Text-Based Sentiment Analysis Using Natural Language Processing

Dibie Chinonso Emmanuel
Department of Computer Science,
Faculty of Science, Benson Idahosa University,
Benin City, Edo State, Nigeria
Corresponding author email: chinonsoemmanueldibie@gmail.com
This article is drawn from the author's
B.Sc. project submitted to the Department of Computer Science,
Faculty of Science, Benson Idahosa University,
Benin City, supervised by Mr. Fred Otaren.
The author is solely responsible for the content of this article.

https://doi.org/10.56201/ijcsmt.vol.12.no11.2026.pg14.21

Abstract
Sentiment analysis, a vital application of Natural Language Processing (NLP), enables the extraction and classification of emotions from textual data, offering valuable insights across domains like business intelligence and social media monitoring. This study aimed to develop an enhanced text-based sentiment analysis model to accurately classify sentiments in diverse English-language textual data, such as social media posts, while addressing challenges like linguistic nuances and context dependency. Using the Sentiment140 dataset, the methodology involved data preprocessing (text normalization, tokenization, lemmatization), feature engineering with TF-IDF, and training a Support Vector Machine (SVM) model as a baseline, with potential for advanced models like bidirectional GRUs or transformers. The system was deployed as a web application with a Flask backend and React frontend, allowing users to input text and receive real-time sentiment predictions with confidence scores. The SVM model achieved an accuracy of 75% and an F1-score of 74% on a test set of 2,000 samples, demonstrating its effectiveness for binary sentiment classification (positive/negative). However, the model struggled with nuanced sentiments like sarcasm, highlighting the need for advanced contextual models. The study contributes to NLP by providing a practical, scalable sentiment analysis solution, with implications for real-time social media monitoring and customer feedback analysis, while identifying future directions for incorporating transformer-based models and data augmentation to enhance performance.

Keywords: *sentiment analysis, natural language processing, support vector machine, TF-IDF, text classification*

1.0 Introduction

Sentiment analysis, a key application of Natural Language Processing (NLP), focuses on identifying and categorising emotions or opinions expressed in text data. Over the past decade, advances in machine learning and deep learning, particularly the development of transformer-based models and recurrent neural networks, have significantly enhanced the accuracy and applicability of sentiment analysis systems. Early work relied heavily on rule-based approaches and simple models that struggled with contextual understanding, while recent innovations have

shifted towards more sophisticated techniques such as Gated Recurrent Units (GRUs) and transformers, which allow for better handling of complex language structures and nuances (Xu et al., 2024). The use of bidirectional GRUs has been particularly effective in capturing contextual dependencies between words in a sentence, improving sentiment classification even when sentiment is expressed indirectly (Xu et al., 2024). The evolution of transformer-based language models, including those based on attention mechanisms, has further revolutionised sentiment analysis by enabling models to process long-range dependencies within text and generate more accurate predictions (Zekaoui et al., 2023). Advancements in tools like ChatGPT have also facilitated more scalable and automated text annotation techniques, essential for training large-scale sentiment analysis models (Belal et al., 2023). In healthcare and other specialised fields, sentiment analysis is increasingly applied to extract insights from clinical notes, patient feedback, and medical records, supporting improved clinical decision-making (Rajput, 2020). Despite this progress, several challenges remain in accurately identifying and classifying sentiment in text. The growing diversity of sources — social media platforms, customer reviews, news articles — introduces significant variability in language use, making it difficult for models to generalise across domains. Sentiment expressions are often indirect or nuanced, with sarcasm, irony, and culturally specific references frequently posing challenges to automated systems. Context handling is a further difficulty: words may convey different sentiments depending on their position within a sentence (e.g., “I didn't like the movie, but I loved the acting”), which traditional models often fail to capture. Sentiment analysis models also require large, labelled datasets for training, which are costly to produce and, where unbalanced, bias models towards majority sentiment classes. The aim of this study is to develop an enhanced text-based sentiment analysis framework using natural language processing techniques. The specific objectives are to: (i) design a sentiment analysis model incorporating advanced language-processing techniques; (ii) implement the proposed model and evaluate its performance on a benchmark dataset; (iii) compare the model's performance with existing sentiment analysis systems; and (iv) assess the model's effectiveness in handling linguistic nuances such as sarcasm and idiomatic expressions. The study focuses on English-language textual data, including social media posts, product reviews and news articles, and is evaluated using a publicly available benchmark dataset to support applicability across domains. Developing an improved sentiment analysis model has implications for business intelligence (deeper insight into customer opinion for product development and marketing), social media monitoring (real-time tracking of public sentiment for reputation management), and academic research (addressing existing gaps in sentiment-analysis methodology).

Review of Related Literature

Conceptual Review

Sentiment analysis refers to the computational task of identifying and categorising the sentiment expressed in text, typically into positive, negative or neutral classes, though more advanced systems support multi-class or multi-label classification capturing finer emotions such as anger, fear, joy or surprise. The classification process typically involves data preprocessing (tokenisation, stemming, lemmatisation), feature extraction (e.g., term frequency–inverse document frequency or word embeddings), and model training using machine learning or deep learning algorithms, and may be applied at document, sentence or aspect level. Opinion mining is closely related but focuses more specifically on extracting subjective viewpoints — who holds an opinion and what it targets — rather than classifying emotion alone, and is especially relevant to customer-review and political-sentiment analysis. Textual data processing — tokenisation, stemming/lemmatisation, stop-word removal, and feature extraction via Bag-of-Words, TF-IDF or embeddings such as Word2Vec and GloVe — converts raw text into the numerical representations that sentiment models require.

Theoretical Review

Sentiment analysis methodology has evolved through three broad phases. Early rule-based systems used predefined sentiment lexicons but struggled with context and ambiguity, including sarcasm and double negatives. The early 2000s introduced supervised machine learning — Support Vector Machines, Naive Bayes and decision trees — which improved on rule-based approaches but still struggled with complex sentence structures. Deep learning subsequently introduced Recurrent Neural Networks and Long Short-Term Memory networks capable of capturing sequential dependencies between words, and this was followed by transformer-based models, particularly BERT (Bidirectional Encoder Representations from Transformers), whose attention mechanisms process text in both directions and have achieved state-of-the-art sentiment-analysis performance (Zekaoui et al., 2023). Most recently, conversational AI models such as ChatGPT have been applied to scalable sentiment annotation, offering an efficient route to labelling large training datasets (Belal et al., 2023).

Empirical Review

Empirical work spans both global and local contexts. Globally, sentiment analysis has been used to track public sentiment on political candidates and to monitor brand sentiment in marketing. In healthcare, sentiment analysis of patient reviews and feedback has helped providers identify service gaps and improve patient satisfaction (Rajput, 2020). Locally, Nigerian studies have applied sentiment analysis to telecommunications customer feedback and to political discourse, gauging public reaction to events and candidates, as well as to public sentiment on government policy in education and healthcare. A substantial methodological literature informs the specific choices made in this study. Xu et al. (2024) demonstrate that bidirectional GRUs outperform traditional models in sentiment classification, while Zekaoui et al. (2023) show transformer-based models deliver superior performance in opinion mining, at the cost of higher computational demand. Belal et al. (2023) show ChatGPT can meaningfully improve the quality and consistency of sentiment annotation. Naithani and Raiwani (2023) provide a comprehensive overview of NLP and machine-learning approaches to sentiment analysis, and Mandloi and Patel (2020) find SVM and Random Forest classifiers perform well on Twitter data despite its informal language. Ombabi et al. (2020) propose a CNN–LSTM hybrid for Arabic sentiment analysis, illustrating the language-specific adaptation such models often require, while Bayer et al. (2023) address the recurring challenge of limited labelled data through text-generation-based augmentation — an approach this study returns to in its recommendations. Hovy and Prabhumoye (2021) catalogue five sources of bias (training data, algorithms, annotators, user interactions, and their interactions) that affect sentiment-analysis accuracy and that any deployed system should account for.

Gap in Literature

Taken together, the literature shows steady progress from lexicon-based methods through classical machine learning to deep learning and transformer architectures, each stage improving the handling of context and nuance at increasing computational cost. However, much of this literature either evaluates sophisticated models in isolation from a deployable system, or evaluates deployed systems using only simple baseline classifiers without situating their limitations against the state of the art. Few studies combine a transparent, reproducible baseline model with an end-to-end, user-facing deployment and an explicit discussion of where that baseline falls short relative to advanced architectures. This study addresses that gap by implementing and evaluating an SVM-based baseline within a complete, deployed web application, and by explicitly identifying — through both quantitative evaluation and qualitative example cases — where such a baseline requires the contextual capacity of models such as bidirectional GRUs or transformers.

2.0 Methodology

The methodology combines established practice in text analytics, machine learning and web-application design to deliver an end-to-end sentiment analysis system: data collection, data cleaning, feature engineering, model selection, model training and evaluation, and deployment.

2.1 Data Collection and Preprocessing

The primary dataset was Sentiment140, a publicly available Kaggle dataset of labelled tweets (binary positive/negative sentiment), supplemented conceptually by the IMDB Reviews dataset (25,000 training and 25,000 test reviews) as a comparator corpus for future work. Given the scale of Sentiment140 (approximately 1.6 million rows), a random sample of 10,000 rows was drawn for development. Preprocessing followed a systematic pipeline: removal of stop words, punctuation, URLs and special characters; tokenisation; lowercasing; removal of rows with missing or empty text after cleaning; removal of non-sentiment-bearing noise such as hashtags and mentions; and lemmatisation to reduce words to their root form. The cleaned dataset was stored with sentiment and cleaned_text columns.

2.2 Feature Engineering and Model Development

Cleaned text was vectorised using Term Frequency–Inverse Document Frequency (TF-IDF) with a maximum of 5,000 features (scikit-learn's TfidfVectorizer). A Support Vector Machine (SVM) with a linear kernel (scikit-learn's SVC, probability=True) was selected as the baseline classifier for its simplicity, interpretability and established effectiveness in text classification, with the model architecture explicitly designed to allow substitution of more advanced classifiers (e.g., bidirectional GRUs or transformer-based models such as BERT) in future iterations.

2.3 Model Training and Evaluation

The cleaned dataset was split 80/20 into training and test sets using scikit-learn's train_test_split. The TF-IDF vectoriser was fitted on the training set and used to transform both splits; the SVM was trained on the resulting training-set features to classify sentiment as positive (1) or negative (0). Model performance was evaluated on the held-out test set (2,000 samples) using accuracy and F1-score, selected for their established suitability to binary classification evaluation. The trained model and vectoriser were serialised (svm_model.pkl, tfidf_vectorizer.pkl) for use at inference time.

2.4 System Architecture and Deployment

The system follows a client–server architecture comprising a React frontend, a Flask backend, and the trained SVM model. The frontend (HTML, CSS, JavaScript, React) presents a text-input field and results display showing the predicted sentiment and a confidence score, and communicates with the backend over HTTP POST requests to a /predict endpoint. The backend (Flask) receives raw text, passes it through a TextProcessor module for cleaning (URL/mention/special-character removal, stop-word removal, lemmatisation), transforms the cleaned text into TF-IDF features using the fitted vectoriser, and returns the SVM's sentiment prediction and confidence score (via predict_proba) as a JSON response. Third-party libraries used include NLTK (text preprocessing), scikit-learn (TF-IDF, SVM) and Flask-CORS (cross-origin requests between frontend and backend).

3.0 Results

3.1 Model Performance

On the held-out test set of 2,000 samples, the SVM model achieved an accuracy of approximately 75% and an F1-score of 74%, indicating balanced, decent performance for a baseline, linear-kernel text classifier.

Table 1. SVM Baseline Model Performance (Test Set, n = 2,000)

Metric	Value
Accuracy	75%
F1-score	74%
Train/test split	80% / 20%
TF-IDF feature limit	5,000 features
Kernel	Linear

Source: Author's computation (2025).

3.2 Frontend Usability and Example Predictions

The deployed React frontend allowed users to input free text and receive real-time sentiment predictions with associated confidence scores. Representative examples from system testing are summarised in Table 2.

Table 2. Example Frontend Predictions

Input Text	Predicted Sentiment	Confidence
"I am very happy today"	Positive	— (not reported)
"Are you stupid."	Negative	85.67%

Source: System testing output (2025).

3.3 Objective Achievement

Against the study's stated objectives: the model was successfully implemented and evaluated, achieving 75% accuracy and a 74% F1-score (Objectives i–ii). Compared with the data-augmentation-based approach of Bayer et al. (2023), the present system prioritises simplicity and real-time deployment over the computational overhead of synthetic data generation, but correspondingly lacks the robustness gains that augmentation can provide (Objective iii). On the objective of handling linguistic nuance, the SVM model struggled with sarcasm — for example, a sarcastic "Yeah, great job!" is liable to be misclassified — indicating that this objective was only partially met and that contextual models are needed to fully address it (Objective iv).

4.0 Discussion

The implemented system achieves its core objective of developing a functional, deployed text-based sentiment analysis pipeline using NLP techniques. The SVM model with TF-IDF features provides a solid, interpretable baseline at 75% accuracy, a competitive result for a linear classifier of this kind, and the real-time web application meaningfully improves usability by giving users immediate feedback with confidence scores — directly supporting the study's practical-deployment aim.

The system's principal shortfall is its limited handling of linguistic nuance: sarcasm and idiomatic expressions require contextual understanding that TF-IDF features, by construction, cannot capture, since TF-IDF represents word frequency rather than relational or sequential meaning. This is consistent with the wider literature's finding that bidirectional GRUs (Xu et al., 2024) and transformer-based models (Zekaoui et al., 2023) substantially outperform classical machine-learning baselines precisely because they model contextual and sequential dependencies between words. The reliance on a 10,000-row sample of the much larger Sentiment140 corpus may also have limited the model's exposure to diverse sentiment patterns, suggesting that data-augmentation techniques such as those proposed by Bayer et al. (2023) — paraphrasing and back-translation — could improve generalisation without the full computational cost of training on the entire 1.6-million-row dataset.

The frontend's React-based design achieved its goal of an interactive, user-friendly interface, though the absence of visual analytics (e.g., sentiment-distribution charts) represents an unrealised element of the original design brief. Overall, the system meets its basic objectives for sentiment analysis — a working, deployed, evaluated classifier — while leaving clear, literature-grounded room for improvement in contextual accuracy, dataset scale, and analytical presentation.

5.0 Conclusion and Recommendations

This study developed and deployed a text-based sentiment analysis system using NLP techniques, classifying English-language social-media text (Sentiment140) into positive or negative sentiment via an SVM classifier trained on TF-IDF features, delivered through a Flask backend and React frontend. The system achieved 75% accuracy and a 74% F1-score on a 2,000-sample test set, successfully fulfilling the objectives of implementation, evaluation, and real-time deployment, while only partially meeting the objective of handling linguistic nuance such as sarcasm. The study concludes that a transparent, classical machine-learning baseline remains a practical and interpretable foundation for deployed sentiment analysis systems, but that advanced contextual models — bidirectional GRUs or transformer architectures such as BERT — are needed to close the remaining gap in nuance handling identified both quantitatively and through example cases. The study recommends that future work: (i) incorporate bidirectional GRUs or transformer-based models (e.g., BERT) to improve contextual understanding and sarcasm handling; (ii) apply data-augmentation techniques such as paraphrasing or back-translation (Bayer et al., 2023) to improve model robustness and generalisation; (iii) add visual analytics (e.g., sentiment-distribution charts, via a library such as Chart.js) to the frontend to enhance interpretability for end users; (iv) extend the model to other languages (following, for example, Ombabi et al.'s 2020 Arabic sentiment-analysis work) to address the current system's English-language dependency; (v) deploy the application on cloud infrastructure (e.g., Heroku for the backend, Netlify for the frontend) to improve scalability and accessibility; and (vi) integrate real-time data sources such as the Twitter API to extend the system's applicability to live social-media monitoring.

References

Alawida, M., Mejri, S., Mehmood, A., Chikhaoui, B., & Isaac Abiodun, O. (2023). A comprehensive study of ChatGPT: Advancements, limitations, and ethical considerations in natural language processing and cybersecurity. *Information, 14*(8), 462.

Bayer, M., Kaufhold, M. A., Buchhold, B., Keller, M., Dallmeyer, J., & Reuter, C. (2023). Data augmentation in natural language processing: A novel text generation approach for long and short text classifiers. *International Journal of Machine Learning and Cybernetics, 14*(1), 135–150.

Belal, M., She, J., & Wong, S. (2023). Leveraging ChatGPT as text annotation tool for sentiment analysis. *arXiv preprint arXiv:2306.17177*.

Berger, J., & Packard, G. (2022). Using natural language processing to understand people and culture. *American Psychologist, 77*(4), 525.

Cambria, E., Schuller, B., Xia, Y., & Havasi, C. (2013). New avenues in opinion mining and sentiment analysis. *IEEE Intelligent Systems, 28*(2), 15–21.

Caucheteux, C., & King, J. R. (2022). Brains and algorithms partially converge in natural language processing. *Communications Biology, 5*(1), 134.

Eddine Zekaoui, N., Yousfi, S., Rhanoui, M., & Mikram, M. (2023). Analysis of the evolution of advanced transformer-based language models: Experiments on opinion mining. *arXiv e-prints, arXiv-2308*.

Guellil, I., Saâdane, H., Azouaou, F., Gueni, B., & Nouvel, D. (2021). Arabic natural language processing: An overview. *Journal of King Saud University–Computer and Information Sciences, 33*(5), 497–507.

Hariri, W. (2023). Unlocking the potential of ChatGPT: A comprehensive exploration of its applications, advantages, limitations, and future directions in natural language processing. *arXiv preprint arXiv:2304.02017*.

Hovy, D., & Prabhumoye, S. (2021). Five sources of bias in natural language processing. *Language and Linguistics Compass, 15*(8), e12432.

Jia, J., Liang, W., & Liang, Y. (2023). A review of hybrid and ensemble in deep learning for natural language processing. *arXiv preprint arXiv:2312.05589*.

Kalyan, K. S., Rajasekharan, A., & Sangeetha, S. (2021). Ammus: A survey of transformer-based pretrained models in natural language processing. *arXiv preprint arXiv:2108.05542*.

Khan, M. T., Durrani, M., Ali, A., Inayat, I., Khalid, S., & Khan, K. H. (2016). Sentiment analysis and the complex natural language. *Complex Adaptive Systems Modeling, 4*, 1–19.

Koroteev, M. V. (2021). BERT: A review of applications in natural language processing and understanding. *arXiv preprint arXiv:2103.11943*.

Li, B., Hou, Y., & Che, W. (2022). Data augmentation approaches in natural language processing: A survey. *AI Open, 3*, 71–90.

Mandloi, L., & Patel, R. (2020, June). Twitter sentiments analysis using machine learning methods. In *2020 International Conference for Emerging Technology (INCET)* (pp. 1–5). IEEE.

Naithani, K., & Raiwani, Y. P. (2023). Realization of natural language processing and machine learning approaches for text-based sentiment analysis. *Expert Systems, 40*(5), e13114.

Narechania, A., Srinivasan, A., & Stasko, J. (2020). NL4DV: A toolkit for generating analytic specifications for data visualization from natural language queries. *IEEE Transactions on Visualization and Computer Graphics, 27*(2), 369–379.

Nasukawa, T., & Yi, J. (2003, October). Sentiment analysis: Capturing favorability using natural language processing. In *Proceedings of the 2nd International Conference on Knowledge Capture* (pp. 70–77).

Olivetti, E. A., Cole, J. M., Kim, E., Kononova, O., Ceder, G., Han, T. Y. J., & Hiszpanski, A. M. (2020). Data-driven materials research enabled by natural language processing and information extraction. *Applied Physics Reviews, 7*(4).

Ombabi, A. H., Ouarda, W., & Alimi, A. M. (2020). Deep learning CNN–LSTM framework for Arabic sentiment analysis using textual information shared in social networks. *Social Network Analysis and Mining, 10*, 1–13.

Pruksachatkun, Y., Phang, J., Liu, H., Htut, P. M., Zhang, X., Pang, R. Y., ... & Bowman, S. R. (2020). Intermediate-task transfer learning with pretrained models for natural language understanding: When and why does it work? *arXiv preprint arXiv:2005.00628*.

Rajput, A. (2020). Natural language processing, sentiment analysis, and clinical analytics. In *Innovation in Health Informatics* (pp. 79–97). Academic Press.

Song, G. (2021). Sentiment analysis of Japanese text and vocabulary learning based on natural language processing and SVM. *Journal of Ambient Intelligence and Humanized Computing*, 1–12.

William, P., Lanke, G. R., Inukollu, V. N. R., Singh, P., Shrivastava, A., & Kumar, R. (2023). Framework for design and implementation of chat support system using natural language processing. In *2023 4th International Conference on Intelligent Engineering and Management (ICIEM)* (pp. 1–7). IEEE.

William, P., Shrivastava, A., Chauhan, P. S., Raja, M., Ojha, S. B., & Kumar, K. (2023). Natural language processing implementation for sentiment analysis on tweets. In *Mobile Radio Communications and 5G Networks: Proceedings of Third MRCN 2022* (pp. 317–327). Springer Nature Singapore.

Xu, W., Chen, J., Ding, Z., & Wang, J. (2024). Text sentiment analysis and classification based on bidirectional Gated Recurrent Units (GRUs) model. *arXiv preprint arXiv:2404.17123*.

Zhang, T., Schoene, A. M., Ji, S., & Ananiadou, S. (2022). Natural language processing applied to mental illness detection: A narrative review. *NPJ Digital Medicine, 5*(1), 1–13.

Zhang, W., Li, X., Deng, Y., Bing, L., & Lam, W. (2021, August). Towards generative aspect-based sentiment analysis. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)* (pp. 504–510).