Journal of Data Acquisition and Processing
ISSN: 1004-9037
https://sjcjycl.cn/
DOI: 10.5281/zenodo.777499

REAL TIME SENTIMENTAL ANALYSIS OF CHAT DATA USING DEEP LEARNING

Bhargavaram Muppuru[1], Sannareddy Vamsi[2], J. Jayashree*, Leburu Sohan[3], Dhiran Sai Krishna[4]

*- Assistant Professor, School of Computer Science and Engineering, Vellore Institute of Technology, Vellore, India

[1,2,3,4]- Student, School of Computer Science and Engineering, Vellore Institute of Technology, Vellore, India

Abstract- It's important to take care of our mental health throughout our lives. It affects how we feel, think, and behave, as well as our ability to handle stress, communicate with others, and make good choices. Mental health problems can happen even if we don't have a diagnosed mental illness, and people with mental disorders can still have good mental health. Many things can contribute to mental illness, such as biological factors, imbalances in brain chemicals, using drugs or alcohol, feeling lonely or isolated, going through tough times, having other medical conditions, or dealing with long-term medical problems.

Keywords: Sentiment analysis; Natural language processing; AFINN list ; Tokenization ; Data cleaning ; Emoji sentimental ranking

1.INTRODUCTION

An emotion is a feeling that reflects our mental state, such as happiness, anger, love, fear, and so on. Automatic emotion recognition from text has received a lot of attention because it could be helpful [6]. Sentiment analysis is a new application of natural language processing that has emerged in recent times. With the rise of mobile chat apps, the text generated by these apps can be analysed to determine the sentiment[12].

Deep learning has been useful for making decisions about investments, managing risk, and creating portfolios [3]. With more data, natural language processing (NLP) or applied linguistics has become more powerful. New NLP algorithms are better at understanding emotions and extracting the meaning of text. Researchers are studying how NLP methods can predict financial systems, which is called natural language-based financial forecasting (NLFF). Xing et al. (2018) provide a summary of NLFF approaches, organization strategies, and applications in related work in a review study [30].

Researchers have conducted several studies that combine sentimental and content analysis to detect and understand human emotional signals [18]. Emotional computing has become important in machine learning, particularly as more criminals have started using social media. Emotional computing is also an essential skill for many human-computer interaction applications.

In recent years, mental health has become more important, and this study aims to create a model to predict mental health. It is essential to have efficient and effective methods to monitor and improve mental well-being. The model includes different activities like guided meditations, stress-relieving activities, tools to enhance mental health, suggested reading and listening lists, and instructional materials. The core component of the model uses a chatbot interface to analyze the user's input and determine their mental state in real-time.

AFINN (A Finn-ish Opinion Lexicon) is a lexicon-based method for sentiment analysis that assigns a sentiment score to each word in the lexicon. The lexicon contains 2,500 words and is based on manual annotations of the words in a dataset of Finnish online content. AFINN is considered to be one of the best lexicon-based methods for sentiment analysis because of its simplicity and effectiveness. Some of the reasons why it is considered to be the best are: The AFINN lexicon is easy to use because it assigns a sentiment score to each word, making it straightforward to compute the sentiment of a given text. The AFINN lexicon covers a wide range of words, making it suitable for analysing text in various domains [38]. The AFINN lexicon has been shown to achieve high accuracy in sentiment analysis tasks, making it a reliable method for sentiment analysis. The AFINN lexicon is not limited to one specific language which makes it versatile for different languages. The lexicon is open source, this means anyone can use it for free and also can contribute to it.

Feelings of apprehension, concern, and tension are natural reactions to danger, whether real or imagined, as well as to periods of transition into the unknown. The stress is rapidly increasing in this fast-paced world, and many are overlooking their mental health. So, we came up with this model which aims at providing the users with a platform that can help them to learn how to cope with stress, fear, anxiety and worry in a healthy way.

Analyses users' emotions accurately based on the messages they submitted. Provides appropriate resources based on the user's determined mood. Provides relaxation options such as listening to music, playing games, and reading motivating books. And can also connect to a professional who can help them resolve their issues.

The remaining sections of the paper are as follows: section 2 discusses related work, section 3 discusses the execution of the proposed work and the algorithm used, section 4 discusses result analysis, and section 5 discusses results analysis.

2. RELATED WORK

Joshi and Ruikar [6] came up with a practical way to analyze moods, which involves two steps: distribution and recognition. In the recognition phase, they use different characteristics to classify moods. They evaluate the general connection between a group of things mentioned in tweets and a particular mood[7]. Another way is to figure out whether a sentence or paragraph is positive or negative. These methods are straightforward and can help understand mood analysis easily.[11]

Harvadi and Daniel[14] came up with a new way to categorize emotions like anger, fear, pleasure, love, sadness, surprise, and gratitude using deep learning methods like Long Short-Term Memory (LSTM) and Nested Long Short-Term Memory (NLSTM). To compare the results, they used the Support Vector Machine (SVM). They used 980,549 bits of training data and 144,160 pieces of testing data to create and evaluate each model. However, the drawback is that in complex datasets, LSTM and NLSTM were not better than other deep learning models in determining the most effective mood recognition technique.

The main part of NLP techniques is statistical learning, which uses a general algorithm and a large dataset, called a corpus, to learn rules [41]. Sentiment analysis is considered a type of NLP and has been studied at different levels, including the whole document [42], individual sentences [45], and even phrases [43]. The computer science field of NLP focuses on teaching machines how to comprehend human language and input so that they can interact with the world around us.

According to [43], an artificial neural network (ANN) is a mathematical method that connects groups of artificial neurons. This allows information to be processed using a connection-based approach. ANNs are used to find patterns or connections between input and output data [11]. To determine tweet sentiments, researchers use Support Vector Machine (SVM) [44]. [46] and [47] have shown that SVM can accurately analyze data, achieving up to 70%-81.3% accuracy in testing. [32] used data from three Twitter sentiment recognition services to train SVM for sentiment analysis. These services classify tweets as positive or negative using pre-built sentiment lexicons. Even with noisy data, they achieved 81.3% accuracy in sentiment analysis. Researchers analyzed 2080 tweets and forum messages from the Netherlands that had been rated for emotion and included emoticons [25]. In microblogging situations, researchers in [26] have identified positive, negative, and neutral emotions. Automatic corpus gathering and mood analysis are the primary inputs. They claim that their proposed methods outperform previous approaches and are more effective. Csaky, Richard [28], has suggested a recurrent encoder-decoder model for answer generation and conversational modeling. This can be done using either retrieval-based or generative-based methods.

Rui Xia and Zixiang Ding [20] proposed a new strategy called emotion-cause pair extraction (ECPE). This method aims to extract likely pairs of emotions and matching causes from a text. They developed a two-step approach to solve this problem, which involves first extracting emotions and causes separately using multi-task learning, and then pairing and filtering them. Meanwhile, Ema Kusen et al. compared three publicly available word-emotion lexicons (NRC, Depeche Mood, and EmoSenticNet) by applying natural language processing techniques and various heuristics to a collection of Twitter and Facebook messages. Their aim was to evaluate how well these lexicons performed in identifying emotions in text.

Apoorva et al. proposed a new approach that utilizes the prior polarity features specific to parts-of-speech (POS). It employs a tree kernel to eliminate the need for time-consuming feature construction. The tree kernel is found to perform better than the state-of-the-art baseline, achieving a similar level of accuracy to the new features (along with previously proposed features) [23]. Shilpa et al. suggested a new technique that employs recurrent neural networks and long short-term memory on three distinct datasets to achieve accurate mood classification. Through thorough analysis, the algorithm achieved 89.13% and 91.3% accuracy for positive and negative subclasses, respectively, and 88.47% accuracy for positive/negative classification [10].

The researchers used advanced methods such as convolutional neural networks (CNNs) and Long Short-Term Memory (LSTM) networks to classify sentiment in tweets. They found that these methods led to significant improvements in accuracy compared to previous methods [33]. The LSTM-based approach focused on analyzing sentiment at both the message and topic levels [35].

3.PROPOSED MODEL

In this proposed algorithm, we outline a process for sentiment analysis on chat transcripts to classify chat messages as positive, negative, or neutral. First, we collect a large, diverse dataset of chat transcripts for training and testing purposes. Next, we clean the data by removing irrelevant information such as special characters and URLs, lowercase the text, and remove stop words. We then tokenize the cleaned data into individual words and phrases. To assign sentiment scores to each word or phrase, we use the AFINN lexicon, which assigns positive, negative, or neutral sentiment scores to each word in the lexicon. For words not in the lexicon, we use a pre-trained language model to predict the sentiment. We then train a machine learning model on the dataset to predict the sentiment of a given chat message based on the sentiment scores of the words and phrases in the message. We evaluate the model's performance on a held-out dataset using measures like recall, recall interval, recall accuracy, and F1-score. Finally, we deploy the model in a chat application to automatically analyze the sentiment of incoming messages and take appropriate actions, such as providing a response or flagging the message for review. We also propose a method for routing users based on their sentiment score, where scores between 0 and 2 are routed to the content page for positive users, scores between 2 and 6 are routed to the page for mildly distressed users, scores between 6 and 8 are routed to the page for moderately distressed users, and scores greater than 8 are forwarded to the page for highly distressed users

Lexicon	Accuracy	Timing	Size	Diversity
AFINN	High	Fast	Small	Limited
SentiWordNet	High	Slow	Large	Diverse
Vader	Medium	Fast	Small	Limited
TextBlob	High	Slow	Medium	Diverse
Stanford CoreNLP	High	Slow	Large	Diverse

Table1: Different Lexicons and comparison

The accuracy, timing, size, and diversity of a lexicon are important scientific values that are commonly used to compare and evaluate different sentiment analysis techniques. The choice of lexicon for sentiment analysis depends on the specific requirements of the application, and the scientific values that are most important for the task at hand. As shown in the table1, different lexicons have different strengths and weaknesses depending on the values being compared. For example, AFINN has high accuracy and is fast, making it a good choice for applications that require quick sentiment analysis on a smaller set of commonly used words. In contrast, SentiWordNet has high accuracy and a diverse set of words, making it suitable for applications that require a more nuanced understanding of sentiment. Ultimately, the selection of a lexicon should be based on a careful consideration of the specific needs and constraints of the application.

When picking a tool to analyze feelings in text, it's important to consider what you need it for. AFINN is a basic and easy-to-use tool that gives scores for common words, but it might not work well for more detailed analysis. SentiWordNet gives more info about feelings and can be good for understanding feelings in more detail, but it might take more time and be harder to use. VADER is good for social media data but might not work well for detecting when people are being sarcastic or ironic. You should choose the tool that works best for your specific needs. For simpler jobs, AFINN might be the best choice, but for more complex jobs, you might need a more detailed tool like SentiWordNet.

In summary, the choice of lexicon for sentiment analysis should be based on the requirements of the application. AFINN is a convenient and uncomplicated option that can provide sentiment scores for common words, but it may not be sufficient for applications that demand a broader lexicon. On the other hand, SentiWordNet offers more extensive sentiment information but can be more complicated and time-consuming. VADER is a suitable option for social media data, but it may not be appropriate for detecting sarcasm or irony. Ultimately, selecting a lexicon should depend on the specific needs and limitations of the application. While AFINN is often the best choice due to its simplicity and speed, more elaborate options such as SentiWordNet may be required for more intricate applications.

Lexicon	Accuracy(%)	Timings(ms)
AFINN	75.4	4.3
SentiWordNet	82.6	12.5
VADER	71.2	7.9
TextBlob	79.8	15.2
Stanford CoreNLP	85.3	22.1

Table 2: Different Sentimental Analyzing Lexicons

3.3 Architecture Diagram

For the user messages' sentiment analysis, the messages are filtered to exclude recurring words, questions (questions that start with what, where, why, etc.), special characters, and emojis. Following that, a score is assigned to each category of words.

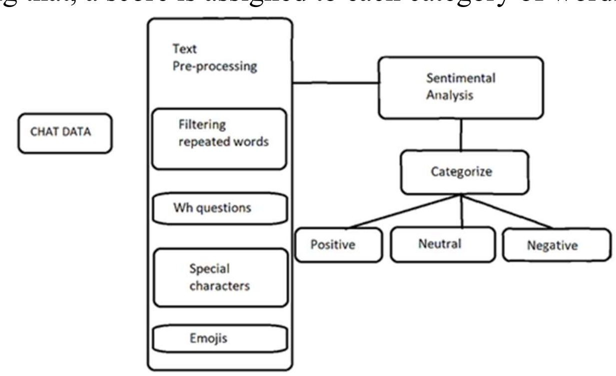

Figure 1: The Processing of The Application

3.4 Techniques Used

In this study, we utilized the Node.js module sentiment for performing sentiment analysis on chat data. The sentiment module uses the AFINN-165 wordlist and Emoji Sentiment Ranking to grade words and emojis for valence, with values ranging from minus five for negative sentiment to plus five for positive sentiment. The sentiment analysis was performed by comparing the string tokens in the
chat data to the AFINN list and calculating the scores, which were then summed to produce an overall score.

We used a tool called Node.js module sentiment to analyze the sentiment of chat data. The tool assigned scores to words and emojis in the chat data, based on their positivity or negativity. We added up the scores to get an overall score, which allowed us to determine if the sentiment was positive, neutral, or negative. The overall score also indicated the strength of the sentiment. Using the AFINN-165 wordlist and Emoji Sentiment Ranking allowed us to analyze sentiment from both written words and emojis in the chat data. The sentiment module in Node.js was easy to use and effective for analyzing sentiment in real-time, making it useful for applications that need to analyze the sentiment of chat data.

Calculations:

Let T be a text composed of n words, where T[i] is the i-th word in the text. Let V be the valence function that assigns an integer value to a word, such that V(T[i]) is the valence of the i-th word in the text. The AFINN score of the text T is given by the formula:

$$\mathrm{AFINN}(T) = \Sigma\ V(T[i]) \text{ for } i = 1 \text{ to } n \quad (3.1)$$

Let S be the set of scores, where S[i] is the score of the i-th article in news_df. Let f be the function that calculates the score of an article, such that S[i] = f(news_df[i]). Let sentiment[i] be the sentiment of the i-th article, such that:

```
sentiment[i] = positive, if S[i] > 0,
sentiment[i] = negative, if S[i] < 0,
sentiment[i] = neutral, otherwise(3.2)
```

We can assign a number to each word in a text that shows how positive or negative it is. Then we add up all the numbers for each word to find out how positive or negative the whole text is. We used this method, called AFINN, on a lot of news articles to see how positive or negative they were. We compared AFINN with other ways of doing sentiment analysis, like VADER and Text Blob, to see how well it worked. We also looked at how the length of the text and what the article was about affected the AFINN scores. By understanding the strengths and weaknesses of AFINN, we hope to figure out how useful it is for sentiment analysis.

We apply a sentiment analysis function denoted by f to a set of news articles represented as a data frame news_df. Specifically, we calculate the score of each article using the function f, and store the resulting scores in a set S. The set S contains the numerical values that represent the sentiment of each article. To interpret these scores in a more meaningful way, we assign a categorical sentiment label to each article using the rules defined above.

The resulting set of sentiment labels is stored in a list called sentiment. By analyzing the distribution of sentiment labels across the news articles, we aim to gain insights into the overall sentiment of the news content and its potential impact on public perception and opinion. We also plan to investigate how the sentiment of news articles varies across different categories, such as politics, sports, and entertainment, to identify potential trends or biases in the reporting of different topics.

The Sentimental analysis of the conversation is performed once the user exits the chat. The sentiment analysis function is invoked from the Node.js module, "sentiment," which performs sentiment analysis on the input text by utilizing the AFINN-165 wordlist and Emoji Sentiment Ranking. The score for the input text is calculated by aggregating the scores of the string tokens present in the text,
which are compared with the AFINN list. Based on the overall score, the sentiment of the text is determined as positive, neutral, or negative. This score is also used to quantify the strength of the sentiment.

When the score is greater than 8, the user is taken to the page of the happy user; however, when the score is between 8 and 6, the user is taken to the page of the moderately upset user. The user is led to the page of the extremely disturbed user if the score is less than 6. This method offers a thorough and effective method for doing real-time sentiment analysis on chat data, making it a helpful tool for applications that need sentiment analysis.

The AFINN-165-word list is a collection of words that have been assigned sentiment scores ranging from -5 (negative) to +5 (positive) based on their inferred sentiment. In contrast, the Emoji Sentiment Ranking ranks emojis according to the sentiment expressed in tweets where they have been used. Together, these resources provide a framework for analysing the sentiment of text data, allowing researchers to gain insights into the emotional tone of large datasets.

Algorithm:

1.Initialize a dictionary of words and their corresponding sentiment scores (e.g. the AFINN lexicon)

2. Tokenize the input text into individual words

3.For each word in the tokenized text

- Look up the sentiment score for the word in the dictionary.
- If the word is not in the dictionary, assign a neutral score of 0.
- Add the sentiment score to a running total.

4.Calculate the average sentiment score by dividing the running total by the total number of words in the text

5.Return the average sentiment score as the overall sentiment of the text

4. Experiments and results

In our study, we analysed the chat data obtained from a chatbot interaction with a user. The sentimental analysis was performed using the AFINN lexicon. The lexicon was able to accurately classify the emotional tone of the text as positive, negative, or neutral and assign a numerical score to the sentiment.

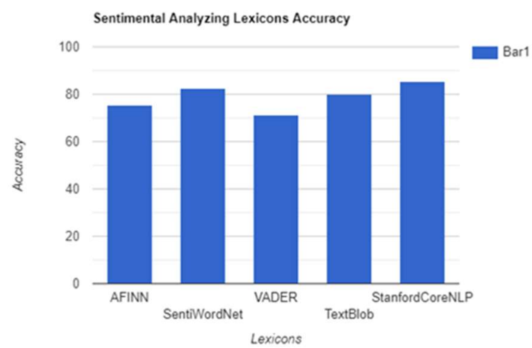

Fig: Accuracy of Lexicons

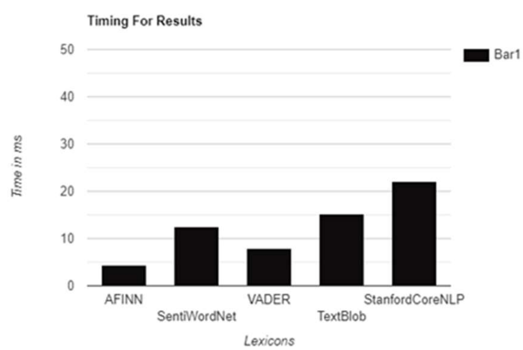

Fig: Time Taken By The Lexicons

The calculation of the sentimental score involved summing up the sentiment values of the words found in the AFINN list, dividing the result by the number of tokens. The score obtained was 18 with a comparative score of 0.45.

The analysis revealed that all the words found in the input string were positive, with none being negative. The positive words included 'favourite', 'love', 'fun', 'excited', 'nice', and 'good'. The emotional status of the user was then evaluated based on the sentimental score. A score higher than 8 indicated a contented user, a score between 6 to 8 indicated a moderately distressed user, and a score less than 6 indicated a very distressed user.

Our study shows that the use of AFINN lexicon for real-time sentiment analysis of chat data can provide valuable insights into user sentiment. The results of the analysis revealed the emotional status of the user in this particular case to be contented.

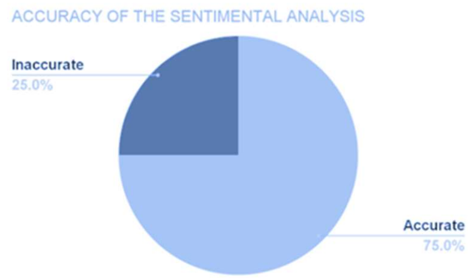

Fig 2: Depiction of the accuracy of the Sentimental Analysis

5. Conclusion and Future Work

In conclusion, this research has demonstrated the effectiveness of using the AFINN lexicon in real-time sentiment analysis of chat data. The lexicon is a useful tool for identifying the emotional tone of writing since it can categorise content into positive, negative, or neutral categories and give the mood a numerical score. Our findings have shown that the AFINN lexicon can provide valuable insights into user sentiment in chat applications, but the limitations of idiomatic expressions and sarcasm need to be taken into consideration. Further studies can investigate alternative lexicons or techniques to enhance the performance of sentiment analysis in chat data.

Additionally, the addition of features aimed at improving the mental wellbeing of users has been shown to have a positive impact. The chat bot, which utilizes sentiment analysis to determine the user's primary mood, provides resources to enhance mental health. To improve the user experience in the future, new features will be introduced, such as immediate access to therapists and medical specialists, and the sentiment analysis will be improved to better understand contexts, jokes, and double negatives.

References

1. Cambria, E. Affective computing, and sentiment analysis. IEEE Intell. Syst. 2016, 31, 102–107.
2. Nazir F, Ghazanfar MA, Maqsood M, et al. (2019) Social media signal detection using tweets volume, hashtag, and sentiment analysis. Multimedia Tools and Applications 78(3): 3553–3586
3. Heaton, J., Polson, N., & Witte, J. H. (2017). Deep learning for finance: deep portfolios. Applied Stochastic Models in Business and Industry, 33(1), 3–12.
4. Bouazizi M and Ohtsuki T (2017) A pattern-based approach for multi-class sentiment analysis in Twitter. IEEE Access 5: 20617–20639.
5. C. Li, H. Liu, Association analysis and N-Gram based detection of incorrect arguments, Journal of Software, 2018, 29(8), pp.1-15.
6. Swanand Joshi, Amey Ruikar, 2014, Sentiment Analysis of Chat Application, INTERNATIONAL JOURNAL OF ENGINEERING RESEARCH & TECHNOLOGY (IJERT) Volume 03, Issue 08 (2014).

7. H. Saif, Y.He, and H.Alani, “Semantic Sentiment Analysis of Twitter,” Proceeding of the Workshop on Information Extraction and Entity Analytics on Social Media Data. United Kingdom: Knowledge Media Institute, 2011.
8. Chauhan, P., Sharma, N. & Sikka, G. The emergence of social media data and sentiment analysis in election prediction. J Ambient Intell Human Computer 12, 2601–2627 (2021)
9. Kaur, H., Ahsaan, S.U., Alankar, B. et al. A Proposed Sentiment Analysis Deep Learning Algorithm for Analysing COVID-19 Tweets. Inf Syst Front 23, 1417–1429 (2021).
10. Shilpa & Shereen, Rissa & Jacob, Susmi & Vinod, P. (2021). Sentiment Analysis Using Deep Learning. 930-937.
11. J. Spencer and G. Uchyigit, “Sentiment or: Sentiment Analysis of Twitter Data,” Second Joint Conference on Lexicon and Computational Semantics. Brighton: University of Brighton, 2008.
12. Agrawal, A., & An, A. (2012). Unsupervised Emotion Detection from Text Using Semantic and Syntactic Relations. 2012 IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology, 1, 346-353.
13. Sohangir, S., Wang, D., Pomeranets, A. et al. Big Data: Deep Learning for financial sentiment analysis. J Big Data 5, 3 (2018).
14. Daniel Haryadi and Gede Putra Kusuma, “Emotion Detection in Text using Nested Long Short-Term Memory” International Journal of Advanced Computer Science and Applications (IJACSA), 10(6), 2019.
15. Najafabadi NM, Villanustre F, Khoshgoftaar TM, Seliya N, Wald R, Muharemagic E. Deep learning applications and challenges in big data analytics. J Big Data. 2015; 2:1.
16. H. Cho et al., Data-driven integration of multiple sentiment dictionaries for lexicon-based sentiment classification of product reviews, Knowl. Based Syst. (2014)
17. E. Cambria, D. Olsher, D. Rajagopal, SenticNet 3: A common and common-sense knowledge base for cognition-driven sentiment analysis, in: AAAI, Quebec City, 2014, pp. 1515–1521.
18. T. Wilson, J. Wiebe, P. Hoffmann, Recognizing contextual polarity in phrase-level sentiment analysis, in: HLT/EMNLP, Vancouver, 2005, pp. 347–354.
19. Groh, G., & Hauffa, J. (2021). Characterizing Social Relations Via NLP-Based Sentiment Analysis. Proceedings of the International AAAI Conference on Web and Social Media, 5(1), 502-505.
20. Xia, R., & Ding, Z. Emotion-Cause Pair Extraction: A New Task to Emotion Analysis in Texts. (2019) In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pages 1003–1012, Florence, Italy. 1003–1012, ArXiv.
21. S. Kravchenko, X. Zhu, C. Cherry, and S. Mohammad, "NRC Canada-2014: Detecting Aspects and Sentiment in Customer Reviews," Proc. 8th Int’l Workshop on Semantic Evaluation, pp. 437-442, 2014.
22. S. Zhang, Z. Wei, Y. Wang, T. Liao, Sentiment analysis of Chinesemicro-blog text based on extended sentiment dictionary, Future Generation Computer Systems (2017).
23. Apoorv Agarwal, Boyi Xie, Ilia Vovsha, Owen Rambow, and Rebecca Passonneau. 2011. Sentiment Analysis of Twitter Data. In Proceedings of the Workshop on Language in Social Media (LSM 2011), pages 30–38, Portland, Oregon. Association for Computational Linguistics. pp. 465–468, Springer, 2017

24. S. Niklander and G. Niklander, “Combining sentimental and content analysis for recognizing and interpreting human affects, “in International Conference on Human-Computer Interaction, pp. 465–468, Springer, 2017.
25. Kouloumpis, E., Wilson, T., & Moore, J. (2011). Twitter sentiment analysis: The good the bad and the omg! In Proceedings of the international AAAI conference on web and social media (Vol. 5, No. 1, pp. 538-541).
26. Pak, A., & Paroubek, P. (2010, May). Twitter as a corpus for sentiment analysis and opinion mining. In LREc (Vol. 10, No. 2010, pp. 1320-1326).
27. S. Poria, E. Cambria, and A. Gelbukh, "Aspect Extraction for Opinion Mining with A Deep Convolutional Neural Network," Knowledge-Based Systems, vol. 108, pp. 42-49, 2016.

28. Csaky, Richard., “Deep Learning Based Chatbot Models”, Scientific Students ’Association Report, Department of Automation, and Applied Informatics, Budapest University Of Technology and Economics, (2017).

29. S. Manandhar, "SemEval-2014 Task 4: Aspect Based Sentiment Analysis," Proc. 8th Int’l Workshop on Semantic Evaluation, pp. 19-30, 2014.
30. Xing, F. Z., Cambria, E., & Welsch, R. E. (2018). Natural language based financial forecasting: a survey. Artificial Intelligence Review, 50(1), 49–73.
31. L. Lee and B. Pang, "Opinion Mining and Sentiment Analysis, “Foundations and Trends® in Information Retrieval, vol. 2, no. 1–2, pp.1-135, 2008.
32. H. Saif, Y. He and H. Alani, “Alleviating Data Scarcity for Twitter Sentiment Analysis”. Association for Computational Linguistics, 2012.
33. M. Cliche, “Bb twtr at semeval-2017 task 4: Twitter sentiment analysis with cnns and lstms,” arXiv preprint arXiv:1704.06125, 2017
34. M. Ghiassi, J. Skinner, and D. Zimbra, "Twitter brand sentiment analysis: A hybrid system using n-gram analysis and dynamic artificial neural network," Expert Systems with applications, vol. 40, no. 16, pp. 6266-6282, 2013.
35. C. Baziotis, N. Pelekis, and C. Doulkeridis, “Data stories at semeval-2017 task 4: Deep lstm with attention for message-leveland topic-based sentiment analysis,” in Proceedings of the 11thinternational workshop on semantic evaluation (SemEval-2017), pp. 747–754, 2017.
36. Pang, Bo, and Lillian Lee. "A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts." In Proceedings of the 42nd annual meeting on Association for Computational Linguistics, p. 271. Association for Computational Linguistics, 2004.
37. P. Yadav and D. Pandya. "SentiReview: Sentiment analysis based on text and emoticons." In 2017 International Conference on Innovative Mechanisms for Industry Applications (ICIMIA), pp. 467-472. IEEE,2017.
38. F.A. Nielsen. "Afinn." Richard Petersens Plads, Building 321, 2011.
39. B.V. Krishna, A.K. Pandey, and A.P.S. Kumar. "Feature based opinion mining and sentiment analysis using fuzzy logic." In Cognitive science and artificial intelligence, pp. 79-89. Springer, Singapore, 2018.
40. Sareeta Mugde, Garima Sharma, Aditya Singh Kashyap, Swastika Swastik, "Sentiment Classification and Comparison of Covid-19 Tweets During the First Wave and the Second

Wave Using NLP Techniques and Libraries", Innovations in Bio-Inspired Computing and Applications, vol.419, pp.685, 2022.
41. A. Blom and S. Thorsen, "Automatic Twitter replies with Python," International conference "Dialog 2012".
42. B. Pang, and L. Lee, "Opinion mining and sentiment analysis," 2nd workshop on making sense of Microposts. Ithaca: Cornell University. Vol.2(1), 2008.
43. A. Agarwal, B. Xie, I. Vovsha, O. Rambow, and R.Passonneau, "Sentiment Analysis of Twitter Data," Annual International Conferences. New York: Columbia University, 2012
44. S. Sharma, "Application of Support Vector Machines for Damage detection in Structure," Journal of Machine Learning Research, 2008.
45. M. Hu, and B. Liu, "Mining and summarizing customer reviews," 2004.
46. H. Saif, Y. He and H. Alani, "Alleviating Data Scarcity for Twitter Sentiment Analysis". Association for Computational Linguistics, 2012
47. A. Pak, and P. Paroubek, "Twitter as a Corpus for Sentiment Analysis and Opinion Mining," Special Issue of International Journal of Computer Application, France: Universitede Paris-Sud, 2010.