INDIAN JOURNAL OF SCIENCE AND TECHNOLOGY

ORIGINAL ARTICLE

# A Machine Learning Framework for Sentiment Classification in the Kashmiri Language

**Sheezan Farooq** [1]*, **Rumaan Bashir**

**1** Department of Computer Science, Islamic University of Science and Technology, Jammu and Kashmir, India

OPEN ACCESS

**Received:** 20/01/2026

**Accepted:** 05/11/2026

**Published:** 21/11/2026

**Citation:** Farooq S, Bashir R (2026) A Machine Learning Framework for Sentiment Classification in the Kashmiri Language. Indian Journal of Science and Technology 19(41): 2824-2831. https://doi.org/10.17485/IJST/v19i41.85

* **Corresponding author**.

sheezanfarooq40@gmail.com

**Funding:** None

**Competing Interests:** None

**Copyright:** © 2026 Farooq & Bashir. This is an open access article distributed under the terms of the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

Published By Indian Society for Education and Environment (iSee)

**ISSN**
Print: 0974-6846
Electronic: 0974-5645

## Abstract

**Objectives:** Sentiment analysis in low-resource languages like Kashmiri is underexplored due to the lack of annotated datasets and computational tools. This study investigates sentiment analysis on Kashmiri sentences using machine learning classifiers: 'Random Forest, Support Vector Machine (SVM), Logistic Regression, Naive Bayes, and K-Nearest Neighbours (KNN)'. **Method:** A manual Kashmiri sentiment dataset was constructed and preprocessed using TFF-IDF vectorization. Standard classifiers 'Support Vector Machine (SVM), Random Forest, Logistic Regression, Naive Bayes and K-Nearest Neighbours (KNN)' were trained and compared with their performance metrics such as accuracy, precision-recall and F1-score. **Findings:** SVM and Random Forest are the top two best performing classifiers, have a reported average accuracy of about 80% and 79%, respectively. With an accuracy of 78%, logistic regression comes in second, while KNN and naive bayes do slightly worse at 74% and 73%, respectively. Classification of neutral sentiments was relatively lower for all models, suggesting that it is one of the main challenges in Kashmiri sentiment analysis. **Novelty:** This study contributes a structured comparison of multiple machine learning classifiers on a dedicated Kashmiri sentiment dataset, offering empirical insights into their strengths and limitations in a low-resource setting. The findings establish a baseline for future research and provide guidance for improving sentiment classification in Kashmiri and similar under-resourced languages.

**Keywords:** Sentiment Analysis; Natural Language Processing (NLP); Machine Learning; Kashmiri Language; TF-IDF Vectorization

## 1 Introduction

Sentiment analysis is a fundamental task in Natural Language Processing (NLP) that focuses on identifying opinions, emotions, and subjective information from textual data. It plays a vital role in applications such as social media monitoring, opinion mining, customer feedback analysis, and decision-support systems [(1)]. 'While significant progress has been achieved in sentiment analysis for high-resource languages such as English and Hindi, low-resource languages continue to remain underexplored due to

the scarcity of annotated datasets, linguistic tools, and computational resources [2].Natural Language Processing is a dynamic field of artificial intelligence that enables computers to process, interpret, and generate human language, thereby facilitating effective human–computer interaction [3]. Recent advances in machine learning and deep learning have substantially improved NLP performance across tasks including sentiment analysis, machine translation, text classification, and language generation [4]. These advancements have led to the development of practical systems such as chatbots, virtual assistants, and intelligent information retrieval platforms [5]. Among these applications, sentiment analysis has gained particular attention as it enables the automatic interpretation of opinions and emotional tones expressed in text [6].

Kashmiri, spoken by more than seven million people, presents unique challenges for sentiment analysis due to its complex morphology, rich vocabulary, and the use of the Perso-Arabic script. Despite its linguistic importance, Kashmiri remains a low-resource language with limited publicly available annotated corpora and computational tools. Existing studies have largely focused on high-resource or moderately resourced languages, while Kashmiri has been comparatively neglected [2]. This gap highlights the need for language-specific resources and systematic evaluation of sentiment analysis techniques tailored to Kashmiri.

Prior research in sentiment analysis has explored a wide range of approaches, including classical machine learning, deep learning, and hybrid models. Supervised machine learning techniques such as Support Vector Machines, Logistic Regression, Naive Bayes, and Bayesian networks have shown strong performance in text classification tasks when sufficient labelled data is available [7],[8]. Ensemble-based methods such as Random Forest and AdaBoost further improve robustness and classification accuracy by combining multiple learners [9]. However, simpler probabilistic models like Naive Bayes often struggle to capture contextual and linguistic nuances, particularly in morphologically rich languages [10]. In recent years, deep learning approaches, including Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), and Deep Neural Networks (DNNs), have significantly advanced sentiment analysis by modelling complex contextual dependencies in text [11],[12]. Transformer-based architectures such as BERT and lightweight variants like DistilBERT have demonstrated promising results, especially for low-resource settings through transfer learning and multilingual representations [13],[14]. Nevertheless, these approaches often require substantial computational resources and large-scale training data, which may not always be feasible for under-resourced languages like Kashmiri. Lexicon-based and hybrid approaches continue to be relevant, particularly for domain-specific sentiment analysis. Dictionary-based and corpus-driven methods leverage predefined sentiment lexicons to assign polarity scores, while hybrid models combine lexicon-based features with machine learning or deep learning techniques to enhance adaptability and performance [15],[16]. For Kashmiri, the development of language-specific sentiment lexicons and annotated corpora remains an open research challenge, further limiting the effectiveness of advanced sentiment classification models [17].

Motivated by these gaps, this study focuses on sentiment analysis for the Kashmiri language using classical machine learning techniques. A manually created and annotated Kashmiri sentiment dataset was developed and processed using TF-IDF vectorization. Five widely used machine learning classifiers Support Vector Machine (SVM), Random Forest, Logistic Regression, Naive Bayes, and K-Nearest Neighbours (KNN) were evaluated for classifying sentiments into positive, negative, and neutral categories. The study aims to identify the most effective classifier for Kashmiri sentiment analysis while highlighting challenges such as neutral sentiment misclassification. By providing a systematic comparison of machine learning models on a dedicated Kashmiri dataset, this work contributes to the development of computational resources and baseline benchmarks for sentiment analysis in Kashmiri and other low-resource languages.

## 2 Methodology

All machine learning models used in this study Support Vector Machine (SVM), Random Forest, Logistic Regression, Naive Bayes, and K-Nearest Neighbours (KNN) were implemented using standard, well-established machine learning libraries. The core algorithms were not structurally changed. Instead, parameter settings were utilized that are generally accepted for low resource text data. TF-IDF vectorization was used as feature extraction and size of maximum featured was 5000. Classifier-specific parameters (e.g., kernel for SVM, number of trees for Random Forest, k value for KNN) were selected according to standard practices known from sentiment analysis literature in order to provide a consistent and reproducible comparison. The Fig. 1 illustrates the methods that is used in the paper:

### 2.1 Data Collection, Preprocessing, and Model Training

2,593 data samples were gathered from benchmark sentiment English datasets, such as SemEval, Yelp, and Amazon reviews [18] and was translated into Kashmiri text using Microsoft Bing translator. The Kashmiri department verified the translations of the benchmark English sentiment datasets into Kashmiri. Reviews were carefully annotated for correctness after being translated

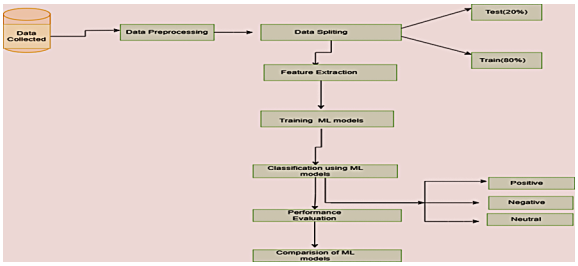


**Fig 1.** Workflow of Sentiment Classification Using Machine Learning Models.

into Kashmiri using Google Translate and verified by linguists. Then a CSV dataset was created using UTF-8 encoding to ensure compatibility with the Kashmiri script. This dataset included three important elements such Serial.no(unique), Kashmiri Sentence and Sentiment Label (Positive=1327, Negative=997, Neutral=269) shown in Fig. 2. Annotation was performed with tight quality control to get accurate dataset. Finally, to ensure data integrity, incomplete information was discarded (e.g. rows with missing labels or sentences). Sentiment indicators were labeled as negative (0), neutral (1) and positive (2). The number of words within the n-grams was tokenised to compute the total count per phrase and features were obtained from TF-IDF vectorisation. An equal training and testing set was generated from the data. Positive and negative sentences (median 10 words) show larger variations (outlier up to 50 words), whereas neutral sentences (median length of 7-8 words) are shorter based on word lengths. The variability of computable features made life difficult for classifiers such as the Naive Bayes. By this pipeline, a high-quality, consistent and clean dataset was ensured which can be used to train machine learning models.

The dataset was spliced into the training (80%) and testing (20%) sets, with stratified sampling being used in order to ensure sentiment distribution (Fig. 2). As a result, five Machine Learning classifiers were trained using the processed training set and evaluated on a testing set with respect to several metrics such as accuracy, precision recall and F1-score. The outcomes were presented in the form of classification reports and confusion matrices. Plotting learning curves and precision-recall curves was another way to evaluate the model's performance.

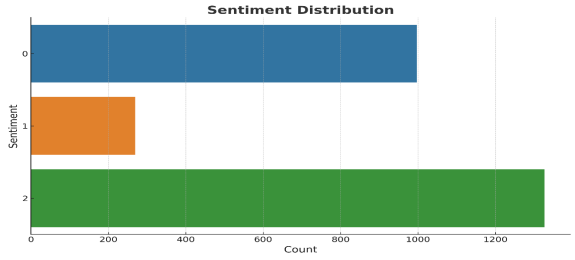


**Fig 2.** Sentiment Distribution in Dataset

## 2.2 Feature Extraction

TF-IDF (Term Frequency-Inverse Document Frequency) vectorization was used to convert Kashmiri sentences into numerical representations, with a maximum of 5000 features. Each document's numerical weight can be determined using the TFIDF approach [19]. The training set's term importance is represented by the TF IDF. In order to convert textual data into numerical representations appropriate for machine learning models, it is used as a feature extraction technique. By integrating two elements, TF-IDF assesses a document's word significance in relation to a corpus. The distribution of non-zero features (TF-IDF values) throughout the training and testing datasets shown in Fig. 3. The y-axis shows the frequency of documents with a particular number of non-zero features, while the x-axis shows the number of non-zero features in each document. The test set (orange line) is more dispersed, indicating a sparser distribution, whereas the training set (blue bars) exhibits a peak at 5–10 non-zero features, suggesting a denser feature representation. This draws attention to how the training and test sets differ in feature sparsity, which may have an impact on the model's capacity for generalisation. The top 5000 most important features from the Kashmiri sentences are converted into TF-IDF feature vectors using the TfidfVectorizer from sklearn. By down-weighting common terms that are less significant for sentiment classification, this transformation allows the machine learning model to

identify the text's underlying patterns. The several classifiers under consideration are then trained and tested using the resultant feature vectors.

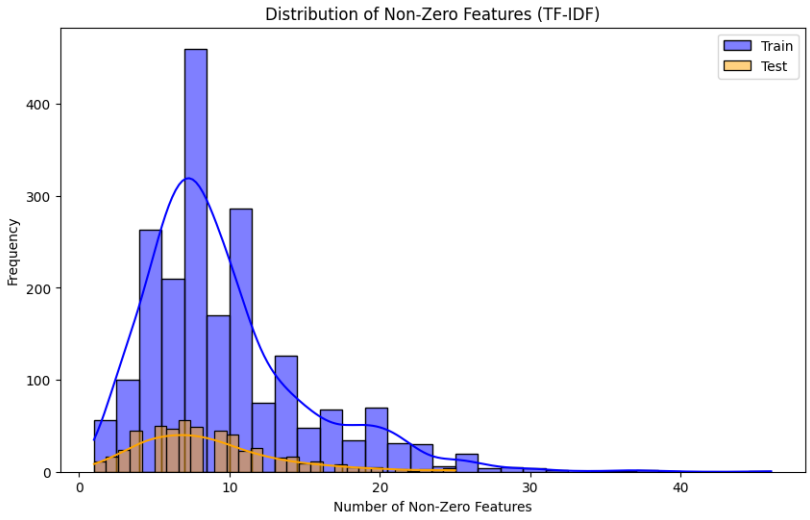


**Fig 3.** Sentiment Distribution of features

## 3 Result and Discussion

Five machine learning classifiers Support Vector Machine (SVM), Random Forest (RF), Logistic Regression (LR), Naive Bayes, and K-Nearest Neighbours (KNN) were examined in this study for Kashmiri sentiment classification using TF-IDF features. The model performance was evaluated by accuracy, precision, recall and F1-score while classification reports, confusion matrices and precision-recall curves were used to study class-wise performances as well as misclassifications. Overall, SVM and Random Forest gave the best prediction, but with SVM demonstrating superior robustness and generalisation capability.

Among the baseline classifiers, Random Forest achieved an overall accuracy of 0.79, showing balanced performance for negative sentiment (0.72 precision, 0.70 recall, F1-score 0.71) and stronger results for positive sentiment (0.75 precision, 0.81 recall, F1-score 0.78), while neutral sentiment remained challenging with high precision (0.97) but lower recall (0.66). The KNN classifier also obtained an accuracy of 0.74, performing relatively well for positive sentiment (0.78 precision, 0.80 recall, F1-score 0.79) but exhibiting limited capability for neutral sentiment detection due to low recall (0.52) despite higher precision (0.85). Logistic Regression demonstrated competitive performance with an accuracy of 0.78, achieving balanced results for negative sentiment (0.73 precision, 0.74 recall, F1-score 0.73) and strong performance for positive sentiment (0.78 precision, 0.85 recall, F1-score 0.81), while neutral sentiment classification remained weaker (precision 1.00, recall 0.50, F1-score 0.67). Naive Bayes achieved an overall accuracy of 0.73, performing well for positive sentiment (0.74 precision, 0.87 recall, F1-score 0.80) but showing significant limitations for neutral sentiment (precision 1.00, recall 0.21, F1-score 0.35), reflecting its difficulty in capturing contextual nuances.

The Support Vector Machine (SVM) was the best classification model in all models. It is linear kernel model used with TF-IDF features and provided the accuracy of 0.81, achieving both strong and balanced results over sentiment classes. For negative sentiments, SVM achieved precision=0.75, recall=0.82, F1-score=0.78. Neutral sentiment classification exhibited high precision (0.91) but comparatively lower recall (0.73), resulting in an F1-score of 0.81, highlighting the inherent difficulty of identifying neutral sentiment in Kashmiri text. The highest performance was observed for positive sentiment, with 0.85 precision, 0.82 recall, and an F1-score of 0.84, as illustrated by the precision–recall curves in Fig. 4. The classification report in Fig. 5 and the confusion matrix analysis in Fig. 6 further confirm that SVM effectively classifies positive and negative sentiments, with most misclassifications occurring in the neutral class.

The observed performance trends of the baseline classifiers are consistent with findings reported in prior sentiment analysis studies on low-resource and morphologically rich languages, where Random Forest and Logistic Regression have been shown to provide stable and competitive performance when combined with TF-IDF features [20], [21]. Similarly, the weaker performance of KNN and Naive Bayes (NB) particularly for neutral sentiment has also been documented in earlier works, which attribute

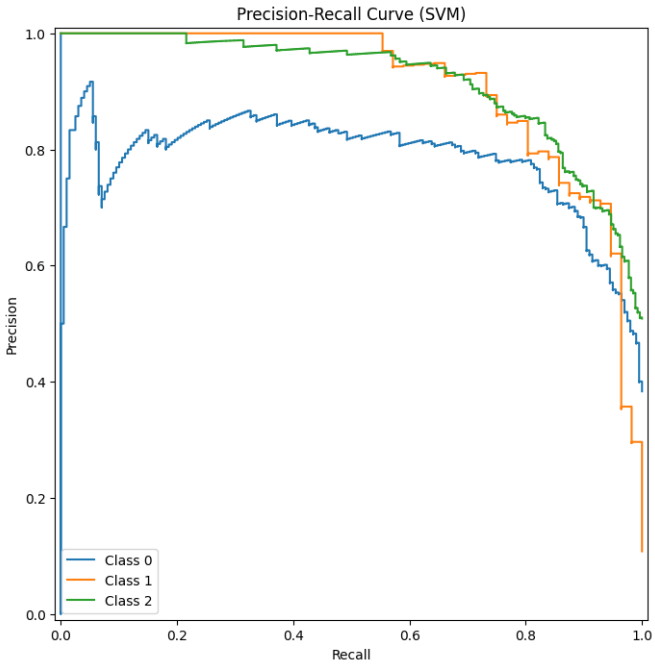


**Fig 4.** Precision-Recall Curve of SVM

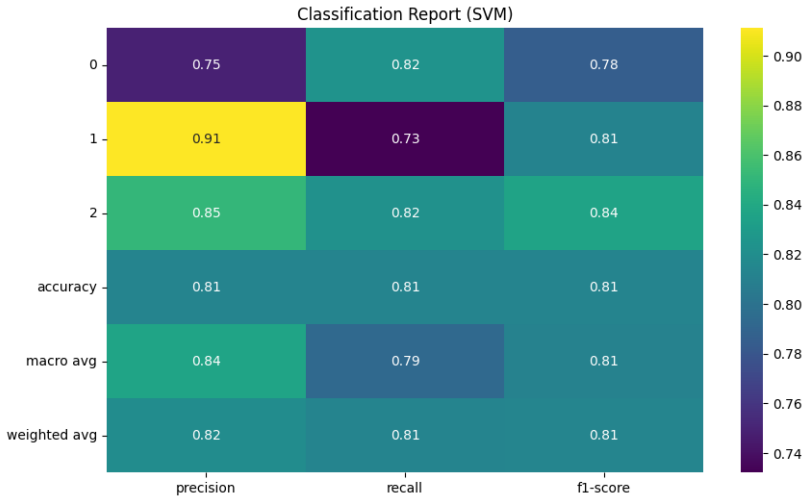


**Fig 5.** Classification Report of SVM

this behaviour to feature sparsity and the limitations of distance-based and probabilistic models in capturing contextual ambiguity[22],[23]. These consistencies validate the reliability of the baseline results obtained in the present study.

The superior performance of SVM observed in this work aligns with several recent studies on low-resource sentiment analysis, which report that margin-based classifiers combined with TF-IDF features remain highly effective under limited data and computational constraints[20]. However, the persistent difficulty in accurately classifying neutral sentiment partially contradicts findings from transformer-based approaches that report improved neutral sentiment recall through contextual embeddings[24]. This discrepancy can be attributed to differences in model architecture, feature representation, and dataset size. Comparable issues of neutral sentiment ambiguity have, however, been recognized in several low-resource language work and this confirms that neutral sentiment classification is still an open research question.

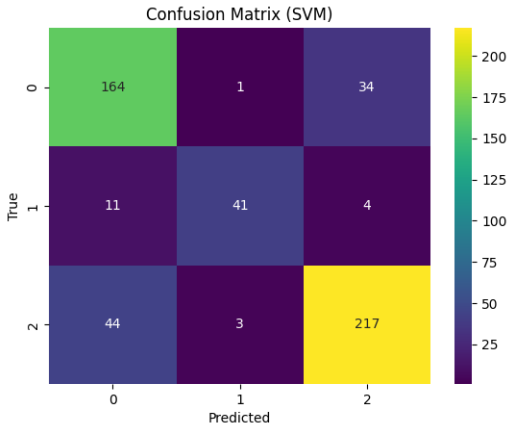


**Fig 6.** Confusion Matrix and Normalized Matrix of SVM

The accuracy comparison of the two models presented Fig. 7 and the comparison results in Table 1 quantitatively prove the performance of the proposed method. In general, from results, it may be concluded that at least all of the considered classifiers are promising for the sentiment classification of Kashmiri, but not as powerful and generalizing solution as SVM. These results demonstrate the potential of classical machine learning models for low resource languages while also indicating future work in using contextual deep learning models, better feature representations and more annotated data to improve neutral sentiment classification and overall generalization.

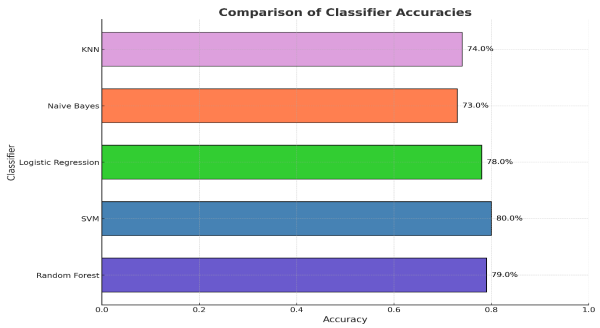


**Fig 7.** Comparison of Accuracies

**Table 1.** Comparison based on performance evaluation

| Model | Accuracy (%) | Macro F1-Score | Weighted F1-Score | Precision (Negative) | Recall (Negative) | F1-Score (Negative) | Precision (Neutral) | Recall (Neutral) | F1-Score (Neutral) | Precision (Positive) | Recall (Positive) | F1-Score (Positive) |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| RF | 79 | 0.76 | 0.75 | 0.72 | 0.70 | 0.71 | 0.97 | 0.66 | 0.79 | 0.75 | 0.81 | 0.78 |
| KNN | 74 | 0.72 | 0.75 | 0.69 | 0.74 | 0.71 | 0.85 | 0.52 | 0.64 | 0.78 | 0.80 | 0.79 |
| LR | 78 | 0.75 | 0.76 | 0.74 | 0.71 | 0.72 | 0.82 | 0.57 | 0.68 | 0.79 | 0.85 | 0.82 |
| Naive Bayes | 73 | 0.61 | 0.71 | 0.69 | 0.68 | 0.69 | 1.00 | 0.21 | 0.35 | 0.74 | 0.87 | 0.80 |
| SVM | 80.6 | 0.81 | 0.81 | 0.75 | 0.82 | 0.78 | 0.91 | 0.73 | 0.81 | 0.85 | 0.82 | 0.84 |

## 4 Conclusion

In conclusion, the sentiment analysis conducted using various machine learning classifiers demonstrated varying levels of efficacy on the Kashmiri sentence dataset. Among the evaluated models, Random Forest and Support Vector Machine (SVM) achieved the highest performance, with SVM attaining the best overall accuracy of 81%, followed by Random Forest with 79% accuracy. These models effectively differentiated between positive, negative, and neutral sentiments, with SVM in particular providing balanced precision and recall across sentiment classes. Class-wise evaluation further revealed that SVM achieved strong F1-scores for positive (0.84) and neutral (0.81) sentiments, confirming its robustness in handling high-dimensional sparse textual features. However, the results also indicate that improvements are required for Naive Bayes and K-Nearest Neighbours (KNN), which showed comparatively lower accuracies of 73% and 74%, respectively, and struggled significantly with neutral sentiment classification. Despite high precision values for the neutral class in some models, recall remained consistently lower, highlighting the inherent ambiguity and lexical overlap associated with neutral expressions in Kashmiri text. This quantitative evidence further reinforces that neutral sentiment classification as a persistent challenge across classical machine learning approaches. The key novelty of this work lies in the systematic evaluation of different classical machine learning classifiers on a manually developed and annotated Kashmiri sentiment dataset using a unified experimental framework. In contrast to many existing works that are either focusing on high resource language(s) or multilingual transfer-based models, this study shows that well engineered classical models such as SVM with TF-IDF features can give reasonable and competitive performance in low resource language setting under limited computational constraints.

For future work, several actionable directions are identified. First, enhancing feature extraction techniques by incorporating sentiment-aware weighting or linguistic cues specific to Kashmiri may help improve neutral sentiment recall. Second, exploring ensemble strategies or integrating contextual deep learning models such as LSTMs or transformer-based architectures could further enhance contextual understanding and classification accuracy. Finally, expanding the dataset with more diverse and naturally occurring Kashmiri text is expected to improve model generalisation across domains'. These steps can strengthen the effectiveness of sentiment analysis systems for Kashmiri and contribute more broadly to research on low-resource language processing.

## References


1) Farooq S, Bashir R. A novel stacked ensemble framework for sentiment classification in Kashmiri. *Journal of Intelligent & Fuzzy Systems*. 2025;p. 18758967251385270.
2) Lone NA, Giri KJ, Bashir R. Natural language processing resources for the Kashmiri language. *Indian Journal of Science and Technology*. 2022;15(43):2275–2281. https://doi.org/10.17485/IJST/v15i43.1964.
3) Azizan NABN, Saee SB, Yusof MAB. Discovering Popular Topics of Sarawak Gazette (SaGa) from Twitter Using Deep Learning. *Communications in Computer and Information Science*. 2023;p. 178–192. https://doi.org/10.1007/978-981-99-0405-1_13.
4) Huang H, Zavareh AA, Mustafa MB. Sentiment Analysis in E-Commerce Platforms: A Review of Current Techniques and Future Directions. *IEEE Access*. 2023. https://doi.org/10.1109/ACCESS.2023.3307308.
5) Vatambeti R, Mantena SV, Kiran KVD, Manohar M, Manjunath C. Twitter sentiment analysis on online food services based on elephant herd optimization with hybrid deep learning technique. *Cluster Computing*. 2024 Feb;27(1):655–671. https://doi.org/10.1007/s10586-023-03970-7.
6) Khan J, Ahmad N, Khalid S, Ali F, Lee Y. Sentiment and Context-Aware Hybrid DNN With Attention for Text Sentiment Classification. *IEEE Access*. 2023;11:28162–28179. https://doi.org/10.1109/ACCESS.2023.3259107.
7) Gutiérrez L, Bekios-Calfa J, Keith B. A review on Bayesian networks for sentiment analysis. In: Proceedings of the International Conference on Software Process Improvement (CIMPS). Cham, Switzerland. Springer. 2019;p. 111–120. https://doi.org/10.1007/978-3-030-58145-5_10.
8) Dhamayanthi N, Lavanya B. The Role Of Naïve Bayes, SVM, And Decision Trees In Sentiment Analysis. *Theory and Practice*. 2024;2024(4):6377–6381. https://doi.org/10.53555/kuey.v30i4.2392.
9) Chen L, Han J, Tian F. Colorization of fusion image of infrared and visible images based on parallel generative adversarial network approach. *Journal of Intelligent & Fuzzy Systems*. 2021;41(1):2255–2264. https://doi.org/10.3233/JIFS-201742.
10) Jofche N, Mishev K, Stojanov R, Jovanovik M, Zdravevski E, Trajanov D. PharmKE: Knowledge Extraction Platform for Pharmaceutical Texts Using Transfer Learning. *Computers*. 2023;12(1):17. https://doi.org/10.3390/computers12010017.
11) Joloudari JH, et al. BERT-deep CNN: state of the art for sentiment analysis of COVID-19 tweets. *Social Network Analysis and Mining*. 2023;13(1). https://doi.org/10.1007/s13278-023-01102-y.
12) Hindarto D. Comparison of RNN Architectures and Non-RNN Architectures in Sentiment Analysis. *Sinkron*. 2023 Oct;8(4):2537–2546. https://doi.org/10.33395/sinkron.v8i4.13048.
13) Tang Z, Li W, Li Y, Zhao W, Li S. Several alternative term weighting methods for text representation and classification. *Knowledge-Based Systems*. 2020;207:106399. https://doi.org/10.1016/j.knosys.2020.106399.
14) Saragih H, Manurung J. Leveraging the BERT model for enhanced sentiment analysis in multicontextual social media content. *Jurnal Teknik Informatika CIT Medicom*. 2024;16(2):82–89.
15) Liu H, Chen X, Liu X. A study of the application of weight distributing method combining sentiment dictionary and TF-IDF for text sentiment analysis. *IEEE Access*. 2022;10:32280–32289. https://doi.org/10.1109/ACCESS.2022.3160416.
16) Salur MU, Aydin I. A Novel Hybrid Deep Learning Model for Sentiment Classification. *IEEE Access*. 2020;8:58080–58093. https://doi.org/10.1109/ACCESS.2020.2982538.

17) Thukroo IA, Bashir R, Giri KJ. A review into deep learning techniques for spoken language identification. *Multimedia Tools and Applications*. 2022 Sep;81(22):32593–32624. https://doi.org/10.1007/s11042-022-13054-0.
18) Blitzer J, Dredze M, Pereira F. Biographies, Bollywood, Boom-boxes and Blenders: Domain Adaptation for Sentiment Classification. In: Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics. 2007. Available from: http://ida.
19) Ahuja R, Chug A, Kohli S, Gupta S, Ahuja P. The impact of features extraction on the sentiment analysis. *Procedia Computer Science*. 2019;p. 341–348. https://doi.org/10.1016/j.procs.2019.05.008.
20) Gosai A, De S, Thankachan K. Sentiment analysis on movie reviews: A deep dive into modern techniques and open challenges. In: arXiv. 2026.
21) Thakur A. Classifying causes of depression from social media posts using machine learning and NLP. 2025.
22) Ding B, Qin C, Zhao R, Luo T, Li X, Chen G, et al. Data augmentation using large language models: Data perspectives, learning paradigms and challenges. In: arXiv. 2024.
23) Klasan A, Hagberg J. Multi-turn contextual sentiment analysis of IT support dialogues: A BERT-based approach. 2025.
24) Naseem U, Razzak I, Musial K, Imran M. Transformer-based deep intelligent contextual embedding for Twitter sentiment analysis. *Future Generation Computer Systems*. 2020;113:58–69.