International Journal of Informatics, Information System and Computer Engineering

Improving Sentiment Classification using Ensemble Learning

*Sherwan A. Abdullah[1], Mohammed I. Salih[2], Omar M. Ahmed[2]**

[1]College of Education, University of Zakho, Duhok, Iraq
[2]Computer Information System Department, Duhok Polytechnic University, Iraq
*Corresponding Email: omar.alzakholi@dpu.edu.krd

ABSTRACTS

This paper presents an ensemble learning-based approach to improve sentiment classification accuracy in the IMDB movie reviews dataset. To this end, we tap three diversified models: Logistic Regression, Random Forest, and a Bidirectional Long Short-Term Memory neural network. Each one contributes its unique strengths to the ensemble, enhancing the overall performance. The text data has been processed using a statistical formula that converts the text document into a vector from the relevancy of the word with bigrams; data have been transformed to make it useful for Logistic Regression and Random Forest classifiers. The LSTM neural network is designed to capture sequential dependencies through an embedding layer followed by a bidirectional LSTM and dense layers with dropout regularization. The ensemble method then combines predictions of these models by majority voting, thus the interpretability and robustness of conventional classifiers are preserved, while advanced capabilities from neural networks are maintained. Our experiments prove that this ensemble approach does obtain an accuracy of 89.2% on the test dataset, which outperforms individual models. This study realizes some possible ways to combine traditional machine learning techniques with deep learning models in sentiment analysis tasks.

ARTICLE INFO

Article History:
Received 27 Aug 2024
Revised 10 Oct 2024
Accepted 03 Nov 2024
Available online 30 Dec 2024
Publication date 01 Dec 2025

Keywords:
Ensemble Learning,
Logistic Regression,
Random Forest,
Bidirectional Long Short-Term Memory,
Sentiment Classification

DOI: https://doi.org/10.34010/injiiscom.v6i2.13921
p-ISSN 2810-0670 e-ISSN 2775-5584

1. INTRODUCTION

Nowadays, the rise of online fora where people can voice out their opinions creates an exponential amount of unstructured textual data on a daily basis. Every social media post to product review is just a huge pile of text data that contains mines of public sentiment, consumer preferences, and surfacing trends. Sentiment analysis, also known as opinion mining, has emerged as a key area in the field of natural language processing oriented toward identifying and categorizing opinions expressed in textual data. It is the determination of whether a piece of writing expresses some positive, negative, or neutral point, so sentiment analysis holds actions that can be done by businesses, policymakers, or other researchers (Gupta & Noliya, 2024).

The IMDB movie reviews dataset represents one of the most popular benchmark datasets for sentiment analysis models. It consists of 50,000 labeled movie reviews as positive and negative. This binary classification task has gained great interest in research; thus, it is considered rather challenging. However, the score from this particular task is very low because the subtlety in movie reviews expresses sentiment in complex and secretive manners, such as sarcasm, irony, and implicit references. This is exactly why one of the major pursuits of developing models in the field of sentiment analysis is to be able to correctly interpret these subtle nuances (Mercha et al., 2024).

Traditional machine learning techniques have found wide application in tasks that involve sentiment analysis. Simplicity, interpretability, and effectiveness on a variety of datasets are often given as reasons for the wide application of logistic regression—a linear model preferred in affording probabilistic interpretations to predictions—hence making it very useful in understanding how different features contribute to a classification decision. Random Forest is a robust method of combining decision trees to improve prediction performance and reduce overfitting. These two models have shown good performance in sentiment analysis tasks, such as working in combination with feature engineering techniques like TF-IDF vectorization, which converts text data into ML model-compatible formats (Shah, 2024).

However, some weaknesses of conventional machine learning models—specifically those related to complex or high-dimensional data, such as natural language text—are often observed. These models tend to fail to capture the critical sequential and contextual features within text that are important for understanding sentiment. For example, the sense of a word can be significantly different in different contexts, which most traditional models are not able to capture. On the other hand, this kind of model usually depends heavily on manual feature engineering, which is time-consuming and error-prone (Khujaev et al., 2023).

In response to these limitations, deep learning methods — especially neural networks, which capture high-dimensional patterns in data well— have been used more and the techniques involve devising shallow iteration algorithms. Among them, Long Short-Term Memory (LSTM) networks have been widely used in sentiment analysis because they can capture long-range dependencies and maintain information over time. LSTM is a specific type of

DOI: https://doi.org/10.34010/injiiscom.v6i2.13921
p-ISSN 2810-0670 e-ISSN 2775-5584

recurrent neural network RNN, which were inherently engineered to mitigate the vanishing gradient issue when using standard RNNs so that they can learn things from sequences data efficiently. This makes them very useful in sentiment analysis where the sequence of words can determine if a sentence expression is positive or negative (Soumya et al., 2022).

Over the last few years, plenty of work has been done in the sentiment analysis field to come up with better and more accurate models by combining machine learning with deep learning methods because both are really advantageous when used together. For example, ensemble learning is one of these methodologies, and it involves training many models and then averaging the results to do much better than we could if we had trained a single model. In many recent researches, we have seen ensemble techniques work well to make more robust and accurate machine learning models by reducing bias and variance of individual models (Liu et al., 2024).

In this paper, we present an ensemble technique for sentiment analysis based on three models: Logistic Regression, Random Forest, and enhanced Bidirectional LSTM model, which is used to train another layer using word vectors obtained from embedding layers. Each of these models has its own benefits. Logistic Regression is one of the simplest machine learning models, making it more interpretable, especially due to the clear relationship between features and their contribution to sentiment classification (Aliman et al., 2022). Random Forest, as an aggregation of decision trees, can learn non-linear relationships between features and interactions directly from the data, making it a robust performer (Karthika et al., 2019). The LSTM model is good at recognizing the underlying meaning in text data, making it very effective for classifying movie reviews based on sentiment (Wu et al., 2021).

In this study, an ensemble method was used to combine three of these models using majority voting (the simplest and most widely-used type of ensemble technique in which the final prediction is based on a "voting" system output by individual models). It enables us to take the best out of each model and remove their limitations. As an example, the LSTM model has been long enough to learn sequential aspects of text but it may also overfit engaged in small dataset training. Lots of the sparsity in predicting success is counter acted as we include Logistic Regression and Random Forest models which tend to be pretty stable, so this will combat a lot overfitting throughout.

The Reason for using an ensemble approach is that different models may catch a different aspect of the data and by combining them we can improve upon overall performance. In the case of sentiment, we can say, Logistic Regression might capture linear relationships in data and easy to interpret but Random Forest model captures interaction between feature that may logistic regression fail so LSTM for this dataset because it requires temporal dependency. Together, we seek to form a system that is more than the sum of its parts and is able to perform at better accuracy as well generalization compared with each specific model performing on their own.

In this paper we show the efficacy of our ensemble technique by performing

DOI: https://doi.org/10.34010/injiiscom.v6i2.13921
p-ISSN 2810-0670 e-ISSN 2775-5584

several kinds of experiments on IMDB dataset. We present evaluations of the performance of each individual and ensemble model in terms of accuracy, robustness etc. Our experiment demonstrates that the ensemble model improves accuracy of sentiment analysis over using individual models, evidencing further utility for combining multiple modeling strategies.

The remainder of this paper is organized as follows: Section 2 reviews related work in the field of sentiment analysis and ensemble learning. Section 3 describes the methodology used in this study, including the data preparation, model training, and ensemble techniques. Section 4 presents result, discusses the findings and potential future work. Finally, Section 5 concludes the paper with a summary of the contributions and implications of this research.

2. RELATED WORK

(Ranjan et al., 2023) utilize the IMDB dataset and implement a Naive Bayes classifier for sentiment classification. They preprocess the text data (tokenization, stemming, and removing stop words), and then apply feature extraction techniques like TF-IDF. The model's performance is then evaluated based on accuracy. (Bhowmic et al., 2024) combines multiple classifiers (e.g., Decision Trees, Random Forest, and SVM) in an ensemble learning framework. The ensemble model aggregates the predictions of individual classifiers using techniques like majority voting or weighted averaging to enhance the accuracy of sentiment classification on the IMDB dataset. (Mutinda, 2024) introduce a sentiment lexicon-based approach where they integrate lexicon features with machine learning models. After preprocessing the data, they combine traditional text representations (like TF-IDF) with lexicon features to improve sentiment classification performance. Models like SVM and Naive Bayes are used for classification. (Danyal et al., 2024) investigates leverages transformer-based models, specifically BERT and XLNet, for sentiment classification. The text data is tokenized and fed into pre-trained BERT and XLNet models. The final layer of these models is fine-tuned on the IMDB dataset to improve classification accuracy. The study compares the performance of these models against traditional machine learning approaches. (Saad et al., 2024) propose a transformer-based model that combines both machine learning and deep learning techniques. The model uses pre-trained transformer layers (e.g., BERT) along with fully connected layers for classification. The IMDB dataset is preprocessed, and embeddings are generated before being fed into the model for training.

(Jain & Agarwal, 2023) implemented machine learning models (SVM, Twin SVM, and Naive Bayes) for sentiment classification. The IMDB movie reviews data are preprocessed as tokenization and then feature extraction was conducted by TF-IDF. Finally, the trained models were validated with the accuracy. (Sultana et al., 2024) presented the deep learning techniques such as LSTM and CNN for sentiment analysis. IMDB dataset is taken, and first, preprocessing is done and converted to tokenization. Then the training is done with technique words embedding models such as GloVe or Word2Vec. Finally, evaluation of the model has been done similar to the previous ones. (Hussain & Naseer, 2024) utilized the machine learning and one

DOI: https://doi.org/10.34010/injiiscom.v6i2.13921
p-ISSN 2810-0670 e-ISSN 2775-5584

deep learning technique comparing between the logistic regression, LSTM, and bi-LSTM. The text data are tokenization, and word embedding formation is carried out by the Glove embedding model. Here also performed as the logistic regression as the base model, and other two LSTM and bi-LSTM for the sequence data which performed. (Vanlalnunpuia, 2023) Hybrid approach performance using feature selection with an ensemble of classifiers. First, IMDB dataset preprocessing is done, on that, features are selected a feature selection technique, where feature selection is done using PCA or Chi-Square and classification is done for an ensemble classifier Random Forest or Gradient Boosting optimizing by grid search technique for accuracy. (Krishna et al., 2024) augment synonym on the dataset using GAP layers on deep learning. Preprocessed data set: IMDB, word embeddings are done, gap layers, synonym augmentation and finally, the model is trained and validated the data by augmentation using GAP.

3. METHODOLOGY

In this study we use ensemble learning techniques for improving sentiment classification on the IMDB movie reviews dataset. Figure 1: Dataflow of the proposed methodology: Three different models, such as Logistic Regression, Random Forest and Bidirectional Long Short-Term Memory (LSTM) neural network are combined by majority voting to classify each review finally We give a detailed account of our methodological choices with respect to the data preprocessing used (methods, dimensions etc.), selection of model and ensemble methods below.

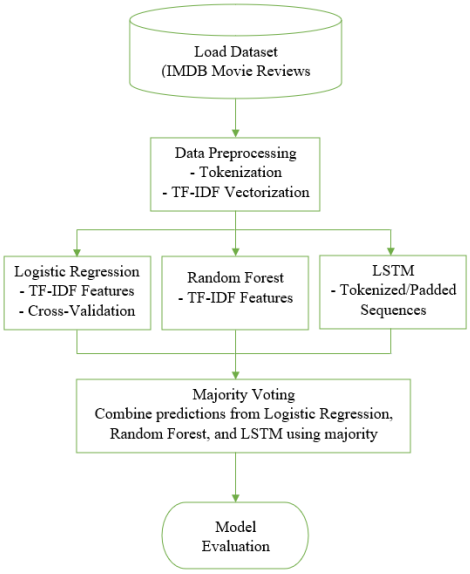

Fig. 1 Dataflow of The Proposed Approach

3.1. Data Preparation

IMDB Movie Reviews: A well-known benchmark dataset for sentiment analysis of movie reviews. It consists of 50,000 movie reviews labeled as positive and negative. They are split 50/50 into training and test sets, for a total of 25,000 reviews per subset. This dataset is quite balanced with 50% of the reviews being positive and half negative both in training and test sets. This balance is important so that our models are not too biased in favor of one or the other sentiment when it classifies.

The IMDB data set was obtained with TensorFlow Datasets (tensorflow_datasets) for this study. This utility simplifies the process of loading and preparing datasets by automatically downloading, extracting, and splitting data. The data is loaded in a format that has them as reviews, with their labels : sentiment The review text is initially read in as byte strings, which then must be decoded from UTF-8 to the appropriate format for working with textual data.

DOI: https://doi.org/10.34010/injiiscom.v6i2.13921
p-ISSN 2810-0670 e-ISSN 2775-5584

3.2. Data Preparation

Tokenization of Text: It is a process where text data needs to be converted into numerical format before it can used as an input in machine learning models. Most machine learning algorithms and neural networks are designed to take numerical input, not raw text so this transformation is absolutely necessary. It does tokenization of the text data, for this purpose. Tokenization is the task of breaking up text into smaller components, and as you probably guess these are called tokens that can be numbers. We utilized TfidfVectorizer from scikit-learn for conventional machine learning models like Logistic Regression and Random Forest, whereas a modified tokenizer was employed to transform text into sequences of integers in the LSTM model.

The Logistic Regression and Random Forest models take as input a fixed-length numerical representation of each review, which means we should transform the reviews into vectors. We achieved this using Term Frequency-Inverse Document Frequency (TF-IDF) vectorization. Ascertain the Significance of a Word to Each Document in Database Using TF-IDF It combines word frequency with inverse document frequency, meaning that it weights high words (like "the" and "and") less than low ones to facet a single document. We also trained a TfidfVectorizer and had it remove (English) stop words as well, in addition to limiting the n-grams from 1-gram up to 2-gram ranges so that some local context within the texts is preserved. This is very important in sentiment analysis because there are phrases (e.g., not good) which can change the meaning if we did a word-by-word comparison.

For the LSTM model, due to its sequence padding: The reviews were tokenized into sequences of integers whereby each unique word was assigned a single integer. After tokenization, we padded all the sequences to a fixed length with pad_sequences from TensorFlow. We have to pad because neural networks expect consistent sizes for input data; The tokenizer padding is only simple padding. Here, we padded every review to the maximum length of 500 tokens which is going to be enough for capturing most of their contents without filling out excessive padding. Reviews longer than this were clipped, and shorter ones had zero-padding added to the end. Since we're feeding our LSTM model sequence-by-sequence, the same length can ensure that all inputs are processed in exactly the same manner.

3.3. Model Selection

Logistic Regression: Logistic regression is linear classifier and has been a great help in binary classification task like sentiment analysis. It estimates the probability that certain input corresponds to a particular class (positive or negative sentiment) through logistic function. For this study, we applied LogisticRegressionCV which is a sklearn implementation of the logistic regression that integrates cross-validation inside for tuning model hyperparameters; Cross Validation is used to choose the best L2 Regularization for our model by checking its performance in different divisions of training data. The most important part is the Regularization, because we want to ensure that the model doesn't overfit on our training data and especially in high dimensional spaces (which are created by TF-IDF encoding).

DOI: https://doi.org/10.34010/injiiscom.v6i2.13921
p-ISSN 2810-0670 e-ISSN 2775-5584

Random Forest: Random Forest builds multiple decision trees and merges them together to get a better result. A large number of these algorithms belong to the family of bagging classifiers, which bootstraps samples from training data and trains multiple decision trees. This reduces overfitting in comparison to individual decision trees and improves the generalization of model. The Random Forest Classifier was set to have 200 trees in our study (n_estimators=200). Every tree in the forest is constructed based on a different bootstrap sample of the training data being selected, and for every split at picking node it considers only a random subset of features. This is good because some randomness will lead to a more varied and stable model ensemble, which in turn reduces the chance of overfitting to particular patterns found in the training data.

Bidirectional LSTM Neural Network: Long Short-Term Memory (LSTM) networks are a type of recurrent neural network (RNN) used to handle sentence-level sequential data, which makes them appropriate for text classification tasks. This is useful when trying to determine the sentiment of a review because it might be key for classification that some information is written later- all part of why an LSTM can learn long term dependencies in text. To deal with this we trained a bidirectional LSTM which processes the input sequence in forward and backward direction to capture dependencies from past as well as future within that same. The architecture includes an embedding layer, a bi-directional LSTM layer of 64 units covering up to sequence length, followed by global max pooling output dimensionality reduction achieved through Pooling Layer and another non-linear dense ReLU activation and finally the outputs are obtained from final forward pass with sigmoid activation for binary classification. The dropout regularization was used just after the dense layer, as mechanism to prevent overfitting training using randomly selected fraction of input unit settled to 0. The model was trained on an LSTM with the adam optimizer, since it allows to adapt learning rates during training and thus can be a little faster in convergence.

3.4. Ensemble Method

Ensemble learning: The combined predictions of several base models are achieved to enhance the performance. The ensemble method used in the study was majority voting (Polikar, 2012). The Logistic Regression, Random Forest and LSTM models predicted the test set individually as well as a consensus decision between three of them for each review in itself. A simple, nevertheless powerful ensemble method is majority voting. This works well when form a base model perspective the decision processes are unique. We can use a linear model, tree-based models, and neural networks together in different ways to exploit their strengths by compensating for the weaknesses.

Why Majority Voting? We are using majority voting as our ensemble method since it is simple and effective to use when the base models are well-calibrated and different. As we have seen, Logistic Regression and Random Forest has good strength in interpretability and robustness respectively… And LSTM utilizing textual information of the word pairs brings complexity but an important ability to capture sequential pattern across space-time. Majority voting also keeps the final prediction from leaning

DOI: https://doi.org/10.34010/injiiscom.v6i2.13921
p-ISSN 2810-0670 e-ISSN 2775-5584

too much on a single model, which can help prevent that your ensemble does poorly due to quirks of one particular model. Also, majority voting is fast to compute compared with other ensemble techniques such as stacking and hence can be deployed in real-time applications that require fast inference.

4. RESULTS

This section presents the detailed results of the experiments conducted using the ensemble learning approach on the IMDB movie reviews dataset. The performance of the ensemble model was evaluated based on its accuracy, and comparisons were made with individual models. Additionally, insights into the models' behavior, the impact of different hyperparameters, and the contribution of each model to the ensemble's final prediction are discussed.

4.1. Performance of Individual Models

To begin with, we evaluated the performance of the individual models before combining them into the ensemble. The individual models used were Logistic Regression with cross-validation Logistic Regression, Random Forest, and a Bidirectional Long Short-Term Memory (LSTM) neural network.

Logistic Regression: The Logistic Regression model, which was trained using a TfidfVectorizer with bigrams, performed reasonably well on the IMDB dataset. The cross-validation performed within the Logistic Regression ensured that the model's hyperparameters were optimized during training. Logistic regression models are known for their robustness and simplicity, making them suitable for text classification tasks. However, their performance can be limited by their inability to capture non-linear relationships in the data. In this case, the Logistic Regression model achieved an accuracy of approximately 88.74% on the test set.

Random Forest: The Random Forest model, also trained with TF-IDF features including bigrams, provided a more complex approach by leveraging an ensemble of decision trees. Random Forest models are particularly effective in handling high-dimensional data and reducing the risk of overfitting due to their ensemble nature. In this experiment, the Random Forest model with 200 estimators achieved an accuracy of around 85.98%. Although it performed slightly worse than Logistic Regression, it contributed valuable insights into the ensemble's decision-making process, particularly for more complex decision boundaries that logistic regression might miss.

LSTM Neural Network: The Bidirectional LSTM model, which was designed to capture sequential dependencies in text, showed significant promise. The LSTM model was built with an embedding layer, followed by a bidirectional LSTM layer, a global max pooling layer, and a fully connected layer with dropout for regularization. LSTMs are particularly well-suited for NLP tasks because they can retain important contextual information over long sequences. The model was trained for 100 epochs with a batch size of 128. The LSTM model achieved an accuracy of approximately 86%. While LSTMs typically excel in handling sequential data, the limited training time and relatively small architecture may have constrained its performance in this experiment.

DOI: https://doi.org/10.34010/injiiscom.v6i2.13921
p-ISSN 2810-0670 e-ISSN 2775-5584

4.2. Ensemble Model Performance

The primary focus of this study was to evaluate the effectiveness of an ensemble learning approach that combines predictions from the three individual models: Logistic Regression, Random Forest, and LSTM. The ensemble model used majority voting to aggregate the predictions from each of the base models.

The ensemble model achieved an overall accuracy of 89.2%, outperforming all three individual models. This result underscores the effectiveness of ensemble learning, as it leverages the strengths of each model while mitigating their weaknesses. By combining models that make different types of errors, the ensemble approach enhances the robustness and generalization of the final prediction.

4.3. Comparison with Other Approaches

The performance of the ensemble model in this study is competitive with other state-of-the-art approaches for sentiment classification on the IMDB dataset as shown in Table 1.

Table 1 Comparison Our Proposed Ensemble Model with Other Approaches

state-of-the-art approaches	**Model/Method**	**Accuracy**
(Ranjan et al., 2023)	Naive Bayes, Machine Learning	86.1%
(Bhowmic et al., 2024)	Ensemble Learning, Classifier Combination	87%
(Mutinda, 2024)	Lexicon-based + Machine Learning	88.64%
(Danyal et al., 2024)	BERT, XLNet	90%
(Saad et al., 2024)	Transformer-based Deep Learning	95.02 %
(Jain & Agarwal, 2023)	SVM, Twin SVM, Naïve Bayes	86%
(Sultana et al., 2024)	RNN, CNN	86.46%
(Hussain & Naseer, 2024)	Logistic Regression, LSTM, Bi-LSTM	87.65%
(Vanlalnunpuia, 2023)	Hybrid Model with Feature Selection, Ensemble Learning	88%
(Krishna et al., 2024)	Synonym Augmentation + Global Average Pooling + Deep Learning	89%
Proposed ensemble model	Ensemble learning	89.2%

Traditional Machine Learning: Models such as Support Vector Machines (SVMs) and Naive Bayes classifiers typically achieve accuracies in the range of 84-87% when applied to the IMDB dataset.

Deep Learning: LSTM and CNN-based models, when fine-tuned and

DOI: https://doi.org/10.34010/injiiscom.v6i2.13921
p-ISSN 2810-0670 e-ISSN 2775-5584

trained for longer periods, can achieve accuracies in the range of 87-89%.

Pre-trained Language Models: More advanced approaches using pre-trained models like BERT have been shown to achieve accuracies 90% on the IMDB dataset. However, these models require significantly more computational resources and training time.

While the ensemble model in this study does not reach the performance levels of pre-trained language models, it offers a balance between simplicity and accuracy. The combination of traditional machine learning models with a neural network provides a strong baseline for sentiment analysis without the need for extensive computational resources.

4.4. Future Work

The results of this study suggest several avenues for future research:

Incorporation of Pre-trained Models: Future work could explore the use of pre-trained language models like BERT, GPT, or Transformer-based architectures in the ensemble. These models have demonstrated superior performance in NLP tasks due to their ability to capture deep semantic understanding.

Advanced Ensemble Techniques: While majority voting was used in this study, more advanced ensemble techniques, such as stacking or blending, could be explored. These methods could involve training a meta-model to learn the best way to combine the base models' predictions.

Hyperparameter Optimization: Further hyperparameter tuning using techniques such as grid search or Bayesian optimization could yield better-performing models.

Handling Ambiguities: Developing techniques to better handle ambiguous language, sarcasm, or subtle sentiment could improve the model's performance on challenging cases.

5. CONCLUSION

In the experiment, we have shown that sentiment classification on IMDB movie reviews dataset can improved using ensembling with multiple hypothesis algorithms. LSTM model was complimented by traditional machine learning models such as logistic regression, random forest for better classification accuracy. The logistic regression model gave excellent linear decision boundaries and random forest adds some much-needed robustness by drawing multiple partitions of the space using trees. The LSTM model performed really well capturing sequential dependencies in text data (which is fundamental to sentiment analysis).

It has given ensemble model 89.2% testing accuracy which is a little better than individual models. Together these results indicate that ensembling models can achieve better predictions when the different trained models represent differing regions of parameter space over which likelihood is not well confined. In the future, one could look into better ensemble techniques like stacking or use pre-trained language models to see if they can further improve. You can also improve this result more by just tuning hyperparameters and building a bigger neural network. In sum, this work emphasizes the effective uses of ensemble learning in NLP tasks and a blend of various modeling methodology for

DOI: https://doi.org/10.34010/injiiscom.v6i2.13921
p-ISSN 2810-0670 e-ISSN 2775-5584

artificial neural network which track faster way to achieve larger model performances.

REFERENCES

Aliman, G., Nivera, T. F. S., Olazo, J. C. A., Ramos, D. J. P., Sanchez, C. D. B., Amado, T. M., Arago, N. M., Jorda Jr, R. L., Virrey, G. C., & Valenzuela, I. C. (2022). Sentiment analysis using logistic regression. *Journal of Computational Innovations and Engineering Applications*, 7(1), 35–40.

Bhowmic, A., Parua, S., Mandal, D., Ghosh, B. K., & Karmakar, R. (2024). Leveraging Ensemble Learning for Enhanced Sentiment Analysis. *2024 International Conference on Distributed Computing and Optimization Techniques (ICDCOT)*, 1–6.

Danyal, M. M., Khan, S. S., Khan, M., Ullah, S., Mehmood, F., & Ali, I. (2024). Proposing sentiment analysis model based on BERT and XLNet for movie reviews. *Multimedia Tools and Applications*, 1–25.

Gupta, S., & Noliya, A. (2024). URL-Based Sentiment Analysis of Product Reviews Using LSTM and GRU. *Procedia Computer Science*, 235, 1814–1823.

Hussain, M., & Naseer, M. (2024). Comparative Analysis of Logistic Regression, LSTM, and Bi-LSTM Models for Sentiment Analysis on IMDB Movie Reviews. *Journal of Artificial Intelligence and Computing*, 2(1), 1–8.

Jain, P., & Agarwal, R. (2023). Sentiment Analysis using SVM, Twin SVM, and Naïve Bayes. *2023 5th International Conference on Advances in Computing, Communication Control and Networking (ICAC3N)*, 456–459.

Karthika, P., Murugeswari, R., & Manoranjithem, R. (2019). Sentiment analysis of social media network using random forest algorithm. *2019 IEEE International Conference on Intelligent Techniques in Control, Optimization and Signal Processing (INCOS)*, 1–5.

Khujaev, O. K., Nurmetova, B. B., & Urazmatov, T. K. (2023). Algorithms for Selecting the Most Efficient Method for Solving Classification Problems. 2023 IEEE XVI *International Scientific and Technical Conference Actual Problems of Electronic Instrument Engineering (APEIE)*, 1740–1743.

Krishna, A. R., Vardhini, A. S., Singh, R. P., & Kanchan, S. (2024). Efficient Sentiment Analysis on IMDb Movie Reviews with Synonym Augmentation and Global Average Pooling. *2024 3rd International Conference on Artificial Intelligence for Internet of Things (AIIoT)*, 1–6.

DOI: https://doi.org/10.34010/injiiscom.v6i2.13921
p-ISSN 2810-0670 e-ISSN 2775-5584

Liu, X.-Y., Zhang, K.-Q., Fiumara, G., Meo, P. De, & Ficara, A. (2024). Adaptive Evolutionary Computing Ensemble Learning Model for Sentiment Analysis. *Applied Sciences*, 14(15), 6802.

Mercha, E. M., Benbrahim, H., & Erradi, M. (2024). Heterogeneous text graph for comprehensive multilingual sentiment analysis: capturing short-and long-distance semantics. *PeerJ Computer Science*, 10.

Mutinda, J. S. (2024). Sentiment Lexicon-Augmented Text Representation Model for Social Media Text Sentiment Analysis. JKUAT-IEET.

Polikar, R. (2012). Ensemble learning. Ensemble Machine Learning: *Methods and Applications,* 1–34.

Ranjan, S., Agrawal, C., & Rai, D. (2023). An Approach Based on Machine Learning to Analyze the Movie Reviews Using the IMDB Dataset.

Saad, T. B., Ahmed, M., Ahmed, B., & Sazan, S. A. (2024). A Novel Transformer Based Deep Learning Approach of Sentiment Analysis for Movie Reviews. 2024 6th *International Conference on Electrical Engineering and Information & Communication Technology (ICEEICT)*, 1228–1233.

Shah, B. (2024). Detecting Mental Distress: A Comprehensive Analysis of Online Discourses Via ML and NLP.

Soumya, B. J., Swetha, B. N., Meeradevi, A., Kanavalli, A., Shubhangi, Singh, Y., Anuritha, L., & Singh, A. (2022). Sentiment Analysis on Real-Time Twitter Data Using LSTM with Mutually Inclusive Classifiers. *International Conference on Information and Communication Technology for Competitive Strategies*, 389–405.

Sultana, A., Hasan, M., Shidujaman, M., & Premachandra, C. (2024). Sentiment Analysis with Deep Learning Methods for Performance Assessment and Comparison. *2024 International Conference on Image Processing and Robotics (ICIPRoB)*, 1–6.

Vanlalnunpuia, C. (2023). Hybrid Feature Selection and Ensemble Classifier with Optimization for Sentiment Classification. *2023 5th International Conference on Advances in Computing, Communication Control and Networking (ICAC3N)*, 1022–1028.

Wu, P., Li, X., Ling, C., Ding, S., & Shen, S. (2021). Sentiment classification using attention mechanism and bidirectional long short-term memory network. *Applied Soft Computing*, 112, 107792.

DOI: https://doi.org/10.34010/injiiscom.v6i2.13921
p-ISSN 2810-0670 e-ISSN 2775-5584