Comparison Study of different ML/DL approach on Twitter Sentiment Analysis For General Election

**Class** : XLS EN\_CSE 4410/6410

**Project Participants** : Md. Asaduzzaman Jabin,

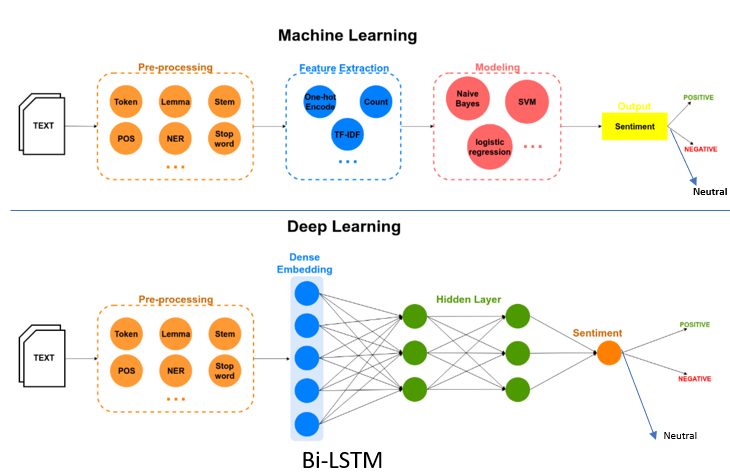
**Student ID(s)** : 811562873 (Graduate Student)

**Date** : Aug 29, 2022

# Introduction

In modern life, the social media craze controls the lives of people, impacts the business, and helps to understand the customer base. Large tech giants – MAANG- Meta, Amazon, and Google built a 30 trillion dollar business every month by only keeping their users engaged in their ecosystem. So, the emotions and sentiments of the user depict a piece of vital information visually rather than articulately in business models, democracy and social rights, and marketing. Also, it plays a crucial role in different areas- e.g, social media interaction, AdSense, client-based business model, suspecting social mob, identifying national security threats, metaverse, and AR/VR game console design, customer’s psychological study like - human behavior in social media understanding, detection of mental breakdown, social media harassment and synthetic social bullying. So, detecting social media sentiment in real-time with a high recognition rate is still challenging. The text-based sentiment is usually performed in NLP using four stages: pre-processing, extraction of features, classification expression in a dynamic manner, and training and testing.

In this project, we applied various combinations of machine learning and deep learning methods- e.g., Naïve Bayes, SVM, and Bi-LSTM architecture. The project’s novelty comes up with the combination of hyperparameters in different layers of the ML model and masks the sentiment with respective analysis. The existing maximum accuracy achieved over this dataset is <70% only. As a result, I found a scope to work on it to get a more accurate prediction of sentiment. We found a dataset using tweets’ API, pre-processed the data, extracted the right features, and applied the Naive Bayes, SVM, and Bi-LSTM Classifier to obtain public opinions. As a result, we identified outliers, analyzed controversial and swing states, and cross-validated election results against sentiments expressed over social media. The results reveal that the election outcomes coincide with the sentiment expressed on social media in most cases. The pre and post-election sentiment analysis results demonstrate the sentimental drift in outliers. Our sentiment classifier shows an accuracy with a precision of 93%.



**Fig 1**: The overall generic model architecture for sentiment analysis, (Top) Machine Learning model architecture- Naïve Bayes, SVM. (Bottom) Deep Learning model architecture- Bi-LSTM.

# Literature Review

Electronic media such as Facebook, Twitter, Instagram, LinkedIn, and discord are common ways to know the opinion and feedback of mass people. Normally, People share rumors, news, political views, social events, political campaigns, business promotions, global events, promoting social works and developments, and expressing sentiment about elections. Social media plays a vital role first time in the US election campaign in 2008. Mr. Obama utilizes the power of social media- Twitter significantly for his political campaign. Even In 2016, Mr. Trump won a victory over Mrs. Clinton, shocking everyone (CBS 16). Pre-election polls suggested the candidate’s dominance over the opponent party. It pulls all ML enthusiasts and business leaders to apply ML to social media data. Even Trump dubbed it a critical tool that played a pivotal role in his victory (CBS, 2016).

Several works on sentiment analysis all around the world have already been carried out in the 2012 U.S. election, employing the Naive Bayes Classifier using Unigram features [1], in 2016 U.S. election- lexicon feature extraction [2], Sentistrenght [3] and other elections, such as Indian PM elections [4,5], Iranian elections [6], Singaporean elections [7], and Colombian elections [8]. These works provide insights into social media sentiments as well as their correspondence with the actual election results; further details are provided in Section 3. Similarly, social media analysis of the 2020 U.S. election can also potentially unveil several hidden sentiments about both candidates. The study can become yet more critical since the shadows of rigging are cast on the elections. The sentiment analysis becomes more interesting since votes were cast via postal services. But, it is hard to recognize sentiment sarcasm trivially. In some cases, the negative sentiments are classified as positive due to their writing styles. Twitter is a much better place for sentiment analysis as compared to other social media [9]. In this sense, some people on social media might not be serious about really expressing their actual feelings. Therefore, their sentiments do not reflect the true picture.

Sentiment analysis involves classifying opinions in text into three main categories, i.e., “positive” “negative” or “neutral” [10]. Sentiment information can be extracted using various ways, including speaker recognition [11], physical activity recognition [12], philological signals [13], human facial features [14], and textual information expressed over social media. Sentiment analysis is employed in numerous fields for opinion mining, such as focusing on multi-level single and multi-word aspects to manifest several domains in Twitter datasets [15], in recommendation systems [16], being employed for business intelligence [29], for finding public opinion about a particular rule before presentation [17], in comments analysis [18], News and print media sentiment Analysis [19], commercial movie reviews analysis [20], publishing or advertising [21-22].

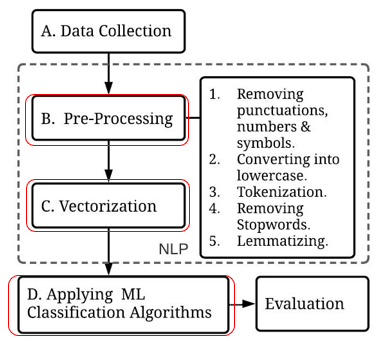
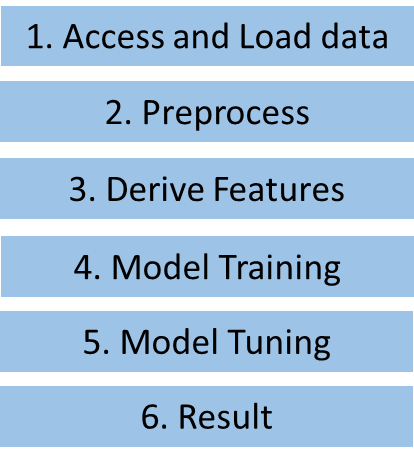
Moreover, social media might not represent complete sentiment in elections, since all voters are not present. While social media might not represent everyone completely, it provides a sample space for people’s opinions. Also, some people might not want to reveal their views due to privacy issues, so even if they are on social media, they might not express their true opinion [18]. Nevertheless, despite all these limitations, social media sentiment analysis provides the nearest approximation of public sentiment. To detect sarcasm, we used Countvectorizer() along with tokenizer(), further details are provided in Section 3.2.

In this project, pre- and post-election sentiments for both candidates in each state for the Indian PM election are considered [23]. Outliers are well known in fundamental data mining tasks to find extreme values laying outside the trends followed by other data samples. For closely contested states, finding outliers is significant for data analysis. Moreover, we analyzed public sentiment rather than election results state-wise.

# Methodology

In this method, we use Pytorch in python as the programming language. Initially, we evaluate the feasibility of our machine learning algorithms – Naïve Bayes, SVM, and Bi-LSTM using the classification learner app toolbox in MatLab to find the quick evaluation of data. Here, Fig. 1 and Fig. 2 depict the overall workflow of the machine learning process to detect the sentiment of the people. The data set consists of a few characteristics-

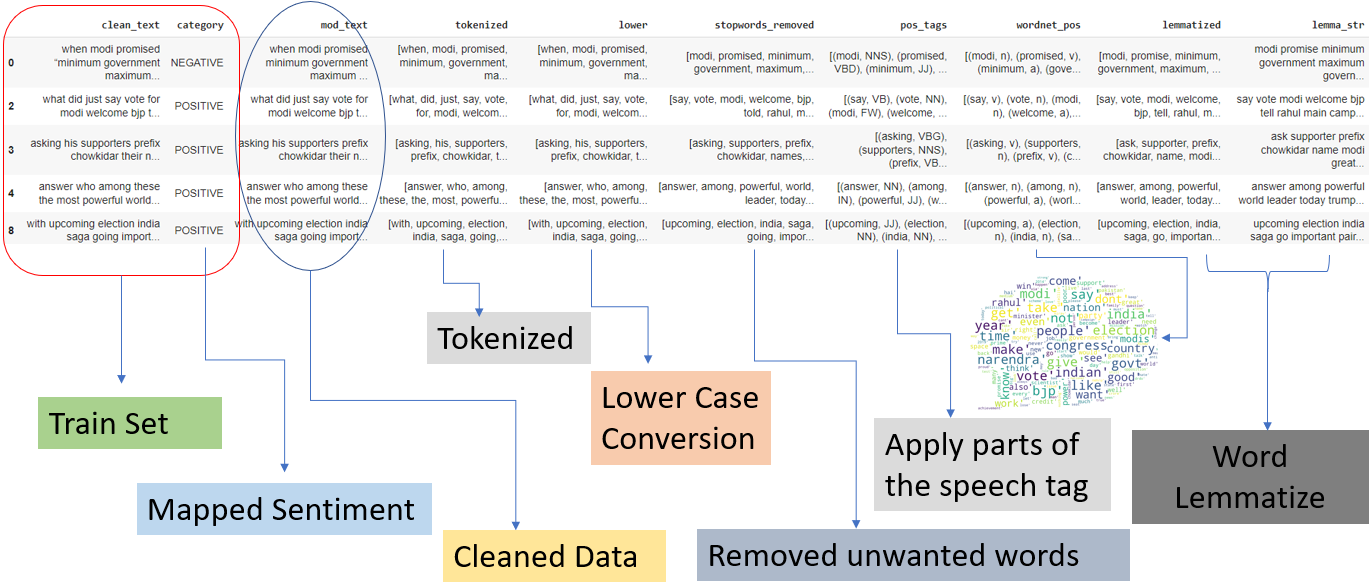
* Kaggle Twitter Data Set- Kaggle Prime Minister Election [23]
* Size- Number of tweets: 162K
* Sentiment Class/ Data Labeling: Negative (-1), Neutral (0), and Positive (+1)
* Class Distribution: Negative (23%), Neutral (33%), and Positive (44%)
* Attributes: 2, Tweet- clean text (string), category (numeric)



**Fig 2**: (Left) The overall workflow of sentiment analysis using ML, (Right) a More detailed version of the proposed NLP workflow (**red area- applied modification/ improvisation in our approach**)

To understand how sentiment is extracted at the human level please refer to Fig. 3. Basically, we need to pick a few influential words from the whole sentence and assign a tag for each part of the speech, the clean text attributes contain the public tweets (string) and category (numerical -1<x<1) and category defines the mapped sentiment (Negative (-1), Positive (+1), Neutral (0)). Then, each adjective word in the text will finally determine the sentiment of the text. However, Table 1. Displays the real-life implementation of NLP preprocessing for sentiment analysis and explained every step using the Kaggle data set [23]. Please be noted that the difference between ML and DL is that, DL

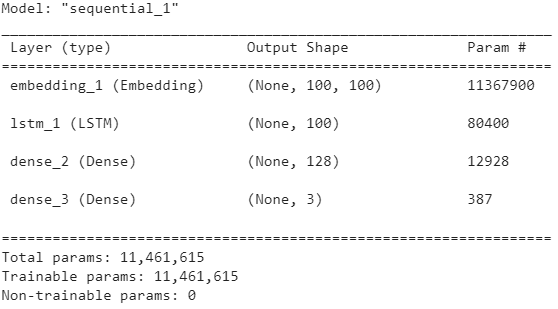
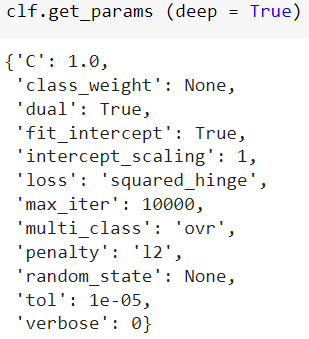
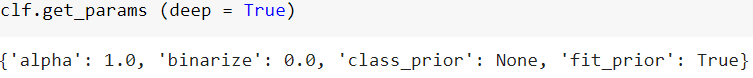
**Table 1**: The generic proposed NLP data preprocessing – sentiment mapping, data cleaning, tokenizer, lowercasing, removing unwanted words, applying parts of speech, Lemmatizing, etc.



does not have any feature extraction process, it is already embedded in the first layer (Embedded) of LSTM. But, for ML, we have the flexibility to choose our feature extractor (Countvectorizer – Acc- 75%). On the other hand, TFIDF and Hash vectorizer shows lower accuracy for machine learning models. One important thing we need to mention is that we need to do data cleaning and drop the missing/ duplicate samples before extracting the features. Then, we need to apply lowercase for the symmetry and do eliminate the stop words (unwanted words – articles, prepositions, conjunctions, etc) from the data by using the regex command in python. To know more, please refer to as Supplementary material slides (Pages 5-8). After feature scaling, we split the data in a 40%: 40%: 20% ratio respectively to train, validation, and test sets for both ML and DL approaches. Let’s summarize the overall approach-

* Data: 40%, 40%, 20%🡪 train, validation, test
* Overall test Accuracy🡪 Bi LSTM gives 93% > SVM (89%) > Naïve Bayes (75%)
* For Deep learning🡪 each layer of Bi-LSTM
  + Test 1: On baseline code (code reference – page 2) we used Dense(256) nodes to give 75% accuracy with the changed data set (Equal sentiment distribution – 33% on each sentiment (positive, neutral, negative)).
  + Test 2: Dense(64) come up with 72%
  + Test 3: Dense (125) come up with 93%
* For Machine Learning
  + SVM
    - Test 1: On baseline code (code reference – page 2) gives for TfidfVectorizer with 73% accuracy, with the changed data set (Equal sentiment distribution – 33% on each sentiment (positive, neutral, negative)).
    - Test 2: HashingVectorizer gives 63% accuracy
    - Test 3: Based on Countvectorizer got an accuracy of 89% (alpha = 1.0, L2 regularize, Tolerance 1e-5)
  + Naïve Bayes
    - Test 1: Baseline code (code reference – page 2) with the changed data set (Equal sentiment distribution – 33% on each sentiment (positive, neutral, negative)) gives 69% accuracy for (alpha =2.0, 5 fold cross validation)
    - Test 2: For changed parameters (alpha =1.0, 10 fold) it gives 75.52% accuracy

The most efficient hyperparameters could be achieved by running a lot of tests and the most updated parameters are depicted in Fig. 3.



(b)

(c)

(a)

**Fig 3**: (a) Tuned hyperparameters for (a) Naïve Bayes, (b) SVM, and (c) Bi-LSTM models.

# Result Analysis

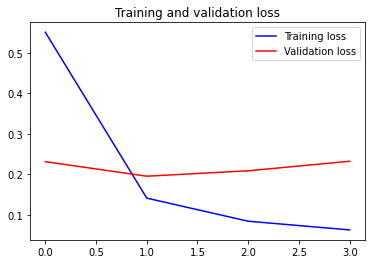
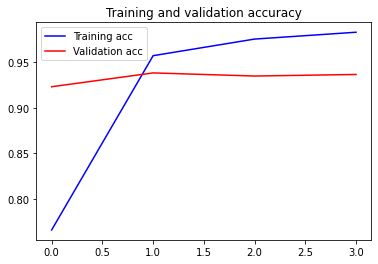
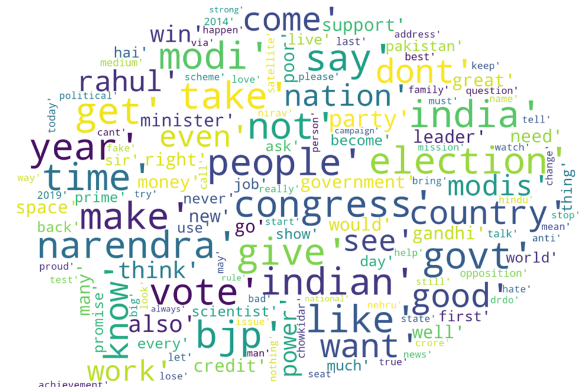
We considered 10-fold cross-validation both for Naïve Bayes and SVM to avoid overfitting problems and used a validation set to tune the model hyperparameters. The tuned hyperparameters for the model have resulted in Fig. 3. And, the overall accuracy comparison is depicted in Table 2. Here, Bi-LSTM is the most accurate model for this data set. So, we displayed the most common buzzwords in Fig. 4 (left) that are found on the entire data set and analyzed the model loss and accuracy trends during the whole training and validation process. Fig. 4 (right), shows the loss and accuracy trends of the whole training and validation process for Bi-LSTM (most accurate at 93% in testing). Here we can notice that the loss (blue) during training goes almost to zero and validation remains flat throughout the whole process (red). However, accuracy goes up to 98% during training and but validation loss remains almost flat. The validation set helps to use precorrection of the hyperparameter again after the training process which ensures the generalization of the data in the future scenario.

After testing the model, we display the confusion matrix for each algorithm. Here we, found that positive and neutral tweets depict more confusion about the model training as depicted in Fig. 5.

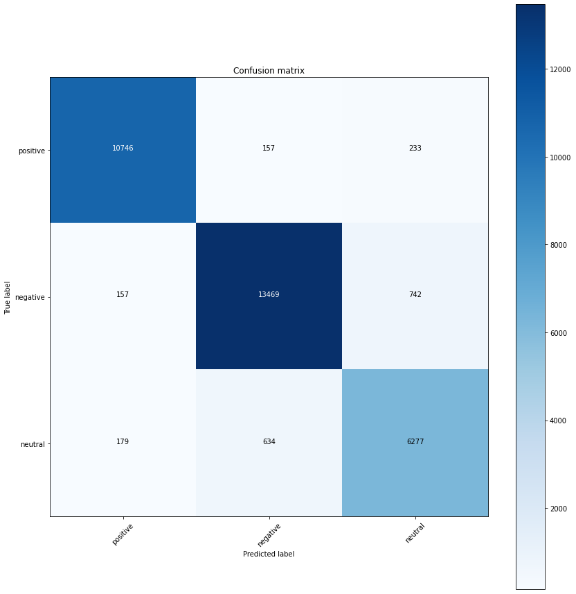
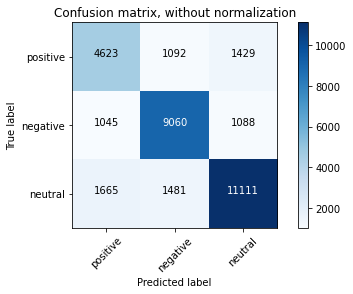
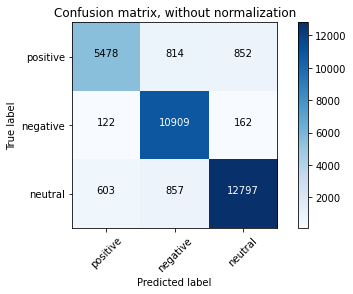
Overall, we compile the code on Core-i9, 16 GB, RTX 3050Ti architecture. All the processing time resulted in the same machine architecture for whole data sets of 162K tweets.

**Table 2**: Accuracy comparison of Naïve Bayes, SVM, and Bi-LSTM algorithms for the testing set. A picture containing text

Description automatically generated



F**ig 4**: (Left) The word cloud for most used buzzwords in tweet sentiments, (Right) The loss and accuracy trends for the highest accurate model (Bi-LSTM 93% testing accuracy)



**Fig 5**: The Confusion matrices for Naïve Bayes, SVM, Bi-LSTM Respectively for sentiment analysis.

# FINAL REPORT - Conclusions

In this project, we have done a comparative study of different ML and DL algorithms for sentiment analysis of election campaign data. We use simplified procedure and model architecture to classify 3 sentiments. From testing results, we can say **SVM is the overall best approach (89%, 600 seconds) when we consider both learning time** and accuracy as crucial parameters, **but in terms of accuracy, LSTM will be the best model (93% accuracy, 14400 seconds)**. Our model will generalize for all future data since we consider holdout validation and 10-fold cross-validation considered during the training process. Moreover, from the confusion matrix, we can say that positive and neutral sentiments are more complex to classify than negative sentiments.

# Appendix

* Our Code: Google drive:
* <https://colab.research.google.com/drive/1Ghc0IgW7HIeUPfO5UTYX7yXxLXzk7rS9?usp=sharing>
* Reference data set:

<https://www.kaggle.com/code/subhadeepdebnath/twitter-sentiment-analysis-lstm-complete-guide/data>

Code Reference 1: <https://github.com/soham2707/Twitter-Sentiment-Analysis->

Code Reference 2: <https://www.kaggle.com/code/moathmohamed/twitter-sentiment-lstm-98>

* Presentation: Video uploaded to google drive

**Reference:**

[1] Wang, H.; Can, D.; Kazemzadeh, A.; Bar, F.; Narayanan, S. A system for real-time sstwitter sentiment analysis of 2012 us presidential election cycle. In Proceedings of the ACL 2012 System Demonstrations, Jeju, Kprea, 8–14 July 2012; pp. 115–120.

[2] Nausheen, F.; Begum, S.H. Sentiment analysis to predict election results using Python. In Proceedings of the 2018 2nd International Conference on Inventive Systems and Control (ICISC), Coimbatore, India, 19–20 January 2018; IEEE: Piscataway, NJ, USA, 2018; pp. 1259–1262.

[3] Caetano, J.A.; Lima, H.S.; Santos, M.F.; Marques-Neto, H.T. Using sentiment analysis to define twitter political users’ classes and their homophily during the 2016 American presidential election. J. Internet Serv. Appl. 2018, 9, 18.

[4] Jose, R.; Chooralil, V.S. Prediction of election result by enhanced sentiment analysis on Twitter data using Word Sense Disambiguation. In Proceedings of the 2015 International Conference on Control Communication Computing India (ICCC), Trivandrum, India, 19–21 November 2015; pp. 638–641.

[5] Singhal, K.; Agrawal, B.; Mittal, N. Modeling Indian general elections: Sentiment analysis of political Twitter data. In Information Systems Design and Intelligent Applications; Springer: Berlin/Heidelberg, Germany, 2015; pp. 469–477.

[6] Salari, S.; Sedighpour, N.; Vaezinia, V.; Momtazi, S. Estimation of 2017 Iran’s Presidential Election Using Sentiment Analysis on Social Media. In Proceedings of the 2018 4th Iranian Conference on Signal Processing and Intelligent Systems (ICSPIS), Tehran, Iran, 25–27 December 2018; IEEE: Piscataway, NJ, USA, 2018; pp. 77–82.

[7] Choy, M.; Cheong, M.L.; Laik, M.N.; Shung, K.P. A sentiment analysis of Singapore Presidential Election 2011 using Twitter data with census correction. arXiv 2011, arXiv:1108.5520.

[8] Cerón-Guzmán, J.A.; León-Guzmán, E. A sentiment analysis system of Spanish tweets and its application in Colombia 2014 presidential election. In Proceedings of the IEEE International Conferences on Big Data and Cloud Computing (BDCloud), Social Computing and Networking (Socialcom), Sustainable Computing and Communications (Sustaincom)(BDCloud-Socialcom Sustaincom), Atlanta, GA, USA, 8–10 October 2016; IEEE: Piscataway, NJ, USA, 2016; pp. 250–257.

[9]. Salloum, S.A.; Al-Emran, M.; Monem, A.A.; Shaalan, K. A survey of text mining in social media: Facebook and twitter perspectives. Adv. Sci. Technol. Eng. Syst. J. 2017, 2, 127–133.

[10] Turney, P.D. Thumbs up or thumbs down? Semantic orientation applied to unsupervised classification of reviews. arXiv 2002, arXiv:cs/0212032.

[11] Cumani, S.; Laface, P.; Kulsoom, F. Speaker recognition by means of acoustic and phonetically informed GMMs. In Proceedings of the Sixteenth Annual Conference of the International Speech Communication Association, Dresden, Germany, 6–10 September 2015; pp. 200–204.

[12] Usman Sarwar, M.; Rehman Javed, A.; Kulsoom, F.; Khan, S.; Tariq, U.; Kashif Bashir, A. Parciv: Recognizing physical activities having complex interclass variations using semantic data of smartphone. Softw. Pract. Exp. 2021, 51, 532–549.

[13] Narejo, S.; Pasero, E.; Kulsoom, F. EEG based eye state classification using deep belief network and stacked autoencoder. Int. J. Electr. Comput. Eng. (IJECE) 2016, 6, 3131–3141. [CrossRef]

[14] Kalsum, T.; Mehmood, Z.; Kulsoom, F.; Chaudhry, H.N.; Khan, A.R.; Rashid, M.; Saba, T. Localization and classification of human facial emotions using local intensity order pattern and shape-based texture features. J. Intell. Fuzzy Syst. 2021, 40, 9311–9331.

[15] Janjua, S.H.; Siddiqui, G.F.; Sindhu, M.A.; Rashid, U. Multi-level aspect based sentiment classification of Twitter data: Using hybrid approach in deep learning. PeerJ Comput. Sci. 2021, 7, e433.

[16] Pradhan, R.; Khandelwal, V.; Chaturvedi, A.; Sharma, D.K. Recommendation System using Lexicon Based Sentimental Analysis with collaborative filtering. In Proceedings of the 2020 International Conference on Power Electronics IoT Applications in Renewable Energy and its Control (PARC), Mathura, India, 28–29 February 2020; pp. 129–132.

[17] Guo, X.; Li, J. A novel twitter sentiment analysis model with baseline correlation for financial market prediction with improved efficiency. In Proceedings of the 2019 Sixth International Conference on Social Networks Analysis, Management and Security (SNAMS), Granada, Spain, 22–25 October 2019; IEEE: Piscataway, NJ, USA, 2019; pp. 472–477.

[18] Cardie, C.; Farina, C.; Bruce, T. Using natural language processing to improve erulemaking: Project highlight. In Proceedings of the 2006 International Conference on Digital Government Research, San Diego, CA, USA, 21–24 May 2006; pp. 177–178.

[19] Mukwazvure, A.; Supreethi, K. A hybrid approach to sentiment analysis of news comments. In Proceedings of the 2015 4th International Conference on Reliability, Infocom Technologies and Optimization (ICRITO)(Trends and Future Directions), Noida, India, 2–4 September 2015; IEEE: Piscataway, NJ, USA, 2015; pp. 1–6.

[20] Štrimaitis, R.; Stefanoviˇc, P.; Ramanauskaite, S.; Slotkien ˙ e, A. Financial Context News Sentiment Analysis for the Lithuanian ˙ Language. Appl. Sci. 2021, 11, 4443. [CrossRef]

[21] Singh, V.K.; Piryani, R.; Uddin, A.; Waila, P. Sentiment analysis of movie reviews: A new feature-based heuristic for aspect-levelsentiment classification. In Proceedings of the 2013 International Mutli-Conference on Automation, Computing, Communication, Control and Compressed Sensing (iMac4s), Kottayam, India, 22–23 March 2013; IEEE: Piscataway, NJ, USA, 2013; pp. 712–717.

[22] Jin, X.; Li, Y.; Mah, T.; Tong, J. Sensitive Webpage Classification for Content Advertising. In Proceedings of the 1st International Workshop on Data Mining and Audience Intelligence for Advertising, ADKDD ’07, San Jose, CA, USA, 12 August 2007; Association for Computing Machinery: New York, NY, USA, 2007; pp. 28–33.

[23] https://www.kaggle.com/code/subhadeepdebnath/twitter-sentiment-analysis-lstm-complete-guide/data