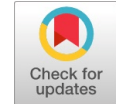# Multiclass Sentiment Analysis of Social Media Data using Neural Networks

Nirmal Varghese Babu, Fabeela Ali Rawther

Abstract: Sentiment analysis, also known as Opinion Mining is one of the hottest topic Nowadays. in various social networking sites is one of the hottest topic and field nowadays. Here, we are using Twitter, the biggest web destinations for people to communicate with each other to perform the sentiment analysis and opinion mining by extracting the tweets by various users. The users can post brief text updates in twitter as it only allows 140 characters in one text message. Hashtags helps to search for tweets dealing with the specified subject. In previous researches, binary classification usually relies on the sentiment polarity(Positive , Negative and Neutral). The advantage is that multiple meaning of the same world might have different polarity, so it can be easily identified. In Multiclass classification, many tweets of one class are classified as if they belong to the others. The Neutral class presented the lowest precision in all the researches happened in this particular area. The set of tweets containing text and emoticon data will be classified into 13 classes. From each tweet, we extract different set of features using one hot encoding algorithm and use machine learning algorithms to perform classification. The entire tweets will be divided into training data sets and testing data sets. Training dataset will be pre-processed and classified using various Artificial Neural Network algorithms such as Reccurent Neural Network, Convolutional Neural Network etc. Moreover, the same procedure will be followed for the Text and Emoticon data. The developed model or system will be tested using the testing dataset. More precise and correct accuracy can be obtained or experienced using this multiclass classification of text and emoticons. 4 Key performance indicators will be used to evaluate the effectiveness of the corresponding approach.

Keywords: Multiclass Sentiment Analysis, Data Pre-processing, Natural Language Processing, Feature Extraction, Classification, Emoticons, Neural Networks

I. INTRODUCTION

1.1 Sentiment Analysis

Sentiment Analysis of various Social Media is considered as one of the hottest topic nowadays. Here, sentiment analysis was done on the famous social media,
Twitter where the data was downloaded for the analysis. Nowadays, Sentiment Analysis is an emerging trend as a lot of organizations or institutions following this procedure to understand the views and opinions of various people[2] . For

example, the usage of a particular product can be analyzed by the way people respond to it[1]. Various Neural Network Algorithms were used for the classification procedure. the inner meaning of the texts and emoticons can be identified using this procedure.

Multiclass Sentiment Analysis provides more precise and accurate accuracy values when compared to the existing methods like Binary and Ternary Classifications. Here, the data will be classified into more than 3 classes so that the deep meaning of each and every word will be found out. Emoticons also plays an important role in the multiclass sentiment analysis. The emoticons can also be classified as same as the texts. For that, the meaning of the emoticons must be found out. The accuracy can get reduced in the multiclass analysis as the classification is getting more intense and accurate. A single word can represents different emotions or feelings but in Multiclass Classification, this problem can be reduced.

Various Hashtags were used to retrieve the required data from the various Social Media using the keyword Matching process, for example #twitter. The maximum word count of data(texts and emoticons) which can be posted online ie tweets is 140. The pre-processing[3] steps like the removal of URL's, Special symbols, Full stops, Stop words etc. were performed where the unimportant or not useful data will be removed[4]-[20]. Tokenization was done to tokenize or categorize the words from the available dataset. Stemming was also performed to remove the suffixes and prefixes from the tokenized words. Various Natural Language tools or packages are imported for this procedure.

One Hot Encoding was used here for the Feature Extraction Procedure. Here, the data will be encoded based on the available data in the unique array created from the datasets. The data will be encoded into either 1 or 0. The encoded data will be given as input to various Neural Network Algorithms like Recurrent Neural Networks, Convolutional Neural Networks, Recurrent Convolutional Neural Networks and Convolutional Neural Networks. The created model was trained using these algorithms and was tested based on the corresponding Sentiments of each tweets. The Accuracy of each model was found out and more values were predicted. Here, the data was classified into 13 sentiment classes.

II. PROPOSED SYSTEM

This Particular System contains various steps which include:
- Data Collection
- Data Pre-Processing
- Feature Extraction
- Text – Emoticon Analysis
- Classification
- Results Evaluation

2.1 Architecture

Here, the datasets which consists of both Texts and Emoticons will be downloaded or extracted from the Twitter Developer App by connecting the PyCharm Interface with the Developer App. The obtained data will be divided into both Training data and Testing data and all the processes will be carried out. In the training dataset and in the Testing dataset, the Data Preprocessing step and Feature Extraction step will be carried out. Make sure that the emoticons are not lost during the data pre-processing step. In the data preprocessing step, the Numbers, Hashtags, White spaces, URL's, Special Symbols were be removed and Stemming and Tokenization were also be done on the dataset.
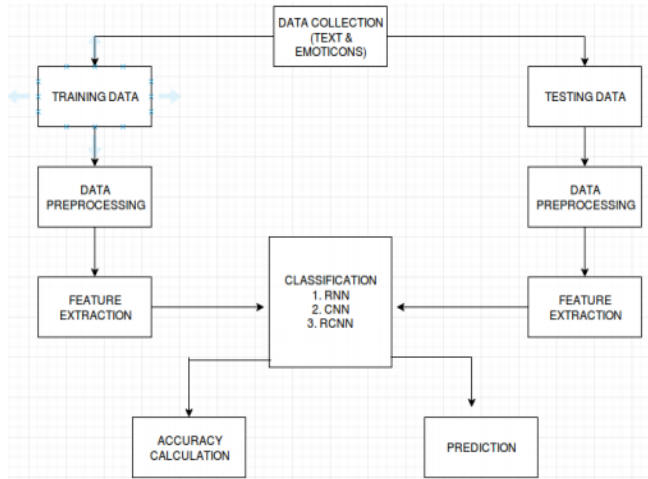

Fig. 1. Proposed System Architecture

In the Feature Extraction, the features are extracted using the one hot encoding procedure. various Classification Algorithms like, Recurrent Neural Networks, Convolutional Neural Networks, Recurrent Convolutional Neural Networks and Convolutional Recurrent Neural Networks were used in the classification procedure. After the classification, the accuracy of each algorithms was calculated from the training dataset. Here, the data will be divided into X and Y. X contains the tweets and Y contains the corresponding sentiments of each tweets whereas the values were predicted or sentiments were predicted using the Testing dataset. Here, the Sentiments were classified into 13 classes like Anger. Neutral, Boredom, Happiness, Enthusiasm, Worry, Relief, Love, Hate, Fun, Empty, Surprise and Sadness. Figure 1 shows the Proposed System Architecture.

Data Collection

Dataset for this work is collected using the Twitter API. And this dataset contains tweets by various people posted in Twitter. The total size of the dataset was 4.2 MB. Around 40, 000 tweets were downloaded from the Twitter by connecting the Twitter with the Twitter Developer App. The dataset is in the CSV Format where the different columns are the Tweet id, Sentiment, Author Name and the Content. The dataset was collected from the Twitter using the Twitter Developer API where the PyCharm IDE, can be connected with the Twitter Account using the various keys like Consumer Key, Consumer Secret Key, Access Token Key and Access Token Secret Key.

Data pre-processing

Data pre-processing[4]-[24] means the procedure of processing of the input data before the analysis process. Make sure that the emoticons are not removed during this step. The various steps involved in this procedure are:

– Removal of Numbers - Numbers present in the data set which is of no use to the analysis will be removed.

- Removal of Hashtags - Hashtags are used to extract or download the data from the Twitter. Using the Keyword Matching, the data will be extracted. After the extraction procedure, these hashtags are of not use so it will be removed during the pre-processing step.

– Removal of White spaces - The White spaces between the words in the tweets will also be removed as it increases the processing time of the entire procedure.

– Removal of URL's - URL's doesn't implies any meaning or doesn't play any part in the sentiment analysis procedure, so it will also be removed before the analysis.

– Removal of Special Symbols - The processing of special symbols takes large amount of time and it doesn't have any inner meaning unless it's an emoticon. Emoticons have an inner meaning and these emoticons only must be kept during the pre-processing step.

– Stemming - Stemming means the process of finding the root forms of the words in the tweets by removing the -es, ed, ing forms ie by removing the suffixes and prefixes. Here, Stemming was performed using both the Porter Stemmer[35] and Snowball Stemmer. Snownall Stemmer is also known as the Porter 2 stemmer as its an advanced version of the Porter Stemmer and the results of using both the stemmers were recorded.

- Porter Stemmer can be importted using :

from nltk.stem.porter import *
stemmer = PorterStemmer()

It is based on the idea that the suffixes in the English language are made up of a combination of smaller and simpler suffixes.

Example: EED - EE means "if the word has at least one vowel and consonant plus EED ending, change the ending to EE" as 'agreed' becomes 'agree'.

– Tokenization - Tokenization means the process of tokenizing or seperating the words in the tweets into different words for the easy processing or analysis of words in the future. The tokenized words are represented using the " ". Here, Tokenization was performed using both the NLTK Word Tokenizer and Word Punctuation Tokenizer[12]. The disadvantage of the Word Tokenizer was that it's considering each and every words separately and the emoticons are getting splitted during the tokenization but whereas in Word Punctuation Tokenizer, the emoticons are also considered as the single word so that they won't get splitted during the procedure.

wordpuncttokenize = WordPunctTokenizer().tokenize

wordpuncttokenize will split pretty much at all special symbols and treat them as separate units. wordtokenize on the other hand keeps things like 're together. It doesn't seem to be all that smart though.

For Example : A : "It was a great day(: "
Using wordpuncttokenize : "It", "was", "a", "great", "day", "(:"
Using wordtokenize : "It", "was", "a", "great", "day", "(", ":"

Feature Extraction

Here, the One Hot Encoding[34] algorithm was used for the Feature Extraction Procedure where the entire data will be represented in the from of 1's and 0's. The entire tweet will be splitted based on the words in the particular tweet. From the whole dataset of the tweets, the unique words will be saved in the Unique Array created. After that, with respect to the unique array created, index values will be given to all the words in the tweets. The process starts with taking the first word or element from the tweet and it will be compared with the Unique Array created. If that particular element is present, '1' will be given to that particular element and rest of the elements or words will be put '0'. The procedure will continues till the entire tweets are encoded. The tweet with largest size or the tweet with maximum number of words were found out and rest all the tweets will be padded based on the largest tweet size. The padding will be based on the largest tweet size. All the tweets less than the size of the largest tweet will be padded with Zeros. The array of the One Hot Encoded data depends on the array of unique data created from the unique words in the Tokenized Data. Also. discrete value encoding can also be used in the Feature Extraction.
For Example:
A : " It was a great day for me"
B : " It is a good day"
Unique Array : ['It', 'was', 'a', 'great', 'day', 'for', 'me', 'a', 'good']
One Hot Encoded: array((1, 0, 0, 0, 0, 0, 0), (0, 1, 0, 0, 0, 0, 0), (0, 0, 1, 0, 0, 0, 0), (0, 0, 0, 1, 0, 0, 0), (0, 0, 0, 0, 1, 0, 0), (0, 0, 0, 0, 0, 1, 0), (0, 0, 0, 0, 0, 0, 1)) , [array((1, 0, 0, 0, 0, 0, 0), (0, 1, 0, 0, 0, 0, 0), (0, 0, 1, 0, 0, 0, 0), (0, 0, 0, 1, 0, 0, 0), (0, 0, 0, 0, 1, 0, 0), (0, 0, 0, 0, 0, 1, 0), (0, 0, 0, 0, 0, 0, 0))]

Text-Emoticon Analysis

Here, there are three types of analysis are done.
– Text Multiclass Classification Analysis
– Emoticon Multiclass Classification Analysis
– Text and Emoticon Multiclass Classification Analysis

A Combination of Sentiment and Emoticon analysis will be done to check the polarity of different texts and emoticons at different points of time. The analysis will be done based on the Artificial Neural Networks for example, Recurrent Neural Networks, Convolutional Neural Network, Recurrent Convolutional Neural Network and Convolutional Recurrent Neural Network. The accuracy of different Neural Network algorithms are shown in the Tables 1, 2, 3 and 4.

Classification

Various Artificial Neural Network algorithms[25]-[32] will be used to classify the tweets and find the Accuracy and Score at different conditions. An algorithm ie the combination of text and emoticons analysis will be found out to check the Accuracy and Score and the values will be compared with the existing machine learning algorithms. The created model will be tested using the Test Dataset. X consists of the content part ie the actual tweet whereas the Y contains the various Sentiments. The System will be trained using both the content part and the Sentiments part.

– **Recurrent Neural Networks** Recurrent Neural Networks[36] is an artificial neural network where connection between nodes from a directed graph along a temoral graph. Here, Long Short Term Memory was used which is a type of RNN which basically extends the memory. The units of LSTM are used as building units for the layers of RNN, which is often called as LSTM Network. LSTM contain their information in the memory which is like the memory of a computer because the LSTM can read, write and delete its memory. LSTM has 3 gates : input, output and forget gates. These gates determines the whether or not to let new input, delete the information or to impact the output at the current time step. These gates are in sigmoid so that it can range from 0 to 1.

– **Convolutional Neural Networks** Convolutional Neural Network[37] is a type of deep network which is commonly used for analyzing visual imagery. These are regularized versions of Multilayer preceptons. The pre-processing required in a ConvNet is much lower as compared to other classification algorithms. ConvNets have the ability to learn these filters/characteristics. A ConvNet is able to successfully capture the Spatial and Temporal dependencies in an image through the application of relevant filters. It performs a better fitting due to the reduction in the number of parameters involved.

– **Recurrent Convolutional Neural Networks** RCNN contains a stack of RCLs[30], optionally interleaved with max pooling layers. Here, the LSTM Layer is followed by a Convo Layer where the data will be first processed in the LSTM layer then in the Covno Layer. These model can be created using all the parameters present in both the Recurrent and Convolutional Neural Networks.

– **Convolutional Recurrent Neural Networks** Here, the Convo Layer is followed by a LSTM Layer where the data will be first processed in the Convo layer then in the LSTM Layer. These model can be created using all the parameters present in both the Recurrent and Convolutional Neural Networks.

Table 1. Accuracy Values of RNN at different conditions

Neural Algorithm	Accuracy % (Texts Only)	Accuracy % (Text + Emoticons)	Accuracy % (Emoticons Only)
RNN	60	75	50

Table 2. Accuracy Values of CNN at different conditions

Neural Algorithm	Accuracy % (Texts Only)	Accuracy % (Text + Emoticons)	Accuracy % (Emoticons Only)
CNN	60	65	60

Table 3. Accuracy Values of RCNN at different conditions

Neural Algorithm	Accuracy % (Texts Only)	Accuracy % (Text + Emoticons)	Accuracy % (Emoticons Only)
RCNN	60	60	50

Table 4. Accuracy Values of CRNN at different conditions

Neural Algorithm	Accuracy % (Texts Only)	Accuracy % (Text + Emoticons)	Accuracy % (Emoticons Only)
RNN	60	60	60

Comparison of Classification Algorithms

Recurrent Neural Network Algorithm showed higher value while handling the Text and Emoticons. The values were less while handling the emoticons only since its hard to handle the emoticons only. Also, the accuracy value was high in Recurrent Neural Networks while Score value was high in Recurrent Convolutional Neural Networks. Also, the Convolutional Neural network had the second highest accuracy. Different algorithms have different accuracy values at different conditions like Text only, Emoticons only and Text + Emoticons. Tables 1, 2, 3 and 4 shows the Accuracy values of different Neural Network Algorithms at different conditions

III. EXPERIMENTAL RESULTS

The Proposed System was implemented using various Neural Network Algorithms like Recurrent Neural Networks, Convolutional Neural Networks, Recurrent Convolutional Neural Networks and Convolutional Recurrent Neural Networks in the classification procedure. After the Analysis and Classification procedure, the data will be plotted based on the corresponding sentiments and the results obtained while using different algorithms or methods will be verified based on the accuracy of the procedure. The total number of tweets belonging to the corresponding sentiments will be classified based on the Sentiments. The Accuracy and Score values of different Neural Network Algorithms were found out. The Accuracy and Score values were recorded using various Neural Network Algorithms for the Multiclass Sentiment Analysis and for Text Classification. The Accuracy value of the model using Recurrent Neural Networks showed the highest while others remained the same. Figure 2 shows the categorization of the tweets based on the sentiments whereas Figures 3 and 4 shows the accuracy and score values of different Neural algorithms.
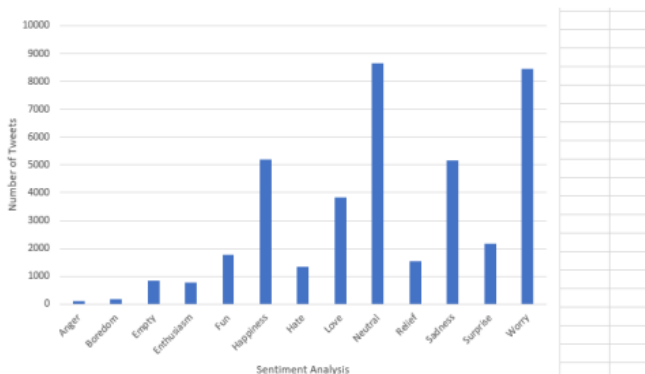

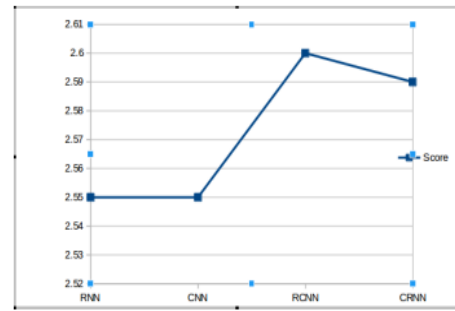**Fig. 2.** Graph Showing the Number of Tweets based on the Sentiments

Fig. 4. Graph Showing the Score values of different Neural Network Algorithms

In case of Score Values, the model with Recurrent Convolutional Neural Networks showed the highest, the model with Convolutional Recurrent Neural Networks showed the Second highest whereas the model with Recurrent Neural Networks and Convolutional Neural Networks remained the constant with lesser values when compared to others. Table 5 shows the Accuracy values of different Neural Network Algorithms.
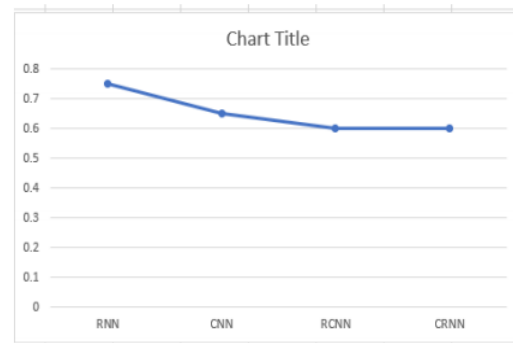

Fig. 3. Graph Showing the Accuracy values of different Neural Network Algorithms

.

As of Discussion, 40,000 tweets were downloaded from the Twitter Developer API. During the analysis, it was observed that the tweets of Neutral sentiments were comparatively high whereas tweets of Anger recorded less in number.

Table 5. Accuracy of different Neural Network Algorithms

Neural Algorithms	Accuracy (%)
RNN	75
CNN	65
RCNN	60
CRNN	60

IV. CONCLUSION

The set of tweets was classified into 13 sub classes. In Multiclass classification, many tweets of one class are classified as if they belong to the others. The two analysis were done: Sentiment Multiclass Classification analysis and Emoticons Multiclass Classification analysis. The system was made based on the algorithm which is made by the combination of Text and Emoticon Analysis.

60

The Sentiment Analysis is based on the sentiment score of both Text and Emoticons. Various Preprocessing steps were done to clean the available data and One Hot Encoding was done to extract the features from the data. Various Neural Network Algorithms like Recurrent Neural Networks, Convolutional Neural Networks, Recurrent Convolutional Neural Networks and Convolutional Recurrent Neural Networks. The Accuracy and Score values of each system or algorithm was found out. When the above 4 algorithms compared, the Recurrent Convolutional Neural Network showed high Score value whereas Recurrent Neural Network showed the high Score value.

VIII. FUTURE SCOPE

As of Future Work, the set of tweets will be classified into more classes. The Spark Architecture can be used as it provides fast streaming of Real Time Data and millions of data can be handled using this architecture. Also, Hadoop Distributed File System can be used as the data can be saved in the system and the needed data can be loaded into the system in use so the redundancy of data can be eliminated. Emojis can also be used for the Sentiment Analysis by giving a score to the emojis and can be used in the analysis. Also, a particular user's Behaviour can also be analyzed by understanding the way he/she responds to various things he/she faces in their day to day life. More inner feelings of the user can be identified. The analysis can be done using the Discrete Value Encoding and Stemming can be done using the other Stemmer Algorithms.

REFERENCES

1. Chhaya Chauhan, Smriti Sehgal "Sentiment Analysis on Product Reviews," International Conference on Computing, Communication and Automation, 2017..
2. ZHU Nanli, ZOU Ping, LI Weiguo, CHENG Meng "Sentiment Analysis: A Literature Review," IEEE ISMOT, 2012.
3. Balakrishnan Gokulakrishnan, Pavalanathan Priyanthan, Thiruchittampalam Ragavan, Nadarajah Prasath, AShehan Perera "Opinion Mining and Sentiment Analysis on a Twitter Data Stream ," The International Conference on Advances in ICT for Emerging Regions , 2012.
4. Huma Parveen, Prof. Shikha Pandey "Sentiment Analysis on Twitter Data-set using Naive Bayes Algorithm," IEEE, 2016
5. . M. Tirupati, Suresh Pabboku, G. Narasimha, "Sentiment Analysis on Twitter using Streaming API," The International Advance Computing Conference , 2017.
6. Kiichi Tago, Qun Jin "Analyzing Influence of Emotional Tweets on User Relationships by Naive Bayes Classification and Statistical Tests," 10th International Conference on Service-Oriented Computing and Applications, IEEE, 2017.
7. Nasser Alsaedi, Pete Burnap "Feature Extraction and Analysis for Identifying Disruptive Events from Social Media," International Conference on Advances in Social Networks Analysis and Mining IEEE/ACM, 2015
8. Mondher Bouazizi, Tomoaki Ohtsuki "Sarcasm Detection in Twitter," IEEE, 2015
9. Hajime Watanabe, Mondher Bouazizi, Tomoaki Ohtsuki "Hate Speech on Twitter: A Pragmatic Approach to Collect Hateful and Offensive Expressions and Perform Hate Speech Detection," IEEE Access, 2018
10. Mondher Bouazizi, Tomoaki Ohtsuki "Sentiment Analysis in Twitter: From Classification to Quantification of Sentiments within Tweets," IEEE, 2016
11. Mondher Bouazizi, Tomoaki Ohtsuki "A Pattern-Based Approach for Multi-Class Sentiment Analysis in Twitter," IEEE Access, 2017
12. Alexandros Baltas, Andreas Kanavos, Athanasios K. Tsakalidis "An Apache Spark Implementation for Sentiment Analysis on Twitter Data," 2017
13. Mondher Bouazizi, Tomoaki Ohtsuki "A Large-Scale Sentiment Data Classification for Online Reviews under Apache Spark," 9th International Conference on Emerging Ubiquitous Systems and Pervasive Networks, 2018
14. Hao Wang, Jorge A. Castanon "Sentiment Expression via Emoticons on Social Media ," International Conference on Big Data,IEEE 2017
15. Wies law Wolny, "Sentiment Analysis of Twitter data using Emoticons and Emoji ideograms ," 2016
16. Pranali Borele, Dilipkumar A. Borikar, "An Approach to Sentiment Analysis using Artificial Neural Network with Comparative Analysis of Different Techniques," 2016
17. Nikolaos Nodarakis, Spyros Sioutas, Athanasios Tsakalidis, Giannis Tzimas "Large Scale Sentiment Analysis on Twitter with Spark," EDBT/ICDT Joint Conference, 2016
18. Andreas Kanavos, Nikolaos Nodarakis, Spyros Sioutas, Athanasios Tsakalidis, Dimitrios Tsolis, Giannis Tzimas "Large Scale Implementations for Twitter Sentiment Classification," MPDI Algorithms, 2018
19. R. Ragupathy, Lakshmana Phaneendra Maguluri "Comparative analysis of machine learning algorithms on social media test," International Journal of Engineering Technology, 2018
20. Ahan M R, Honnesh Rohmetra, Ayush Mungad "Social Network Analysis using Data Segmentation and Neural Networks," International Research Journal of Engineering and Technology (IRJET), 2018
21. Georgios S. Solakidis, Konstantinos N. Vavliakis, Pericles A. Mitkas "Multilingual Sentiment Analysis using Emoticons and Keywords," IEEE/WIC/ACM International Joint Conferences on Web Intelligence (WI) and Intelligent Agent Technologies (IAT), 2014
22. N. AZMINA M. ZAMANI, SITI Z. Z. ABIDIN, NASIROH OMAR1, M. Z. Z.ABIDEN "Sentiment Analysis: Determining People's Emotions in Facebook ," Applied Computational Science
23. Wies law Wolny "Emotion Analysis of Twitter Data That Use Emoticons and Emoji Ideograms," 25TH INTERNATIONAL CONFERENCE ON INFORMATION SYSTEMS DEVELOPMENT, 2016
24. Ga¨el Guibon, Magalie Ochs, Patrice Ballot "From Emojis to Sentiment Analysis," https://hal-amu.archives-ouvertes.fr/hal-01529708, 2017
25. Xi Ouyang, Pan Zhou, Cheng Hua Li, Lijun Liu "Sentiment Analysis Using Convolutional Neural Network," IEEE International Conference on Computer and Information Technology, 2015
26. Aliaksei Severyn, Alessandro Moschitti "Twitter Sentiment Analysis with Deep Convolutional Neural Networks," ACM, 2015
27. Isabel Segura-Bedmar1, Antonio Quiros, Paloma Martınez1 "Exploring Convolutional Neural Networks for Sentiment Analysis of Spanish tweets", ACL,15th Conference of the European Chapter of the Association for Computational Linguistics, 2017
28. Dinh Nguyen, Khoung An Vo, Dang Pham, Mao Nguyen "A Deep Architecture for Sentiment Analysis of News Articles", Advances in Intelligent Systems and Computing, 2017
29. Qurat Tul Ain, Mubashir Ali, Amna Riaz, Amna Noureen, Muhammad Kamran, Babar Haya, A. Rehman "Sentiment Analysis Using Deep Learning Techniques: A Review", (IJACSA) International Journal of Advanced Computer Science and Applications, 2017
30. Siwei Lai, Liheng Xu, Kang Liu, Jun Zhao "Recurrent Convolutional Neural Networks for Text Classification", Twenty-Ninth AAAI Conference on Artificial Intelligence, 2015
31. Pranali Borele, Dilipkumar A. Borikar "An Approach to Sentiment Analysis using Artificial Neural Network with Comparative Analysis of Different Techniques", IOSR Journal of Computer Engineering (IOSR-JCE) , 2016
32. Shiyang Liao, Junbo Wang, Ruiyun Yu, Koichi Sato, Zixue Cheng "CNN for situations understanding based on sentiment analysis of twitter data", 2017
33. https://hackernoon.com/what-is-one-hot-encoding-why-and-when-do-you-have-touse-it-e3c6186d008f
34. https://kite.com/python/docs/nltk.tokenize.regexp.WordPunctTokenizer
35. https://tartarus.org/martin/PorterStemmer/
36. https://towardsdatascience.com/recurrent-neural-networks-and-lstm4b601dd822a5
37. https://medium.com/@RaghavPrabhu/understanding-of-convolutional-neuralnetwork-cnn-deep-learning-99760835f148

AUTHORS PROFILE

Nirmal Varghese Babu, Post Graduate Engineering student in Computer Science and Engineering from Amal Jyothi College of Engineering, Kanjirappally, Kerala, India

Fabeela Ali Rawther, Assistant Professor in Computer Science and Engineering from Amal Jyothi College of Engineering, Kanjirappally, Kerala, India