



A Hybrid Machine Learning and Deep Learning-Based System for Text Sentiment Analysis of Movie Review Classification

Praveen Sharma^{1*}, Divyarth Rai²

¹Research Scholar, Dept. of CSE, LNCT University, Bhopal, India

²Department of CSE, LNCT University, Bhopal, India

¹p2raveen@gmail.com

²divyarthrai@gmail.com

Abstract. Sentiment Analysis is a significant NLP task that focuses on identifying polarity (sentiment) and intention for an identifiable piece of textual data. Movie reviews are important in determining whether the movie is successful or unsuccessful. To analyze this unstructured data, we employ newly developed technological algorithms. Machine learning is a transformative technology that is used in practically every industry, with increasingly sophisticated algorithms that provide better, perfect outcomes. Previous research used NLP and ML to extract information from customer reviews. The IMDB 50k movie review dataset is used in this work to perform text sentiment analysis (SA) and assess the model's performance using hybrid ML and DL approaches. The study focuses on two types of models: the utilizes a hybrid technique combining XGBoost and CNN, also single CNN model. Performance indicators like as f1-score, accuracy, precision, and AUC offer a thorough picture of each model's performance. Experimental results show that CNN model get 94.21% and hybrid (XGBoost and CNN) achieved 96.59%, respectively, in text SA of IMDB 50k movie reviews. Researchers may use this study to find the optimal method for SA. Hybrid algorithms outperform single models according to accuracy and efficiency when compared to both the current and suggested methods.

Keywords: Sentiment Analysis, NLP, Machine Learning, Deep Learning, XGBoost, CNN, IMDB 50k.

1 Introduction

A new generation of researchers has devoted themselves to sentiment analysis in response to the rising tide of online social networks and societies built around electronic media [1]. An analysis of people's opinions and sentiments on topics, goods, services, organizations, individuals, problems, events, and all the characteristics connected to them is referred to as opinion mining or SA [2]. The process of applying NLP to obtain semantic quantification of the data under investigation through text evaluation is called opinion mining, or SA [3]. The

enormous development of records and the amount of data generated and exchanged every two days have greatly increased the need to comprehend, mine, and investigate this reality. Consequently, DL took over from traditional ML and neural networks as the primary means of acquiring this type of massive data. Consumers use reviews to determine if a movie is worth seeing. Finding the summary of every review for a film might save users time and effort and enable them to make an educated decision. To assist moviegoers, decide if a film is worth time, critics frequently use review sites that assign ratings to films. Based on criticisms, sentiment analysis can deduce the critics' mindset. By analyzing the review's tone, one may determine the review's overall quality and whether the review is positive or negative. Users may now submit direct reviews of things, corporations, etc., owing to technological improvements [4].

The quantity of textual information available online has skyrocketed due to the widespread use of both fixed and mobile Internet connections. As an example, the IMDB dataset includes user reviews of various films. Assuming the reviews are good, everyone wants to watch the movie. As a result, there is a current uptick in efforts to dissect these evaluations and make error-free sentiment predictions [5]. Conversational models are an essential part of AI, ML, DL, and NLP systems. Computer vision [6]. The use of deep learning methods has led to impressive results on several natural language processing jobs as of late. When analyzing text sentiment, most SA models turn to ML and deep learning techniques [7]. Sentiment analysis is research that uses consumer feedback to categorize text, which can help businesses improve. Analyses were previously conducted utilizing NLP and ML based on customer-provided data [8][9]. In this research, SA to the IMDB movie reviews dataset and evaluate the model's performance using ML and DL techniques. In contrast to deep machine learning techniques, which operate on several layers to provide superior results, typical ML algorithms only use one. To find the most effective algorithm for SA, this article is useful for academics. When compared to deep learning and ML, hybrid algorithms provide more effective and efficient outcomes.

1.1 Contribution of this Work

The goal of this study was to develop a machine-deep learning-based system that could effectively analyze text sentiment. The primary contribution that fulfills in this study are as follow:

- To collect the dataset for analysis and classification of the particular methods and techniques.
- To perform pre-processing for select best features and remove null values and data balancing.
- To explore, analyze and study existing sentiment analysis detection techniques based on text classification.
- To elevate the model performance with the performance parameter.

- to enhance the efficiency of classification in sentiment analysis by employing various machine-deep approaches. This will be accomplished by identifying sentiment from the text by utilizing previous knowledge.

The paper is divided into five primary components. In Section 2 by analyzing the literature and assessing how well the models performed on similar tasks. Section 3 offer an overview of the study's research strategy and procedures. Section 4 presents the study's findings and analysis. The study is concluded and its practical implications for further research are discussed in Section 5.

2 Literature Survey

In this part, cover some of the work that has already been done, as well as the advantages of suggested paradigm.

Gohil et al. (2023) examined and contrasted many ML methods, such as ensemble models, logistic regression (LR), XGBoost, Grid SearchCV, and Gradient Boosting (GB)Classifier. They proved that their ensemble model was superior to ML algorithms like XGBoost and GB Classifier with an accuracy of 85% [10]. Demir and Bilgin (2023) used BERT-based language models conduct SA on the data they extracted from Turkish news texts. Machine learning methods that were utilized comprised NBs, RF, and SVM. The Roberta, DistilBERT, and ALBERT language models all get success rates of 80%, 77%, and 80%, respectively. With scores of 71%, 68%, and 68%, respectively, RF, NBs, and SVM round out list of ML methods [11]. Başa and Basarslan (2023) the Internet Movie Database (IMDb) dataset, the frequency-based TF-IDF approach was used to produce word vectorization. ML methods were then utilized for classification. SVM was found to be the most successful with an accuracy of 90% [12]. Kumar and Uddin Haider (2023) used ensemble learning techniques, SVM and LSTM due to the low sentiment prediction accuracy of the lexicon-based methodology. Ensemble learning, in particular LSTM with SVM, outperforms other ML algorithms according to acc and ROC-AUC. With a 93% ROC-AUC, the acc is 87% [13]. Rajath et al. (2023) methods used include GB, LR, K-nearest neighbors (KNNs), NB Classifier, Decision Tree (DT) Classifier, Bagging Classifier, Ada Boosting classifier, SVM, and Bagging Classifier. At 58% accuracy and F1-scores of 0.62 for LR and 0.57 for SVM, respectively, they discover that these techniques provide the best results. After much deliberation, they decide that their ML approach could provide a reliable and efficient technique for analyzing the tone of tweets [14]. Vinothkumar et al. (2022) evaluate the accuracy of SVM, NBs, and LR, all of which are supervised ML models using the Indian False New identification dataset. The hyper-tuning SVM model is more accurate than the other supervised learning ML models, with a 93% accuracy rate [15]. Al-Hagree and Al-Gaphari (2022) study has focused on these mobile applications for SA tasks based on user feedback. ML approaches, specifically the NBs, KNN, DT, and SVM models, are used to accomplish the Arabic SA procedure. The best findings with respect of accuracy (89.65%), rec (88.08%), prec (88.25%), and Fscore (88.25%) were obtained by the NB model [16]. Jayakody and Kumara (2021) used the KNN, count vectorizer,

and TF-IDF methods to transform the words into vectors, and the SVM, logical regression, and KNN machine learning methods to analyze product review-based tweets. "LR with Count Vectorizer" obtained the best accuracy score at 88.26%. genuine positive and genuine negative rates will be calculated at last [17].

Developing reliable systems in the field of SA is limited by several continuing challenges. It is still challenging to acquire high-quality labeled datasets, particularly for domains and languages other than English, and correct sentiment interpretation is complicated by the contextual sensitivity and intrinsic ambiguity of language. While aspect-based sentiment analysis necessitates advanced methods for detecting and evaluating sentiment towards certain parts of text, multilingual analysis adds complexity owing to cultural variances and linguistic variations. Also, models have a hard time picking up on figurative language, irony, and sarcasm, and there are ethical challenges when trying to adapt to new domains while still being fair and transparent. The difficulty of understanding the logical structure hidden in the DL models decreases the chance of realizing their decision-making processes. Researchers of other domains must form ties, spearhead the development of new computational methodologies, and recognize the intricacies of cultures and languages if they are to cope with these difficulties, move the area of sentiment analysis forward, and hallmark systems that are more dependable.

3 Research Methodology

The recommended method for analyzing mood of reviews utilizes a fusion of DL and ML techniques. ML-based text SA on a dataset of movie reviews is depicted in Figure 1, which is the first of its kind. The whole methodology and flowchart process described below very briefly:

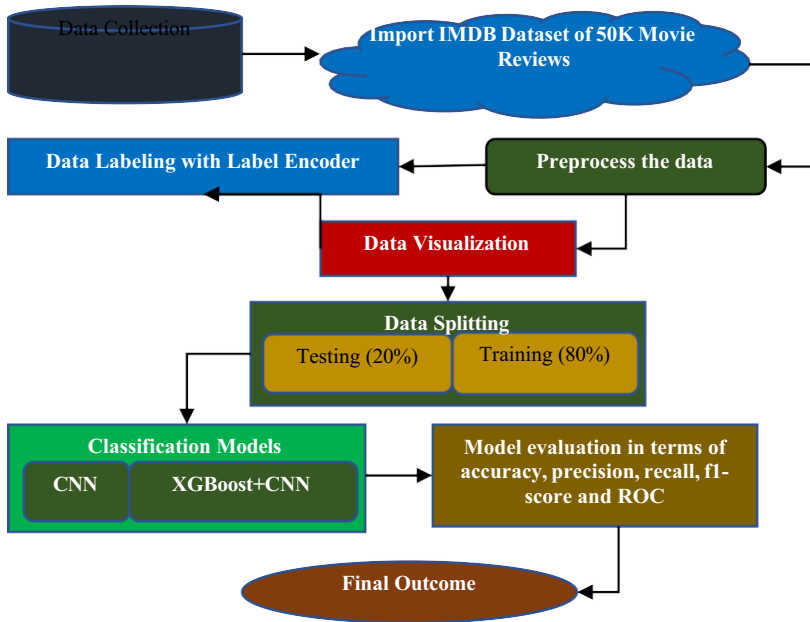


Fig. 1. Flow Chart for Proposed Methodology

3.1 Data Collection and Preprocessing

This research made use of the IMDB Dataset of 50,000 Movie Reviews, which is housed in the Kaggle repository. 50K movie reviews from IMDB for use in NLP and text analytics. After collecting dataset have applied data preprocessing techniques for the data cleaning and prepare for the models training. Here is a rundown of the preprocessing methods used for this research:

- **Remove URL:** Textual references to URLs in a document don't give any more information; they only point to specific locations on the web. Consequently, we eliminate them as well by making use of the mentioned library, which offers regular expression matching operations.
- **Remove Punctuation:** During this step, we normalized the data and eliminated all punctuation from the text. The resulting text is condensed and made easier to understand. The document was stripped of any punctuation.
- **Removal of Stop Words:** Repeated appearances of words are common in text files. So, getting rid of the stop words is a key priority. The use of stop words never adds value to anything written. Countless instances of these terms may be found throughout the book.
- **Tokenization:** Locating non-fragmentable tokens or basic units is the initial stage in natural language processing (NLP). Tokens are the name given to these smaller units. Any combination of letters, numbers, and punctuation marks might constitute a token.

- **Noise removal:** By filtering out irrelevant data, we improved the dataset. Symbols and letters that are not Bangla, as well as emoticons and other non-Bangla characters, are regarded as noise.
- **Label Encoding (LE):** One easy way to describe nominal qualities as class-transferable continuous integers is via Label Encoding (LE).

3.2 Data Splitting

In order to maintain class distribution, the algorithm does a Stratified Fold cross-validation on a dataset. For an 80-20 split between train-validation and test-validation, it uses `train_test_split` within each fold.

3.3 Classification Models

The goal of this research is to create an ML and DL model that can correctly predict text SA. There are several approaches that may be used for text classification. The following techniques have been selected for categorization based on efficacy. This study uses a hybrid model of CNN and XGBoost techniques for classification, which are described below:

CNN Model. The input, convolutional, pooling, full connection, and output layers make up the CNN, a kind of FFNN. The convolutional and pooling layers alternate positions periodically [18]. CNNs analyze data using one or more feature planes. Neurons on the same data plane share weights, while each neuron on a feature plane has a unique pattern. The pooling layer is added after the convolutional layer to decrease the dimensionality of the input data and accelerate algorithm training. The second is to get rid of extraneous characteristics and prevent overfitting the network. Every neuron in the layer below it is coupled to every other neuron in the connection layer. All of the local characteristics that were acquired in the previous layer may be integrated throughout the whole connection layer to create the overall features. The output layer receives the activation functions utilized by every neuron in the whole connection layer.

XGBoost model. XGBoost algorithm was proposed by [19] according to the GBDT framework. Due to its outstanding results in Kaggle's ML contests, it has garnered a lot of attention. In contrast to the GBDT, XGBoost mitigates overfitting by incorporating a regularization component into the objective function.

Hybrid model of XGBoost and CNN. One strong DL model is the CNN, which learns and extracts high-quality data fast at each layer thanks to its hierarchical structure. With the help of the model's local sensing fields and the weight transfer and pooling operation, may simplify the network architecture and decrease the number of variables. As an outcome, it has been extensively utilized for image classification

issues and has produced excellent outcomes. When using CNN with traditional classifiers, the retrieved characteristics are misunderstood. At the same time, XGBoost will get over the problems mentioned since it is a gradient-boosting integrated learning method that effectively achieves better classification accuracy [20]. For hybrid CNN-XGBoost model, exchanged out the CNN model's last fully connected layer for an XGBoost classifier. The calculated possibilities for the input sample are the output units in the CNN's final layer. It takes into account the likelihood of each output using an activation function. The activation function receives the bias term as an input after it has been added to the outputs of the hidden layer with trainable weights and one additional term. Although the CNN network does make sense when looking at the hidden layers' output values, these values may be used as features by other classifiers.

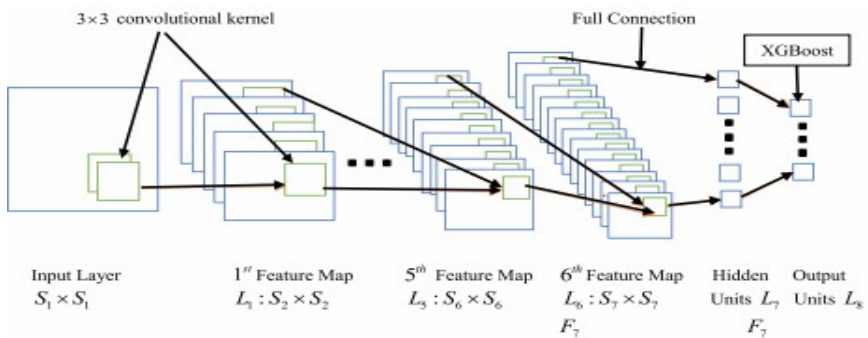


Fig. 2. Hybrid CNN-XGBoost model

Figure 2 illustrates the framework of the recently created hybrid CNN-XGBoost model. The initial step is to feed the input layer with decentered and normalised pictures. Once the training procedure merges or the training error is resolved, the first in-sequence CNN model with a fully linked output layer is learned. Tweaking period follows in which we trim the last FC layer. Similarly, the flat model from the first dense layer is transferred to the XGBoost classifier to make the next layer. The summary of the model identifies the thick layer by name. By passing the name of the dense layer as an argument to the `model.get_layer()` function, we can obtain its trained layers: `(new_model = model.get_layer('dense_n').output)`. Training, validation, and testing datasets are forecasted using this trained model. The XGBoost model is fitted with an input feature vector that contains the predicted values from both the training and validation datasets. After 100 cycles of training, the XGBoost model is ready to go. The purpose was to avoid overfitting, hence 70 early stopping rounds were specified. Categorization reports also make use of the predicted testing dataset.

capacity to reflect realism emotional can be as certain as possible. Thus, it an integral part of the model assessment on four instances.

- **True Positive (TP):** Despite the model's negative prediction, the instance turned out to be positive.
- **True Negative (TN):** When the scenario corresponded exactly to the way the model had imagined it would be—negative.
- **False Positive (FP):** In the example, the situation was bad, although its future was estimated to be good.
- **False Negative (FN):** However, the model predicted a negative outcome which in the end turned out to be a positive one.

Accuracy (Acc): Accuracy is the number of accurately predicted classes out of all of them. It gives as Equation (1):

(1)

Precision (Prec): Accuracy is the ratio of "TP" to the sum of "TP" and "FP," or the quotient of the two. It alludes to how accurate the optimistic prediction is, in simple terms. Equation (2) represent precision:

(2)

Recall (Rec): The ratio that is derived by splitting the total number of 'True Positives' by the sum of 'TP' and 'FN' is precise. It displays the proportion of correctly classified positive instances. The Formula of recall in Equation (3):

(3)

F1-Score: The combination and evaluation of prec and rec may be achieved via the use of the F1score. The harmonic means of prec and rec is calculated in Equation (4).

(4)

ROC-AUC: One measure of separability is the AUC, while the other is the probability curve (ROC). This metric shows the model's class discrimination capability.

Training and Validation Loss: Loss functions are used in training and validation to quantify the difference between the values in a dataset that are expected to be there and the actual values. Loss is calculated at each training iteration in the model-training framework and used to modify the model's parameters to minimize mistakes. While the training loss shows the mistake on the training dataset, the validation loss quantifies the error on a separate validation dataset.

4.3 Experimental Results

This section provides a discussion of the outcomes derived from the first proposed models. The proposal involves using CNN and XGBoost models to assess the sentiment inside textual data.

Table 1. PERFORMANCE OF THE CNN CLASSIFIER

Parameters	Training Accuracy	Training Loss	Testing Accuracy	Testing Loss
CNN	95.83	12.85	94.21	17.42

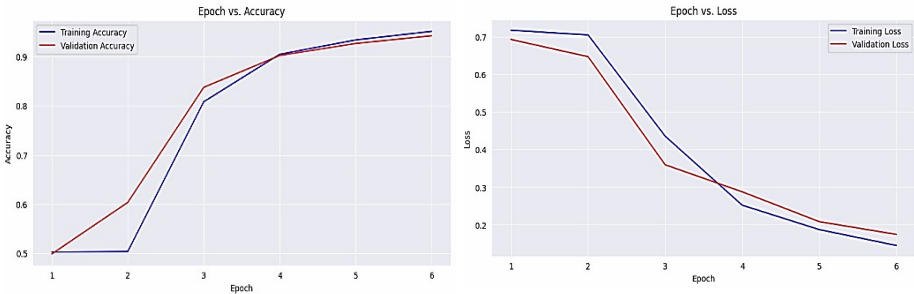


Fig. 4. Training and Validation Accuracy/Loss Graph

Figure 4, displays a line graph that illustrates an acc of a training and validation data. The acc score of the recommended CNN is displays on a y-axis, while an x-axis shows the number of epochs, ranging from 0 to 6. The model achieves 95.83% training accuracy and 94.21% testing accuracy. Also, another line graph shows the training and validation losses. The acc score of the suggested CNN is displays on a y-axis, while an x-axis represents the epochs from 0 to 6. The model shows a similar training loss of 12.85% and testing loss of 17.42%

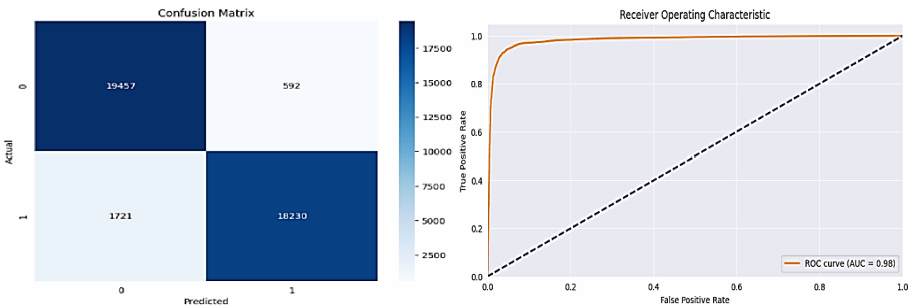


Fig. 5. Confusion Matrix and ROC curve of CNN Classifier

Figure 5 shows the confusion matrix demonstrates how well the CNN classifier performs. The confusion matrix accurately predicts 19457 negative reviews for class 0 and 18230 positive feedback for class 1. The FP rate is shown on an x-axis and the

TP rate on a y-axis of the ROC curve. In the figure, CNN model gets an acc of 94%, respectively.

```

Classification Report:
                precision    recall  f1-score   support

   negative      0.96      0.97      0.97     20049
   positive      0.97      0.96      0.97     19951

 accuracy              0.97     40000
 macro avg              0.97     40000
 weighted avg          0.97     40000
    
```

Fig. 6. Hybrid XGBoost +CNN Classifier Classification Report

Figure 6 shows the classification report for the proposed hybrid XGBoost+CNN classifier, demonstrating its efficacy in sentiment analysis. As for the negative class in the classification report, the suggested hybrid model achieves 97% recall, 96% accuracy, and 97% f1-score. The accuracy, rec, and f1score for the positive class are 93%, 89%, and 91%, correspondingly.

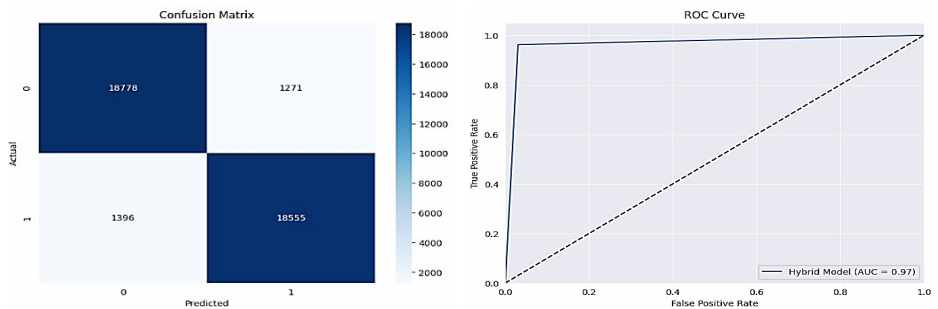


Fig. 7. Confusion Matrix and ROC of Hybrid XGBoost +CNN Classifier

The efficacy of the hybrid XGBoost+CNN classifier in SA is seen in Figure 7, which displays the confusion matrix. The confusion matrix accurately predicts 18778 negative reviews for class 0 and 18555 positive reviews for class 1. The ROC curve in Figure 7 illustrates the performance of the hybrid XGBoost +CNN classifier throughout the testing phase for text sentiment classification. The ROC curve shows the TPrate on a y-axis and the FPrate on an x-axis. The graphic shows that the hybrid model achieves an accuracy of 97%.

4.4 Comparison of Proposed Models with Base Models and Discussion

This section compares the basic model with the suggested models in order to determine which model is best for SA categorization; the comparison is displayed in Table 2.

Table 2. COMPARISON OF THE BASE AND PROPOSED MODELS

	LSTM	GRU	CNN	Hybrid XGBoost+CNN
Training Accuracy	91.13	85.32	95.83	98.17
Testing Accuracy	87.63	85.21	94.21	96.59
AUC	96.00	93	98.00	83.45

Table 2 shows that the proposed Hybrid XGBoost + CNN model outperforms the baseline LSTM, GRU, and CNN models, achieving the highest training (98.17%) and testing accuracy (96.59%). While CNN captures local text features effectively, combining it with XGBoost enhances robustness and handles complex feature interactions. The main advantage of the hybrid approach is its ability to leverage deep learning for feature extraction and gradient boosting for accurate classification, resulting in more reliable sentiment prediction.

5 Conclusion and Future Scope

This paper proposes a complete framework for analyzing sentiment in movie reviews that smoothly integrates ML and DL techniques. The dataset underwent substantial preprocessing, including text cleaning and structure for analysis. Two models were created: a CNN and a hybrid model that combines XGBoost with CNN. The CNN model has the remarkable training accuracy (95.83%), as well as the testing accuracy (94.21%), and its prec, rec, and F1-score values are ranging from 91% to 97%. The hybrid XGBoost+ CNN model surpassed it in all the parameters. It achieved training and testing acc of 98.17% & 96.59%, respectively, with prec, rec, and the F1scores in a range between 89% and 97%. The research results above compared favorably to other comparable models like LSTM and GRU, thus exhibiting how machine learning has been combined with deep learning algorithms to better call performances. In addition, the models showed profound ROC-AUC values which signify the ability for negative and positive classification. The methodology we proposed for the sentiment analysis of movie reviews showed satisfactory performance considering it's washed up through multiple evaluation parameters.

The limitation of the proposed sentiment analysis is its extremely data-driven/model-dependent nature and, hence, it may limit its functionality and feasibility in different scenarios. The IMDB dataset was not only main data set for training and assessment but also, we have to integrate other models which had less coverage over other sources and domains. In order to overcome these constraints, it would be necessary to investigate a variety of datasets, improve model designs, and maximize the combination of several approaches to improve overall performance and flexibility in practical applications.

References

- [1] J. Singh, G. Singh, and R. Singh, "Optimization of sentiment analysis using machine

- learning classifiers,” *Human-centric Comput. Inf. Sci.*, 2017, doi: 10.1186/s13673-017-0116-3.
- [2] J. Zhao, K. Liu, and L. Xu, “Sentiment Analysis: Mining Opinions, Sentiments, and Emotions,” *Comput. Linguist.*, vol. 42, no. 3, pp. 595–598, Sep. 2016, doi: 10.1162/COLI_r_00259.
- [3] B. L. Devi, V. V. Bai, S. Ramasubbareddy, and K. Govinda, “Sentiment Analysis on Movie Reviews,” in *Advances in Intelligent Systems and Computing*, 2020, pp. 321–328. doi: 10.1007/978-981-15-0135-7_31.
- [4] R. Kumar and A. Benitta, “IMDB Movie Reviews Sentiment Classification Using Deep Learning,” 2022.
- [5] O. Habimana, Y. Li, R. Li, X. Gu, and G. Yu, “Sentiment analysis using deep learning approaches: an overview,” *Sci. China Inf. Sci.*, vol. 63, no. 1, Jan. 2020, doi: 10.1007/s11432-018-9941-6.
- [6] R. Vidhya, P. Gopalakrishnan, and N. Vallamkondu, “Sentiment Analysis Using Machine Learning Classifiers: Evaluation of Performance,” in *Proceedings of the First International Conference on Computing, Communication and Control System, I3CAC 2021, 7-8 June 2021, Bharath University, Chennai, India*, EAI, 2021. doi: 10.4108/eai.7-6-2021.2308565.
- [7] M. M and S. Mehla, “Sentiment Analysis of Movie Reviews using Machine Learning Classifiers,” *Int. J. Comput. Appl.*, vol. 182, no. 50, pp. 25–28, Apr. 2019, doi: 10.5120/ijca2019918756.
- [8] K. Amulya, S. B. Swathi, P. Kamakshi, and Y. Bhavani, “Sentiment Analysis on IMDB Movie Reviews using Machine Learning and Deep Learning Algorithms,” in *2022 4th International Conference on Smart Systems and Inventive Technology (ICSSIT)*, IEEE, Jan. 2022, pp. 814–819. doi: 10.1109/ICSSIT53264.2022.9716550.
- [9] A. Samyuktha, M. Pallavi, L. Jagan, D. . Y. Srinivasulu, and M. M. Rakesh, “Sentiment Analysis Using Machine Learning,” *Int. J. Res. Appl. Sci. Eng. Technol.*, 2023, doi: 10.22214/ijraset.2023.48706.
- [10] R. Gohil, S. Deepa, M. Vinay, and J. Jayapriya, “Election Forecasting with Machine Learning and Sentiment Analysis: Karnataka 2023,” in *IEEE 1st International Conference on Ambient Intelligence, Knowledge Informatics and Industrial Electronics, AIKIIIE 2023*, 2023. doi: 10.1109/AIKIIE60097.2023.10390333.
- [11] E. Demir and M. Bilgin, “Sentiment Analysis from Turkish News Texts with BERT-Based Language Models and Machine Learning Algorithms,” in *2023 8th International Conference on Computer Science and Engineering (UBMK)*, 2023, pp. 1–4. doi: 10.1109/UBMK59864.2023.10286719.
- [12] S. N. Başa and M. S. Basarslan, “Sentiment Analysis Using Machine Learning Techniques on IMDB Dataset,” in *2023 7th International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT)*, 2023, pp. 1–5. doi: 10.1109/ISMSIT58785.2023.10304923.
- [13] S. Kumar and M. T. Uddin Haider, “Sentiment Analysis of Political Party Twitter Data using Ensemble Learning Classifier,” in *2023 14th International Conference on Computing Communication and Networking Technologies (ICCCNT)*, 2023, pp. 1–5. doi: 10.1109/ICCCNT56998.2023.10308149.
- [14] N. Rajath, V. Kushal, A. Kodipalli, T. Rao, V. Pushpalatha, and B. R. Rohini,

- “Sentiment Analysis of Twitter Using Machine Learning Algorithms,” in *2023 International Conference on Computational Intelligence for Information, Security and Communication Applications (CIISCA)*, 2023, pp. 250–254. doi: 10.1109/CIISCA59740.2023.00056.
- [15] S. Vinothkumar, S. Varadhaganapathy, M. Ramalingam, D. Ramkishore, S. Rithik, and K. P. Tharanies, “Fake News Detection Using SVM Algorithm in Machine Learning,” in *2022 International Conference on Computer Communication and Informatics, ICCCI 2022*, 2022. doi: 10.1109/ICCCI54379.2022.9740886.
- [16] S. Al-Hagree and G. Al-Gaphari, “Arabic Sentiment Analysis Based Machine Learning for Measuring User Satisfaction with Banking Services’ Mobile Applications : Comparative Study,” in *2022 2nd International Conference on Emerging Smart Technologies and Applications, eSmarTA 2022*, 2022. doi: 10.1109/eSmarTA56775.2022.9935486.
- [17] J. P. U. S. D. Jayakody and B. T. G. S. Kumara, “Sentiment analysis on product reviews on twitter using Machine Learning Approaches,” in *2021 International Conference on Decision Aid Sciences and Application, DASA 2021*, 2021. doi: 10.1109/DASA53625.2021.9682291.
- [18] C. Wigington, S. Stewart, B. Davis, B. Barrett, B. Price, and S. Cohen, “Data Augmentation for Recognition of Handwritten Words and Lines Using a CNN-LSTM Network,” in *Proceedings of the International Conference on Document Analysis and Recognition, ICDAR*, 2017. doi: 10.1109/ICDAR.2017.110.
- [19] T. Chen and C. Guestrin, “XGBoost: A scalable tree boosting system,” in *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2016. doi: 10.1145/2939672.2939785.
- [20] X. Ren, H. Guo, S. Li, S. Wang, and J. Li, “A novel image classification method with CNN-XGBoost model,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2017. doi: 10.1007/978-3-319-64185-0_28.
- [21] I. Ahmed, “A Study Of Heart Disease Diagnosis Using Machine Learning and Data Mining,” California State University, 2022.

Open Access This chapter is licensed under the terms of the Creative Commons Attribution-NonCommercial 4.0 International License (<http://creativecommons.org/licenses/by-nc/4.0/>), which permits any noncommercial use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.

