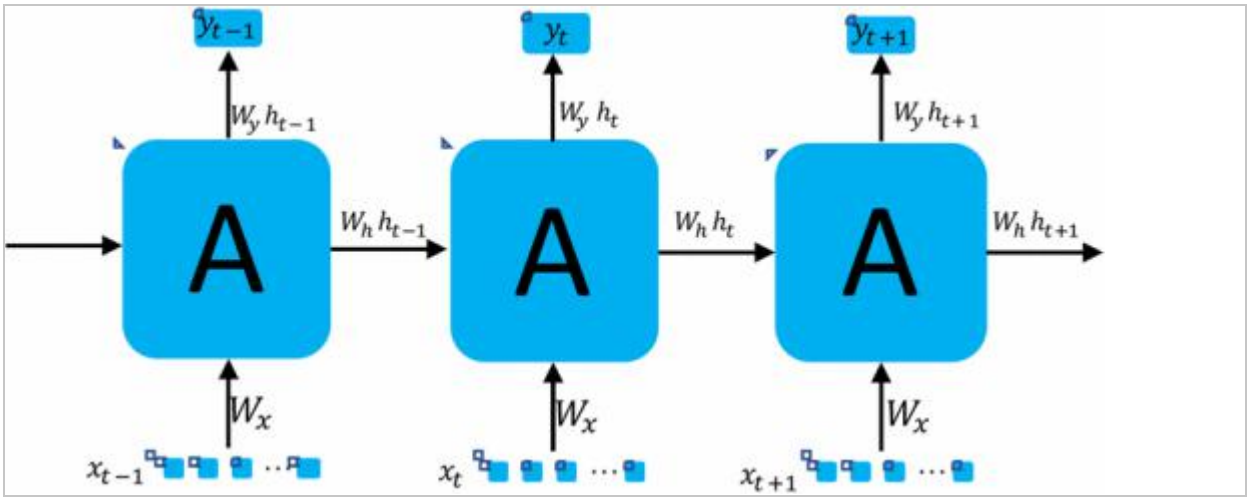


Fig. 1.

RNN architecture

The recursive formulas of RNN are shown in equations:

$$\mathbf{ht=tanh(Whht-1+Wxxt)yt=Wyht}$$

where yt is output vector, ht is hidden layer vector, xt is input vector, and Wh is weighting matrix.

Although theoretical RNN can handle such long-term dependencies (Long Dependencies) problem, the longer the interval time step (the data to be referenced is at a longer time point), the Wh and Wx (weighting matrix) will continue to multiply recurrently with previous output. This will cause the vanishing gradient and exploding gradient problem. To solve this problem, we use Long Short-Term Memory (LSTM) networks to improve this problem.

Using LSTM

According to the Wikipedia's definition, long short-term memory (LSTM) units, as shown in Fig. 2, are a building unit for layers of a (RNN). A RNN composed of LSTM units is often called an LSTM network. The difference between LSTM and

traditional RNN neural networks is that each neuron in LSTM is a memory cell. The LSTM links the previous data information to the current neurons. Each neuron contains three gates: input gate, forget gate, and output gate. Using the internal gate, the LSTM can solve the problem of long-term dependence of the data. Next we present the internal gates of LSTM and describe how the LSTM architecture solves long-term dependency problems.

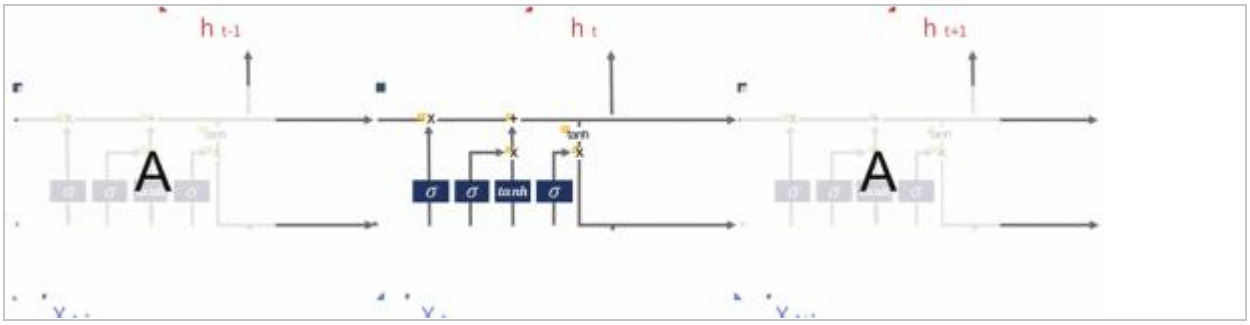


Fig. 2.

LSTM architecture

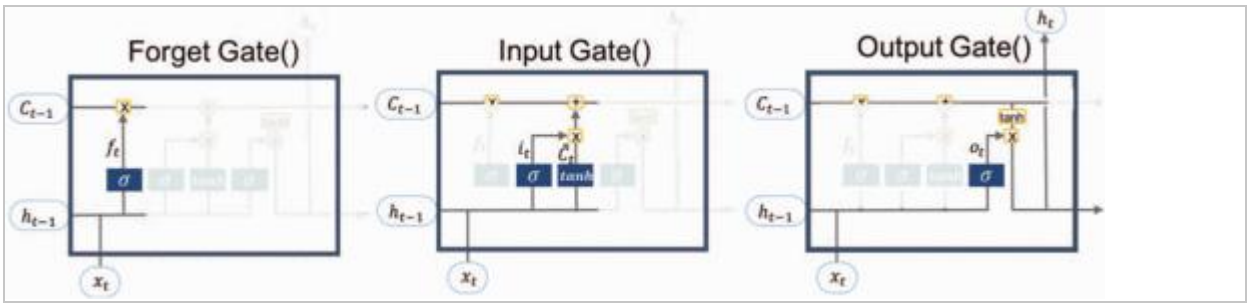


Fig. 3.

Project Timeline-

- **January- February, 2020-**

Creating Sentiment Analysis Model

Collecting User Generated Data

Data Pre-processing

- **March, 2020-**

Creating RNN model

Creating LSTM architecture

- **April ,2020-**

Implementing the dataset over the model and making chances as per the requirement.

Training and Testing the model

Final Validation of the Result over various dataset for performance evaluation.

5.Tools and technology to be used (hardware and software)

Software requirements:

- PYTHON 3
- PYTHON LIBRARIES FOR SENTIMENT ANALYSIS : NLTK , TEXTBLOB
- MODELS FOR TIME SERIES ANALYSIS : RNN USING LSTM (USING KERAS) / ARIMA MODEL
- BASIC PYTHON LIBRARIES FOR DATA HANDLING AND DATA PREPROCESSING : PANDAS, NUMPY, MATPLOTLIB
- PYTHON IDE : SPYDER(ANACONDA)

Hardware requirements:

- LAPTOP
- INTEL I- 5 8 GEN PROCESSOR
- RAM 8 GB
- OS : WINDOWS 10
- GPU

6. CONCLUSION

In this study we explored the usability of time series methods on sentiment tracking. We studied various research works done by professionals for predicting sentiments over the period of time. We studied various dataset for implementing time series with sentiments. We are going to plot frequencies and decomposed data from various data sets to identify patterns, trends and seasonality. Also, we will find peaks in topic's popularity and in sentiments while we investigated their usability in detecting important events or the reasons of sentiment change. We believe that this is a good start for the development of methods that could automatically detect a sentiment change and the reasons that caused it. In future we plan to explore these problems more thoroughly.

7. References

Research Papers:-

- Time Series Forecasting using a hybrid ARIMA and Neural Network Model by G.Peter Zhang
- Time+User Dual Attention Based Sentiment Prediction for Multiple Social Network Texts With Time Series by(Lei Li ; Yabin Wu ; Yuwei Zhang ; Tianyuan Zhao).
- Stock Market Trend Prediction with Sentiment Analysis based on Neural Network by(Xu Jiawei, Tomohiro Murata)

Internet :-

- Wikipedia
- Towardsdatascience
- Analytics Vidhya
- Kaggle