Classification of movie reviews using term frequency-inverse document frequency and optimized machine learning algorithms

Muhammad Zaid Naeem [1] , **Furqan Rustam** [1] , **Arif Mehmood** [2] , **Mui-zzud- din** [1] , **Imran Ashraf** [Corresp., 3] , **Gyu Sang Choi** [Corresp. 3]

[1] Department of Computer Science, Khwaja Fareed University of Engineering and Information Technology, Rahim Yar Khan, Pakistan

[2] Department of Computer Science & Information Technology,, The Islamia University of Bahawalpur, Bahawalpur, Pakistan

[3] Information and Communication Engineering, Yeungnam University, Gyeongsan si, Daegu, South Korea

Corresponding Authors: Imran Ashraf, Gyu Sang Choi
Email address: imranashraf@ynu.ac.kr, castchoi@ynu.ac.kr

The internet movie database (IMDB) being one of the popular online databases for movies and personalities, provides a wide range of movie reviews from millions of users. This provides a diverse and large dataset to analyze users' sentiments about various personalities and movies. Despite being helpful to provide the critique of movies, the reviews on IMDB can not be read as a whole and requires automated tools to provide insights on the sentiments in such reviews. This study provides the implementation of various machine learning models to measure the polarity of the sentiments presented in user reviews on the IMDB website. For this purpose, the reviews are first preprocessed to remove redundant information and noise, and then various classification models like support vector machines (SVM), Naïve Bayes classifier, random forest, and gradient boosting classifiers are used to predict the sentiment of these reviews. The objective is to find the optimal process and approach to attain the highest accuracy with the best generalization. Various feature engineering approaches such as term frequency-inverse document frequency (TF-IDF), bag of words, global vectors for word representations, and Word2Vec are applied along with the hyperparameter tuning of the classification models to enhance the classification accuracy. Experimental results indicate that the SVM obtains the highest accuracy when used with TF-IDF features and achieves an accuracy of 89.55\%. The sentiment classification accuracy of the models is affected due to the contradictions in the user sentiments in the reviews and assigned labels. For tackling this issue, TextBlob is used to assign a sentiment to the dataset containing reviews before it can be used for training. Experimental results on TextBlob assigned sentiments indicate that an accuracy of 92\% can be obtained using the proposed model.

Classification of Movie Reviews Using Term Frequency-Inverse Document Frequency and Optimized Machine Learning Algorithms

Muhammad Zaid Naeem[1†]**, Furqan Rustam**[1†]**, Arif Mehmood**[2]**, Mui-zzud-din**[1]**, Imran Ashraf**[3*]**, and Gyu Sang Choi**[3*]

[1]**Department of Computer Science, Khwaja Fareed University of Engineering and Information Technology, Rahim Yar Khan, 64200, Pakistan**
[2]**Department of Computer Science & Information Technology, The Islamia University of Bahawalpur, Bahawalpur 63100**
[3]**Department of Information & Communication Engineering, Yeungnam University, Gyeongbuk, Gyeongsan-si 38541, Republic of Korea**

Corresponding author:
Gyu Sang Choi, Imran Ashraf

Email address: {castchoi,imranashraf}@ynu.ac.kr

ABSTRACT

The internet movie database (IMDB) being one of the popular online databases for movies and personalities, provides a wide range of movie reviews from millions of users. This provides a diverse and large dataset to analyze users' sentiments about various personalities and movies. Despite being helpful to provide the critique of movies, the reviews on IMDB can not be read as a whole and requires automated tools to provide insights on the sentiments in such reviews. This study provides the implementation of various machine learning models to measure the polarity of the sentiments presented in user reviews on the IMDB website. For this purpose, the reviews are first preprocessed to remove redundant information and noise, and then various classification models like support vector machines (SVM), Naïve Bayes classifier, random forest, and gradient boosting classifiers are used to predict the sentiment of these reviews. The objective is to find the optimal process and approach to attain the highest accuracy with the best generalization. Various feature engineering approaches such as term frequency-inverse document frequency (TF-IDF), bag of words, global vectors for word representations, and Word2Vec are applied along with the hyperparameter tuning of the classification models to enhance the classification accuracy. Experimental results indicate that the SVM obtains the highest accuracy when used with TF-IDF features and achieves an accuracy of 89.55%. The sentiment classification accuracy of the models is affected due to the contradictions in the user sentiments in the reviews and assigned labels. For tackling this issue, TextBlob is used to assign a sentiment to the dataset containing reviews before it can be used for training. Experimental results on TextBlob assigned sentiments indicate that an accuracy of 92% can be obtained using the proposed model.

INTRODUCTION

Social media has become an integral part of human lives in recent times. People want to share their opinions, ideas, comments, and daily life events on social media. In modern times, social media is used for showcasing one's esteem and prestige by posting photos, text, and video clips, etc. The rise and wide usage of social media platforms and microblogging websites provide the opportunity to share as you like where people share their opinions on trending topics, politics, movie reviews, etc. Shared opinions on social networking sites are generally known as short texts (ST) concerning the length of the posted text (Sahu and Ahuja, 2016). ST has gained significant importance over traditional blogging because of its simplicity and effectiveness to influence the crowd. These ST take the form of jargon and are even used by

search engines as queries. Apart from being inspiring, the ST contains users' sentiments about a specific personality, topic, or movie and can be leveraged to identify the popularity of the discussed item. The process of mining the sentiment from the texts is called sentiment analysis (SA) and has been regarded as a significant research area during the last few years (Hearst, 2003). Sentiments given on social media platforms like Twitter, Facebook, etc. can be used to analyze the perception of people about a personality, service, or product, as well as, used to predict the outcome of various social and political campaigns. Thus, SA helps to increase the popularity and followers of political leaders, as well as, other important personalities. Many large companies like Azamon, Apple, and Google use the reviews of their employees to analyze the response to various services and policies. In the business sector, companies use SA to derive new strategies based on customer feedback and reviews (Hand and Adams, 2014; Alpaydin, 2020).

Besides the social media platforms, several websites serve as a common platform for discussions about social events, sports, and movies, etc., and the internet movie database (IMDB) is one of the websites that offer a common interface to discuss movies and provide reviews. Reviews are short texts that generally express an opinion about movies or products. These reviews play a vital role in the success of movies or sales of the products (Agarwal and Mittal, 2016). People generally look into blogs, review sites like IMDB to know the movie cast, crew, reviews, and ratings of other people. Hence it is not only the word of mouth that brings the audience to the theaters, reviews also play a prominent role in the promotion of the movies. SA on movie reviews thus helps to perform opinion summarization by extracting and analyzing the sentiments expressed by the reviewers (Ikonomakis et al., 2005). Being said that the reviews contain valuable and very useful content, the new user can't read all the reviews and perceive the positive or negative sentiment. The use of machine learning approaches proves to ease this difficult task by automatically classifying the sentiments of these reviews. Sentiment classification involves three types of approaches including the supervised machine learning approach, using the semantic orientation of the text, and use of SentiWordNet based libraries (Singh et al., 2013a).

Despite being several approaches presented, several challenges remain unresolved to achieve the best possible accuracy for sentiment analysis. For example, a standard sequence for preprocessing steps is not defined and several variations are used which tend to show slightly different accuracy. Bag of words (BoW) is widely used for sentiment analysis, however, BoW loses word order information. Investigating the influence of other feature extraction approaches is of significant importance. Deep learning approaches tend to show better results than the traditional machine learning classifiers, but the extent of their better performance is not defined. This study uses various machine learning classifiers to perform sentiment analysis on the movie reviews and makes the following contributions

- This study proposes a methodology to perform the sentiment analysis on the movie reviews taken from the IMDB website. The proposed methodology involves preprocessing steps and various machine learning classifiers along with several feature extraction approaches.

- Both simple and ensemble classifiers are tested with the methodology including decision trees (DT), random forest (RF), gradient boosting classifier (GBC), and support vector machines (SVM). In addition, a deep learning model is used to evaluate its performance in comparison to traditional machine learning classifiers.

- Four feature extraction techniques are tested for their efficacy in sentiment classification. Feature extraction approaches include term frequency-inverse document frequency (TF-IDF), BoW, global vectors (GloVe) for word representations, and Word2Vec.

- Owing to the influence of the contradictions in users' sentiments in the reviews and assigned labels on the sentiment classification accuracy, in addition to the standard dataset, TextBlob annotated dataset is also used for experiments.

- The performance of the selected classifiers is analyzed using accuracy, precision, recall, and F1 score. Additionally, the results are compared with several state-of-the-art approaches to sentiment analysis.

The rest of this paper is organized as follows. Section 2 discusses a few research works which are closely related to the current study. The selected dataset, machine learning classifiers, and preprocessing procedure, and the proposed methodology are described in Section 3. Results are discussed in Section 4 and finally, Section 5 concludes the paper with possible directions for future research.

RELATED WORK

A large amount of generated data on social media platforms on Facebook, and Twitter, etc. are generating new opportunities and challenges for the researchers to fetch useful and meaningful information to thrive business communities and serve the public. As a result, multidimensional research efforts have been performed for sentiment classification and analysis. Various machine learning and deep learning approaches have been presented in the literature in this regard. Few research works which is related to the current study are discussed here; we divide the research works into two categories: machine learning approaches and deep learning approaches.

The use of machine learning algorithms has been accelerated in several domains including image processing, object detection and natural language processing tasks, etc. (Ashraf et al., 2019a; Khalid et al., 2020; Ashraf et al., 2019b). For example, The study (Hakak et al., 2021) uses a machine learning approach for the fake new classification. The study proposes a feature selection technique and an ensemble classifier using three machine learning classifiers including DT, RF, and extra tree classifier. The proposed model achieves a good accuracy score on the 'Liar dataset' as compared to the ISOT dataset.

The authors implement several machine learning classification models for sentiment classification of IMDB reviews into positive and negative sentiments in (Pang et al., 2002a). For this purpose, a dataset containing 752 negative reviews and 1,301 positive reviews from the IMDB website is used. The research aims at finding the suitable model with the highest F1 score and best generalization. Various combinations of features and hyperparameters are used for training the classifiers for better accuracy. K-fold cross-validation is used for evaluating the performance of the classifiers. Naive Bayes tend to achieve higher accuracy of 89.2% than the SVM classifier which achieves 81.0% accuracy.

Similarly, the study (Singh et al., 2013a) conducts experimental work on performance evaluation of the SentiWordNet approach for classification of movie reviews. The SentiWordNet approach is implemented with different variations of linguistic features, scoring schemes, and aggregation thresholds. For evaluation, two large datasets of movie reviews are used that contain the posts on movies about revolutionary changes in Libya and Tunisia. The performance of the SentiWordNet approach is compared with two machine learning approaches including NB and SVM for sentiment classification. The comparative performance of the SentiWordNet and machine learning classifiers show that both NB and SVM perform better than all the variations of SentiWordNet.

A hybrid method is proposed in (Singh et al., 2013b) where the features are extracted by using both statistical and lexicon methods. In addition, various feature selection methods are applied such as Chi-Square, correlation, information gain, and regularized locality preserving indexing (RLPI) for the features extraction. It helps to map the higher dimension input space to the lower dimension input space. Features from both methods are combined to make a new feature set with lower dimension input space. SVM, NB, K- nearest neighbor (KNN), and maximum entropy (ME) classifiers are trained using the IMDb movie review dataset. Results indicate that using hybrid features of TF and TF-IDF) with Lexicon features gives better results.

The authors propose an ensemble approach to improve the accuracy of sentiment analysis in (Minaee et al., 2019). The ensemble model comprises convolutional neural network (CNN) and bidirectional long short term memory (Bi-LSTM) networks and the experiments are performed on IMDB review and Stanford sentiment treebank v2 (SST2) datasets. The ensemble is formed using the predicted scores of the two models to make the final classification of the sentiment of the reviews. Results indicate that the ensemble approach performs better than the state-of-the-art approaches and achieves an accuracy of 90% to classify the sentiment from reviews.

The authors investigate the use of three deep learning classifiers including multilayer perceptron, CNN, and LSTM for sentiment analysis in (Ali et al., 2019). Besides, experiments are also carried using a hybrid model CNN-LSTM for sentiment classification, and the performance of these models is compared with support vector machines and Naive Bayes. Multilayer Perceptron (MLP) is developed as a baseline for other networks' results. LSTM network, CNN, and CNN-LSTM are applied on the IMDB dataset consisting of 50,000 movies reviews. The word2vec is applied for word embedding. Results indicate that higher accuracy of 89.2% can be achieved from the hybrid model CNN-LSTM. Individual classifiers show a lower accuracy of 86.74%, 87.70%, and 86.64% for MLP, CNN, and LSTM, respectively.

Similarly, an ensemble classifier is proposed in (Yenter and Verma, 2017) which comprises CNN and LSTM networks. The model aims at the word-level classification of the IMDB reviews. The output of the CNN network is fed into an LSTM before being concatenated and sent to a fully connected layer to

produce a single final output. Various regularization techniques, network structures, and kernel sizes are used to generate five different models for classification. The designed models can predict the sentiment polarity of IMDB reviews with 89% or higher accuracy.

The study (Jain and Jain, 2021a) conducts experiments using the IMDB review dataset with deep learning models for sentiment classification. It uses a convolutional neural network (CNN) and long short-term memory (LSTM) with different activation functions. The highest accuracy of 0.883 is achieved with CNN using the ReLU activation function. Similarly, (Nafis and Awang, 2021) proposes a hybrid approach for IMDB review classification using TF-IDF and SVM. The approach called SVM-RFE uses important feature selection to train the SVM model. Feature selection helps in boosting the performance of SVM and increases the accuracy to 89.56% for IMDB reviews sentiment classification. The study (Giatsoglou et al., 2017) proposed an approach for sentiment analysis using a machine learning model. A hybrid feature vector is proposed by combining word2vec and BoW technique and experiments are performed using four datasets containing online user reviews in Greek and English language. In a similar fashion, (Mathapati et al., 2018) performs sentiment analysis on IMDB reviews using a deep learning approach. The study used a CNN and LSTM recurrent neural network to obtain significant accuracy on the IMDB reviews dataset.

The study (Jain and Jain, 2021b) uses a machine learning approach for IMDB reviews classification. The study performs preprocessing of data and proposes a feature selection technique using association rule mining (ARM). Results show that Naive Bayes (NB) outperforms all other used models by achieving a 0.784 accuracy score using the proposed features. The study (Qaisar, 2020) presents an approach using LSTM for IMDB review sentiment classification. LSTM achieves an 0.899 accuracy score on the IMDB dataset. Along the same lines, (Shaukat et al., 2020) performs experiments on the IMDB reviews dataset using a supervised machine learning approach. The study proposed neural network can achieve a 0.91 accuracy score.

From the above-discussed research works, it can be inferred that supervised machine and deep learning approaches show higher performance than lexicon-based approaches. Additionally, the accuracy offered by machine learning approaches requires further improvement, as shown in Table 1. This study focus on using several machine learning classifiers for this purpose, in addition to three feature extraction, approaches for enhanced classification performance. This study contributes to filling the literature gap which is accuracy and efficiency for IMDB review sentiment classification using state-of-the-art techniques.

MATERIALS AND METHODS

This section describes the dataset used for the experiments, machine learning classifiers selected for review classification, as well as, the proposed methodology and its working principles.

Data description

This study uses the 'IMDB Reviews' from Kaggle which contains users' reviews about movies (dat, 2018). The dataset has been largely used for text mining and consists of reviews of 50,000 movie reviews of which approximately 25,000 reviews belong to the positive and negative classes, respectively. Table 2 shows samples of reviews from both negative and positive classes.

TextBlob

TextBlob is a Python library that we used to annotate the dataset with new sentiments (Tex, 2020; Loria, 2018). TextBlob is used for labeling as the possibility of contradiction between the review text and label can not be ignored. TextBlob finds the polarity score for each word and then sums up these polarity scores to find the sentiment. TextBlob assigns a polarity score between -1 and 1. A polarity score greater than 0 shows the positive sentiment, a polarity score less than 0 shows a negative sentiment while a 0 score indicates that the sentiment is neutral. In the dataset used in this study, 23 neutral sentiments are found after applying TextBlob. Pertaining to the low number of neutral sentiments which can cause class imbalance, only negative and positive sentiments are used for experiments. Contradiction in TextBlob annotated label and original dataset label is shown in Table 3.

Table 1. Comprehensive summary of research works discussed in the related work.

Reference	Approach	Model	Aim
(Singh et al., 2013a)	Lexicon-Based	SentiWordNet	Movie review classification
(Singh et al., 2013b)	Machine Learning	RLPI , Hybrid Features, KNN	IMDB reviews classification
(Yenter and Verma, 2017)	Deep Learning	CNN LSTM	IMDB reviews classification
(Giatsoglou et al., 2017)	Machine Learning	BoW-DOUBLE and Average emotion-DOUBLE	IMDB reviews classification
(Mathapati et al., 2018)	Deep Learning	CNN	IMDB reviews classification
(Ali et al., 2019)	Deep Learning	Multilayer perceptron, CNN and LSTM	IMDB reviews classification
(Minaee et al., 2019)	Deep Learning	Bi-LSTM	IMDB review and Stanford sentiment treebank v2 (SST2)
(Qaisar, 2020)	Deep Learning	LSTM	IMDB reviews classification
(Shaukat et al., 2020)	Deep & Machine Learning	NN	IMDB reviews classification
(Jain and Jain, 2021a)	Deep Learning	CNN	IMDB reviews classification
(Nafis and Awang, 2021)	Machine Learning	SVM + (SVM-RFE)	IMDB reviews classification
(Jain and Jain, 2021b)	Machine Learning	NB+ ARM	IMDB reviews classification

Table 2. Description of IMDB dataset variables

Review	Label
Gwyneth Paltrow is absolutely great in this mo..	0
I own this movie. Not by choice, I do. I was r.	0
Well I guess it supposedly not a classic becau..	1
I am, as many are, a fan of Tony Scott films. ..	0
I wish "that '70s show" would come back on tel..	1

Table 3. Contradiction in TextBlob and original dataset labels.

Review	TextBlob	Original
movie makers always author work mean yes things condensed sake viewer interest look anne green gables wonderful job combining important events cohesive whole simply delightful believe chose combine three novels together anne avonlea dreadful mess look missed paul irving little elizabeth widows windy poplars anne college years heaven sake delightful meet priscilla rest redmond gang kevin sullivan taken things one movie time instead jumbling together combining characters events way movie good leave novels montgomery beautiful work something denied movie let seeing successful way brough anne green gables life	Positive	Negative

Feature Engineering Methods

Identification of useful features from the data is an important step for the better training of machine learning classifiers. The formation of secondary features from the original features enhances the efficiency of machine learning algorithms (Oghina et al., 2012; Mujahid et al., 2021). It is one of the critical factors to increase the accuracy of the learning algorithm and boost its performance. The desired accuracy can be

achieved by excluding the meaningless and redundant data. Less quantity of meaningful data is better than having a large quantity of meaningless data (Prabowo and Thelwall, 2009). So, feature engineering is the process of extracting meaningful features from raw data which helps in the learning process of algorithms and increases its efficiency and consistency (Lee et al., 2016).

Bag of Words

The BoW is simple to use and easy to implement for finding the features from raw text data (Rustam et al., 2021; Rupapara et al., 2021a). Many language modeling and text classification problems can be solved using the BoW features. In Python, the BoW is implemented using the CountVectorizer. BoW counts the occurrences of a word in the given text and formulates a feature vector of the whole text comprising of the counts of each unique word in the text. Each unique word is called 'token' and the feature vector is the matrix of these tokens (Liu et al., 2008). Despite being simple, BoW often surpasses many complicated feature engineering approaches in performance.

Term Frequency-Inverse Document Frequency

TF-IDF is another feature engineering method that is used to extract features from raw data. It is mostly deployed in areas like text analysis and music information retrieval (Yu, 2008). In this approach, weights are assigned to every term in a document based on term frequency and inverse document frequency (Neethu and Rajasree, 2013; Biau and Scornet, 2016). Terms having higher weights are supposed to be more important than terms having lower weights. The weight for each term is based on the equation 1.

$$W_{i,j} = TF_{t,d}\left(\frac{N}{D_t}\right) \tag{1}$$

where $TF_{t,d}$ is the number of occurrences of term t in document d, $D_{f,t}$ is the number of documents having the term t and N is the total number of documents in the dataset.

TF-IDF is a kind of scoring measurement approach which is widely used in summarization and information retrieval. TF calculates the frequency of a token and gives higher importance to more common tokens in a given document(Vishwanathan and Murty, 2002). On the other hand, IDF calculates the tokens which are rare in a corpus. In this way, if uncommon words appear in more than one document, they are considered meaningful and important. In a set of documents D, IDF weighs a token x using the equation 2.

$$IDF(x) = N/n(x) \tag{2}$$

Where $n(x)$ denotes frequency of x in D and $N/n(x)$ denotes the inverse frequency. TF-IDF is calculated using TF and IDF as shown in equation 3.

$$TF - IDF = TF \times IDF \tag{3}$$

TF-IDF is applied to calculate the weights of important terms and the final output of TF-IDF is in the form of a weight matrix. Values gradually increase to the count in TF-IDF but are balanced with the frequency of the word in dataset (Zhang et al., 2008).

Word2Vec

Word2Vec is one of the widely used NLP techniques for feature extraction in text mining that transforms text words into vectors (Wang et al., 2016). Given a corpus of text, Word2Vec uses a neural network model for learning word associations. Each unique word has an associated list of numbers called 'vector'. The cosine similarity of the vectors represents the semantic similarity between the words that are represented by vectors.

GloVe

GloVe from Global Vectors is an unsupervised model used to obtain words' vector representation (Bhoir et al., 2017). The vector representation is obtained by mapping the words in a space such that the distance between the words represents the semantic similarity. Developed at Stanford, GloVe can determine the similarity between words and arrange them in the vectors. The output matrix by the GloVe gives vector space of word with linear substructure.

Supervised machine learning Models

Several machine learning classifiers have been selected for evaluating the classification performs in this study. A brief description of each of these classifiers is provided in the following sections.

Random Forest

Rf is based on combining multiple decision trees on various subsamples of the dataset to improve classification accuracy. These subsamples are the combination of randomly selected features which are the size of the original dataset to form a bootstrap dataset. The average of predictions from these models is used to obtain a model with low variance. Information gain ratio and Gini index are the most frequently used feature selection parameters to measure the impurity of feature (Agarwal et al., 2011).

$$\sum \sum_{j \neq i} (\frac{f(C_i, T)}{|T|})(\frac{f(C_j, T)}{|T|}) \tag{4}$$

where $\frac{f(C_i,T)}{|T|}$ indicates the probability of being a member of class C_i.

The decision trees are not pruned upon traversing each new training data set. The user can define the number of features and number of trees on each node and set the values of other hyperparameters to increase the classification accuracy (Biau and Scornet, 2016).

Gradient Boosting Classifier

GBC is an ensemble classifier used for classification tasks with enhanced accuracy based on boosting (Ayyadevara, 2018). It combines many weak learners sequentially to reduce the error gradually. This study uses the GBC with decision tree as a weak learner. GBC performance depends on the loss function and mostly the logarithmic loss function is used for classification. In addition, weak learners and adaptive components are important parameters of GBC. The hyperparameters setting of GBC used in this study is shown in Table 4. GBC is used with 300 n_estimators indicating that 300 weak learners (decision trees) are combined under boosting method and each tree is restricted to 300 max_depth. The learning_rate is set to 0.2 which helps to reduce model overfitting (Rustam et al., 2020).

Decision Tree

DT is one of the most commonly used models for classification and prediction problems. DT is a simple and powerful tool to understand data features and infer decisions. The decision trees are constructed by repeatedly dividing the data according to split criteria. There are three types of nodes in a decision tree: root, internal, and leaf. The root node has no incoming but zero or more outgoing edges, the internal node has exactly one incoming but two or more outgoing edges while the leaf node has one incoming while no outgoing edge (Bakshi et al., 2016; Tan et al., 2006). Nodes and edges represent features and decisions of a decision tree, respectively. A decision tree can be binary or non-binary depending upon the leaves of a node. The gain ratio is one of the commonly used split criteria for DT.

$$Gain\ ratio = \frac{\Delta_{info}}{Split\ Info} \tag{5}$$

where split info is defined as in equation 6.

$$Split\ Info = -\sum_{i=1}^{k} P(v_i) log_2 P(v_i) \tag{6}$$

where k indicates the total number of splits for DT which is hyperparameter tuned for different datasets to elevate the performance. DT is non-parametric, computationally inexpensive, and shows better performance even when the data have redundant attributes.

Support Vector Machine

Originally proposed by Cortes and Vapnik in 1995 for binary classification, SVM is expanded for multi-class classification (Cortes and Vapnik, 1995). SVM is a widely used approach for non-linear classification, regression, and outlier detection (Bennett and Campbell, 2000). SVM has the additional

advantage of examining the relationship theoretically and performs distinctive classification than many complex approaches like neural networks (Agarwal and Mittal, 2016). SVM separates the classes by distinguishing the optimal isolating lines called hyperplane by maximizing the distance between the classes' nearest points (Neethu and Rajasree, 2013). Different kernels can be used with SVM to accomplish better performance such as radial, polynomial, neural, and linear (Guzman and Maalej, 2014). SVM is preferred for several reasons including the lack of local minimal, structural risk minimization principle, and developing more common classification ability (Visa et al., 2011; Vishwanathan and Murty, 2002).

For optimizing the performance of the machine learning models used in this study, several hyperparameters have been fine-tuned according to experience from the literature on text classification tasks. A list of the parameters and corresponding values used for experiments in this study is provided in Table 4.

Table 4. Hyperparameters used for optimizing the performance of models.

Model	Hyperparameters	Values range used for tuning
RF	n_estimators=300, random_state=50, max_depth=300	n_estimators = {50 to 500}, random_state = {2 to 60}, max_depth = {50 to 500}
SVM	kernel= 'linear' , C=3.0, random_state=50	kernel= {'linear', 'poly', 'sigmoid'} , C={1.0 to 5.0}, random_state={2 to 60}
DT	random_state=50, max_depth=300	random_state = {2 to 60}, max_depth = {50 to 500}
GBC	n_estimators=300, random_state=50, max_depth=300, learning_rate=0.2	n_estimators = {50 to 500}, random_state = {2 to 60}, max_depth = {50 to 500}, learning_rate = {0.1 to 0.8}

Proposed methodology

With the growing production of movies over the last two decades, a large number of opinions and reviews are posted on various social media platforms and websites. Such reviews are texts that show explicit opinions about a film or product. These opinions play an important part in the success of film or sales of the products (Agarwal and Mittal, 2016). People search blogs, and evaluation sites like IMDB to get the likes and dislikes of other people about films, the cast, and team, etc. but it is very difficult to read every review and comment. Evaluation of these sentiments becomes beneficial to assisting people in this task. Sentiments expressed in such reviews are important regarding the evaluation of the movies and their crew. Automatic sentiment analysis with higher accuracy is extremely important in this regard and this study follows the same direction and proposes an approach to perform the sentiment analysis of movie reviews. In addition, since the contradictions in the expressed sentiments in movie reviews and their assigned labels can not be ignored, this study additionally uses TextBlob to determine the sentiments. Two sets of experiments are performed using the standard dataset and TextBlob annotated dataset to fill in the research gap as previous studies do not consider the contradictions in the sentiments and assigned labels. Figure 1 shows the flow of the steps carried out for sentiment classification.

As a first step, the reviews are preprocessed using a sequence of operations. Preprocessing is critical to boosting the training of the classifiers and enhancing their performance. The purpose of preprocessing is to clean the data by removing unnecessary, meaningless, and redundant text from reviews. For this purpose, the six steps are carried sequentially, as shown in Figure 2.

Punctuation is removed from IMDB text reviews because punctuation does not add any value to text analysis (Guzman and Maalej, 2014). Sentences are more readable for humans due to punctuation, however, it is difficult for a machine to distinguish punctuation from other characters. Punctuation distorts a model's efficiency to distinguish between entropy, punctuation, and other characters (Rupapara et al., 2021b; Liu et al., 2008). So, punctuation is removed from the text in pre-processing to reduce the complexity of the feature space. Table 5 shows the text of a sample review, before and after the punctuation has been removed.

Once the punctuation is removed, the next step is to find numerical values and remove them as they are not valuable for text analysis. Numerical values are used in the reviews as an alternative to various English words to reduce the length of reviews and ease of writing the review. For example, 2 is used for 'to' and numerical values are used instead of counting like 1 instead of 'one'. Such numerals are
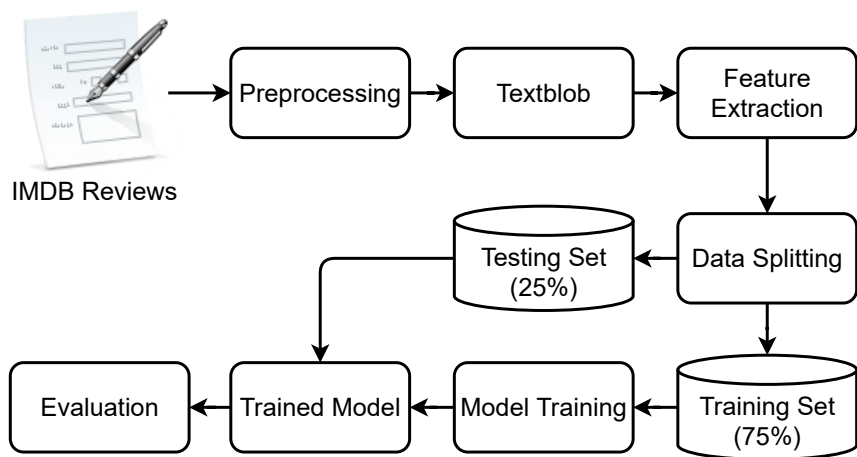
Figure 1. The work flow of proposed methodology for movie review classification.
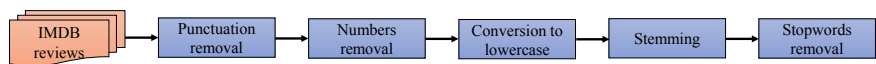
Figure 2. Preprocessing steps for movies review dataset.

Table 5. Text from sample review before and after punctuation removal.

Before puncutation removal	After punctuation removal
@Gwyneth Paltrow is absolutely... !!!great in this movie.	Gwyneth Paltrow is absolutely great in this movie
I own this movie. This is number 1 movie... I didn't like by choice, I do.	I own this movie This is number 1 movie I didnt like by choice I do
I wish "that '70s show" would come back on tel.	I wish that 70s show would come back on tel

329 convenient for humans to interpret, yet offer no help in the training of machine learning classifiers. Table
330 6 shows text from sample reviews after the numeric values are removed.

Table 6. Sample text from movie reviews after removing numeric values.

Input Data	After Numeric Removal
Gwyneth Paltrow is absolutely great in this movie.	Gwyneth Paltrow is absolutely great in this movie
I own this movie This is number 1 movie I didnt like by choice I do.	I own this movie This is number movie I didnt like by choice I do
I wish that 70s show would come back on tel.	I wish that s show would come back on tel

331   In the subsequent step of numbers removal, all capital letters are converted to lower form. Machine
332 learning classifiers can not distinguish between lower and upper case letters and consider them as different
333 letters. For example, 'Health', and 'health' are considered as two separate words if conversion is not
334 performed from uppercase to lowercase. It may reduce the significance of most occurred terms and
335 degrade the performance (Liu et al., 2010). It increases the complexity of the feature space and reduces
336 the performance of classifiers. So, converting the upper case letters to lower form helps in increasing the
337 training efficiency of the classifiers. Table 7 shows the text after the case is changed for the reviews.
338   Stemming is an important step in pre-processing because eliminating affixes from words and changing
339 them into their root form is very helpful to enhance the efficiency of a model (Goel et al., 2016). For

Table 7. Sample output of the review text after changing the case of review text.

Input Data	After Case Lowering
Gwyneth Paltrow is absolutely great in this movie. I own this movie This is number movie I didnt like by choice I do. I wish that s show would come back on tel.	gwyneth paltrow is absolutely great in this movie i own this movie this is number movie i didnt like by choice i do i wish that s show would come back on tel

example, 'help', 'helped', and 'helping' are altered forms of 'help', however, machine learning classifiers consider them as two different words (Singh et al., 2013b). Stemming changes these different forms of words into their root form. Stemming is implemented using the PorterStemmer library of Python (Pang et al., 2002b). Table 8 shows the sample text of review before and after stemming.

Table 8. Text from sample review before and after stemming.

Input Data	After Stemming
gwyneth Paltrow is absolutely great in this movie. i own this movie this is number movie i didnt like by choice I do. i wish that s show would come back on tel.	gwyneth paltrow is absolute great in this movie i own this movie this is number movie i didnt like by choice i do i wish that s show would come back on tel

The last step in the preprocessing phase is the removal of stop words. Stop words have no importance concerning the training of the classifiers. Instead, they increase the feature vector size and reduce the performance. So they must be removed to decrease the complexity of feature space and boost the training of classifiers. Table 9 shows the text of the sample review after the stopwords have been removed.

Table 9. Sample reviews before and after the stop words removal.

Input Data	After Stopwords Removal
gwyneth Paltrow is absolutely great in this movie. i own this movie this is number movie i didnt like by choice I do. i wish that s show would come back on tel.	gwyneth paltrow absolute great movie own movie number movie didnt like choice do wish show would come back tel

After the preprocessing is complete, feature extraction takes place where BoW, TF-IDF, and GloVe are used. Feature space for the sample reviews is given in Tables 10 and 11 for BoW and TF-IDF features, respectively. Experiments are performed with the standard dataset, as well as, the TextBlob annotated dataset to analyze the performance of the machine learning and proposed models.

The data are split into training and testing sets in a 75 to 25 ratio. Machine learning classifiers are trained on the training set while the test set is used to evaluate the performance of the trained models. For evaluating the performance, standard well-known parameters are used such as accuracy, precision, recall, and F1 score.

Evaluation Parameters

Performance evaluation of the classifiers requires evaluation metrics for which accuracy, precision, recall, and F1 score are selected concerning their wide use. The introduction of the confusion matrix is necessary to define the mathematical formulas for these evaluation metrics. The confusion matrix as shown in Figure 3 can be considered as an error matrix that indicates four quantities. The confusion matrix shows true positive (TP), false positive (FP), true negative (TN), and false-negative (FN). Each row of the matrix represents the actual labels while each column represents predicted labels (Landy and Szalay, 1993).

Table 10. BoW features from the preprocessed text of sample reviews.

No.	absolute	back	choice	come	didnt	do	great	gwyneth	like
1	1	0	0	0	0	0	1	1	0
2	0	0	1	0	1	1	0	0	1
3	0	1	0	1	0	0	0	0	0
No.	movie	number	own	paltrow	show	tel	wish	would	
1	1	0	0	1	0	0	0	0	
2	2	1	1	0	0	0	0	0	
3	0	0	0	0	1	1	1	1	

Table 11. TF-IDF features from the preprocessed text of sample reviews.

No.	absolute	back	choice	come	didnt	do	great	gwyneth	like
1	0.467351	0.000000	0.000000	0.000000	0.000000	0.000000	0.467351	0.467351	0.000000
2	0.000000	0.000000	0.346821	0.000000	0.346821	0.346821	0.000000	0.000000	0.346821
3	0.000000	0.408248	0.000000	0.408248	0.000000	0.000000	0.000000	0.000000	0.000000
No.	movie	number	own	paltrow	show	tel	wish	would	
1	0.355432	0.000000	0.000000	0.467351	0.000000	0.000000	0.000000	0.000000	
2	0.527533	0.346821	0.346821	0.000000	0.000000	0.000000	0.000000	0.000000	
3	0.000000	0.000000	0.000000	0.000000	0.408248	0.408248	0.408248	0.408248	

TP indicates that the classifier predicted the review as positive and the original label is also positive. A review is TN if it belongs to the negative class and the real outcome is also negative. In the FP case, the review is predicted as positive, but the original label is negative. Similarly, a review is called FN if it belongs to the positive class but the classifier predicted it as negative (Rokach and Maimon, 2005).

Accuracy is a widely used evaluation metrics and indicates the ratio of true predictions to the total predictions. It has a maximum value of 1 for 100 percent correct prediction and the lowest value of 0 for 0 percent prediction. Accuracy can be defined as

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \tag{7}$$

Precision focuses on the accuracy of predicting the positive cases. It shows what proportion of the positively predicted cases is originally positive. It is defined as

$$Precision = \frac{TP}{TP+FP} \tag{8}$$

Recall calculates the ratio of correct positive cases to the total positive cases. To get the ratio, the total number of TP is divided by the sum of TP and FN as follows

$$Recall = \frac{TP}{TP+FN} \tag{9}$$

F1 score is considered an important parameter to evaluate the performance of a classifier and has been regarded as more important than precision and recall. It defines how precise and robust is the classifier by incorporating precision and recall (Bruce et al., 2002). F1 score value varies between 0 and 1 where 1 shows the perfect performance of the classifier. F1 score is defined as

$$F1Score = 2 \times \frac{precision \times recall}{precision+recall} \tag{10}$$

Predicted Labels

		Negative (0)	Positive (1)
Actual Labels	**Negative (0)**	True Negative(TN)	False Positive (FP)
	Positive (1)	False Negative(FN)	True Positive (TP)

Figure 3. Confusion matrix.

RESULTS AND DISCUSSION

This study uses four machine learning classifiers to classify movie reviews into positive and negative reviews, such as DT. SVM, RF, and GBC. Four feature extraction approaches are utilized including TF-IDF, BoW, Word2Vec, and GloVe on the selected dataset to extract the features. Results for these feature extraction approaches are discussed separately. Similarly, the influence of TextBlob annotated data on the classification accuracy is analyzed. The contradictions in the sentiments expressed in the reviewers and the assigned sentiments cannot be ignored, so TextBlob is used to annotate the labels. Several experiments are performed using the standard, as well as, the TextBlob annotated dataset.

Results using BoW Features

Table 12 shows the classification accuracy of the machine learning classifiers when BoW features to train and test the classifiers. Results indicate that SVM can achieve an accuracy of 0.87 with BoW features. Overall, the performance of all the classifiers is good except for DT whose accuracy is 0.72.

Table 12. Accuracy of the selected models with BoW features.

Classifier	Accuracy
DT	0.72
RF	0.86
GBC	0.85
SVM	0.87

Performance of the classifiers is given in Table 13 in terms of precision, recall, and F1 score. The F1 score indicates that its value is the same with both positive and negative classes for all the classifiers, except for GBC who has F1 scores of 0.86 and 0.85 for positive and negative classes, respectively. Precision values are slightly different for positive and negative classes; for example, SVM has a precision of 0.88 and 0.90 for positive and negative classes. Similarly, although precision, recall, and F1 score of DT are the lowest but the values for positive and negative classes are almost the same. An equal number of the training samples in the dataset makes a good fit for the classifiers, and their accuracy and F1 scores are in agreement.

Table 13. Performance evaluation metrics using BoW features.

Model	Precision			Recall			F1 score		
	Pos.	Neg.	W avg.	Pos.	Neg.	W avg.	Pos.	Neg.	W avg.
DT	0.71	0.72	0.72	0.72	0.71	0.72	0.72	0.72	0.72
RF	0.85	0.87	0.86	0.88	0.84	0.86	0.86	0.86	0.86
GBC	0.83	0.87	0.85	0.88	0.82	0.85	0.86	0.85	0.85
SVM	0.86	0.88	0.87	0.88	0.86	0.87	0.87	0.87	0.87

Results using TF-IDF Features

Table 14 contains the accuracy results for the classifiers using the TF-IDF features. It shows that the performance of the SVM has been elevated with an accuracy of 0.89 which is 2.29% higher than that of using BoW features. Unlike BoW which counts only the frequency of terms, TF-IDF also records the importance of terms by assigning higher weights to rare terms. So, the performance is improved when TF-IDF features are used as compared to BoW features.

Table 14. Accuracy of models with TF-IDF features.

Classifier	Accuracy
DT	0.71
RF	0.86
GBC	0.86
SVM	0.89

Results for precision, recall, and F1 score are given in Table 15. Experimental results indicate that the F1 score is the same for positive and negative classes for all classifiers which indicates the good fit of the modes on the training data. On the other hand, precision for positive and negative classes is slightly different. For example, GBS has a precision of 0.84 and 0.87 while SVM has a precision of 0.88 and 0.90 for positive and negative classes, respectively.

Table 15. Performance evaluation metrics using TF-IDF features.

Model	Precision			Recall			F1 score		
	Pos.	Neg.	W avg.	Pos.	Neg.	W avg.	Pos.	Neg.	W avg.
DT	0.72	0.71	0.71	0.70	0.72	0.71	0.71	0.71	0.71
RF	0.86	0.86	0.86	0.86	0.85	0.86	0.86	0.86	0.86
GBC	0.84	0.87	0.86	0.88	0.83	0.86	0.86	0.85	0.86
SVM	0.88	0.90	0.89	0.90	0.88	0.89	0.89	0.89	0.89

SVM performs better for text classification than other supervised learning models, especially in the case of large datasets as this algorithm is derived from the theory of structural risk minimization (Mouthami et al., 2013).

Classifiers Results using GloVe Features

Experimental results using GloVe features are shown in Table 16 for the selected classifiers. Results suggest that the performance of all the classifiers has been degraded when trained and tested on GloVe features. Glove features are based on the global word-to-word co-occurrence and count the co-occurred terms from the entire corpus. GloVe model is traditionally used with deep learning models where it helps to better recognize the relationships between the given samples of the dataset. In machine learning models, its performance is poor than that of TF-IDF features (Dessi et al., 2020). SVM and RF outperform other models using GloVE features.

Table 16. Performance of classifiers using GloVe features.

Model	Accuracy	Precision			Recall			F1 Score		
		Pos.	Neg.	W avg.	Pos.	Neg.	W avg.	Pos.	Neg.	W avg.
DT	0.65	0.64	0.65	0.65	0.64	0.65	0.65	0.65	0.65	0.65
RF	0.74	0.75	0.74	0.74	0.72	0.77	0.74	0.73	0.75	0.74
GBC	0.65	0.65	0.65	0.65	0.65	0.66	0.65	0.65	0.65	0.65
SVM	0.75	0.75	0.75	0.75	0.75	0.75	0.75	0.75	0.75	0.75

Results using Word2Vec Features

Performance elevation metrics for all the classifiers using the Word2Vec features are given in Table 17. Results indicate that the performance of the classifiers is somehow better when trained and tested on

423 Word2Vec features in comparison with GloVe features results. The performance of classifiers is not
424 significant using Word2Vec features in comparison to the results of the classifiers using BoW and TF-IDF
425 features. SVM achieved the highest accuracy of 0.88 with Word2Vec features as compared to other
426 models because Word2Vec gives linear features set which is more suitable for SVM as compared to RF,
427 GBC, and DT.

Table 17. Performance evaluation of classifiers using Word2Vec features.

Model	Accuracy	Precision			Recall			F1 Score		
		Pos.	Neg.	W avg.	Pos.	Neg.	W avg.	Pos.	Neg.	W avg.
DT	0.65	0.65	0.65	0.65	0.65	0.65	0.65	0.65	0.65	0.65
RF	0.80	0.80	0.80	0.80	0.80	0.80	0.80	0.80	0.80	0.80
GBC	0.65	0.65	0.65	0.65	0.65	0.65	0.65	0.65	0.65	0.65
SVM	0.88	0.88	0.88	0.88	0.88	0.88	0.88	0.88	0.88	0.88

428 The comparison between machine learning models results on original dataset sentiment with BoW,
429 TF-IDF, GloVe, and Word2Vec features are shown in Figure 4. SVM is significant with all features and
430 achieved the best score with BoW, TF-IDF, and Word2Vec. This significant performance of SVM is
431 because of its linear architecture and binary classification problem. SVM is more significant on linear
432 data for binary classification with its linear kernel as shown in this study.
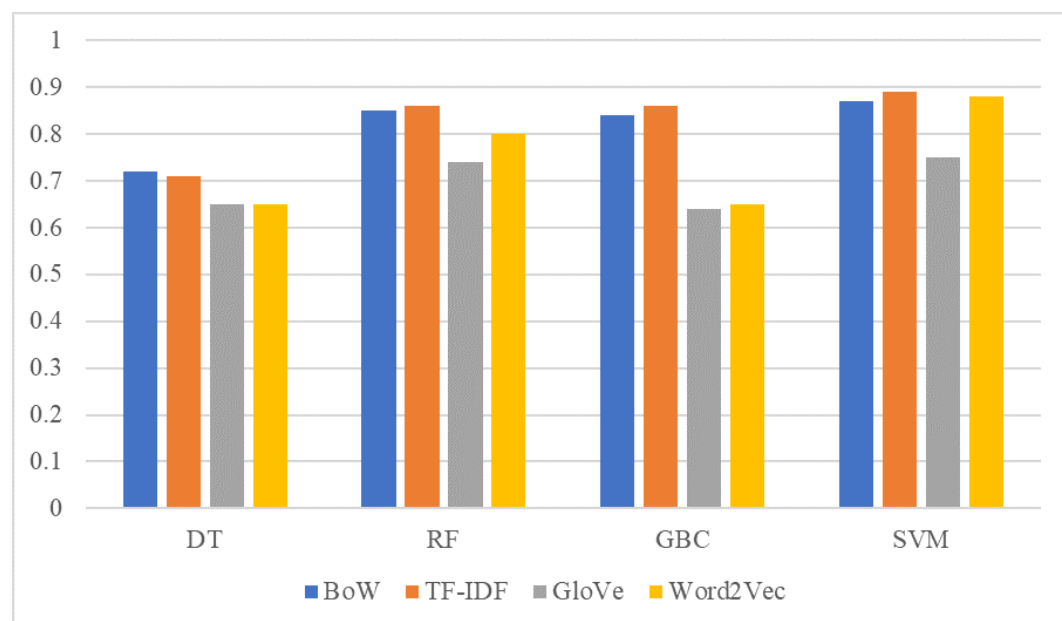
Figure 4. Performance comparison between machine learning models using original dataset and BoW,
TF-IDF, GloVe, Word2Vec features.

Results on TextBlob annotated dataset

434 The contradictions in the users' expressed sentiments in the reviews and assigned labels can influence the
435 sentiment classification accuracy of the models. To resolve this issue, TextBlob annotated data are used
436 for the performance evaluation of the models. Results suggest that the performance of machine learning
437 models is better when used with TextBlob labels than the original labels.

Results using BoW Features

439 The performance of models with BoW and TextBlob sentiments are shown in Table 18. Results indicate
440 that SVM achieved its highest accuracy of the study 0.92 with TextBlob sentiments and BoW features.
441 While the performance of other models such as RF, GBC, and DT has also been improved.

Table 18. Performance evaluation of classifiers using BoW features on TextBlob annotated dataset.

Model	Accuracy	Precision			Recall			F1 Score		
		Pos.	Neg.	W avg.	Pos.	Neg.	W avg.	Pos.	Neg.	W avg.
DT	0.79	0.85	0.61	0.73	0.87	0.57	0.72	0.87	0.59	0.73
RF	0.85	0.84	0.90	0.87	0.98	0.47	0.72	0.90	0.62	0.76
GBC	0.82	0.85	0.70	0.78	0.92	0.55	0.73	0.98	0.62	0.75
SVM	0.92	0.94	0.84	0.89	0.94	0.84	0.89	0.94	0.84	0.89

Results using TF-IDF Features

The performance of models with TF-IDF features and TextBlob sentiments are shown in Table 19. SVM achieves its highest accuracy score of 0.92 with TextBlob sentiment and TF-IDF features. While other models such as RF, GBC, and DT repeat their performances with TF-IDF features.

Table 19. Performance evaluation of classifiers using TF-IDF features on TextBlob annotated dataset.

Model	Accuracy	Precision			Recall			F1 Score		
		Pos.	Neg.	W avg.	Pos.	Neg.	W avg.	Pos.	Neg.	W avg.
DT	0.79	0.85	0.62	0.73	0.87	0.58	0.72	0.86	0.60	0.73
RF	0.84	0.85	0.88	0.87	0.98	0.51	0.74	0.91	0.65	0.78
GBC	0.83	0.86	0.73	0.79	0.92	0.57	0.75	0.89	0.64	0.77
SVM	0.92	0.92	0.88	0.90	0.96	0.78	0.87	0.94	0.82	0.88

Results using GloVe Features

Table 20 shows the performance comparison of models using GloVe features and TextBlob sentiments and it indicates that using GloVe features and TextBlob sentiments is better as compared to their performance on the original sentiments and GloVe features. Compared to the performance on the original dataset, the accuracy of the models has been improved significantly when used with TextBlob assigned sentiments. For example, the highest accuracy with GloVe features and TextBlob sentiment is 0.81 which was only 0.75 on the original sentiments. However, the performance of the machine learning models is inferior to that of BoW and TF-IDF.

Table 20. Performance evaluation of classifiers using GloVe features on TextBlob annotated dataset.

Model	Accuracy	Precision			Recall			F1 Score		
		Pos.	Neg.	W avg.	Pos.	Neg.	W avg.	Pos.	Neg.	W avg.
DT	0.72	0.81	0.47	0.64	0.81	0.48	0.64	0.81	0.48	0.64
RF	0.80	0.71	0.81	0.76	0.94	0.39	0.67	0.87	0.51	0.69
GBC	0.72	0.81	0.47	0.64	0.81	0.48	0.65	0.81	0.48	0.64
SVM	0.81	0.83	0.71	0.77	0.93	0.46	0.70	0.88	0.56	0.72

Results using Word2Vec Features

Table 21 shows the performance of machine learning models with Word2Vec features and TextBlob sentiments. SVM achieves significantly better accuracy with Word2Vec features as compared to GloVe features. It gives a 0.88 accuracy score which is more than GloVe features but lower than BoW and TF-IDF features. RF and GBC achieve 0.79 and 0.70 accuracy scores, respectively. The performance of DT is degraded when used with Word2Vec features.

The comparison between machine learning model results on TextBlob sentiment dataset with BoW, TF-IDF, GloVe, and Word2Vec features is given in Figure 5. SVM obtains better results with TextBlob sentiments using BoW and TF-IDF features as compared to GloVe and Word2Vec features. Similarly, the performance of other models such as RF, GBC, and DT has also been improved with the TextBlob sentiments. For the given dataset, the performance of models is good using TextBlob but these results can not be generalized for every other dataset. It is possible that a few original labels are not correct and using the TextBlob label can show better performance.

Table 21. Performance evaluation of classifiers using Word2Vec features on TextBlob annotated dataset.

Model	Accuracy	Precision			Recall			F1 Score		
		Pos.	Neg.	W avg.	Pos.	Neg.	W avg.	Pos.	Neg.	W avg.
DT	0.69	0.78	0.87	0.83	0.99	0.24	0.62	0.87	0.48	0.62
RF	0.79	0.78	0.87	0.83	0.99	0.24	0.62	0.87	0.38	0.63
GBC	0.70	0.80	0.44	0.62	0.80	0.44	0.62	0.80	0.44	0.62
SVM	0.88	0.90	0.83	0.87	0.95	0.71	0.83	0.92	0.77	0.84
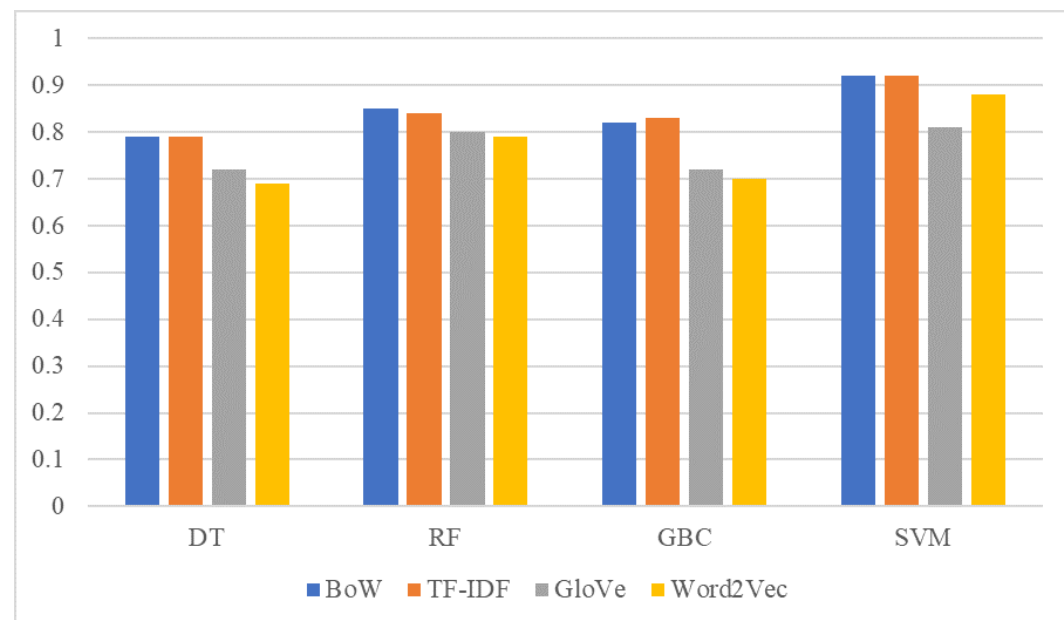
Figure 5. Performance comparison between machine learning models using TextBlob dataset and BoW, TF-IDF, GloVe, Word2Vec features.

The performance of machine learning models is good when used with TF-IDF features extracted from the original dataset and SVM outperforms with a significant 0.89 accuracy score. TF-IDF generates a weighted feature set as compared to BoW, GloVe, and Word2Vec features which helps to improve the accuracy of learning models. On the other hand, the accuracy of DT is reduced by 1% from 72% to 71% because DT is a rule-based model that performs well on simple term frequency as compared to weighted features. Weighted features introduce complexity in the DT learning process. SVM performs well because TF-IDF provides a linear feature set with the binary class which is more suitable for SVM that performs better being the linear model. The performance of machine learning models is improved with TextBlob data annotation. Machine learning models perform well with TF-IDF and BoW features and SVM obtains the highest accuracy of 0.92 accuracy score using TextBlob labels.

Performance of Deep Learning Models

To compare the performance of the proposed approach with the latest deep learning approach, experiments have been performed using several deep learning models. For this purpose, state-of-the-art deep learning models are used such as LSTM, CNN-LSTM, and Gated Recurrent Unit (GRU). The architecture of used deep learning models is provided in Figure 6. These models are used with the TextBlob annotated dataset owing to the superior results on the dataset from machine learning models.

The performance of deep learning models is also good similar to machine learning models, as shown in Table 22. CNN-LSTM achieves a 0.90 accuracy score while GRU has a 0.86 accuracy. The results of deep learning models are shown in Table 6. Overall, the performance of deep learning models is slightly lower than the machine learning models. Regarding the machine learning model, SVM gives the highest accuracy of 0.92 while the deep learning model CNN-LSTM achieves a 0.90 accuracy. The significant performance of machine learning models is because of handcrafted TF-IDF weighted features.
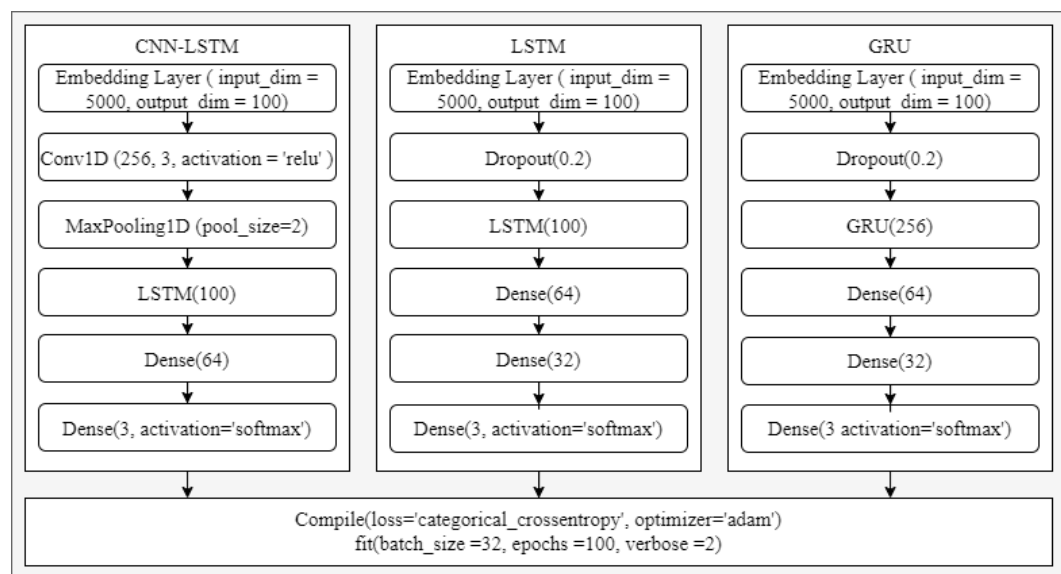
Figure 6. LSTM, CNN-LSTM, and GRU architectures .

Table 22. Performance analysis of deep learning models

Model	Accuracy	Class	Precision	Recall	F1 Score
LSTM	0.80	Neg.	0.83	0.79	0.81
		Pos.	0.93	0.93	0.94
		Avg.	0.88	0.87	0.87
CNN-LSTM	0.90	Neg.	0.78	0.88	0.83
		Pos.	0.96	0.91	0.93
		Avg.	0.87	0.90	0.88
GRU	0.86	Neg.	0.84	0.88	0.86
		Pos.	0.88	0.83	0.85
		Avg.	0.86	0.86	0.86

Performance Analysis with State-of-The-Art Approaches

Performance analysis has been carried out to analyze the performance of the proposed approach with other state-of-the-art approaches that utilize the IMDB movie reviews analysis. Comparison results are provided in Table 23. Results indicate that the proposed methodology can achieve competitive results to that of state-of-the-art approaches. The use of SVM with TF-IDF and BoW using the TextBlob technique provides an accuracy of 92% which is better than the state-of-the-art approaches.

Statistical T-test

The T-test is performed in this study to show the statistical significance of the proposed approach (Omar et al., 2021). The T-test is applied to SVM results with the proposed approach and original dataset. The output from the T-test favors either null hypothesis or alternative hypothesis.

- Accept Null Hypothesis: This means that the compared results are statistically equal.

- Reject Null Hypothesis: This means that the compared results are not statistically equal.

The output values of T-test in terms of T-statistic and critical Value are shown in Table 24. T-test infers that the null hypothesis can be rejected in favor of the alternative hypothesis because the T-statistic value is less than the critical value indicating that the compared values are statistically different from each other.

CONCLUSION

With an ever-growing production of cinema movies, web series, and television dramas, a large number of reviews can be found on social platforms and movies websites like IMDB. Sentiment analysis of such

Table 23. Performance analysis of the proposed methodology.

Year	Reference	Model	Accuracy
2016	(Sahu and Ahuja, 2016)	RF	0.90
2017	(Yenter and Verma, 2017)	CNN+LSTM	0.895
2017	(Giatsoglou et al., 2017)	BoW-DOUBLE and Average emotion-DOUBLE	0.83
2018	(Mathapati et al., 2018)	CNN	0.89
2019	(Ali et al., 2019)	CNN+LSTM	0.89
2019	(Bodapati et al., 2019)	LSTM + DNN	0.885
2020	(Tripathi et al., 2020)	TF-IDF +LR	0.891
2020	(Qaisar, 2020)	LSTM	0.899
2020	(Shaukat et al., 2020)	NN	0.91
2021	(Jain and Jain, 2021a)	CNN	0.883
2021	(Nafis and Awang, 2021)	SVM + (SVM-RFE)	0.895
2021	(Jain and Jain, 2021b)	NB+ ARM	0.784
2021	Proposed	SVM + TextBlob +BoW & TF-IDF	0.92

Table 24. Statistical T-test output values

Student T-test output parameters	Output value
T-statistic	-0.182
Critical Value	0.000

reviews can provide insights about the movies, their team, and cast to millions of viewers. This study proposes a methodology to perform sentiment analysis on the movie reviews using supervised machine learning classifiers to assist the people in selecting the movies based on the popularity and interest of the reviews. Four machine learning algorithms including DT, RF, GBC, and SVM are utilized for sentiment analysis that is trained on the dataset preprocessed through a series of steps. Moreover, four feature extraction approaches including BoW, TF-IDF, GloVe, and Word2Vec are investigated for their efficacy in extracting the meaningful and effective features from the reviews. Results indicate that SVM achieves the highest accuracy among all the classifiers with an accuracy of 89.55% when trained and tested using TF-IDF features. The performance using BoW features is also good with an accuracy of 87.25%. Contrary to BoW which counts the occurrence of unique tokens, TF-IDF also records the importance of rare terms by assigning a higher weight to rare terms and perform better than BoW. However, the performance of the classifiers is greatly affected by GloVe and Word2Vec features which suggest that word embedding does not work well with the movie review dataset. For improving the performance of models and reducing the influence of contradictions found in the expressed sentiments and assigned labels, TextBlob is used for data annotation. Experimental results on TextBlob annotated dataset indicates that SVM achieves the highest accuracy of 92% with TF-IDF features. Compared to the standard dataset, the TextBlob assigned labels result in better performance from the models. The performance of deep learning models is slightly lower than machine learning models with the highest accuracy of 0.90 by the CNN-LSTM. Despite the equal number of positive and negative reviews used for training, the prediction accuracy for the positive and negative classes is different. Precision, recall, and F1 score indicate the models have a good fit, and performance evaluation metrics are in agreement. The current study excludes the neutral class due to a low number of samples and experiments are performed using positive and negative classes. Consequently, the accuracy may have been higher as compared to that with three classes. Similarly, probable class imbalance by adding neutral class samples is not investigated and is left for the future. We intend to perform further experiments using movie reviews datasets from other sources in the future. Furthermore, the study on finding the contradictions in the sentiments expressed in the reviews and the assigned labels is also under consideration.

Financial disclosure

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (NRF-2019R1A2C1006159) and MSIT(Ministry of Science and ICT), Korea, under the ITRC(Information Technology Research Center) support pro-

538 gram (IITP-2020-2016-0-00313) supervised by the IITP(Institute for Information & communications
539 Technology Promotion).

REFERENCES

541 (2018). Imdb dataset. https://www.kaggle.com/lakshmi25npathi/imdb-dataset-of-50k-movie-reviews.
542   Accessed: 30 October 2021.
543 (2020). Textblob documentation. https://textblob.readthedocs.io/en/dev/. Accessed: 22 December 2021.
544 Agarwal, A., Xie, B., Vovsha, I., Rambow, O., and Passonneau, R. J. (2011). Sentiment analysis of twitter
545   data. In *Proceedings of the workshop on language in social media (LSM 2011)*, pages 30–38.
546 Agarwal, B. and Mittal, N. (2016). Machine learning approach for sentiment analysis. In *Prominent
547   feature extraction for sentiment analysis*, pages 21–45. Springer.
548 Ali, N. M., Abd El Hamid, M. M., and Youssif, A. (2019). Sentiment analysis for movies reviews dataset
549   using deep learning models. *International Journal of Data Mining & Knowledge Management Process
550   (IJDKP) Vol*, 9.
551 Alpaydin, E. (2020). *Introduction to machine learning*. MIT press.
552 Ashraf, I., Hur, S., Shafiq, M., Kumari, S., and Park, Y. (2019a). Guide: Smartphone sensors-based
553   pedestrian indoor localization with heterogeneous devices. *International Journal of Communication
554   Systems*, 32(15):e4062.
555 Ashraf, I., Hur, S., Shafiq, M., and Park, Y. (2019b). Floor identification using magnetic field data with
556   smartphone sensors. *Sensors*, 19(11):2538.
557 Ayyadevara, V. K. (2018). Gradient boosting machine. In *Pro machine learning algorithms*, pages
558   117–134. Springer.
559 Bakshi, R. K., Kaur, N., Kaur, R., and Kaur, G. (2016). Opinion mining and sentiment analysis. In *2016
560   3rd International Conference on Computing for Sustainable Global Development (INDIACom)*, pages
561   452–455. IEEE.
562 Bennett, K. P. and Campbell, C. (2000). Support vector machines: hype or hallelujah? *Acm Sigkdd
563   Explorations Newsletter*, 2(2):1–13.
564 Bhoir, S., Ghorpade, T., and Mane, V. (2017). Comparative analysis of different word embedding models.
565   In *2017 International Conference on Advances in Computing, Communication and Control (ICAC3)*,
566   pages 1–4. IEEE.
567 Biau, G. and Scornet, E. (2016). A random forest guided tour. *Test*, 25(2):197–227.
568 Bodapati, J. D., Veeranjaneyulu, N., and Shaik, S. (2019). Sentiment analysis from movie reviews using
569   lstms. *Ingénierie des Systèmes d Inf.*, 24(1):125–129.
570 Bruce, L. M., Koger, C. H., and Li, J. (2002). Dimensionality reduction of hyperspectral data using
571   discrete wavelet transform feature extraction. *IEEE Transactions on geoscience and remote sensing*,
572   40(10):2331–2338.
573 Cortes, C. and Vapnik, V. (1995). Support-vector networks. *Machine learning*, 20(3):273–297.
574 Dessi, D., Helaoui, R., Kumar, V., Reforgiato Recupero, D., and Riboni, D. (2020). Tf-idf vs word
575   embeddings for morbidity identification in clinical notes: An initial study. In *1st Workshop on Smart
576   Personal Health Interfaces, SmartPhil 2020*, volume 2596, pages 1–12. CEUR-WS.
577 Giatsoglou, M., Vozalis, M. G., Diamantaras, K., Vakali, A., Sarigiannidis, G., and Chatzisavvas,
578   K. C. (2017). Sentiment analysis leveraging emotions and word embeddings. *Expert Systems with
579   Applications*, 69:214–224.
580 Goel, A., Gautam, J., and Kumar, S. (2016). Real time sentiment analysis of tweets using naive bayes.
581   In *2016 2nd International Conference on Next Generation Computing Technologies (NGCT)*, pages
582   257–261. IEEE.
583 Guzman, E. and Maalej, W. (2014). How do users like this feature? a fine grained sentiment analysis
584   of app reviews. In *2014 IEEE 22nd international requirements engineering conference (RE)*, pages
585   153–162. IEEE.
586 Hakak, S., Alazab, M., Khan, S., Gadekallu, T. R., Maddikunta, P. K. R., and Khan, W. Z. (2021). An
587   ensemble machine learning approach through effective feature extraction to classify fake news. *Future
588   Generation Computer Systems*, 117:47–58.
589 Hand, D. J. and Adams, N. M. (2014). Data mining. *Wiley StatsRef: Statistics Reference Online*, pages
590   1–7.
591 Hearst, M. (2003). What is text mining. *SIMS, UC Berkeley*, 5.

Ikonomakis, M., Kotsiantis, S., and Tampakas, V. (2005). Text classification using machine learning techniques. *WSEAS transactions on computers*, 4(8):966–974.

Jain, A. and Jain, V. (2021a). Effect of activation functions on deep learning algorithms performance for imdb movie review analysis. In *Proceedings of International Conference on Artificial Intelligence and Applications*, pages 489–497. Springer.

Jain, A. and Jain, V. (2021b). Efficient framework for sentiment classification using apriori based feature reduction. *EAI Endorsed Transactions on Scalable Information Systems*, 8(31):e3.

Khalid, M., Ashraf, I., Mehmood, A., Ullah, S., Ahmad, M., and Choi, G. S. (2020). Gbsvm: Sentiment classification from unstructured reviews using ensemble classifier. *Applied Sciences*, 10(8):2788.

Landy, S. D. and Szalay, A. S. (1993). Bias and variance of angular correlation functions. *The Astrophysical Journal*, 412:64–71.

Lee, S. H., Cui, J., and Kim, J. W. (2016). Sentiment analysis on movie review through building modified sentiment dictionary by movie genre. *Journal of intelligence and information systems*, 22(2):97–113.

Liu, B. et al. (2010). Sentiment analysis and subjectivity. *Handbook of natural language processing*, 2(2010):627–666.

Liu, Y., Huang, X., An, A., and Yu, X. (2008). Modeling and predicting the helpfulness of online reviews. In *2008 Eighth IEEE international conference on data mining*, pages 443–452. IEEE.

Loria, S. (2018). textblob documentation. *Release 0.15*, 2:269.

Mathapati, S., Adur, A. K., Tanuja, R., Manjula, S., and Venugopal, K. (2018). Collaborative deep learning techniques for sentiment analysis on imdb dataset. In *2018 Tenth International Conference on Advanced Computing (ICoAC)*, pages 361–366. IEEE.

Minaee, S., Azimi, E., and Abdolrashidi, A. (2019). Deep-sentiment: Sentiment analysis using ensemble of cnn and bi-lstm models. *arXiv preprint arXiv:1904.04206*.

Mouthami, K., Devi, K. N., and Bhaskaran, V. M. (2013). Sentiment analysis and classification based on textual reviews. In *2013 international conference on Information communication and embedded systems (ICICES)*, pages 271–276. IEEE.

Mujahid, M., Lee, E., Rustam, F., Washington, P. B., Ullah, S., Reshi, A. A., and Ashraf, I. (2021). Sentiment analysis and topic modeling on tweets about online education during covid-19. *Applied Sciences*, 11(18):8438.

Nafis, N. S. M. and Awang, S. (2021). An enhanced hybrid feature selection technique using term frequency-inverse document frequency and support vector machine-recursive feature elimination for sentiment classification. *IEEE Access*, 9:52177–52192.

Neethu, M. S. and Rajasree, R. (2013). Sentiment analysis in twitter using machine learning techniques. In *2013 Fourth International Conference on Computing, Communications and Networking Technologies (ICCCNT)*, pages 1–5.

Oghina, A., Breuss, M., Tsagkias, M., and De Rijke, M. (2012). Predicting imdb movie ratings using social media. In *European Conference on Information Retrieval*, pages 503–507. Springer.

Omar, B., Rustam, F., Mehmood, A., Choi, G. S., et al. (2021). Minimizing the overlapping degree to improve class-imbalanced learning under sparse feature selection: application to fraud detection. *IEEE Access*, 9:28101–28110.

Pang, B., Lee, L., and Vaithyanathan, S. (2002a). Thumbs up? sentiment classification using machine learning techniques. *arXiv preprint cs/0205070*.

Pang, B., Lee, L., and Vaithyanathan, S. (2002b). Thumbs up? sentiment classification using machine learning techniques. *arXiv preprint cs/0205070*.

Prabowo, R. and Thelwall, M. (2009). Sentiment analysis: A combined approach. *Journal of Informetrics*, 3(2):143–157.

Qaisar, S. M. (2020). Sentiment analysis of imdb movie reviews using long short-term memory. In *2020 2nd International Conference on Computer and Information Sciences (ICCIS)*, pages 1–4. IEEE.

Rokach, L. and Maimon, O. (2005). Clustering methods. In *Data mining and knowledge discovery handbook*, pages 321–352. Springer.

Rupapara, V., Rustam, F., Amaar, A., Washington, P. B., Lee, E., and Ashraf, I. (2021a). Deepfake tweets classification using stacked bi-lstm and words embedding. *PeerJ Computer Science*, 7:e745.

Rupapara, V., Rustam, F., Shahzad, H. F., Mehmood, A., Ashraf, I., and Choi, G. S. (2021b). Impact of smote on imbalanced text features for toxic comments classification using rvvc model. *IEEE Access*.

Rustam, F., Khalid, M., Aslam, W., Rupapara, V., Mehmood, A., and Choi, G. S. (2021). A performance

comparison of supervised machine learning models for covid-19 tweets sentiment analysis. *Plos one*, 16(2):e0245909.

Rustam, F., Mehmood, A., Ullah, S., Ahmad, M., Khan, D. M., Choi, G. S., and On, B.-W. (2020). Predicting pulsar stars using a random tree boosting voting classifier (rtb-vc). *Astronomy and Computing*, 32:100404.

Sahu, T. P. and Ahuja, S. (2016). Sentiment analysis of movie reviews: A study on feature selection & classification algorithms. In *2016 International Conference on Microelectronics, Computing and Communications (MicroCom)*, pages 1–6. IEEE.

Sahu, T. P. and Ahuja, S. (2016). Sentiment analysis of movie reviews: A study on feature selection classification algorithms. In *2016 International Conference on Microelectronics, Computing and Communications (MicroCom)*, pages 1–6.

Shaukat, Z., Zulfiqar, A. A., Xiao, C., Azeem, M., and Mahmood, T. (2020). Sentiment analysis on imdb using lexicon and neural networks. *SN Applied Sciences*, 2(2):1–10.

Singh, V., Piryani, R., Uddin, A., and Waila, P. (2013a). Sentiment analysis of movie reviews and blog posts. In *2013 3rd IEEE International Advance Computing Conference (IACC)*, pages 893–898. IEEE.

Singh, V. K., Piryani, R., Uddin, A., and Waila, P. (2013b). Sentiment analysis of movie reviews: A new feature-based heuristic for aspect-level sentiment classification. In *2013 International Mutli-Conference on Automation, Computing, Communication, Control and Compressed Sensing (iMac4s)*, pages 712–717. IEEE.

Tan, P.-N., Steinbach, M., and Kumar, V. (2006). Introduction to data mining, pearson education. *Inc., New Delhi*.

Tripathi, S., Mehrotra, R., Bansal, V., and Upadhyay, S. (2020). Analyzing sentiment using imdb dataset. In *2020 12th International Conference on Computational Intelligence and Communication Networks (CICN)*, pages 30–33. IEEE.

Visa, S., Ramsay, B., Ralescu, A. L., and Van Der Knaap, E. (2011). Confusion matrix-based feature selection. *MAICS*, 710:120–127.

Vishwanathan, S. and Murty, M. N. (2002). Ssvm: a simple svm algorithm. In *Proceedings of the 2002 International Joint Conference on Neural Networks. IJCNN'02 (Cat. No. 02CH37290)*, volume 3, pages 2393–2398. IEEE.

Wang, Z., Ma, L., and Zhang, Y. (2016). A hybrid document feature extraction method using latent dirichlet allocation and word2vec. In *2016 IEEE first international conference on data science in cyberspace (DSC)*, pages 98–103. IEEE.

Yenter, A. and Verma, A. (2017). Deep cnn-lstm with combined kernels from multiple branches for imdb review sentiment analysis. In *2017 IEEE 8th Annual Ubiquitous Computing, Electronics and Mobile Communication Conference (UEMCON)*, pages 540–546. IEEE.

Yu, B. (2008). An evaluation of text classification methods for literary study. *Literary and Linguistic Computing*, 23(3):327–343.

Zhang, C., Zuo, W., Peng, T., and He, F. (2008). For chinese reviews using machine learning methods based on string kernel. In *2008 Third International Conference on Convergence and Hybrid Information Technology*, volume 2, pages 909–914. IEEE.