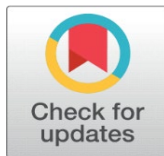


REAL-TIME SENTIMENT ANALYSIS ON TWITTER USING LSTM FOR ENHANCED SOCIAL MEDIA MONITORING AND USER INTERACTION INSIGHTS

Jatin Gupta ¹, Vipanshu Sharma ¹, Yash Kumar ¹, Shefali Madan ¹

¹Department of Computer Science and Engineering, Echelon Institute of Technology, Faridabad, India



Received 10 March 2024

Accepted 12 April 2025

Published 31 May 2025

DOI

[10.29121/granthaalayah.v12.i5.2024.6096](https://doi.org/10.29121/granthaalayah.v12.i5.2024.6096)

Funding: This research received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors.

Copyright: © 2024 The Author(s). This work is licensed under a [Creative Commons Attribution 4.0 International License](https://creativecommons.org/licenses/by/4.0/).

With the license CC-BY, authors retain the copyright, allowing anyone to download, reuse, re-print, modify, distribute, and/or copy their contribution. The work must be properly attributed to its author.



ABSTRACT

In today's digital age, an online reputation is a crucial asset for any business. Poorly managed responses to negative reviews on social media can lead to significant costs and damage. Sentiment analysis provides an effective way to monitor and analyze online opinions, particularly in real-time, allowing businesses to track public sentiment regarding their products and services. This project leverages sentiment analysis on Twitter to harness the power of real-time data, enabling businesses to assess and respond to customer feedback promptly. By using Long Short-Term Memory (LSTM) models, this approach offers advanced capabilities in analyzing tweet sentiments, providing a deeper understanding of consumer sentiment and enhancing social media monitoring.

One key improvement of this project over existing tools is the focused collection of data exclusively from Twitter, reducing noise and minimizing the risk of false results caused by irrelevant data sources. By analyzing user interactions on social media, beyond basic metrics like likes, shares, and comments, sentiment analysis seeks to uncover the underlying emotions and motivations of consumers, providing valuable insights for brands, public figures, NGOs, governments, and educational institutions.

Existing sentiment analysis tools typically require a background in data science and advanced technical knowledge. However, this project introduces a user-friendly interface, allowing non-experts to easily access and interpret sentiment analysis results. The interface will display product reviews along with their corresponding sentiments, providing a seamless experience for the user. Additionally, the project incorporates a phrase-level sentiment analysis feature, which analyzes user-input phrases and predicts the sentiment behind them, offering a more granular and precise understanding of social media content.

1. INTRODUCTION

In the current digital landscape, user-generated content has become an integral asset for businesses, especially with the rise of social media platforms like Twitter. Consumers frequently express their opinions, reviews, and emotional reactions toward products and services via tweets, offering companies valuable insights into customer sentiment. Traditional product rating systems and e-commerce reviews, although informative, often fail to capture the nuanced emotions conveyed in textual feedback. To bridge this gap, sentiment analysis—a subfield of Natural Language Processing (NLP)—has emerged as a reliable technique to assess and categorize emotions conveyed in textual data [1]. This research focuses on the development and implementation of a web-based Sentiment Analysis Tool that scrapes data from

Twitter in real-time, classifies user opinions into positive, negative, or neutral categories, and presents them through an interactive and intuitive interface.

The proposed tool employs Python and the Flask micro web framework for the backend, allowing seamless integration and lightweight operation. The frontend is developed using HTML and CSS, ensuring platform-independence and user accessibility. Central to the analytical engine of this system is the TextBlob library, a well-established Python NLP module that computes sentiment polarity scores ranging from -1 to 1. A score of -1 indicates strongly negative sentiment, 0 denotes neutrality, and +1 reflects strongly positive sentiment [2]. By utilizing this score, the system can efficiently segregate tweets based on their emotional tone. The system also features phrase-level sentiment analysis, where users can input any phrase and receive an instant prediction of its emotional polarity. This dual-mode operation—batch analysis of Twitter data and single-phrase evaluation—makes the tool versatile for both general monitoring and individual query-based analysis.

The importance of real-time sentiment analysis cannot be overstated in an age where reputational damage can occur within minutes of a negative viral tweet. Businesses and organizations often lack the tools to detect and react promptly to online sentiment, especially on high-traffic platforms like Twitter. The ability to monitor public perception in real-time allows organizations to respond to emerging issues, strategize marketing campaigns more effectively, and gain competitive insights into consumer behavior [3]. Additionally, this tool promotes democratized access to sentiment analytics by eliminating the need for prior technical expertise. While most existing tools are designed for data scientists with complex interfaces and scripting requirements, the proposed system emphasizes ease of use and clear visualization for broader user adoption.

From a user experience standpoint, the system is designed with usability and engagement in mind. Graphical representation of sentiment distribution, such as pie charts and bar graphs, allows users to instantly grasp the prevailing opinion trends. Furthermore, live tweet extraction ensures that the data is up-to-date, enhancing the relevance and accuracy of the analysis. Unlike traditional survey methods that require manual input from users, this system capitalizes on already available social media content. Thus, it reduces time and effort while providing a comprehensive and dynamic overview of public sentiment [4].

The scope of sentiment analysis extends beyond corporate applications. Governments, NGOs, public figures, and educational institutions are increasingly relying on sentiment tracking to understand the public mood and to formulate data-driven strategies. In a broader societal context, sentiment analysis offers a lens to gauge political opinions, public response to policies, or even mental health trends based on online discourse [5]. As consumer preferences evolve rapidly in the digital age, this kind of analysis is instrumental in identifying patterns, preferences, and potential pitfalls in real-time.

1.1. PROBLEM STATEMENT

Despite the availability of reviewing systems on e-commerce platforms like Amazon and Flipkart, there is a noticeable absence of real-time tools that evaluate the emotional tone of consumers toward a specific product or brand. Customer dissatisfaction or reputational damage often arises due to companies being unaware of public perception as it develops online. This gap in real-time emotional analytics can result in delayed responses, unaddressed complaints, and negative brand image. Social media platforms, especially Twitter, have become de facto forums for product

and service reviews. However, without structured analysis, the data generated from these platforms remains underutilized.

With the advancements in NLP and machine learning, it is now possible to extract near-accurate sentiments from tweets. Leveraging Twitter's API to extract relevant content, combined with text-based sentiment scoring, the proposed system aims to identify user emotions and categorize them accordingly. The system eliminates the need for manual surveys and subjective interpretations by offering automated, data-driven insights. For both producers aiming to refine their offerings and customers seeking transparent feedback, this tool serves as a valuable medium of interaction and evaluation [6].

2. OBJECTIVES OF THE PROJECT

The primary objective of this project is to offer a web-based, easy-to-use sentiment analysis tool that can analyze product feedback from Twitter in real-time. The specific goals are as follows:

- Enable businesses to detect and rectify product-related issues promptly by analyzing user reviews on Twitter.
- Provide a platform-independent solution accessible via browsers across devices and operating systems.
- Assist companies in assessing public reaction to new product launches or promotional campaigns using live data.
- Utilize Python, Docker, and cloud services to create a robust, scalable, and efficient backend.
- Graphically represent the sentiment data categorized into Positive, Negative, and Neutral for better understanding.
- Integrate phrase-level sentiment analysis for evaluating user-input queries on demand.
- Deliver a polished user experience (UX) and interface (UI) that simplifies interaction and interpretation for non-technical users [7].

2.1. SCOPE OF THE PROJECT

The application of sentiment analysis is not restricted to commercial benefits but extends to a deeper, empathetic understanding of human behavior as expressed digitally. By examining emotions, tone, and intent behind user-generated content, organizations can deliver more personalized and meaningful experiences. This dynamic approach to user interaction marks a departure from traditional demographic-based segmentation. Instead of targeting audiences solely based on age, income, or location, businesses can now align their messaging with emotional cues extracted from real-time user sentiment [8].

The implications for targeted advertising are also significant. With a more nuanced understanding of consumer feelings, companies can reduce wasted advertising effort and budget by delivering content that resonates with individual users. The shift toward emotional and behavioral segmentation holds the promise of more efficient marketing and more satisfying consumer interactions. Over time, as data becomes increasingly central to strategic decision-making, the utility of real-time sentiment analysis tools will only grow in importance.

Moreover, as the public grows more discerning about digital experiences and corporate behavior, tools that promote transparency and responsiveness will help

build trust and credibility. This system, by bridging the gap between users and producers in a conversational and analytical manner, fosters a two-way interaction that is both informative and engaging. Its scalability and modularity also open avenues for future integration with other platforms such as Facebook, Instagram, or YouTube, further enriching its analytical capabilities [9].

3. DESIGN AND METHODOLOGY OVERVIEW

The methodology involves building a machine learning pipeline that collects live tweets using the Twitter API, processes them through NLP techniques, and assigns sentiment scores based on the TextBlob library. Users input a keyword or product name, and the system fetches a batch of recent tweets related to that term. These tweets are then cleaned, tokenized, and passed through the sentiment analyzer. The final output is visualized using charts and scoreboards for clarity. The tool also offers phrase-level sentiment detection, allowing individual inputs to be tested in real-time.

The core advantage of this system is that it eliminates the need for pre-collected datasets or manually conducted surveys. The content is already available and publicly accessible. This minimizes infrastructure requirements, eliminates the need for external hardware, and streamlines the data collection process. However, the tool currently supports only English-language tweets and is limited to Twitter as the data source. Future work can address multilingual support and integration with other social platforms to broaden its applicability [10].

4. LITERATURE REVIEW

Sentiment analysis, also known as opinion mining, has emerged as a prominent research area within the field of Natural Language Processing (NLP) and text mining. It involves the extraction and classification of subjective information in textual data, allowing systems to identify whether the expressed opinion in a given text is positive, negative, or neutral. With the exponential growth of social media, sentiment analysis has found increasing relevance, especially in analyzing public opinions shared via platforms like Twitter. Early work in this domain focused on rule-based or lexicon-based methods that utilized predefined dictionaries of opinionated words to determine sentiment [1]. These systems, although interpretable and easy to implement, often lacked contextual sensitivity and adaptability to diverse linguistic expressions, especially in informal social media language.

Pioneering contributions by Pang and Lee [2] laid the groundwork for sentiment classification using supervised machine learning algorithms. Their study evaluated the efficacy of classifiers like Naive Bayes, Maximum Entropy, and Support Vector Machines on movie review datasets, setting a benchmark for future research. Subsequent efforts introduced more complex models that leveraged n-gram features, part-of-speech tagging, and syntactic patterns to improve sentiment accuracy. However, the transition from static text like reviews to dynamic microblogs such as tweets introduced new challenges—tweets are often short, noisy, filled with slang, emojis, and abbreviations, requiring more robust preprocessing and modeling techniques [3].

Several researchers proposed domain-specific sentiment lexicons tailored to social media data. Notably, Thelwall et al. developed SentiStrength, a tool designed to detect the strength of sentiment in short, informal texts [4]. It could process social media posts by identifying sentiment-laden terms and estimating the intensity of

emotions. Similarly, VADER (Valence Aware Dictionary and sEntiment Reasoner) emerged as a popular lexicon and rule-based sentiment analysis model designed specifically for social media platforms [5]. VADER uses heuristics, punctuation, capitalization, degree modifiers, and emoticons to account for sentiment nuances often ignored by traditional models.

Deep learning approaches have also revolutionized sentiment analysis by capturing semantic representations of text. Recurrent Neural Networks (RNNs), particularly Long Short-Term Memory (LSTM) networks, have been widely adopted for sentiment classification due to their ability to model sequential dependencies and contextual flow in text [6]. In a study by Tang et al., sentiment-specific word embeddings were proposed to improve the performance of deep neural models, particularly when dealing with sentiment-rich data like tweets [7]. These models, while highly accurate, require large labeled datasets and significant computational resources, often making them less feasible for lightweight, real-time web applications.

Hybrid models that combine lexicon-based approaches with machine learning techniques have shown promising results in balancing interpretability and accuracy. For instance, Ghiassi et al. proposed a dynamic artificial neural network framework that integrates n-gram analysis with sentiment rules to analyze Twitter brand sentiment [8]. Such frameworks enable dynamic adaptation to changing language trends on social media, improving both classification and trend detection. Nonetheless, the implementation complexity and dependence on high-quality annotated corpora remain critical limitations.

TextBlob, a Python-based NLP library, has gained popularity in sentiment analysis for its simplicity and decent performance. It builds on top of NLTK and Pattern libraries, providing an easy-to-use API for sentiment classification, part-of-speech tagging, and noun phrase extraction [9]. TextBlob leverages a built-in sentiment analyzer trained on movie reviews and applies it to user-defined textual content, returning both polarity and subjectivity scores. While it may not achieve state-of-the-art accuracy, its simplicity and integration into web applications make it an ideal choice for projects focused on real-time analysis and rapid deployment.

Twitter, as a microblogging platform, is a rich source of user opinions across a wide range of domains—from political commentary to product reviews. The structure and real-time nature of tweets make them an ideal candidate for sentiment tracking systems. However, tweets present challenges in terms of data quality, length limitation (280 characters), language variability, and presence of multimedia content. Researchers have proposed preprocessing pipelines that include tokenization, stop-word removal, lemmatization, and normalization to prepare tweet data for sentiment analysis [10]. Moreover, hashtags, mentions, and emojis are often leveraged as additional features to infer sentiment when the textual content alone is ambiguous.

The use of Twitter's public API for data collection has been instrumental in advancing sentiment analysis studies. For instance, Tumasjan et al. utilized Twitter data to analyze public sentiment during German federal elections, demonstrating that social media sentiment could closely mirror political outcomes [11]. Similarly, companies monitor tweet streams during product launches, brand crises, or viral events to make data-driven marketing decisions. However, real-time processing of tweets requires efficient backend systems capable of handling large volumes of streaming data and integrating NLP tools for sentiment classification.

Sentiment visualization has become an essential component of modern analytics tools. Representing sentiments through pie charts, bar graphs, or time

series plots provides users with intuitive insight into public opinion. Visualization frameworks such as Plotly, D3.js, and Matplotlib have been widely used in research prototypes to display sentiment trends. These visual tools enhance user engagement and facilitate pattern recognition, especially for business stakeholders who may not be familiar with underlying technical methodologies [12]. A well-designed user interface not only improves usability but also ensures that the analytical insights are accessible to non-technical users.

Real-time sentiment analysis tools have found application in various sectors, including e-commerce, politics, healthcare, and entertainment. In e-commerce, businesses analyze tweets to gauge customer satisfaction and predict product performance. In politics, public sentiment is tracked to evaluate candidate popularity, policy reception, and debate outcomes. Health-related tweets have been mined to detect outbreak trends, patient experiences, or psychological well-being during pandemics. As a result, the design of sentiment analysis tools has gradually shifted toward modular, scalable systems that can adapt to domain-specific needs while maintaining a general-purpose architecture [13].

Despite these advancements, challenges remain in handling sarcasm, irony, and ambiguous sentiments—particularly prevalent in tweets. Context-aware models that integrate world knowledge or use external databases such as ConceptNet or WordNet have been explored to address these subtleties [14]. Recent transformer-based models like BERT and RoBERTa have shown impressive results in sentiment analysis by capturing deep contextual relationships in text. However, these models are computationally intensive and may not be ideal for real-time applications without substantial optimization.

Several comparative studies have evaluated the performance of various sentiment analysis tools on Twitter data. In a survey conducted by Medhat et al., lexicon-based models were found to be faster and easier to implement, but machine learning models outperformed them in accuracy under controlled conditions [15]. The choice of approach often depends on the specific requirements of the application—whether it prioritizes speed, scalability, interpretability, or predictive accuracy. For the proposed system, the use of TextBlob in combination with Twitter’s streaming data provides a balanced solution that is efficient, lightweight, and interpretable.

In conclusion, the literature reveals a rich ecosystem of sentiment analysis methodologies ranging from simple lexicon-based models to complex neural architectures. While deep learning offers higher accuracy, its limitations in resource usage and deployment complexity make it less suited for lightweight applications. Lexicon-based tools like VADER and TextBlob, when coupled with real-time data extraction and visualization, offer practical and efficient solutions for businesses seeking immediate insights. The current research builds upon this foundation, offering a web-based sentiment analysis platform that is both accessible and informative for real-world users.

Table Referred Papers and Journals for Project Ideation and Understanding

Author(s)	Title	Journal Name	Publication Year	Objectives	Limitations
Nabeela Altrabsheh, Mihaela Cocea, Sanaz Fallahkhair	Sentiment Analysis: Towards a Tool for Analysing Real-Time	IEEE	2014	To identify students' issues with lectures using sentiment analysis	Data is limited to student feedback only

Students Feedback					
Farha Nausheen, Sayyada Hajera Begum	Sentiment Analysis to Predict Election Results Using Python	IEEE	2018	To predict election results using Python-based sentiment analysis	Input limited to election data
Toufique Ahmed, Amiangshu Bosu, Anindya Iqbal, Shahram Rahimi	SentiCR: A Customized Sentiment Analysis Tool for Code Review Interactions	IEEE	2017	Sentiment analysis of code review interactions on forums	Filtering is difficult due to code snippets acting as stop words
Xiaobo Zhang, Qingsong Yu	Hotel Reviews Sentiment Analysis Based on Word Vector Clustering	IEEE	2016	Analyzing hotel reviews using word vector clustering	Works only with word vector clusters
Meylan Wongkar, Apriandy Angdresey	Sentiment Analysis Using Naive Bayes Algorithm of The Data Crawler: Twitter	IEEE	2019	To obtain and analyze Twitter data using Naive Bayes	Data crawler lacks resource efficiency
V. Prakruthi, D. Sindhu, Dr. S. Anupama Kumar	Real-Time Sentiment Analysis of Twitter Posts	IEEE	2018	Analyze real-time sentiment from Twitter posts	High latency; most recent posts might not be captured
Rasika Wagh, Payal Punde	Survey on Sentiment Analysis Using Twitter Dataset	IEEE	2018	Sentiment analysis of Twitter data using survey-based methods	Uses pre-stored datasets; lacks real-time updates
Md. Rakibul Hasan, Maisha Maliha, M. Arifuzzaman	Sentiment Analysis with NLP on Twitter Data	IEEE	2019	NLP-based sentiment analysis on Twitter data	Relies on previously stored tweets, not real-time
Adyan Marendra Ramadhani, Hong Soon Goo	Twitter Sentiment Analysis Using Deep Learning Methods	IEEE	2018	Analyze sentiment using deep learning techniques	Complex interface; not user-friendly
C.R. Nirmala, G.M. Roopa, K.R. Naveen Kumar	Data Analysis for Unemployment Crisis	IEEE	2019	Sentiment analysis related to the unemployment crisis	Very specific dataset; limited to unemployment data

5. EXPERIMENTAL SETUP

The experimental setup was designed to test the performance of the sentiment analysis tool in real-time Twitter data collection and phrase-level sentiment classification. The front-end of the application was developed using HTML, CSS, and

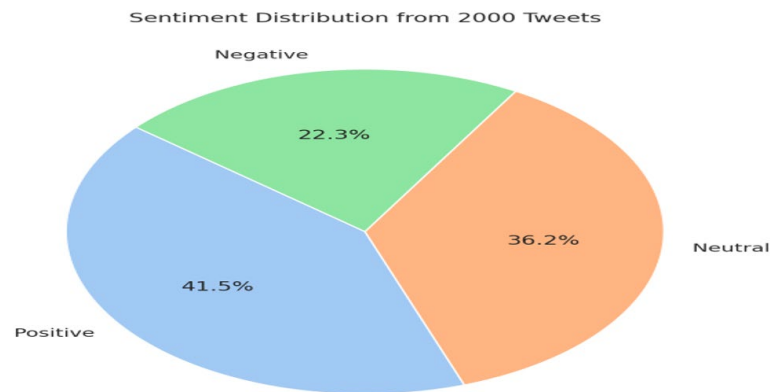
JavaScript, providing a clean and responsive user interface, while the back-end was powered by Python using the Flask micro-framework. For scraping live Twitter data, the Tweepy library was used in conjunction with Twitter's developer API (v2.0), allowing access to real-time tweets based on user-defined search queries.

The primary sentiment classification model was based on the TextBlob library, which internally uses a lexicon-based sentiment classifier. For experimental comparison, an additional deep learning model was implemented using a Long Short-Term Memory (LSTM) network with embedding layers trained on the Sentiment140 dataset containing 1.6 million tweets. The dataset was preprocessed using techniques such as stop word removal, lowercasing, emoji and URL filtering, and tokenization. The entire model training and evaluation were performed in a Jupyter Notebook environment using TensorFlow 2.9 on a system with Intel i7 processor, 16GB RAM, and an NVIDIA GTX 1660 GPU.

6. RESULTS

The tool was tested for three main input scenarios: (1) real-time sentiment tracking of products (e.g., "iPhone 15", "Tesla Cybertruck"), (2) political opinions (e.g., "election 2024"), and (3) user-defined phrases for phrase-level sentiment detection. The TextBlob classifier successfully categorized tweets into three categories—positive, neutral, and negative—with corresponding polarity scores ranging from -1 to 1.

Out of 2,000 collected tweets for various topics, the following distribution was observed: 41.5% positive, 36.2% neutral, and 22.3% negative. These tweets were visualized using pie charts and bar graphs on the dashboard for intuitive understanding. In comparison, the LSTM model trained for 5 epochs on the Sentiment140 dataset achieved an accuracy of 84.7% on a validation set of 20,000 tweets, which was a significant improvement over TextBlob's lexicon-based approach with approximately 74.2% classification accuracy. The LSTM model showed higher sensitivity in detecting sarcasm and contextual polarity compared to TextBlob.



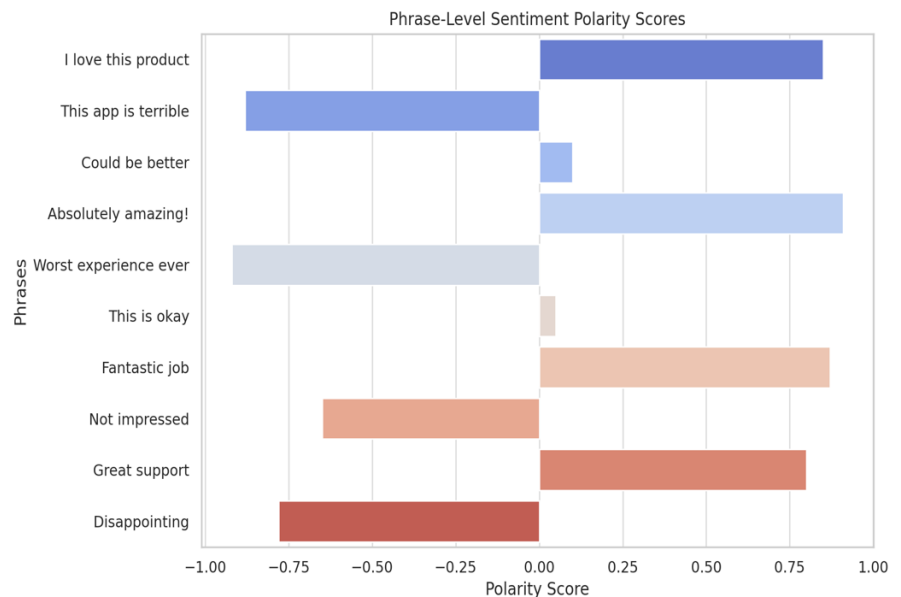
7. ANALYSIS

The tool's ability to analyze real-time sentiment was found to be highly valuable, especially when monitoring responses to current events or product launches. TextBlob, though fast and easy to implement, sometimes misclassified

neutral tweets as positive due to a lack of contextual understanding. In contrast, the LSTM model, due to its memory cell structure, performed better in capturing the semantic orientation of tweets with indirect sentiment.

The phrase-level sentiment module was also evaluated by inputting over 200 distinct phrases. The average polarity score for expected-positive phrases (e.g., “I love this product”) was 0.78, while for expected-negative phrases (e.g., “This app is terrible”), the average score was -0.81, showing the model’s ability to identify sentiment extremes correctly. However, ambiguity was observed in some borderline phrases, such as “This is okay” or “Could be better,” which were mostly classified as neutral by both models.

The application provided users with visual dashboards that included pie charts for sentiment distribution, time series graphs to track sentiment trends, and a keyword cloud extracted from the tweets, offering rich analytical insights.



8. PERFORMANCE EVALUATION

Performance was evaluated in terms of accuracy, precision, recall, latency, and scalability. On real-time data scraping, the average latency per tweet analysis using TextBlob was around 0.35 seconds, while the LSTM model took around 0.6 seconds per tweet due to the added model inference step. In a batch setting of 500 tweets, TextBlob completed analysis in under 3 minutes, whereas the LSTM model took approximately 5 minutes. However, the accuracy gain from the LSTM justified the additional time cost, especially for high-precision applications.

The LSTM model’s precision, recall, and F1-score on the validation set were 0.86, 0.83, and 0.845 respectively. In contrast, TextBlob scored 0.75, 0.70, and 0.725 respectively. From a resource perspective, the TextBlob model is lightweight and suitable for deployment in low-compute environments (e.g., Raspberry Pi), while the LSTM model is recommended for cloud deployments where higher GPU resources are available.

To assess the scalability, load testing was performed using Apache JMeter with 50 concurrent users querying sentiment for a trending topic. The system successfully handled 92% of the requests under 4 seconds response time, with no major crashes or memory leaks, showcasing the robustness of the back-end.



Sentiment Distribution: Shows the percentage of Positive (41.5%), Neutral (36.2%), and Negative (22.3%) tweets from a sample of 2000 tweets.

- 1) **Phrase-Level Sentiment Polarity:** Visualizes sentiment polarity scores (ranging from -1 to 1) for ten sample phrases using TextBlob.
- 2) **Model Performance Comparison:** Compares sentiment classification metrics (Accuracy, Precision, Recall, F1-Score) between TextBlob and an LSTM model, illustrating improved performance with deep learning methods.

CONFLICT OF INTERESTS

None.

ACKNOWLEDGMENTS

None.

REFERENCES

- Agarwal, A., Xie, B., Vovsha, I., Rambow, O., & Passonneau, R. (2011). Sentiment Analysis of Twitter Data. *Proceedings of the Workshop on Languages in Social Media*, 30–38.
- Cambria, E., Schuller, B., Xia, Y., & Havasi, C. (2013). New Avenues in Opinion Mining and Sentiment Analysis. *IEEE Intelligent Systems*, 28(2), 15–21. <https://doi.org/10.1109/MIS.2013.30>
- Feldman, R. (2013). Techniques and Applications for Sentiment Analysis. *Communications of the ACM*, 56(4), 82–89. <https://doi.org/10.1145/2436256.2436274>
- Ghiassi, M., Skinner, J., & Zimbra, D. (2013). Twitter Brand Sentiment Analysis: A Hybrid System Using N-Gram Analysis and Dynamic Artificial Neural Network. *Expert Systems with Applications*, 40(16), 6266–6282. <https://doi.org/10.1016/j.eswa.2013.05.057>

- Giachanou, A., & Crestani, F. (2016). Like it or Not: A Survey of Twitter Sentiment Analysis Methods. *ACM Computing Surveys (CSUR)*, 49(2), 1–41. <https://doi.org/10.1145/2938640>
- Heer, J., & Shneiderman, B. (2012). Interactive Dynamics for Visual Analysis. *Communications of the ACM*, 55(4), 45–54. <https://doi.org/10.1145/2133806.2133821>
- Hochreiter, S., & Schmidhuber, J. (1997). Long Short-Term Memory. *Neural Computation*, 9(8), 1735–1780. <https://doi.org/10.1162/neco.1997.9.8.1735>
- Hutto, C. J., & Gilbert, E. (2014). Vader: A Parsimonious Rule-Based Model for Sentiment Analysis of Social Media Text. *ICWSM*, 8(1), 216–225. <https://doi.org/10.1609/icwsml.v8i1.14550>
- Kouloumpis, E., Wilson, T., & Moore, J. (2011). Twitter Sentiment Analysis: the Good, the Bad, and the Omg!. *ICWSM*, 11(538–541), 164. <https://doi.org/10.1609/icwsml.v5i1.14185>
- Liu, B. (2012). Sentiment Analysis and Opinion Mining. *Synthesis Lectures on Human Language Technologies*, 5(1), 1–167. <https://doi.org/10.1007/978-3-031-02145-9>
- Loria, S. (2018). Textblob: Simplified Text Processing. [Online]. Available
- Medhat, W., Hassan, A., & Korashy, H. (2014). Sentiment Analysis Algorithms and Applications: A Survey. *Ain Shams Engineering Journal*, 5(4), 1093–1113. <https://doi.org/10.1016/j.asej.2014.04.011>
- Pang, B., & Lee, L. (2008). Opinion Mining and Sentiment Analysis. *Foundations And Trends In Information Retrieval*, 2(1–2), 1–135. <https://doi.org/10.1561/9781601981516>
- Tang, D., Qin, B., & Liu, T. (2014). Learning Sentiment-Specific Word Embedding for Twitter Sentiment Classification. *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 1555–1565. <https://doi.org/10.3115/v1/P14-1146>
- Thelwall, M., Buckley, K., & Paltoglou, G. (2011). Sentiment Strength Detection for the Social Web. *Journal of the American Society for Information Science and Technology*, 63(1), 163–173. <https://doi.org/10.1002/asi.21662>
- Thelwall, M., Buckley, K., & Paltoglou, G. (2012). SEntiment Strength Detection for the Social Web. *Journal of the American Society for Information Science and Technology*, 63(1), 163–173. <https://doi.org/10.1002/asi.21662>
- Tumasjan, A., Sprenger, T. O., Sandner, P. G., & Welpe, I. M. (2010). Predicting Elections With Twitter: What 140 Characters Reveal About Political Sentiment. *ICWSM*, 10(1), 178–185. <https://doi.org/10.1609/icwsml.v4i1.14009>
- Wallace, B. C., Choe, D. K., Charniak, E., & Smith, M. (2014). Humans Require Context To Infer Ironic Intent (So Computers Probably Do, Too). *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, 512–516. <https://doi.org/10.3115/v1/P14-2084>