Sentiment Analysis of Social Media Data

Snehal Salve[1], Swapnal Thube[2], Amruta Warghade[3], Prof. Snehal Umare[4]
[1, 2, 3, 4] Dept of Computer
[1, 2, 3, 4] MAEER's MITCOE Kothrud,Pune

Abstract- *Sentiment analysis is the computational study of opinions, sentiments, evaluations, attitudes, views and emotions expressed in text. It refers to a classification problem where the main focus is to predict the polarity of words and then classify them into positive, negative or neutral sentiments. Sentiment analysis over Social media offers people a fast and effective way to measure the public's feelings towards other people and whether their attitude towards other people is positive or negative. In the process of analysis many preprocessing techniques can be applied on the data that is generated by various social media, applying machine learning algorithms like KNN(K- nearest neighbors) to classify the comments as positive, negative and so on. By using Naive Bayes, K-NN and Deep learning algorithms the results were generated. Then the results were measured in terms of accuracy with respect to Naïve Bayes,KNN and Deep learning and accuracy of Naïve Bayes was found more than KNN and Deep learning.*

I. INTRODUCTION

In today's digital and technology world , it is necessary to know customer reviews or opinion about certain issues or product .Social media and shopping websites like amazon ,flip cartetc. create lot of such data of reviews and opinions . As this data is very much large it is required to develop some method to read , analyze and classify the opinions as positive , negative and neutral, this is known as sentiment analysis. By analyzing this reviews , it provides important observation about success or failure of product . It helps companies or owners to take future decisions. Sentiment analysis has mainly three levels : Document level , Sentence level , Aspect level. Sentence and document level analysis provides overall sentiment about text and aspect level analysis deals with identifying and classifying the sentiment of semantically distinct and identifiable aspects within the text . Aspect-level analysis requires natural language processing to identify where aspect is located. In this paper they developed an approach using small dataset for aspect-level sentiment analysis. They have used two datasets with Portuguese opinions annotated at document and aspect level to train and test our approach and used an English dataset to gauge their approach for larger datasets.

R is a popular programming language which is generally embraced by information researchers. In any case, normal R must be executed in a solitary machine environment. As the volume of accessible information proceeds to quickly develop from an assortment of sources, versatile and execution investigation arrangements have turned into a fundamental device to upgrade business profitability and income. The R environment provides enormous built-in functions in the package "base", most of which are generally required for elementary data analysis.

II. LITERATURE SURVEY

In paper "A two step method for sentiment analysis of tweets" published in 2016 it used Naive Bayes, SVM ,lexicon and lexicon SVM for sentiment analysis. In this paper Lexicon was coupled with standard machine learning method which improved performance .This method lacked performance when POS tags were added and additional computational resources were added.

In Paper "Min deep learning" published in 2018 it used SVM and CNN to perform sentiment analysis. As it used deep learning it outperformed retain global metrics and also revealed deeper semantic relationships.But it suffered from overfitting and main issue was scarcity and non-disclosure of Tibetan Corpus.

In Paper "A Deep Learning Approach to Classify Aspect Level Sentiment using Small Datasets" which was published in 2018. It used two architectures for sentiment analysis , in first architecture, they classified the aspect sentiment using a document-level model that receives a sentence as input. In the second architecture, they applied a preprocessing step over the input and classify the aspect sentiment training a model for each aspect. The second architecture achieved higher accuracy when compared to the first one, which confirmed that using a preprocessing step plus a model by aspect improved the results.

In Paper "Text Understanding from Scratch" which was published in 2015. It used convolutional Neural Networks to find positivity and negativity of amazon reviews and this method worked with 96% of accuracy.

In Paper ”Convolutional Neural Networks for Sentence Classification ” which was published in 2014.It used Convolutional Neural trained on top of pre-trained word vectors for sentence-level classification tasks. simple CNN with little hyper parameter tuning and static vectors achieved excellent results on multiple benchmarks. Learning task-specific vectors through fine-tuning offered further gain in performance. They additionally proposed a simple modification to the architecture to allow for the use of both task-specific and static vectors. The CNN models discussed herein improved upon the state of the art on 4 out of 7 tasks, which included sentiment analysis and question classification.

In Paper ”Sentiment Analysis of tweets using Semantic Analysis” which was published in 2017.In this paper Naïve Bayes ,Maximum Entropy and Negation were used for sentiment analysis and Naive Bayes outperformed Maximum Entropy and Negation.

In paper ”Sentiment Analysis of Top Colleges” which was published in 2018 .In this paper Naive Bayes and K-NN were used for sentiment analysis and Naive Bayes out performed K-NN

III. RELATED WORK

A. Brief Description associated with the various tweets.

- Emoticons: The expressions which are used to describe the users emotions or feelings for an issue or his personal issues.[3]
- Target: The Twitter users will use the special characters“@” symbol to simply refer to other users on the various social media sites which continuously alerts them.
- Hash tags: The Users usually use The hash tags ”#” to refer to various topics. This is done to increase the views of their personal tweets.

B. The steps proposed are :

- Data collection using Twitter API: Large sets of twitter data is not available publicly. Hence we first extract the twitter data from the Twitter API.
- Data Preprocessing: This involves cleaning and simplifying the data by performing spell correction, punctuation handling etc. so as to remove noise from the data.
- Applying Classification algorithms: The classification algorithms are applied on these tweets in order to categorize them. Different models provide different accuracy and we choose the model with highest accuracy.
- Classified tweets: The results of the above step is classifies
- tweets which may belong to any of the categories
- according to the need of data.

C. Design and Functionality Designing functional classifier for analysis can be divided into four steps. They are:

1) Acquisition of Data:

Data is collected in the raw format from twitter using library called twee stream of python programming and twee R for R programming. The library twee stream provides an API for streaming data . There are two streams for accessing the tweets by using this API they are Filter and Sample streams. The behavior of the Filter stream is as follows:

- To search Specific keywords in the tweets.
- Search according the user ID’s.

2) Feature Extraction :

A tweet acquired has a lot of raw information which may or may not be useful for application. It comes in the form of python “dictionary” , these dictionaries can be obtained by NLTK tool ,it contains data in form of key-value pairs . A list of some key valued pairs is given below:

- User ID.
- Screen name of user.
- Original text of tweet.
- Presence of hash tags.
- Whether it is a re-tweet.
- Geo-tag location of tweet.
- Date and time tweet was created.

There are some criteria to filter the tweets and the criteria are as followed

- Remove re-tweets.
- Remove very short tweets.
- Remove non-English tweets.
- Remove similar tweet.

Twitter sustains have huge amounts of extra fields and implanted pointless data. We utilize the gettext()capacity to remove the content fields and appoint the rundown to a variable tweet T. The capacity is connected to every one of the 5000 tweets. The code beneath likewise indicates consequences of extraction for the initial 5

sustains[2].tweetT=lapply(tweet,function(t)$getText())head(tweetT,5)[2]

3) Human Labeling :

Tweets were classified into four classes they are: positive, negative, neutral and ambiguous. The labels that were suggested to the labelers were :

- Positive: Positive words/sentences have a positive sentiment attached to them. For example, when some text indicates happiness, enthusiasm, kindness etc., they're generally classified as having a positive sentiment.
- Negative: Negative sentences have a negative sentiment attached to them .For example, when some text indicates sadness, hate, violence, discrimination etc., they're generally classified as having a negative sentiment.
- Neutral: If creator of tweet expresses no personal opinion in the tweet and merely transmits information.
- Blank: Leave the tweet unlabeled if the language used is other than English on the data trained.

4) Classification : Here prediction is important with respect to the data. Basically, in R console the inbuilt classification algorithm is Naive Bayes theorem but we can also apply different kinds of classification algorithms. At first, we access the twitter data using API's and store it in the R-console and use packages in the R console to classify the data based on the polarity by applying respective algorithms and the finally finding the emotion.

D. Classification Models

Classification models predict categorical class labels. For example, we can build a classification model to categorize bank loan applications as either safe or risky. There are different types of classification models, such as :

1) Naive Bayes :

It is a classification technique form on Bayes' Theorem with associate assumption of independence among predictors. In straightforward terms, a Naive Bayes classifier assumes that the presence of a specific feature during a class is unrelated to the presence of the other feature. For example, a fruit could also be thought-about to be associate apple if it's red, round, and regarding three inches in diameter. Even if these options rely on one another or upon the existence of the opposite options, all of those properties severally contribute to the chance that this fruit is an apple

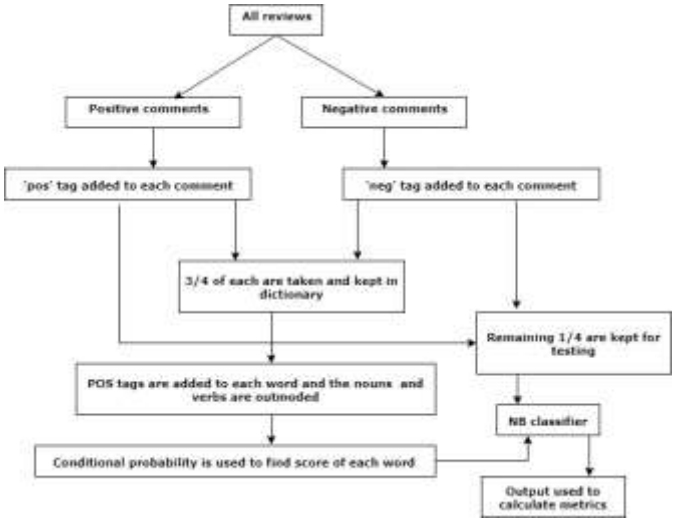
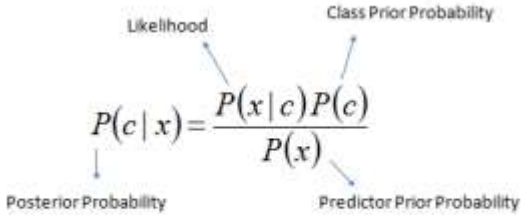


Fig.1.Flow Diagram which represents Naive Bayes Classifier process[2]

Which is why it's referred to as 'Naive'. Naive Bayes model is simple to create and notably helpful for very large data sets. Along with simplicity, Naive Bayes is understood to outdo even extremely subtle classification strategies. Bayes theorem provides a way of calculating posterior probability P(c—x) from P(c), P(x) and P(x—c). Look at the equation below[14]:



$$P(c \mid X) = P(x_1 \mid c) \times P(x_2 \mid c) \times \cdots \times P(x_n \mid c) \times P(c)$$

- 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.

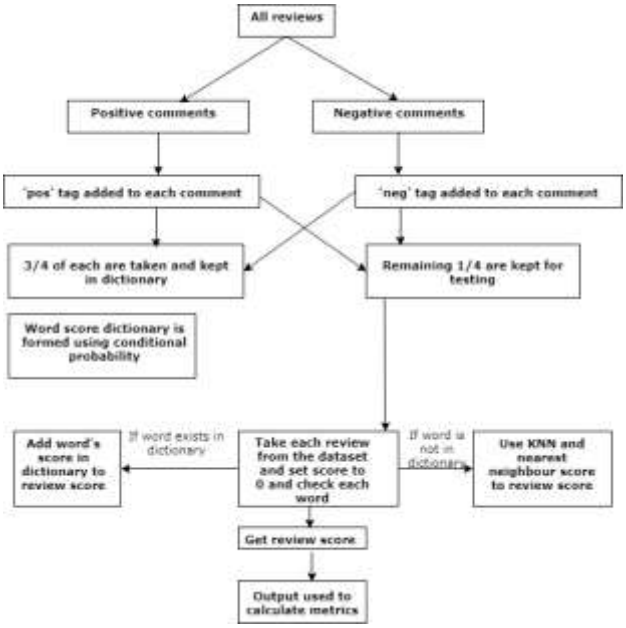


Fig.2.Flow Diagram which represents KNN Classifier process[2]

2) K-NN :

K-Nearest Neighbors is one of the most basic yet essential classification algorithms in Machine Learning. It belongs to the supervised learning domain and finds intense application in pattern recognition, data mining and intrusion detection. It is widely disposable in real life scenarios since it is non-parametric, meaning, it does not make any underlying assumptions about the distribution of data (as opposed to other algorithms such as GMM, which assume a Gaussian distribution of the given data). We are given some prior data (also called training data), which classifies coordinates into groups identified by an attribute. Algorithm Let m be the number of training data samples. Let p be an unknown point[15].

a) Store the training samples in an array of data pointsarr[]. This means each element of this array represents a tuple (x, y).
b) For I to m: Calculate Euclidean distance d(arr[i],p).
c) Make set S of K smallest distances obtained. Each of these distances correspond to an already classified data point.
d) Return the majority label among S.

3) Deep learning approaches:

Deep learning is a branch of machine learning that deals with deep neural networks (DNNs).In structure of deep neural networks it has many hidden layers following a hierarchy for features extraction .for this method more the layers, richer information can be exploited from input[8]. Deep learning can extract features from raw data as well. In current proposed work convolution neural network and recurrent neural network is involved.

Word embedding

Sentence						
Really	0.9	0.5	. . .	0.6	0.1	0.2
Loved	0.1	0.2	. . .	0.5	0.5	0.6
The	0.1	0.3	. . .	0.2	0.9	0.1
New	0.7	0.3	. . .	0.4	0.1	0.2
Feature	0.6	0.4	. . .	0.8	0.3	0.7

Fig.3.Sentence to matrix conversion[6]

a) Convolutional Neural Networks: Convolution neural network has one or more layers performing a convolution operation. CNN are mainly used to process images and videos, as it can receive multidimensional arrays as inputs. CNNs are widely used for image processing but in this case it is used for text processing. There are two-main approaches namely Zhang and LeCun[7] and Kim[6] . In Zhang and LeCun approach CNNs were used to solve natural language processing tasks such as sentiment analysis and text categorization. For this in preprocessing step string to matrix structure uses sentence characters as rows and the alphabet letters as columns. Then fill the matrix with one(1) where cells have same letters in rows and columns and zero(0) otherwise.

Kim[6] applied a CNN to analyze sentiment at the sentence level. Each row in the matrix is a word embedding. Therefore matrix is of n*m dimension where n is number of words in the sentence and m is length of word embedding Kim's CNNs extracts features from relation between word in sentence. By the help of word embedding. Kim[6] obtained semantic information to classify the sentiment from text.

Alphabet

Sentence	a	b	. . .	x	y	z
a	1	0	. . .	0	0	0
b	0	1	. . .	0	0	0
o	0	0	. . .	0	0	0
. . .	. . .	. . .	. . .	. . .	. . .	. . .
e	0	0	. . .	0	0	0

Fig.4.Sentence to matrix conversion[6]

b) Recurrent Neural Network (RNNs): This is type of DNN which deals with inputs , that makes them useful to deal with

speech and language problems. RNN keeps a history with information about previous inputs in its hidden units[11].While dealing with long sequences , common RNNs have problem. To avoid such problems Long Short Term Memory(LSTM) network[13] .a type of RNN that manages the internal information with a series of memory mechanism.

G. ASPECT - LEVEL SENTENCE ANALYSIS METHOD :

While performing aspect level sentiment analysis first identification of a certain aspect target in text and then its classification[8]. As it involves dealing with natural language , an aspect may be written in many ways , makes it difficult to identify in order to avoid creation of rules to identify aspect as well as which parts of text to consider in the sentiment classification , they have proposed two architectures. In their first architecture (SA I), they trained a model by aspect to make the identification of aspect in the sentence as input for a second model that makes an overall classification of the sentence. When they found aspect, they passed the sentence as input for second model that makes overall classification. There second architecture (SA II) has a structure similar to first one consisting of two modules. They had main difference as SA II has separately trained models for each aspect , making the classification strictly specific to the selected aspects. In SA II model input is first processed and then they had obtained meaningful worlds from it. These words are given as input to first module (Aspect Identifier) and identifying aspect, input goes to second module(Aspect Classification).

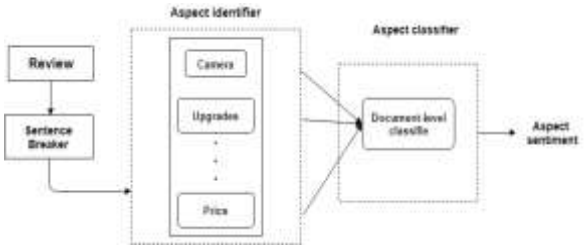


Fig.5.SA I Architecture[8]

1) Aspect As identifying sentiment at aspect – level involves the specific definition of aspect targets. They selected 10 cellphone – related aspects. Their aspects were: battery ,camera, design, display , memory, price, processor, sales and services , temperature, upgrades.

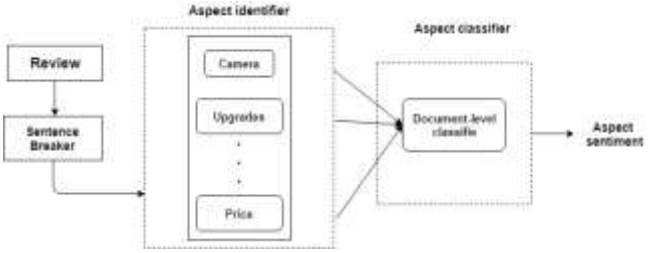


Fig.6.SA II Architecture[8]

2) Dataset : They used multiple manually- annotated datasets to train and test their models. These datasets were results of their efforts to create a Portuguese dataset for document and aspect-level sentiment classification. First dataset contains 32,000 reviews. They used a second dataset (DS II) containing 30,000 reviews with aspect annotated , by single professional within the company as the testing benchmark against which their approaches were compared.

3) Text Preprocessing Both approaches preprocessed the input text prior to its classification. In SA I they used a sentence splitter to divide a document into sentences. They used three punctuation marks (.?!) as sentence breakers. As result, each sentence of a review is an individual input to the architecture[11]. In SA II, they performed a series of NLP operations in order to do a data text cleaning on the user reviews.

4) Aspect Identification : The aspect identification step is part of both approaches but has small differences in its definition depending on the architecture. In SA I, they trained a Support Vector Machine (SVM) classier for each aspect resulting in 10 different classifiers using DS II. The input reviews are the entire document, which may contain additional aspects beyond the target aspects. They divided the dataset by aspect and performed a 10-fold cross-validation. they showed the results of this training in the Identification column in the "SA I" column. In the pipeline, they used the trained models to discover the aspects in a review. The model's input is the resulting sentences from the preprocessing step. Thus, when a sentence is classified as containing an aspect, they used it as input to the aspect classification step. In SA II, they have a similar approach in which they have the same number of models as the aspects to perform the identification. In this case, they train an LSTM network with a single layer containing 100 nodes for 40 epochs.

5) Aspect classification : Once the previous process identified the specific aspect from a piece of text, their architectures proceed to classify the sentiment polarity of the text with regards to the identified aspect. In SA I, they used a single classification model to perform the sentiment classification. They trained this model to perform document-level sentiment

analysis. Using the data from DS I, which contains 32 thousand annotated reviews, they tested several model combinations. For each model, they performed a 10-fold cross validation, and trained them over 20 epochs. Table I shows a comparison between the models applied to binary classification (positive and negative classes only) and multi-class classification (positive, negative, and neutral classes) for the following models: an SVM; an LSTM with one layer containing 256 nodes; the CNN proposed by Zhang and LeCun(Z CNN)[7]; the CNN proposed by Kim[6] with an LSTM layer containing 256 nodes (K CNN + LSTM); and a preprocessing step (described in Section III-C as the one used in SA II) followed by an LSTM with one layer containing 100 nodes (P + LSTM). We use the best model (P + LSTM) as our sentiment classier, it receives a sentence selected from the aspect identification step and classifies attributing to it a positive, negative, or neutral sentiment. Unlike SA I, the SA II classification process has an independently trained model per aspect, thus, they have a total of 10 sentiment classier models. The classier model employed is the same as the identification process, i.e., an LSTM network with one layer containing 100 nodes trained over 40 epochs

TABLE I
COMPARISON BETWEEN THE RESULTING ACCURACIES OF DOCUMENT-LEVEL APPROACHES DIVIDED BY BINARY AND MULTI-CLASS SENTIMENT CLASSIFICATION[8]

CLASSIFICATION	SVM	LSTM	Z-CNN	K-CNN+LSTM	P+LSTM
Binary	96.3%	96.35%	93.73%	96.65%	98.3%
Multi-class	91.52%	91.1%	93.65%	93.39%	93.78%

TABLE II
MOST FOUND ASPECTS IN DS III

Aspect	# of reviews
Battery	95,370
Camera	94,725
Design	50,249
Display	47,698
Memory	40,502
Price	143,072
Processor	14,232
Sales and Services	60,964
Temperature	33,563
Upgrades	36,601

IV. RESULTS AND OBSERVATIONS

In first approach the tweets of the users in twitter were extracted through APIs of python. The information that we extracted from the dataset for particular person had 399 tweets that were into the category positive based on the sentiments of the end users, and there were 231 tweets which were considered a negative and the remaining 665 emotionswere considered a neutral. Similarly with another person 564 tweets were classified as positive, 364 as negative and 866 as neutral. Accuracy in terms of first person versus Naïve Bayes and KNN were as follows accuracy under Naive Bayes is 89.6, K-NN it is 86.2. Similarly for second person the accuracy under Naive Bayes is 86.8, K-NN it is 87.6 and in case of third person it is 87.2 under Naive Bayes and 84.6 under K-NN. In second approach the comparison between SA I and SA II involves evaluating the two steps (identification and classification) in both architectures. They used DS II to compare the results and which architecture has the best performance.

Table II shows the comparison between SA I and SA II for each aspect in their list of selected ones. As results show ,SA II has a better performance over SA I in almost every aspect. Although the classification step in SA I is not restricted to a selected list of aspects, the results have shown accuracies lower than 70%, which means that the aspect specialization in classification improves task performance. Results for both identification and classification in SA II indicate accuracies over 90%, which suggests that the use of singular LSTM networks for each aspect works better for the aspect-level classification task when compared to SA I. In order to test their best approach over a larger dataset and in a different language, they used a third dataset with manually annotated aspects in English. This third dataset (DS III)

contains a total of 616,976 reviews and they described the distribution between aspects in Table III The results for both identification and classification are displayed in the column SAII English in Table II. Sentiment analysis for this dataset has accuracy above 90%, indicating that our approach can be used in different contexts obtaining similar results (i.e. increased data availability did not improve results).

TABLE III
ACCURACY COMPARISON BETWEEN APPROACH 1 AND 2 FOR THE TASK FOR ASPECT-LEVEL CLASSIFICATION

	SA I		SA II		SA II English	
Aspect	Identification	Classification	Identification	Classification	Identification	Classification
Battery	93%	86%	99%	99%	99%	93%
Camera	83%	88%	99%	98%	98%	89%
Design	96%	89%	99%	99%	99%	91%
Display	98%	80%	99%	98%	99%	90%
Memory	98%	84%	99%	98%	99%	94%
Price	93%	94%	99%	99%	98%	95%
Processor	99%	85%	99%	100%	99%	93%
Sales and Services	94%	72%	99%	98%	99%	92%
Temperature	99%	62%	99%	98%	99%	88%
Upgrades	89%	78%	98%	93%	99%	88%

V. CONCLUSION

The project helps us to analyze huge amount of data and process it. The data will be collected by the twitter streaming

API. The data collected will be analyzed, based on score we analyze how users are feeling about particular

person. We can also use this to visualize the users opinion towards other people by drawing a bar graph. According to papers referred. In case of deep learning approach second architecture gave high accuracy but Naive Bayes gave higher accuracy than both deep learning and K-NN.

REFERENCES

[1] NageswaraRaoMoparthi,Dr. N.Geethanjali Design and implementation of hybrid phase based ensemble technique for defect discoveryusing SDLC software metrics An International Conference by IEEE, AEEICB16(978-1-4673-9745-2) PP. 269-276, 2016 IEEE

[2] Ch. Nanda Krishna , Dr. P. VidyaSagar Dr. NageswaraRaoMoparthi Sentiment Analysis of Top Colleges An International Conference by IEEE,AEEICB18(978-1-5386-9), 2018 IEEE

[3] Snehal Kale VijayaPadmadas Sentiment Analysis of Tweets Using Semantic Analysis An International Conference by IEEE,AEEICB17(978-1-5386-4008-1), 2017 IEEE

[4] SemirSalawu, Yulan He, and Joanna Lumsden ,Approaches to Automated Detection of Cyberbullying ,IEEE Transformation on affective Computing

[5] Joao Paulo Aires, Carlos Padilha, Christian Quevedo, Felipe Meneguzzi ”A Deep Learning Approach to Classify Aspect-Level Sentiment using Small Datasets”,an International Conference by IEEE (978-1-5090-6014-6/18) 2018 IEEE

[6] Y. Kim, Convolutional neural networks for sentence classification, in Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, October 25-29, 2014, Doha, Qatar,A meeting of SIGDAT, a Special Interest Group of the ACL, 2014, pp.17461751.

[7] X. Zhang and L. Cun, Text understanding from scratch, CoRR, vol.abs/1502.01710, 2015.

[8] Joao Paulo Aires, Carlos Padilha, Christian Quevedo, Felipe Meneguzzi,”A Deep Learning Approach to Classify Aspect-Level Sentiment using Small Datasets”,2018 International Joint Conference on Neural Networks (IJCNN)

[9] NimitBindalNiladriChatterjee, ”A two step method for sentiment analysis of tweets”,2016 International Conference on Information Technology.

[10] Benwang Sun1, Fang Tian2, Li Liang1,”Tibetan Micro-Blog SentimentAnalysis Based on Mixed Deep Learning”,an International Conferenceby IEEE (978-1-5386-5195-7/18) 2018 IEEE

[11] P.VidyaSagar, Dr.N.Geethanjali, An Improved Parallel Activity scheduling algorithm for large datasets, International Journal of Engineering Research and Applications, Vol. 5, Issue 7, pp.23-29, 2015.

[12] Y. LeCun, Y. Bengio, and G. Hinton, Deep learning, Nature, vol. 521,no. 7553, pp. 436444, 2015.

[13] S. Hochreiter and J. Schmidhuber, Long short-term memory, Neural Comput., vol. 9, no. 8, pp. 17351780, Nov. 1997.

[14] https://www.geeksforgeeks.org/naive-bayes-classifiers/

[15] https://www.geeksforgeeks.org/k-nearest-neighbours/