doi:10.21608/esju.2024.252649.1024

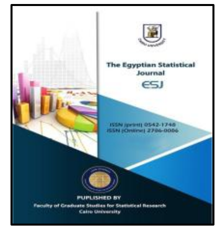
Research Article

The Egyptian Statistical Journal (ESJ), 68 (1): 1-14

Homepage: https://esju.journals.ekb.eg/

The Egyptian Statistical Journal

Print ISSN 0542-1748– Online ISSN 2786-0086



Sentiment analysis for movie recommendations: harnessing opinion mining systems to analyze user reviews

Doha Taha Nor El-Deen[1*], Rania Salah El-Sayed[2], Ali Mohamed Hussein[3], Mervat S. Zaki[4]

Received 11 December 2023; revised 23 January 2024; accepted 2 February 2024

Keywords

Artificial Neural Network; Long Short-Term Memory; Convolutional Neural Networks; Sentiment Analysis.

Abstract

Opinion mining systems now require sentiment analysis because of the massive amounts of data and opinions that are generated, shared, and sent every day through the Internet and other media. The major topic of this study is sentiment analysis for movie recommendations. There are too many reviews and comments to manually process. As a result, to process it successfully, we used user reviews of films (whether they were positive or negative) to create an overall assessment of reviews. A strategy must be developed for extracting knowledge from the existing reviews and applying it more effectively. In this research work, two machine learning approaches are adopted, applied, and tested for the analysis and classification of user reviews. The first approach involves some supervised machine learning algorithms, namely Support Vector Machine (SVM), and Naïve Bayes (NB), which are applied based on feature selection algorithms, namely Term Frequency–Inverse Document Frequency (TF-IDF). The second approach is concerned with presenting a proposed model based on deep learning algorithms such as Artificial Neural Networks (ANN), Convolutional Neural Networks (CNN), and Long Short-Term Memory (LSTM) that are applied based on Word embedding techniques such as Glove that enable deep learning models to capture semantic relationships and contextual information. This enhances the models' ability to understand and analyze textual data. The test results demonstrated that LSTM outperforms other approaches. Despite CNN reporting accuracy better than ANN. Our models outperformed Support Vector Machine and Naive Bayes, as well.

1. Introduction

People's behaviors have started to shift in recent years, particularly for technical and environmental causes. Socialization has largely shifted to social media programs on cell phones following technical advancements. People are now making and sharing videos that reflect their own experiences in addition to the usual images and videos. In restaurants, cities, businesses, and public institutions, people readily express their happiness and appreciation for a movie or a service. In addition to social media platforms like Twitter and Facebook, they also post videos on YouTube channels with their thoughts.

Daily Twitter and Facebook both acquire about 10 TB and 7 TB of data, respectively (D. Chaffe, 2023). New technologies have emerged because of the limitations of old computer systems'

✉ Corresponding author*: doha.taha.1911@gmail.com

[1,3] Misr University for Science and Technology, Giza, Egypt.

[2,4] Faculty of Science (Girls Branch), Al-Azhar University, Cairo, Egypt

processing power and the traditional machine-learning techniques used to analyze these massive volumes of data. Deep learning techniques are becoming more popular for data analysis while cloud computing technologies are becoming more common for higher processing power (W. Tian et al.,2020). The sentiment analysis problem of natural language processing (NLP) is one of the application areas for deep learning approaches to texts.

Sentiment analysis (SA) aids in a better understanding of user opinions on other reviews or social media. Recent years have seen significant advancements in the study of sentiment analysis, opinion mining, and emotion recognition. Artificial neural network technology, which is based on an upgraded version and is inspired by the structure and operation of the human brain, is the foundation of the machine learning technique known as deep learning. It has a significantly higher number of hidden layers than artificial neural networks. Consequently, by developing more intricate networks that resemble the human brain, more thorough and effective learning can be accomplished. Many fields, including image processing (A. T. Kabakus et al., 2022; A. Sevik et al., 2018; A. Alhudhaif et al., 2021; N. Calik et al., 2022), text classification (B. Pang et al., 2004), speech recognition (Y. Zhang et al. .2022; M. F. Mridha et al., 2022), and NLP (S. Meera et al., 2022; K. Chowdhary et al., 2020), use deep learning techniques. Social networks, online discussion boards, blogs and other platforms have had a big impact on daily life lately, especially in terms of how individuals express their thoughts. Most businesses and organizations must extract relevant data from huge unstructured data sets (such as consumer perceptions of company branding). SA is used for more than just rating products or films. It has also been used by researchers in news, politics, and sports. SA can be used to analyze online political discussions to determine people's opinions about a political system (S. Nisar et al., 2019).
Sentiment analysis was done to categorize the sentiments at the document or sentence levels (as positive or negative) (K. Dashtipour et al., 2021). Two techniques for emotion analysis are frequently used: (1) A technique for determining polarity based on the use of lexicons (dictionaries of words and their polarities); (2) Techniques based on learning for huge datasets. However, both traditional and cutting-edge machine learning methods have been widely used for sentiment analysis in the English language.

A lot of machine learning techniques heavily rely on training features, which can be expensive, time-consuming, and require customization. Some machine learning methods also depend on manual dictionaries. Some other lexicographical methods analysis the textual sentiment with the position-based technique of grammar word relations. There are also some methods in machine learning named K-means, NB, and SVM, those approaches use frequency and features for classification. Summary writings contain low sentence validity, non-standard syntactic constructions, and a colloquial tone as compared to traditional terminology. This is also more difficult to correctly determine the polarity of the sentiment. Deep learning methods have recently been proven to surpass cutting-edge machine learning-based categorization thanks to automated feature engineering. To perform sentiment analysis on a database, we employed ANN and deep learning techniques like CNN and LSTM.

This is how the rest of this paper is organized. Section 2 defends other works that have been cited, Section 3 presents proposed strategies for categorizing sentiments, Section 4 shows experimental results, and Section 5 explains the paper's conclusions.

2. Related work

A description of some research that has already been done on SA tasks using various methodologies, depending on the context in which sentiment analysis is used (M. Birjali et al., 2021). It goes by a variety of titles, including aspect-based, opinion extraction, sentiment mining, subjective analysis, and effect analysis. The terms "sentiment analysis" and "idea mining" are interchangeable (T. A. Rana et al., 2016). There is no clear distinction between the phrase's mood, opinion, and impact. We can categorize emotions using machine learning and dictionary-based methods, respectively. Figure 1 shows that the four primary categories of sentiment analysis methodologies are lexicon-based approaches, machine learning approaches and their hybrid applications, and additional approaches (M. Birjali et al., 2021). In our investigation, we used supervised learning. one of the machine learning techniques, alongside deep learning techniques.

In (A. Kamal, 2015), a framework for opinion mining was developed that makes it easier to analyze objectivity or subjectivity, extract features, and summarise reviews at the intersection of both machine learning and natural language processing approaches. The authors classified the subjectiveness and objectivity of reviews using supervised machine learning, and (A. Kamal, 2013), they employed several methods, including NB, Decision Trees, Multilayer Perceptron, and Bagging, for extracting subjective sentences from customer evaluations for mining user opinions and product features at the nexus of rule-based and machine learning techniques. He increased mining efficiency.

Numerous algorithms have been developed by researchers for the analysis of online users' opinions. The classifiers chosen are k-Nearest Neighbour (KNN), RF, and Naive Bayes (NB). With the help of the String To Word Vector filter, the phrase embedding was created. Sentiment Polarity Dataset model 2.0, a study based on the two-magnitude dataset from the IMDB website, was provided by authors (P. Baid et al., 2017).

The authors of (G. K. Pitsilis et al., 2018) developed a system called Classifying Sentiments from Movie Reviews Using Deep Neural Networks 3 that uses a Recurrent Neural Network (RNN) to identify offensive content on the internet. The attributes were obtained by the authors through Twitter statistics, which they then fed into an RNN classifier. The results showed that RNN outperformed SVM in terms of success rate, reaching up to 95.33%. The actual model architecture and supporting hyper-parameters, such as the filter region size, must be specified by experts for these models. To reduce the loss of local information, the authors (A. Hassan et al., 2017) ignored the pooling layer in the convolutional network and replaced it with a single LSTM layer. A single recurrent layer is sufficient to capture long-term dependencies in the ConvLSTM model, ConvLSTM model combines the convolutional and recurrent layers into a single model. The combined CNN and LSTM versions outperformed the individual versions. It reached an accuracy of up to 88.3%.

(D. Nguyen et al., 2018), over 5,000 real articles were collected by You-Net Media in the domain of Internet-Telecom by the authors, who then entered pre-processed embedding features into CNN, LSTM, and a combination of CNN and LSTM to employ a solitary technique for identifying polarity in news articles. In simulations, the CLSTM model outperformed CNN and LSTM, achieving an accuracy trend of up to 96.52 percent as opposed to 91.19 percent and 92.3 percent for LSTM and CNN.

By utilizing deep neural networks to occupy a product, the authors of (S. Liao et al., 2017) offered a mechanism to gauge customer satisfaction. The Twitter API was abused to gather the tweets. The CNN classifier created with this design used one convolutional layer and one pooling layer. According to the researchers, the proposed CNN outperformed SVM and Naive Bayes with an accuracy of 95%, as opposed to 62.25 percent and 70 percent for Naive Bayes and SVM, respectively.

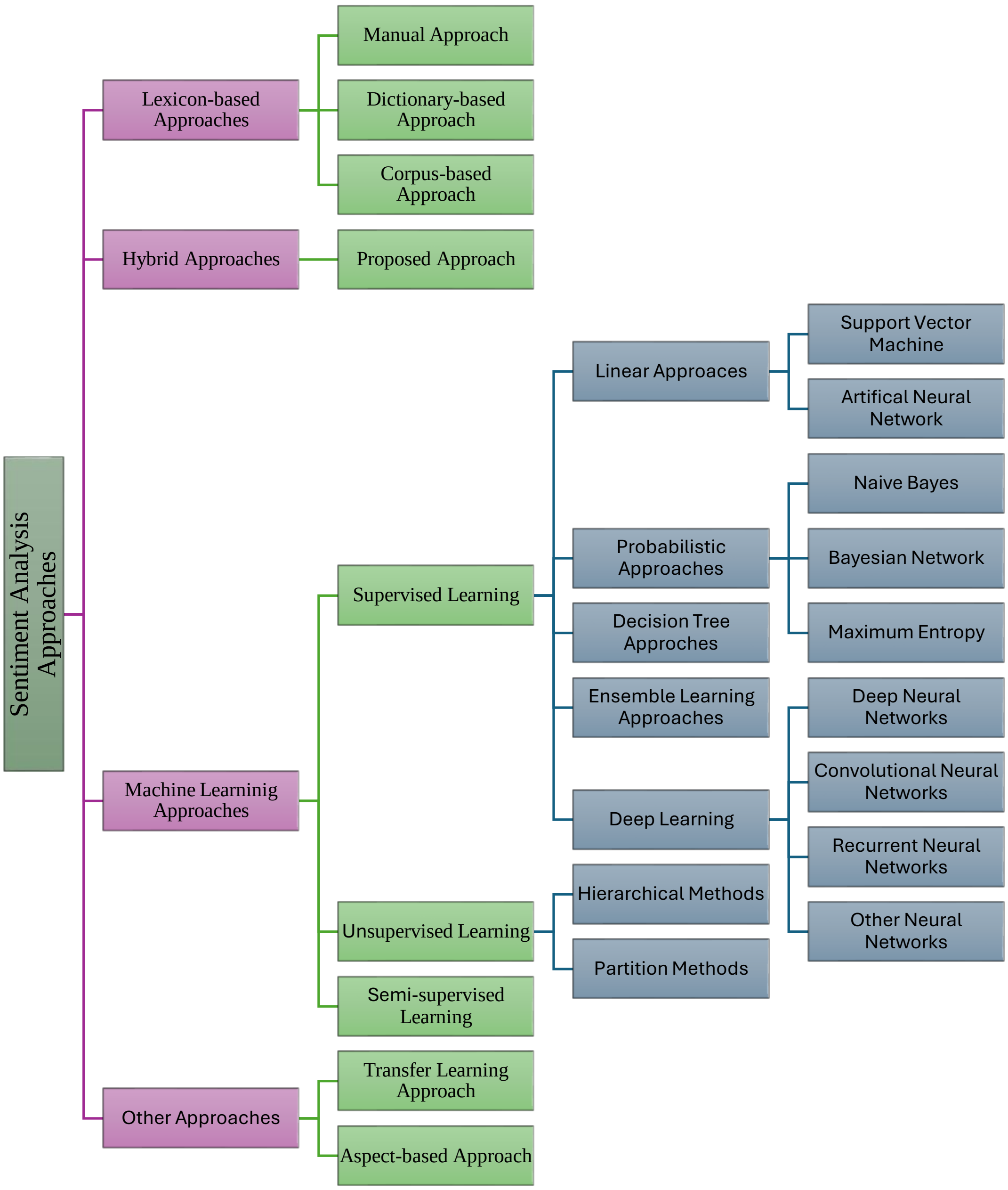


Figure 1: Methods of Sentiment analysis (M. Birjali et al., 2021)

The authors (W. Zhao et al., 2017) suggested a novel deep learning framework for product review sentiment classification which employs prevalently available ratings as weak supervision signals. Weakly-supervised Deep Embedding (WDE) is the name of the framework. They constructed a

dataset containing 1.1M weakly labeled review sentences and 11,754 labeled review sentences from Amazon. WDE-LSTM outperformed WDE-CNN in the testing, achieving up to 87.9% Macro-f1.
Considering that the site of rotten tomatoes was used to gather movie reviews, authors in (X. Ouyang et al., 2015) directed a technique for police work polarity in English film reviews. The 5 classes in the sample were negative, positive, neutral, somewhat negative, and somewhat positive. The RNN achieved an accuracy of 78.34% when compared to SVM. In the study by (R. Andrade-Gonzalez et al., 2021), Twitter comments about Mexican video streaming are extracted, the text is cleaned, and a text mining supervised support model is built to classify the sentiments of the tweets.

3. Methodology

This section introduces the study's dataset and discusses word embedding, preprocessing, and deep learning techniques. Figure 2 shows the steps for categorizing the sentiment of movie reviews.

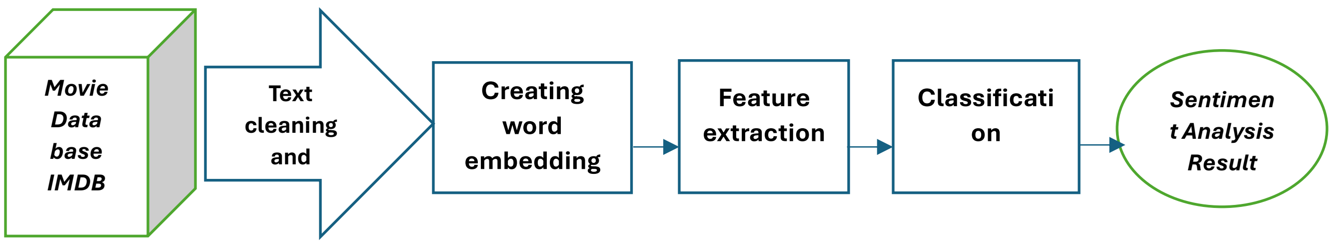


Figure 2: The steps of classifying movie review sentiment.

3.1 Preprocessing

Preparing texts for processing is known as text preprocessing. It entails actions like the elimination of stop words and special letters. It is employed to get the data ready for NLP categorization. Only the most crucial terms are selected from HTML tags, or existing words are cleaned up, and all are then transformed into normalized form to produce better and more trustworthy categorization results.
Preprocessing operation steps:
1. HTML strips were eliminated.
2. Square brackets, URLs, Email addresses, and special characters were eliminated.
3. The omission of numbers.
4. White space was eliminated.
5. Punctuation was eliminated.
6. Tokenization took place.
7. To get rid of word variances, all text data was transformed to lowercase.
8. Stop words were eliminated.

3.2 Feature extraction

After pre-processing, we convert movie reviews into numerical features to classify them. Word embedding is used for deep learning (S. M. M. Hossain et al., 2021), while the Term Frequency–

Inverse Document Frequency (TF-IDF) vector is used for machine learning (S. M. M. Hossain et al.,2023).

3.2.1 Term Frequency– Inverse Document Frequency

In this section, we discuss the TF-IDF algorithm, which is widely used for feature selection in various natural language processing (NLP) tasks. Feature selection plays a crucial role in reducing the dimensionality of textual data and extracting meaningful and informative features for analysis and modeling.
Before beginning the TF-IDF algorithm, pre-processing is carried out (S. M. M. Hossain et al.,2023). (i) Sentences rather than words are tokenized after the pre-processing stage. The weight value is then given. ii) The word frequency is then calculated. iii) In this phase, the formula in Eq. (1) is used to calculate TF (Term Frequency).

$$TF(x) = \frac{No. of\ times\ words\ x\ in\ a\ doc}{Total\ no. of\ words\ in\ it} \quad (1)$$

iv) The frequency of each word in each phrase is then recorded in a table. IDF (Inverse Document Frequency) is then determined by Eq. (2).

$$IDF(x) = \log_e \frac{Total\ no. of\ docs}{No. of\ docs\ with\ x\ in\ that\ doc} \quad (2)$$

vi) The result of multiplying equations (1) and (2) yields the TF-IDF. vii) In this phase, the threshold value is determined by averaging the scores of all the terms. If the appropriate score is higher than the threshold value, any words are then chosen. The general steps of the TF-IDF technique are summarised in Figure 3.

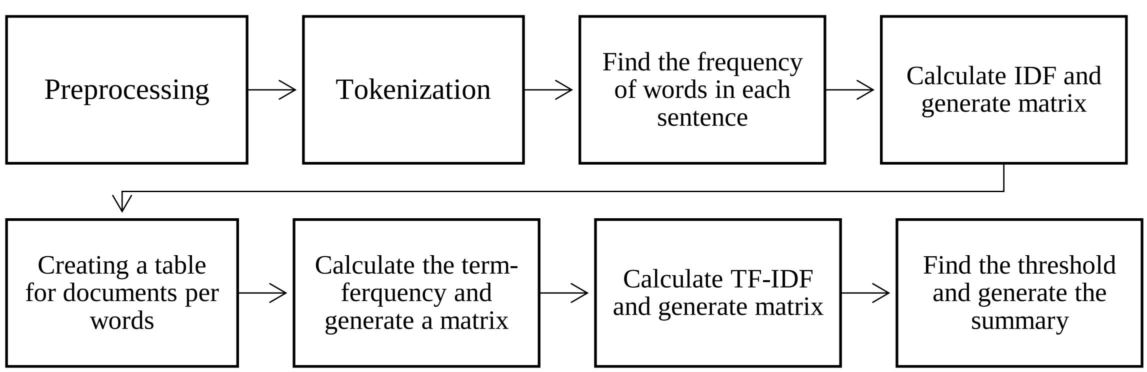


Figure 3: The TF-IDF technique's overall procedure.

3.2.2 Word embedding

word embedding techniques to extract meaningful features and enhance the capabilities of their deep learning models in user reviews. As a word representation input to the deep learning model, the word embedding is quite helpful. Nevertheless, we decided to train our word embedding model for the database (S. M. M. Hossain *et al.*, 2021). As seen in Figure 4, the word embedding design advises converting the text data into a format compatible with learning models. The initial stage seeks to completely transform the corpus text into a series because the input data we have consists of P text reviews of movie collections. Each token is then given an integer index in a lexicon. The result is a list of different sequence lengths for having different word counts. The deep learning model with the word embedded layer is then trained using primer data.

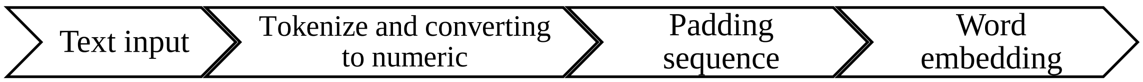


Figure 4: Word embedding method.

4. Classifiers

For the classification of sentiment. Firstly, we applied One of the supervised algorithms such as SVM, NB (S. Liao et al., 2017). Secondly, we also applied deep learning algorithms such as LSTM, CNN, and ANN.

4.1 Naïve Bayes (NB)

The computationally effective NB technique is utilized in text analysis. NB tracks every phrase in a text by assigning equal weight to each word (I. Rish, 2001). The Bayes theorem is applied in NB, where features are independent of one another. In other words, the likelihood of one trait is independent of the likelihood of the other. In Eq. (3), the probability model is shown.

$$P(A|B) = P(A) \times P(B|A)/P(A) \quad (3)$$

where $P(B)$ is prior probability of B, $P(A)$= prior probability of A and $P(B|A)$ = occurrence of predictor B given class A probability.

4.2 Support Vector Machines (SVM)

Improving classification accuracy using SVM as a supervised machine learning model that is employed in many machine learning applications (T. Joachims, 1998). The optimal hyperplane that separates all the data points of one class from those of the other is chosen using SVM to categorize the data in a binary classification issue. SVM uses the formula in Eq. (4) to maximize the separated hyperplane to distinguish between the classes:

$$wx_i + b = 0 \quad (4)$$

if x is the sample i feature vector, w is the weight factor, and b is the bias.

4.3 Convolutional Neural Network (CNN)

One of the most well-liked deep learning models is the convolution neural network (CNN). This model was the one to which digital signal processing was first applied (Y. Kim, 2014).

- **Convolution layer**: this layer processes data by surfing a fixed-size filter, or kernel, over the input data to produce more refined output. It does this by using convolution operations.

- **Pooling layer:** The pooling layer's primary function is to pool the vectors that the convolution layer produces so that only the most significant vectors are kept.
- **Fully connected layer:** Following the convolution layer in a CNN, there is always one or more fully connected layers. This is a standard perceptron-based method to generate the output, which is then utilized in a back-propagation fashion to retrain the entire system.
- **Dropout:** Overfitting can be avoided using this method. We employ a probability p to randomly stop some specific weights during the training phase.

4.4 Recurrent Neural Network (RNN)

One input layer, two hidden layers, and one output layer make up the RNN architecture. These layers each function on their own. Every layer has a set of weighted structures, and each layer has its own set of threshold values. The system will produce more accurate findings in this way. These iterative processes lead to the association of the newly obtained input with the previous input by storing the old input state and combining it with the newly obtained input value (Q. Yao et al., 2022). Due to the problems of RNN, various networks of later variants, such as LSTM.

4.4.1 Long-Term Memory Networks (LSTM)

LSTMs, also known as long-term memory networks, are a unique type of (RNN) that can learn long-term dependencies (H. Sak et al.,2014). All recurrent neural networks take the form of a chain of repeating modules of neural networks, and in standard RNNs, this repeating module will have a very simple structure, such as a single tanh layer. LSTMs also have this chain-like structure, but the repeating module has a different structure: instead of having a single neural network layer, there are four, interacting in a very unique way. The LSTM design includes a range of multiple units for each step, each of which is made up of a cell c_t, the memory component of the LSTM, and several gates that regulate the flow of information within the LSM unit. The following are the transition functions between the LSTM units (C. Zhou et al., 2015), visualized in these equations from (5) to (10).

$$i_t = \sigma\,[\,w_i \cdot (\,h_{t-1} + b_i)]\ (5)$$

$$f_t = \sigma\,[w_f \cdot (\,h_{t-1}, x_t) + b_f]\ (6)$$

$$q_t = \tanh\,[w_q \cdot (\,h_{t-1}, x_t) + b_q]\ (7)$$

$$O_t = \sigma\,[w_o \cdot (\,h_{t-1}, x_t) + b_o]\ (8)$$

$$c_t = q_t \odot i_t + f_t \odot c_{t-1}\ (9)$$

$$h_t = o_t \odot \tanh[c_t]\ (10)$$

Where i_t is an input gate, O_t is an output gate O_t, and f_t is a forget gate at time step t. These gates determine how the current memory cell c_t and the current hidden state h_t are updated together. Here, x_t is the input vector fed into LSTM. Whereas tanh represents the hyperbolic tangent function and σ represents the sigmoid activation function. The weight matrices and bias, however, are denoted by w and b. In addition, the operator designates the elementwise product.

5. Results and Analysis

Python 3.7 was employed. On an AMD Ryzen 7 CPU with 32 GB of RAM, an Nvidia RTX 2060 Super with 8 GB of graphics memory, 240 GB of SSD storage, and a 1 TB HDD, all the tests were conducted.

5.1 Dataset description

We made use of the 50k movie reviews from IMDB (IMDB Dataset of 50K Movie Reviews at site: https://www.kaggle.com/datasets/lakshmi25npathi/imdb-dataset-of-50k-movie-reviews). There are 50,000 elements in the file, divided into two columns: "review" and "sentiment." The review's content is listed in the review column, and the sentiment that goes along with it is listed in the sentiment column. There are just two possible values for the "sentiment" column: positive or negative. The description of the database IMDB is shown in Table 1. There are twenty-five thousand positive reviews and twenty-five thousand negative reviews, all in English.
We used 5039 positive ratings and 4961 negative reviews for testing. The movie review was split into two sets: a training set and a test set. 20% for testing and 80% for training. Additionally, 20% of the total training movie review is used to validate the model and lessen model bias.

Table 1: Description of IMDB database.

<table>
<tr><th colspan="2">Class</th><th>Positive class</th><th>Negative class</th></tr>
<tr><td>Training data</td><td rowspan="2">80%</td><td>16,000</td><td>16,000</td></tr>
<tr><td>Validation data</td><td>3,961</td><td>4,039</td></tr>
<tr><td>Test data</td><td>20%</td><td>5,039</td><td>4,961</td></tr>
</table>

5.2 Performance measures

we take accuracy and F1-score for evaluating our detection model as formulated in Equations (11) to (17) (Ü. Ağbulut et al., 2021).

$$Recall\ of model = \frac{\sum_j Recall\ of\ each\ class\ \times\ N_j}{N} \quad (11)$$

$$Rcall\ of\ class = \frac{true\ positive}{true\ negative + false\ negative} \quad (12)$$

$$Precision\ of model = \frac{\sum_j Precision\ of\ each\ class\ \times\ N_j}{N} \quad (13)$$

$$Precision\ of\ class = \frac{true\ positive}{true\ negative + false\ positive} \quad (14)$$

$$Accuracy\ of\ model = \frac{\sum_j Recognotion\ rate\ of\ each\ class \times N_j}{N} \quad (15)$$

$$Recognotion\ rate\ of\ class = \frac{\text{true negative+true positive}}{\text{No.of all samples}} \quad (16)$$

$$F_1 score = 2 \times \frac{precision \times Recall}{precision + Recall} \quad (17)$$

where j represents each of the class, Nj indicates the number of samples in class j, and N is the total number of samples used to test the model.

5.3 Performance Evaluation for different machine learning algorithms.

On the 50k movie reviews from IMDB, we applied SVM and NB and we used TF-IDF (S. M. M. Hossain et al.,2023), Due to conditional probability for independent occurrences, NB outperformed SVM in terms of accuracy. The results of the experiment are summarized in Table 2.

Table 2: performance of machine learning algorithms

Classifier	**Numeric conversion**	$F_1 score$	**Accuracy**
SVM	TF-IDF	72 %	71.25 %
NB	TF-IDF	72 %	72 %

5.4 Performance Evaluation for deep learning algorithm

For the most effective tests, we used a batch size of 128 and an input length of 100. With the help of a callback function and terminating criteria, up to 50 epochs were used to train the model. The last layer was applied with the Sigmoid activation function, which yields a value between 0 and 1. The loss function employed was Binary Cross-Entropy. The learning rates for the Adam optimizer were 0.001, β_1 of 0.9, and β_2 of 0.999. Table 3 displays the hyper-parameters that work best for our models.

Table 3: Hyper-parameters

Hyper-parameters	**Value**
Input length	100
Optimizer	Adam
Loss	Binary cross-entropy
$\boldsymbol{\beta_1}$	0.9
$\boldsymbol{\beta_2}$	0.999
Epoch	50
Learning rate	0.001
Batch size	128

5.4.1 Performance Evaluation for ANN

We must provide 100 words as input to the embedding layer because the word embedding length is 100. The vocabulary has 90,598 words in it. The GloVe embedding was used. We performed the training and validation. After testing, we acquired an F_1-score of 74%. Table 4 shows the trainable parameters of the proposed model. Table 7 shows the accuracy of training and accuracy of validation for ANN, CNN, and LSTM.

Table 4: Trainable parameters of the proposed Simple Neural Network model

Layers	**Embedding**	**Flatten**	**Dense**
Parameters #	9059800	0	10001
Output Shape	(100,100)	10000	1

5.4.2 Performance Evaluation for CNN

The embedding layer's input length and output vector are both 100. Then, a one-dimensional convolutional layer with 128 features and a kernel size of 5 with ReLU activation function was added. We then implemented a max-pooling layer to reduce feature size. Finally, we added a dense layer with sigmoid activation to the output. We performed the training and validation and finally, after testing, we obtained an F1-score of 85%. Table 5 shows the trainable parameters for the suggested model.

Table 5: trainable parameters of the proposed CNN model

Layers	**Embedding**	**Conv-1D**	**Global Max Pooling**	**Dense**
Parameters #	9059800	64128	0	129
Output Shape	(100,100)	(96,128)	128	1

5.4.3 Performance Evaluation for LSTM

The embedding layer's input length and output vector are both 100. We then include a 128-neuron LSTM layer after that. Table 6 shows the trainable parameters for the suggested model.

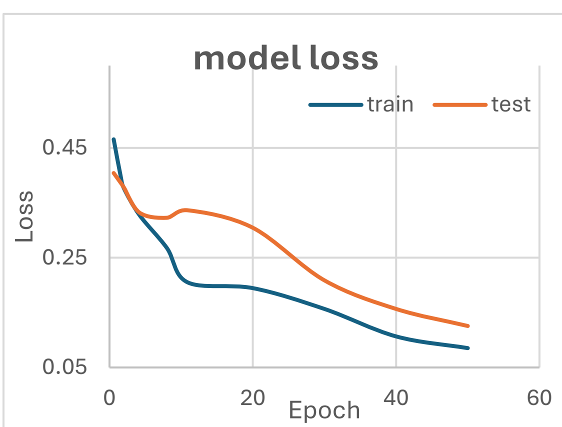
Table 6: Trainable parameters of the proposed LSTM model

Layers	**Embedding**	**LSTM**	**Dense**
Parameters #	9059800	117248	129
Output Shape	(100,100)	128	1

We performed the training and validation. After testing, we obtained an F_1-score of 87% and 87.11% accuracy. With the help of a callback function and terminating criteria, we trained the model for up to 50 epochs, as seen in Figures 5 and 6.

Table 7: Training and Validation accuracy of ANN, CNN and LSTM.

Classifier	**Training accuracy**	**Validation accuracy**
ANN	85.49%	74.91%
CNN	99.66%	85.25%
LSTM	92.25%	87.15%



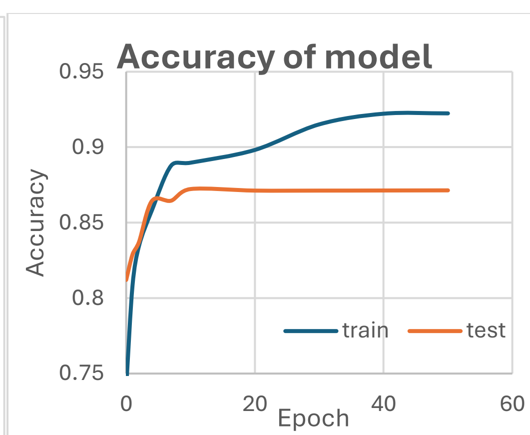


Figure 5: Loss curve for LSTM model. Figure 6: Accuracy curve for LSTM model.

The LSTM performance curves are shown in Figures 5 and 6. Figure 7 showes the ROC curve for the LSTM. We obtain 4295 true negatives, 4416 true positives, 666 false negatives, and 623 false positives, according to Figure 7, the AUC for each class is 0.87.

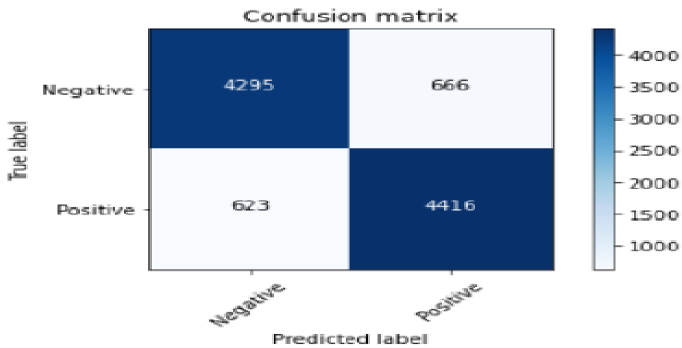


Figure 7: Confusion matrix of LSTM Model.

5.5 Measurement accuracy of a deep learning model for test data

Table 2 and table 8 make it clear that deep learning algorithms outperform machine learning algorithms in terms of performance. The accuracy of the LSTM, one of the three deep learning models, was higher thanks to extended sequence modeling, as seen in Table 8.

Table 8: Performance analyses of all deep learning techniques

Technique	Accuracy (%)	F_1-score (%)	Recall (%)	Precision (%)
ANN	73.95	74.00	74.00	74.00
CNN	85.17	85.00	85.00	85.00
LSTM	87.11	87.00	87.00	87.00

6. Conclusion

Due to the daily production, distribution, and transfer of enormous volumes of data and views via the Internet and other media, sentiment analysis has become essential for the development of opinion mining systems. The way people express their thoughts has been greatly impacted in recent years by social networks, online forums, blogs, and other platforms. Politics, sports, the entertainment business, and other industries routinely employ sentiment analysis to categorize these feelings as positive or negative. The same is true for movie recommendations. This study examines well-known machine learning algorithms including SVM and NB. Additionally, neural networks like ANN, CNN, and LSTM, a type of RNN, are used. With an accuracy of 87.11% and an F1-score of 87%, the LSTM model outperforms them all. LSTM With hyper-parameter tuning outperforms than state–of–the–art LSTM models. However, we might use the unique dataset even more for sentiment analysis based on several targets.

References

A. Alhudhaif, K. Polat, and O. Karaman, 2021, “Determination of COVID-19 pneumonia based on generalized convolutional neural network model from chest X-ray images,” *Expert Systems with Applications,* vol. 180, p. 115141.

A. Hassan and A. Mahmood, 2017, “Deep learning approach for sentiment analysis of short texts,” in *2017 3rd international conference on control, automation and robotics (ICCAR)*, pp. 705–710.

A. Kamal, 2013, "Subjectivity classification using machine learning techniques for mining feature-opinion pairs from web opinion sources," *arXiv Prepr. arXiv1312.6962*.

A. Kamal, 2015, "Review mining for feature based opinion summarization and visualization," *arXiv Prepr. arXiv1504.03068*.

A. Sevik, P. Erdogmus, and E. Yalein, 2018, "Font and Turkish letter recognition in images with deep learning," in *2018 International Congress on Big Data, Deep Learning and Fighting Cyber Terrorism (IBIGDELFT)*, pp. 61–64.

A. T. Kabakus and P. Erdogmus, 2022 ,"An experimental comparison of the widely used pre-trained deep neural networks for image classification tasks towards revealing the promise of transfer-learning," *Concurrency and Computation: Practice and Experience,* vol. 34, no. 24, p. e7216,.

B. Pang and L. Lee, 2004, "A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts," *arXiv Prepr. cs/0409058*.

C. Zhou, C. Sun, Z. Liu, and F. Lau, 2015, "A C-LSTM neural network for text classification," *arXiv Prepr. arXiv1511.08630*.

D. Chaffe, 2023, "Global social media statistics research summary 2023," *smart insights*.

D. Nguyen, K. Vo, D. Pham, M. Nguyen, and T. Quan, 2018, "A deep architecture for sentiment analysis of news articles," in *Advanced Computational Methods for Knowledge Engineering: Proceedings of the 5th International Conference on Computer Science, Applied Mathematics and Applications, ICCSAMA 2017 5*, pp. 129–140.

G. K. Pitsilis, H. Ramampiaro, and H. Langseth, 2018, "Detecting offensive language in tweets using deep learning," *arXiv Prepr. arXiv1801.04433*.

H. Sak, A. W. Senior, and F. Beaufays, 2014, "Long short-term memory recurrent neural network architectures for large scale acoustic modeling," *INTERSPEECH*, pp. 338-342.

I. Rish, "An empirical study of the naive Bayes classifier, 2001, " in *IJCAI 2001 workshop on empirical methods in artificial intelligence*, vol. 3, no. 22, pp. 41–46.

K. Chowdhary and K. R. Chowdhary, 2020, "Natural language processing," *Fundamentals of artificial intelligence*, pp. 603–649.

K. Dashtipour, M. Gogate, E. Cambria, and A. Hussain, 2021, "A novel context-aware multimodal framework for persian sentiment analysis," *Neurocomputing*, vol. 457, pp. 377–388.

M. Birjali, M. Kasri, and A. Beni-Hssane, 2021, "A comprehensive survey on sentiment analysis: Approaches, challenges and trends," *Knowledge-Based Systems*, vol. 226, p. 107134.

M. F. Mridha, A. Q. Ohi, M. A. Hamid, and M. M. Monowar, 2022, "A study on the challenges and opportunities of speech recognition for Bengali language," *Artificial Intelligence Review*, pp. 1–25.

N. Calik and B. U. Toreyin, 2022, "Image Analysis," in *Business Analytics for Professionals*, Springer, pp. 201–238.

P. Baid, A. Gupta, and N. Chaplot, 2017, "Sentiment analysis of movie reviews using machine learning techniques," *International Journal of Computer Applications*, vol. 179, no. 7, pp. 45–49.

Q. Yao, D. D.-C. Lu, and G. Lei, 2022, "A surface temperature estimation method for lithium-ion battery using enhanced GRU-RNN," *IEEE Transactions on Transportation Electrification*, vol. 9, no. 1, pp. 1103–1112.

R. Andrade-Gonzalez and R. Rodriguez-Aguilar, 2021, "Sentiment analysis model on twitter about video streaming platforms in Mexico," in *Computer Science and Engineering in Health*

Services: 5th EAI International Conference, COMPSE 2021, Virtual Event, July 29, 2021, Proceedings 5, pp. 73–87.

S. Liao, J. Wang, R. Yu, K. Sato, and Z. Cheng, 2017, “CNN for situations understanding based on sentiment analysis of twitter data,” *Procedia Computer Science*, vol. 111, pp. 376–381.

S. M. M. Hossain, J. A. Sumon, A. Sen,M. I. Alam, K. M. A. Kamal, H. Alqahtani, I. H. Sarker, 2021, “Spam filtering of mobile SMS using CNN–LSTM based deep learning model,” in *International Conference on Hybrid Intelligent Systems*, pp. 106–116.

S. M. M. Hossain, K. M. A. Kamal, A. Sen, and I. H. Sarker, 2023, “TF-IDF Feature-Based Spam Filtering of Mobile SMS Using a Machine Learning Approach,” in *Applied Intelligence for Industry 4.0*, Chapman and Hall/CRC, pp. 162–175.

S. Meera and S. Geerthik, 2022, “Natural language processing,” *Artificial Intelligent Techniques for Wireless Communication and Networking*, pp. 139–153.

S. Nisar, M. Tariq, A. Adeel, M. Gogate, and A. Hussain, 2019, “Cognitively inspired feature extraction and speech recognition for automated hearing loss testing,” *Cognitive Computation*, vol. 11, pp. 489–502.

T. A. Rana and Y.-N. Cheah, 2016, “Aspect extraction in sentiment analysis: comparative analysis and survey,” *Artificial Intelligence Review*, vol. 46, pp. 459–483.

T. Joachims, 1998, “Text categorization with support vector machines: Learning with many relevant features,” in *European conference on machine learning*, pp. 137–142.

Ü. Ağbulut, A. E. Gürel, and Y. Biçen, 2021, “Prediction of daily global solar radiation using different machine learning algorithms: Evaluation and comparison,” *Renewable and Sustainable Energy Reviews*, vol. 135, p. 110114.

W. Tian, L. Yi, W. Liu, W. Huang, G. Ma, and Y. Zhang, 2020 “Ground radar precipitation estimation with deep learning approaches in meteorological private cloud,” *Journal of cloud computing*, vol. 9, no. 1, pp. 1–12,.

W. Zhao, Z. Guan, L. Chen, X. He, D. Cai, B. Wang, Q. Wang, 2017, “Weakly-supervised deep embedding for product review sentiment analysis,” *IEEE Transactions on Knowledge and Data Engineering*, vol. 30, no. 1, pp. 185–197.

X. Ouyang, P. Zhou, C. H. Li, and L. Liu, 2015, “Sentiment analysis using convolutional neural network,” in *2015 IEEE international conference on computer and information technology; ubiquitous computing and communications; dependable, autonomic and secure computing; pervasive intelligence and computing*, pp. 2359–2364.

Y. Kim, 2014, “Convolutional neural networks for sentence classification,” *arXiv Prepr. arXiv1408.5882*.

Y. Zhang, D.S. Park, W. Han, J. Qin, A. Gulati, J. Shor, A. Jansen, Y. Xu, Y. Huang, S. Wang and Z. Zhou, 2022, “Bigssl: Exploring the frontier of large-scale semi-supervised learning for automatic speech recognition,” *IEEE Journal of Selected Topics in Signal Processing*, vol. 16, no. 6, pp. 1519–1532.

S. Khalid, T. Khalil and S. Nasreen, *2014,* "A survey of feature selection and feature extraction techniques in machine learning," *Science and Information Conference*, London, UK, pp. 372-378, doi: 10.1109/SAI.2014.6918213.