Educational Administration: Theory and Practice
2023, 29(4), 1724-1734
ISSN: 2148-2403
https://kuey.net/

Research Article

Sentiment Analysis Of Movie Review Using Machine Learning

Anshika Gupta[1], Sweta Pandey[2], Mandakini Priyadarshani Behera[3], Subhashree Darshana[4*], Adyasha Dash[5]

[1,2,3,4*,5]School of Computer-Engineering, KIIT (Deemed) University, Bhubaneswar, Odisha, India 751024

***Corresponding author:** Subhashree Darshana
*School of Computer-Engineering, KIIT (Deemed) University, Bhubaneswar, Odisha, India 751024 Email:-subhashree.darshanafcs@kiit.ac.in

Citation: Subhashree Darshana et.al (2023), Sentiment Analysis Of Movie Review Using Machine Learning*Educational Administration: Theory and Practice, 29(4), 1724-1734*
Doi: 10.53555/kuey.v29i4.6616

ARTICLE INFO **ABSTRACT**

This research applies sentiment analysis techniques to classify movie reviews from IMDB as positive or negative. The research involved analyzing sentiment analysis papers, studying classification algorithms, and collecting IMDB movie review data. The study utilizes or implements machine learning models like KNN, SVM, Logistic Regression, and Random Forest on preprocessed movie review text. Key steps include feature extraction, sentiment expression examination, feature ranking, and training a multilevel classifier. By leveraging these methods, the approach achieves great accuracy in correctly classifying the sentiment polarity of IMDB movie reviews, demonstrating its effectiveness for this opinion-mining task on review data.

Keywords: Sentiment analysis, Movie reviews, KNN, SVM, Logistic Regression, Tf-idf ,n-gram, Random Forest.

Introduction:

The rise of social media and online platforms has led to a massive collection of user-generated opinionated text data, including movie reviews. Sentiment analysis is the computational study of identifying emotions and polarities in text and enables us to extract valuable information from data. In, the movie industry, sentiment analysis of user reviews can provide insights into audience reactions, predict box office performance, and inform marketing strategies. This research aims to develop and evaluate robust sentiment analysis models tailored for movie review data. Using the techniques of Natural Language Processing and algorithms of machine learning, we examine different approaches for text data preprocessing, relevant feature identification, and accurate sentiment categorization of reviews. We aim to achieve high classification accuracy levels, determine the key factor affecting model performance, and provide insights into the advantages and limitations of various techniques when applied to the domain of movie review sentiment analysis.
Sentiment analysis involves three key phases:

Data Preprocessing: Cleaning raw text data to remove noise and inconsistencies.
Feature Extraction: Identifying and numerically representing sentiment-bearing words or phrases.
Classification: Using machine learning algorithms to categorize text into sentiment classes (positive, negative, neutral) based on extracted features.

Literature Review : Sentimental Analysis using Machine Learning approaches offer classification solutions through training data and model-based classification using Naive Bayes for unrevealed data.

$$P_{NB}(c/d) = \frac{P(c)(\pi_{i=1}^{m} P(f_i|c)\, n_i^{(d)})}{P(d)}$$

Naive Bayes uses data from different social media platforms like YouTube, Twitter, and Facebook to calculate accuracy, recall, and precision.

Manu Kumar and Manju Bala in 2016 [2] their study suggests that analyzing large amounts of unstructured data from social media platforms like YouTube, Twitter, and Facebook is challenging. To address this, they utilized cloud services and Hadoop for intelligent data analysis, specifically focusing on Twitter sentiment analysis.
The study uses Naive Bayes classifier for sentiment analysis on Big Data using Hadoop, dividing problems into Training, combining and classifying jobs. The results are compared to a virtual Hadoop cluster constructed on the cloud, albeit with a weaker cloud.

Joscha et. al, in their paper [1] compared Bag of words models and n-grams techniques for improving sentiment analysis performance by considering semantic associations between sentences and document parts and highlighting their limitations.

Minhoe Hur et al. in 2016 [3] in their paper proposed a system to predict box-office collection using movie reviews, using viewer opinions as input and machine learning algorithms like ANN (Artificial Neural Network), Regression, and SVM (Support Vector Machine) to establish a non-linear relationship between the box office and its collection predictors.

Ahmad Kamal in his paper [4] developed an opinion-mining framework using supervised machine learning for objectivity and subjectivity analysis, feature extraction, and review summarization.

Humera Shaziya et al. in their paper [5] used WEKA Tool to analyze the movie reviews for sentiment analysis, enhancing previous work on categorizing opinions. They considered multiple-person reviews and found that Naïve Bayes performed better than Support Vector Machine (SVM) for movie reviews and text.

Pang & Lee work [6] work is a standard in sentimental analysis of movie reviews, focusing on overall sentiment rather than topic. They suggest that traditional machine learning methods provide better outcomes than human-created baselines. However, their three machine learning techniques (Naive Bayes, Maximum entropy Classification, and Support vector machines) do not offer as effective sentiment grouping outcomes as traditional classifier-based classification.

Machine Learning Methods:

1. Term Frequency — Inverse Document Frequency (TFIDF):

It is a vectorization technique based on Bag of Words (BoW). It is more efficient than the BoW model because it takes into account the importance of each word in the document.

$$TF = \frac{Number\ of\ times\ a\ word\ "X"\ appears\ in\ a\ Document}{Number\ of\ words\ present\ in\ a\ Document}$$

$$IDF = log\left(\frac{Number\ of\ Documents\ present\ in\ a\ Corpus}{Number\ of\ Documents\ where\ word\ "X"\ has\ appeared}\right)$$

$$TF\,IDF = TF * IDF$$

2. N-gram model:

N-grams in Natural Language Processing (NLP) is a sequence of n words extracted from text for analysis, storing information about the content and relationships between words, and can be as short as one word (unigram) or 2 words (bigram) or 3 words (trigram), etc.

The N-gram model is a probabilistic language for N-1 words that predicts the most frequently used words in a sequence and can be used in speech recognition and machine translation.(Fig-6, Fig-7, Fig-8).

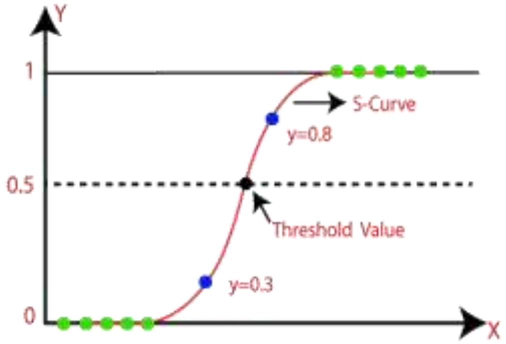
We used Bag of Word (BoW) for this model. It is a statistical language model that analyzes text and information by word count, regardless of the word order.

3. Random Forest:

Random Forest is an ensemble learning method that employs multiple decision trees to make predictions. It falls under supervised learning domain and can be used both in classification and regression problems. Instead of relying on a single decision tree, it makes predictions from each tree and predicts the final outcome based on the maximum number of votes predicted. The more trees in the forest, the more they are given, and the problem of overfitting is prevented.

4. Logistic Regression:

Logistic regression is a method used to create training models that describe data and relationships between one variable and another, or multiple variables.

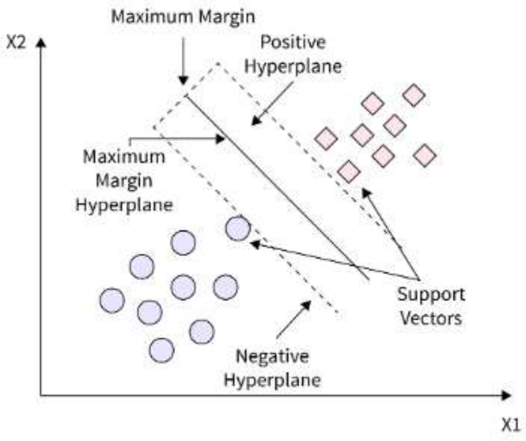


5. K-Nearest Neighbour:

K-NN finds the K points closest to the new point and predicts the label or value based on the labels or values of the K people nearby. It usually uses the Euclidean distance to calculate the distance, and the distribution or average (regression) returns the labels from the neighbors K.

6. Support Vector Machine (SVM):

It is a supervised machine learning algorithm for classification and regression. It works by finding the best hyperplane that separates data points into different groups. This hyperplane is created in a high-dimensional space to distinguish between groups. By determining support vectors (data points closest to the plane), SVM performs a classification or reclassification function to show the separation of classes.



Methodology :

We have used a dataset that has 40k movie reviews from IMDB that have been labeled with a positive (1) or negative (0) label. First, we have reduced the dataset to 10 percent, keeping 50 percent for the training set and 25 percent for the validation and test sets.

The accuracy of a system is measured by its recognition accuracy as a percentage of test text input to trained emotional text data.

The precision rate represents the proportion of emotions accurately identified within a particular class, relative to the total number of emotions classified across all classes.

Recall measures the accuracy of a machine learning model in identifying true positives from all actual positive samples in a dataset.

The F-measure represents a combined metric that weighs both the precision rate and the recall rate, providing an assessment of the system's overall performance by factoring in correct identifications while disregarding incorrect recognition instances.

Pre-processing:

We have created a function that is pre-processing and cleaning the data using lemmatization.

Lemmatization

This function clean, preprocesses text data for natural language processing tasks using the NLTK library. The clean function It initializes the WordNet lemmatizer, tokenizes sentences, removes capital letters, removes stopwords, removes non-alphanumeric characters, and lemmatizes the tokens. (Fig-4).

After cleaning, we created lists for positive and negative words, and then we visualized the most common ones with WordClouds(Fig-5).

TF-IDF

After we explored the data, we used TfidfVectorizer to turn our data into a matrix for analysis. The text uses the term frequency and inverse document frequency (TF-IDF) techniques to convert text into numerical representations. This technique assigns higher weights to frequent and rare terms, capturing the discriminative power of words and identifying key features.

It reduces the influence of non-informative words, improving the accuracy and effectiveness of tasks like document classification, information retrieval, and text clustering.

The model is trained using random forest, KNN, SVM, and logistic regression and evaluated using cross-validation scores.

Random Forest Classifier

It is used to construct a model for classification tasks.For assessing the model's effectiveness, cross-validation methodologies are employed. The model exhibiting the best performance metrics is selected as the optimal choice. This selection is done by using estimators.

Then GridSeachCV is leveraged for hyperparameter tuning of the RandomForestClassifier. This systematic approach finds the best hyperparameter.

Logistic Regression

Evaluating a logistic regression model's performance on test data typically involves calculating the accuracy score and generating a confusion matrix. The accuracy score gives an overall measure of the model's performance.

The confusion matrix provides more detailed insights by displaying the breakdown of true positives, false positives, true negatives, and false negatives. Analyzing the confusion matrix allows us to find specific types of errors the model is making.

K-Nearest Neighbour

After training the K-Nearest Neighbors (KNN) classifier on the given dataset, its performance is evaluated. A confusion matrix is generated to visually compare the model's predicted labels against truth values, enabling the identification of specific misclassification patterns.

Additionally, a classification report is generated which quantifies metrics like precision, recall, and F1 score of each class enabling a better understanding of the classifier's efficiency in handling different categories of data. By generating both these reports, we can get comprehensive insights into the KNN model's predictive capabilities and strengths.

Support Vector Machine

Comparing the performance of SVM models with and without scaling, we uncover how preprocessing techniques like feature scaling can influence the classification accuracy and robustness of the model.

Leveraging classification matrix and visualizations, we are carefully examining the effects of scaling, which offers a deeper understanding of its effectiveness in enhancing SVM performance across diverse classification tasks.

Results:

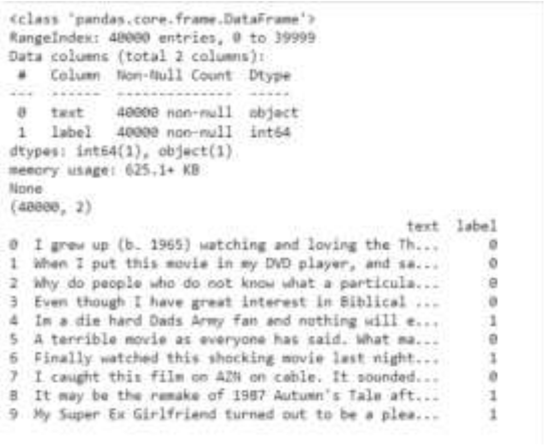
Exploratory data analysis:

Pre-processing

We reduced the data set to 4000 and then we kept 2000 for the training set and 1000 for both the validation set and the test set each.

The train_test_split function is a tool in Scikit-Learn that splits the dataset into a training subset and a testing subset.

The test_size parameter is used to determine the proportion of the original dataset to be included in the test split.



```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 40000 entries, 0 to 39999
Data columns (total 2 columns):
 #   Column  Non-Null Count  Dtype 
---  ------  --------------  ----- 
 0   text    40000 non-null  object
 1   label   40000 non-null  int64 
dtypes: int64(1), object(1)
memory usage: 625.1+ KB
None
(40000, 2)
                                                text  label
0  I grew up (b. 1965) watching and loving the Th...      0
1  When I put this movie in my DVD player, and sa...      0
2  Why do people who do not know what a particula...      0
3  Even though I have great interest in Biblical ...      0
4  Im a die hard Dads Army fan and nothing will e...      1
5  A terrible movie as everyone has said. What ma...      0
6  Finally watched this shocking movie last night...      1
7  I caught this film on AZN on cable. It sounded...      0
8  It may be the remake of 1987 Autumn's Tale aft...      1
9  My Super Ex Girlfriend turned out to be a plea...      1
```

Fig-1: Original Dataset

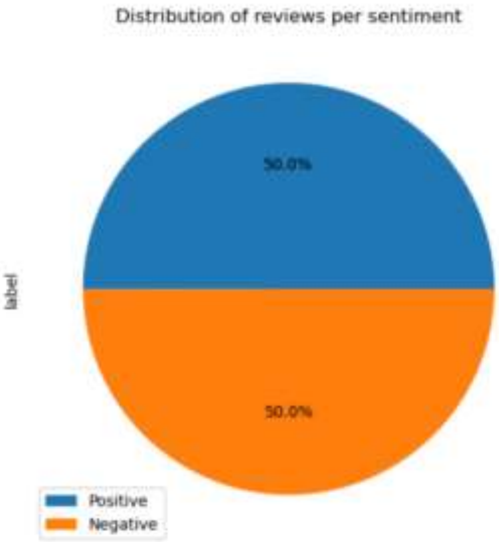
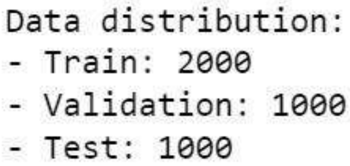


Fig-2: Distribution of positive and negative reviews

After train_test_split:

Below is the splitted data that we will use for the main analysis part.



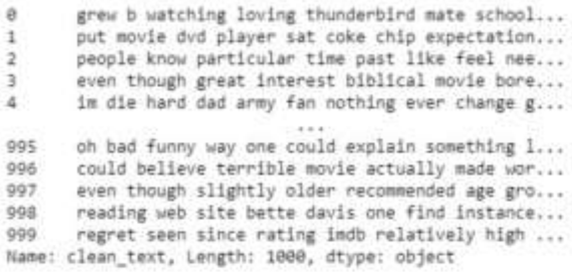
```
Data distribution:
- Train: 2000
- Validation: 1000
- Test: 1000
```

Fig-3: Data for evaluating the models

Lemmatization:

The NLTK library was used to preprocess text data for natural language processing tasks, including initializing the WordNet.

After tagging the sentences, removing capital letters, removing words, and non-alphanumeric characters, and lemmatizing the tags, we can see the finished text below.



```
0      grew b watching loving thunderbird mate school...
1      put movie dvd player sat coke chip expectation...
2      people know particular time past like feel nee...
3      even though great interest biblical movie bore...
4      im die hard dad army fan nothing ever change g...
                              ...
995    oh bad funny way one could explain something l...
996    could believe terrible movie actually made wor...
997    even though slightly older recommended age gro...
998    reading web site bette davis one find instance...
999    regret seen since rating imdb relatively high ...
Name: clean_text, Length: 1000, dtype: object
```

Fig-4: After Lemmatization

Then we created two different WordClouds for positive and negative reviews.
We visualized the most common words using WordCloud.

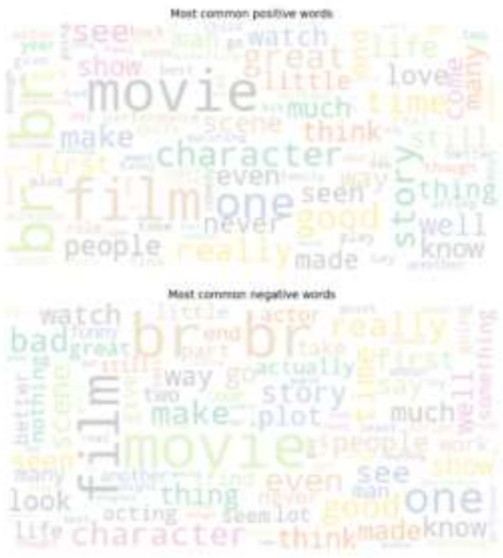


Fig-5: WordCloud for positive and negative words.

The n-gram model has been implemented to predict the most frequently occurring words in positive and negative reviews following specific sequences.

We used Bag of Word (BoW) for this. It is a statistical language model that analyzes text and information by word count, regardless of the word order. It can be used as a Python dictionary, where each key represents a word and the numerical occurrence of the value.

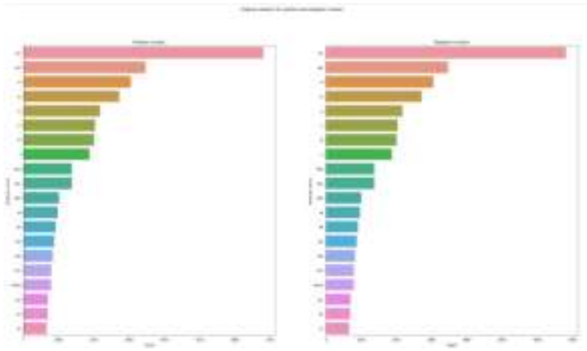


Fig-6: Unigram analysis for positive and negative reviews.

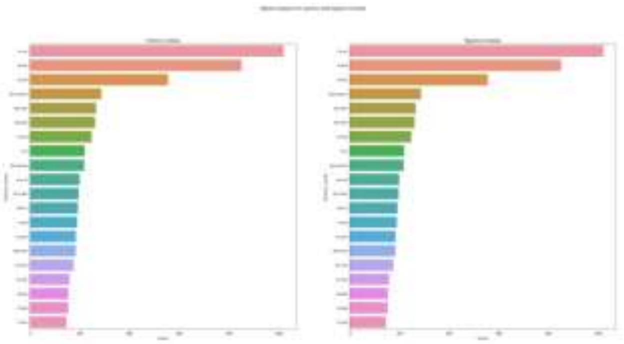


Fig-7: Bigram analysis for positive and negative reviews.

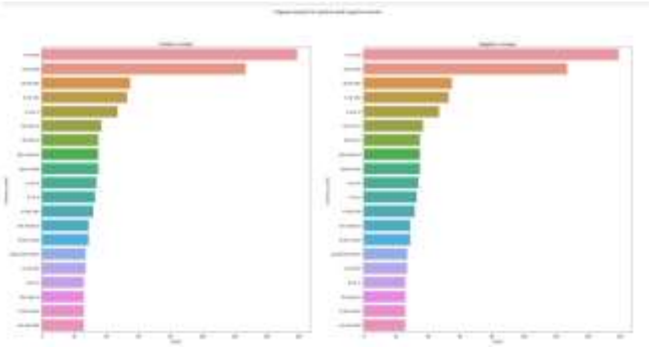
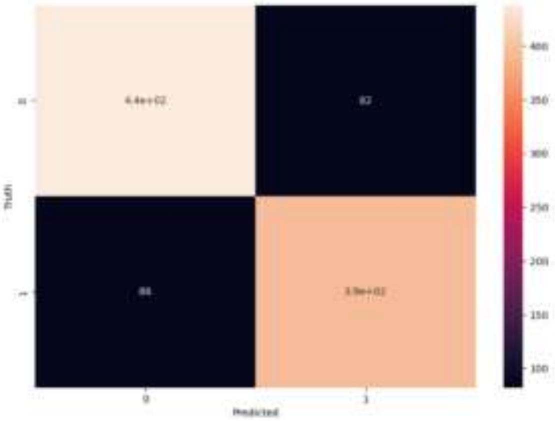
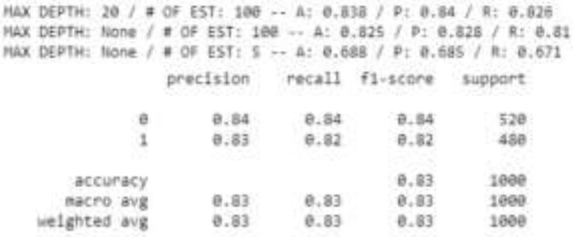


Fig-8: Trigram analysis for positive and negative reviews.

Random Forest:

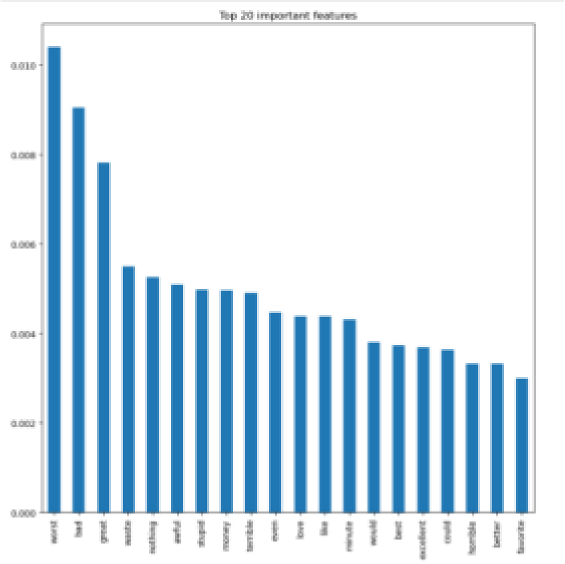


We did hyperparameter tuning using GridSearchCV. We evaluated different trees using different estimators and then choosing the best one.

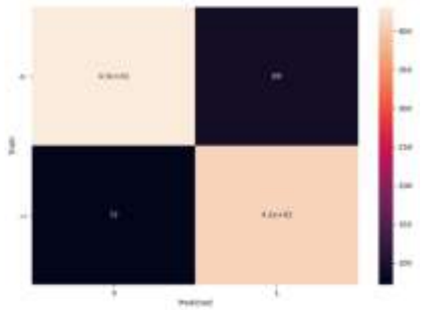


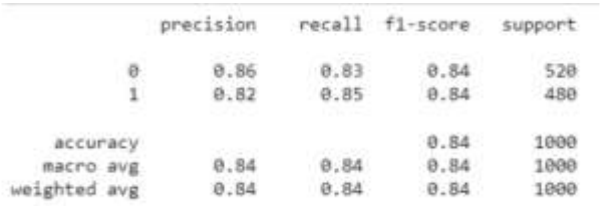
```
MAX DEPTH: 20 / # OF EST: 100 -- A: 0.838 / P: 0.84 / R: 0.826
MAX DEPTH: None / # OF EST: 100 -- A: 0.825 / P: 0.828 / R: 0.81
MAX DEPTH: None / # OF EST: 5 -- A: 0.688 / P: 0.685 / R: 0.671
              precision    recall  f1-score   support

           0       0.84      0.84      0.84       520
           1       0.83      0.82      0.82       480

    accuracy                           0.83      1000
   macro avg       0.83      0.83      0.83      1000
weighted avg       0.83      0.83      0.83      1000
```

We got the accuracy as: 0.83, precision as: 0.83, recall as: 0.83, and F1-score as: 0.83.
Now we can analyze the most important words to predict the correct results.



Logistic Regression :

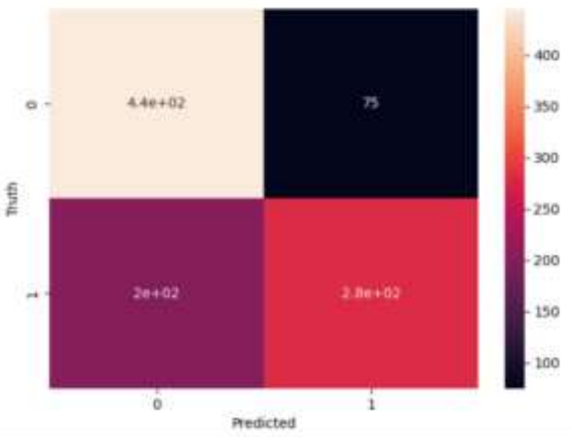


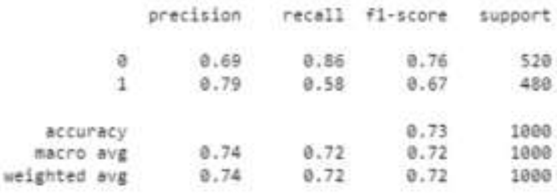


```
              precision    recall  f1-score   support

           0       0.86      0.83      0.84       520
           1       0.82      0.85      0.84       480

    accuracy                           0.84      1000
   macro avg       0.84      0.84      0.84      1000
weighted avg       0.84      0.84      0.84      1000
```

We got the accuracy as: 0.84, precision as: 0.84, recall as: 0.84, and F1-score as: 0.84.

KNN:



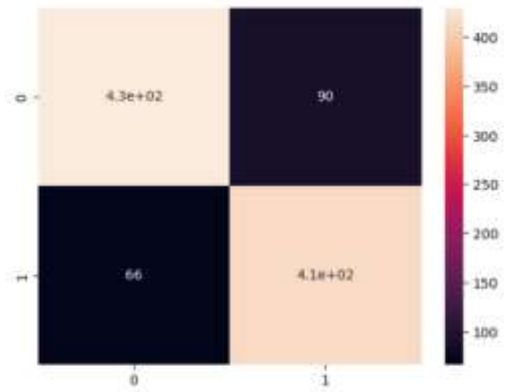


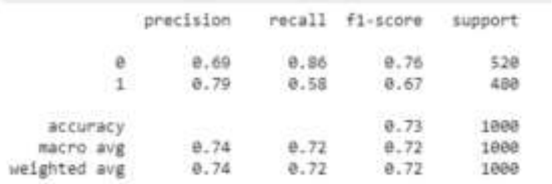
```
              precision    recall  f1-score   support

           0       0.69      0.86      0.76       520
           1       0.79      0.58      0.67       480

    accuracy                           0.73      1000
   macro avg       0.74      0.72      0.72      1000
weighted avg       0.74      0.72      0.72      1000
```

We got the accuracy as: 0.73, precision as: 0.74, recall as: 0.72, and F1-score as: 0.72.

SVM

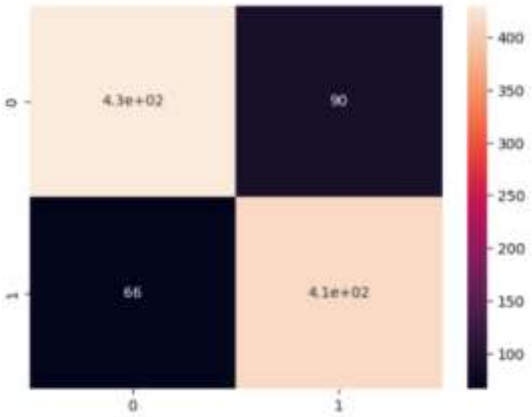
On unscaled data:

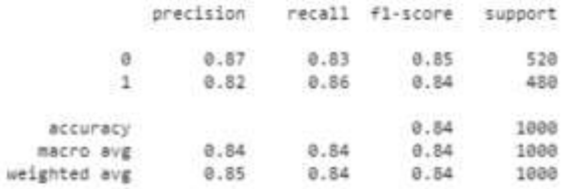




```
              precision    recall  f1-score   support

           0       0.69      0.86      0.76       520
           1       0.79      0.58      0.67       480

    accuracy                           0.73      1000
   macro avg       0.74      0.72      0.72      1000
weighted avg       0.74      0.72      0.72      1000
```

On scaled data:





```
              precision    recall  f1-score   support

           0       0.87      0.83      0.85       520
           1       0.82      0.86      0.84       480

    accuracy                           0.84      1000
   macro avg       0.84      0.84      0.84      1000
weighted avg       0.85      0.84      0.84      1000
```

We got the accuracy as: 0.84, precision as: 0.84, recall as: 0.84, and F1-score as: 0.84.

Now, we do a comparative analysis of these measuring metrics with the different models evaluated above and analyse which model performs best for all these metrics.

Algorithm	**Accuracy**	**Precision**	**Recall**	**F1-score**
Random Forest with estimator 100	0.83	0.83	0.83	0.83
Random Forest with estimator 5	0.68	0.68	0.67	0.68
Multinomial Logistic Regression	0.84	0.84	0.84	0.84
KNN	0.73	0.74	0.72	0.72
SVM	0.84	0.84	0.84	0.84

Table 1: Classification report.

Conclusion:

In this research, various techniques were used to classify the movie reviews from IMDB as positive or negative. The algorithms performed were machine KNN, SVM, Logistic Regression, Tf-idf, and Random Forest on preprocessed movie review text. Both SVM and logistic regression yielded identical and optimal outcomes in the experiment (score is-0.84). The precision of SVM and logistic regression is also equal to 0.84. These two models performed equally well on this dataset. And KNN performed the least.

To enhance the accuracy, it is necessary to evaluate additional algorithms besides the set examined, such as ANN, LSTM, BERT, or develop hybrid methodologies that combine multiple approaches. We can also use sentiment analysis on images by using Deep learning, VGGImageNet , ResNet-50 architecture. Analyzing the sentiment of the review can offer valuable insights in a variety of fields. Intelligent systems can be developed that can provide users with comprehensive reviews on their own without requiring the user to go through individual reviews.

Acknowledgments:

Thank you to all the people who have worked directly or indirectly on this paper.

References :

[1] Joscha Markle-Huß, Stefan Feuerriegel, Helmut Prendinger. 2017 Improving Sentiment Analysis with Document-Level Semantic Relationships from Rhetoric Discourse Structures, Proceedings of the 50th Hawaii International Conference on System Sciences.

[2] Monu Kumar Thapar University, Patiala “Analyzing Twitter sentiments through big data” , IEEE, 2016.

[3] Minhoe Hur Seoul National University “Box-office forecasting based on sentiments of movie reviews and Independent subspace method”, Information Sciences, 2016.

[4] Kamal A., 2015, Review Mining for Feature Based Opinion Summarization and Visualization.

[5] Humera Shaziya, G.Kavitha, Raniah Zaheer, 2015, Text Categorization of Movie Reviews for Sentiment Analysis , International Journal of Innovative Research in Science, Engineering and Technology, Vol. 4, ssue11.

[6] Pang, Bo, Lillian Lee, and ShivakumarVaithyanathan. "Thumbs up?: sentiment classification using machine learning techniques." Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10. Association for Computational Linguistics, 2002.

[7] Vryniotis Vasils and Vasilis Vryniotis” Machine Learning Tutorial: The Multinomial Logistic Regression (Softmax Regression) May 2017.

[8] Ankur Goel, Jyoti Gautam, Sitesh Kumar, “Real-time sentiment analysis of tweets using Naïve Bayes”, Next Generation Computing Technologies (NGCT), Electronic ISBN: 978-1-5090-32570 , 2016.

[9] Kamal, A. 2013 Subjectivity Classification using Machine Learning Techniques for Mining Feature-Opinion Pairs from Web Opinion Sources. International Journal of Computer Science Issues 10(5), 191-200.

[10] Tirath Prasad Sahu and Sanjeev Ahuja, "Sentiment Analysis of Movie Reviews: A study on Feature Selection & Classification Algorithms", IEEE, 2016.

[11] Bingwei Liu∗, Erik Blasch†, Yu Chen‡, Dan Shen∗ and Genshe Chen, "Scalable Sentiment Classification for Big Data Analysis Using Na¨ıve Bayes Classifier" Big Data, IEEE International Conference,2103 Electronic ISBN: 978-1-4799-1293-3, December 2013

[12] Shweta Rana, Archana Singh "Comparative analysis of sentiment orientation using SVM and Naïve Bayes techniques", Next Generation Computing Technologies(NGCT), Electronic ISBN: 978-1-5090-3257-0, 2016.

[13] J. S., Yang, M. H., Hwang, Y. J., Jeon, S. H., Kim, K. Y., Jung, I. S., ... & Na, J. H. (2012, November). Customer preference analysis based on SNS data. In 2012 Second International Conference on Cloud and Green Computing (pp. 609-613). IEEE.

[14] Saravanan, M., Sundar, D., & Kumaresh, V. S. (2013, December). Probing of geospatial stream data to report disorientation. In 2013 IEEE Recent Advances in Intelligent Computational Systems (RAICS) (pp. 227-232). IEEE.

[15] Chirgaiya, Sachin et al. 'Analysis of Sentiment Based Movie Reviews Using Machine Learning Techniques'. 1 Jan. 2021 : 5449 – 5456.

[16] Nassar, N., Jafar, A., & Rahhal, Y. (2020). A novel deep multi-criteria collaborative filtering model for recommendation system. Knowledge-Based Systems, 187, 104811.

[17] Sharma, S., Rana, V., & Malhotra, M. (2022). Automatic recommendation system based on hybrid filtering algorithm. Education and Information Technologies, 27(2), 1523-1538.

[18] Singh, R. H., Maurya, S., Tripathi, T., Narula, T., & Srivastav, G. (2020). Movie recommendation system using cosine similarity and KNN. International Journal of Engineering and Advanced Technology, 9(5), 556-559.

[19] Tran, D.D., Nguyen, T.T., Dao, T. (2022). Sentiment Analysis of Movie Reviews Using Machine Learning Techniques. In: Yang, XS., Sherratt, S., Dey, N., Joshi, A. (eds) Proceedings of Sixth International Congress on Information and Communication Technology. Lecture Notes in Networks and Systems, vol 235. Springer, Singapore.

[20] M. Pandey, S. Nayak and S. S. Rautaray, "An Analysis on Sentiment Analysis and Stock Market Price Prediction," 2023 2nd International Conference on Automation, Computing and Renewable Systems (ICACRS), Pudukkottai, India, 2023, pp. 367-370,

[21] Darshana, S., Rautaray, S.S., Pandey, M. (2021). AI to Machine Learning: Lifeless Automation and Issues. In: Pandey, M., Rautaray, S.S. (eds) Machine Learning: Theoretical Foundations and Practical Applications. Studies in Big Data, vol 87. Springer, Singapore.

[22] S. Darshana and K. Soumyakanta, "A Revolutionary Machine-Learning based approach for identifying Ayurvedic Medicinal Plants," 2022 International Conference on Advancements in Smart, Secure and Intelligent Computing (ASSIC), Bhubaneswar, India, 2022, pp. 1-5,

[23] Adyasha Dash, Subhashree Darshana, Devendra Kumar Yadav, Vinti Gupta,A clinical named entity recognition model using pretrained word embedding and deep neural networks, Decision Analytics Journal,Volume 10, 2024,100426, ISSN 2772-6622

[24] Jena, J.J., Satapathy, S.C. A new adaptive tuned Social Group Optimization (SGO) algorithm with sigmoid-adaptive inertia weight for solving engineering design problems. Multimed Tools Appl 83, 3021–3055 (2024).

[25] Ajeet Ram Pathak, Manjusha Pandey, Siddharth Rautaray, Application of Deep Learning for Object Detection, Procedia Computer Science, Volume 132, 2018, Pages 1706-1717, ISSN 1877-0509

[26] Mohanty D., Jena, B. Khuntia,T., Mohanty, P.K., Mohapatra, S., Behera, S. (2024). Green Transit: Harnessing Renewable Energy For Sustainable Integration. Educational Administration: Theory and Practice, 30(4), 7242–7254. https://www.kuey.net/index.php/kuey/article/view/2552