IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

A Mood Based Movie Recommendation System Using Machine Learning

[1]Anand Prakash Srivastava, [2]Dr Sanjeev Kumar Sharma,

[1]Research Scholar, [2]Professor,

[1]Computer Science & Engineering,

[1]Sanskriti University, Mathura, India

Abstract: This study presents a movie recommendation system tailored to the user's mood, using the Positive and Negative Affect Schedule (PANAS) and cosine similarity for recommendations. The system categorizes movies into genres that evoke positive or negative emotions, matching the user's current emotional state. The user's mood is determined by their responses to a PANAS questionnaire, guiding the selection of the appropriate genre dataset. From this dataset, a random movie is chosen, and ten similar movies are recommended using cosine similarity. The system's effectiveness was evaluated using three machine learning algorithms: Naive Bayes (Gaussian, Multinomial, Bernoulli), Support Vector Machine (SVM) with linear and radial basis function (rbf) kernels, and Decision Tree (DT) with log loss, gini, and entropy criteria. Models were trained on datasets comprising 70%, 75%, and 80% of the available data, and their performance was assessed. Results indicated that the Decision Tree method, particularly with the gini criterion, achieved the highest accuracy, while the SVM also performed well. Naïve Bayes showed the lowest accuracy. The Decision Tree algorithm's consistent and superior performance highlights its suitability for this recommendation system, whereas Naive Bayes was less effective for this application.

Index Terms **-** Movie recommendation system, PANAS, Mood-based recommendations, Genre categorization, User mood recommendation

I. INTRODUCTION

Recommendation systems [1] are adopted in various sectors like e-commerce, retail banking and entertainment. The main goal of these systems is to provide recommendations to users based on their data, which are collected constantly and analyzed. The most popular methods for recommendation system [2] are Content-based Filtering (CBF), Collaborative Filtering (CF) and Hybrid Filtering. With the use of CBF (Content based filtering) technique [3] which checks about the features of each item and suggests other items that have similar attributes. By exploring the correspondence between users and items, CF [4, 5] improves some drawbacks of CBF and provides suggestions. It uses the data of user's past selection and other likeminded users' choice to offer personalized recommendations. Many existing recommendation systems (RS) use a hybrid-filtering (HF) technique [6] that mixes the strengths of Content-Based filtering (CBF) and Collaborative Filtering (CF). A film rides on audience's feedback. Other users rely so much on these reviews while making their own choices. People tend to be more likely to choose a movie that has been well-received by the majority, rather than one that has been mostly disliked. Considering these reviews, excluding those that contain misleading information, also adds to the intricacy of decision-making. There is a potential solution to this problem through sentiment analysis. Utilizing Natural Language Processing (NLP), Sentiment Analysis [7] allows for the extraction of information from textual sources and the classification of statements, words, or documents as positive or negative. Understanding the author's perspective and sharing one's own experiences can be highly valuable. Opinion mining utilizes the principles of data mining to extract and categorize the viewpoints expressed in diverse online forums or venues. This facilitates a more comprehensive comprehension of the user's sentiments or emotions pertaining to a specific topic. [8]. Understanding the

differences between emotions and affect is essential when discussing moods. Psychological research differentiates between emotions and moods by explaining that emotions are strong sentiments directed towards someone or something, while moods are less intense feelings that don't require a specific stimulus. [9] provide a comprehensive definition of affect, which includes both emotions and moods. Several efforts were made to classify emotions into separate dimensions [10]. Overall, it was concluded that mood can be classified into either a positive affect or a negative affect dimension. According to [11], positive affect is mainly related to how enthusiastic, active, and alert a person feels. However, the concept of negative affect is a wide-ranging aspect of personal distress and unpleasant involvement that includes different types of unfavorable emotional states, as explained by Watson, Clark, and Tellegen [11]. Both of these dimensions have been further divided into a list of 10 items, each representing a unique mood state. A widely studied and commonly used scale is PANAS (Positive Affect - Negative Affect Scale). When evaluating a person's mood with the PANAS, users rate each item on a scale from 1 to 5, indicating the level of presence or intensity [10]. The responses given are combined to create a score that represents the user's current mood. Considering that this generated score disregards other factors, such as dependencies within the questionnaire, it is worth exploring the necessity of considering all individual answers [11, 12]. This paper proposed mood-based movie recommendation system based on PANAS and cosine similarity. The recommendation system is then evaluated using three different machine learning algorithms namely Naive Bayes (NB) using gaussian nb, multinomial nb and Bernoulli nb, Support Vector Machine (SVM) using linear and rbf kernel and Decision Tree (DT) using criterion log loss, gini and entropy. Rest of the paper is organized as follows. Section 2 introduces Dataset, PANA Scale, Cosine Similarity, SVM, DT and NB. Section 3 introduces Result and Section 4 introduces Conclusion.

II. RESEARCH METHODOLOGY

2.1 Dataset

Kaggle provided two datasets, namely tmdb 5000 movies and tmdb 5000 movies. Both csv files namely movies and credits [13] were used with each file containing 20 and 4 characteristics, respectively. Both datasets have been used for Movie Recommendation system. Movies dataset consists of features namely budget, genre, homepage, id, keywords, original language, original title, overview, popularity, production company, production countries, release date, revenue, runtime, spoken language, status, tagline, title, vote average and vote count. Credits dataset consists of features namely movie id, title, cast and crew. The two datasets used in movie recommendation are merged to form a single dataset shown in Figure 1. The columns kept under it include the movie ID, title, genre and tags.

	genres	movie_id	title	\
0	[action, crime, drama]	49026	The Dark Knight Rises	
1	[adventure, drama, action]	254	King Kong	
2	[drama, romance, thriller]	597	Titanic	
3	[action, drama, horror]	72190	World War Z	
4	[drama, romance]	64682	The Great Gatsby	

	tags
0	dccomics crimefighter terrorist christianbale ...
1	filmbusiness screenplay showbusiness naomiwatt...
2	shipwreck iceberg ship katewinslet leonardodic...
3	dystopia apocalypse zombie bradpitt mireilleen...
4	basedonnovel infidelity obsession leonardodica...

Fig. 1 Merged Dataset

2.2 PANA Scale

The Positive and Negative Affect Schedule (PANAS), created in 1988 by psychologists David Watson, Lee Anna Clark, and Auke Tellegen, is a psychometric scale that aims to assess positive and negative affect [11]. The study includes a set of 20 items that capture a range of emotions. Participants are asked to rate these items on a 5-point Likert scale, ranging from Very Slightly or Not at All" to Extremely." Positive Affect (PA) encompasses feelings of enthusiasm, alertness, and energy, while Negative Affect (NA) includes distress and unpleasurable engagement. The PANAS is widely utilized [14] in both clinical and community settings to evaluate emotional states and their correlation with personality traits. Its uses encompass tracking shifts in clients' emotions over time, assessing the effectiveness of therapeutic interventions, and capturing immediate affect [15]. The scale is renowned for its robust psychometric properties, showcasing exceptional internal consistency with Cronbach alpha coefficients ranging from 0.86 to 0.90 for positive affect (PA) and 0.84 to 0.87 for negative affect (NA). It also demonstrates consistent test-retest reliability over an 8-week period, with slightly stronger reliability for shorter time frames. The PANAS demonstrates strong convergent validity, as PA is positively associated with social activity and mood fluctuations, while NA is linked to stress, depression, and overall distress. The discriminant validity is robust, as PA demonstrates minimal correlation with stress and depression, while NA exhibits minimal correlation with social activity and mood fluctuations. The PANAS is highly valuable tool in the fields of psychological research and clinical practice. They provide reliable and valid assessments of positive and negative affect, which play a crucial role in understanding and improving emotional wellbeing and personality traits. Same has been used as a proposed method to calculate either the positive affect or negative affect of a user according to his/her mood, shown in Figure 2.

2.3 Cosine Similarity

Cosine similarity [16, 17] is a metric commonly used to evaluate the similarity between two vectors. It disregards the magnitude of the vectors and focuses on calculating the cosine of the angle between them. In the context of movie recommendation systems, it is used to measure the similarity between users or movies based on ratings or other features. The similarity measure plays a vital role in collaborative filtering techniques, which serve as the basis for numerous recommendation systems. Mathematically, the cosine similarity between two vectors C and D is defined as:

$$\text{cosine\_similarity}(C, D) = \frac{C \cdot D}{\|C\|\|D\|} = \frac{\sum_{i=1}^{n} C_i D_i}{\sqrt{\sum_{i=1}^{n} C_i^2}\sqrt{\sum_{i=1}^{n} D_i^2}} \quad (1)$$

where, C ・ D is the dot product of vectors C and D. // C // and // D // are the magnitudes (Euclidean norms) of vectors C and D, respectively. Ci and Di are the components of vectors C and D at dimension i. Cosine similarity is a measure that goes from -1 to 1. A value of 1 implies that the vectors being compared are equal. The value of 0 implies that the vectors are orthogonal, meaning they have no resemblance. The value of -1 signifies that the vectors are completely opposite in direction. When it comes to movie recommendation systems, cosine similarity can be applied in two primary approaches: user-based and item-based collaborative filtering. The objective of user-based collaborative filtering is to identify individuals with similar preferences. Similarity between users is determined by comparing their rating vectors. For example, if users U1 and U2 exhibit similar rating patterns, it is probable that they share similar preferences. The algorithm suggests movies to a user U1 by considering the preferences of other users who have similar tastes. When it comes to item-based collaborative filtering, the main objective is to identify similarities between movies by analyzing users' ratings. Comparisons are made between the rating vectors of movies. If two movies are found to be similar, it is likely that users who enjoyed one movie will also enjoy the other. This approach proves to be highly beneficial when incorporating new users into the system, as it does not depend on having a vast amount of user data.

```
Choose the emotion you have felt in the past week for 'Interested':
 1.Very Slightly or not at all  2.A little      3.Moderately    4.Quite a bit   5.Extremely
 1
Choose the emotion you have felt in the past week for 'Excited':
 1.Very Slightly or not at all  2.A little      3.Moderately    4.Quite a bit   5.Extremely
 2
Choose the emotion you have felt in the past week for 'Strong':
 1.Very Slightly or not at all  2.A little      3.Moderately    4.Quite a bit   5.Extremely
 5
Choose the emotion you have felt in the past week for 'Enthusiastic':
 1.Very Slightly or not at all  2.A little      3.Moderately    4.Quite a bit   5.Extremely
 1
Choose the emotion you have felt in the past week for 'Proud':
 1.Very Slightly or not at all  2.A little      3.Moderately    4.Quite a bit   5.Extremely
 3
Choose the emotion you have felt in the past week for 'Inspired':
 1.Very Slightly or not at all  2.A little      3.Moderately    4.Quite a bit   5.Extremely
 1
Choose the emotion you have felt in the past week for 'Determined':
 1.Very Slightly or not at all  2.A little      3.Moderately    4.Quite a bit   5.Extremely
 5
Choose the emotion you have felt in the past week for 'Attentive':
 1.Very Slightly or not at all  2.A little      3.Moderately    4.Quite a bit   5.Extremely
 4
Choose the emotion you have felt in the past week for 'Active':
 1.Very Slightly or not at all  2.A little      3.Moderately    4.Quite a bit   5.Extremely
 2
Choose the emotion you have felt in the past week for 'Alert':
 1.Very Slightly or not at all  2.A little      3.Moderately    4.Quite a bit   5.Extremely
 4
Choose the emotion you have felt in the past week for 'Distressed':
 1.Very Slightly or not at all  2.A little      3.Moderately    4.Quite a bit   5.Extremely
 1
Choose the emotion you have felt in the past week for 'Upset':
 1.Very Slightly or not at all  2.A little      3.Moderately    4.Quite a bit   5.Extremely
 2
Choose the emotion you have felt in the past week for 'Guilty':
 1.Very Slightly or not at all  2.A little      3.Moderately    4.Quite a bit   5.Extremely
 4
Choose the emotion you have felt in the past week for 'Scared':
 1.Very Slightly or not at all  2.A little      3.Moderately    4.Quite a bit   5.Extremely
 1
Choose the emotion you have felt in the past week for 'Hostile':
 1.Very Slightly or not at all  2.A little      3.Moderately    4.Quite a bit   5.Extremely
 2
Choose the emotion you have felt in the past week for 'Irritable':
 1.Very Slightly or not at all  2.A little      3.Moderately    4.Quite a bit   5.Extremely
 4
Choose the emotion you have felt in the past week for 'Ashamed':
 1.Very Slightly or not at all  2.A little      3.Moderately    4.Quite a bit   5.Extremely
 1
Choose the emotion you have felt in the past week for 'Nervous':
 1.Very Slightly or not at all  2.A little      3.Moderately    4.Quite a bit   5.Extremely
 2
Choose the emotion you have felt in the past week for 'Jittery':
 1.Very Slightly or not at all  2.A little      3.Moderately    4.Quite a bit   5.Extremely
 4
Choose the emotion you have felt in the past week for 'Afraid':
 1.Very Slightly or not at all  2.A little      3.Moderately    4.Quite a bit   5.Extremely
 2
Positive affect mean score: 2.8
Negative affect mean score: 2.3
```

Fig. 2 PANA Scale

2.4 Support Vector Machine

The Support Vector Machine (SVM) [18, 19] is a robust supervised learning algorithm that was developed by Vladimir Vapnik in the 1990s. It is commonly used for classification tasks and can also be applied to regression problems. The SVM algorithm
operates by discerning the hyperplane that efficiently segregates data points belonging to distinct groups. When dealing with linearly separable data, the task at hand is to find a hyperplane that can maximize the margin. The margin is the distance between the hyperplane and the nearest data points from each class, known as support vectors. The support vectors are crucial in defining the precise location and orientation of the hyperplane. In an n-dimensional space, the equation of a hyperplane can be expressed as:

$$a * q + e = 0 \quad (2)$$

The weight vector is denoted by a, the input features are represented by q, and the bias term is denoted by e. The objective is to optimize the margin, which is defined as the ratio of 2 divided by the magnitude of vector a. This optimization is subject to the condition that the data points be accurately identified:

$$y_i(a * q_i + e) \geq 1 \quad (3)$$

In situations where it is challenging to achieve a complete separation of classes due to noise or overlapping data points, SVM introduces the concept of a soft margin. This requires the incorporation of slack variables ξ_i to the optimization problem, which permits a certain degree of misclassification while also imposing a penalty through a regularization parameter C. The optimization objective changes:

$$min\frac{1}{2}||a||^2 + C\sum_{i=1}^{n}(\xi_i) \quad (4)$$

$$s.t\begin{cases} y_i(a * q_i + e) \geq 1 - \xi_i \\ \xi_i \geq 0 \end{cases} \quad (5)$$

The linear kernel is the simplest type of kernel, where the decision boundary is a straight line (or hyperplane in higher dimensions). It is given by

$$K(q_i, q_j) = q_i * q_j \quad (6)$$

The RBF kernel, also known as the Gaussian kernel, is a popular choice for non-linear data. It is defined as

$$K(q_i, q_j) = exp(-\gamma||q_i - q_j||^2) \quad (7)$$

where γ is a parameter that determines the extent of the kernel’s distribution.

2.5 Decision Tree

A Decision Tree [20-24] is a powerful tool in the field of machine learning, capable of handling both classification and regression tasks with ease. The structure of this. resembles a tree, with internal nodes making decisions based on attribute values,
branches showing the outcomes of these decisions, and leaf nodes representing the final output, which can be a class label or a continuous value. At the top of the tree, the root node symbolizes the complete dataset. Decision trees are constructed using a top-down, iterative partitioning method known as recursive binary splitting. This involves dividing the dataset into subsets based on the attribute that offers the most effective separation, as determined by a selected criterion. Log loss is a metric that evaluates the effectiveness of a classification model by considering the probability values it generates, ranging from 0 to 1. The objective of utilizing log loss is to reduce the disparity between the predicted probability and the true class label. The log loss for a binary classification problem is defined as follows:

$$logloss = -\frac{1}{K}\sum_{i=1}^{k}[y_i \log_2(q_i) + (1 - y_i)\log_2(1 - q_i)] \quad (8)$$

The Gini Impurity quantifies the impurity level of a dataset by calculating the likelihood of randomly choosing an incorrect class, based on the distribution of classes within the dataset. The calculation is as follows:

$$Gini(Dataset) = 1 - \sum_{i=1}^{k} q_i^2 \quad (9)$$

where q_i is the ratio of instances belonging to class i in the dataset. The reduction in entropy or impurity after a dataset is split on an attribute is measured by Information Gain. The calculation involves determining the discrepancy between the entropy of the initial dataset and the combined entropy of the subsets following the division. The calculation of the entropy for a dataset is as follows:

$$Entropy(D) = -\sum_{i=1}^{k}(q_i)\log_2(q_i) \quad (10)$$

The calculation for determining the information gain (IG) of an attribute A is as follows:

$$IG(D, A) = Entropy(D) - \sum_{v}\frac{D_v}{D}Entropy(D_v) \quad (11)$$

where D_v is the subset of D where attribute A has value v.

2.6 Naive Bayes

Naive Bayes [25 – 28] is a straightforward yet remarkably powerful probabilistic classifier that relies on Bayes' theorem. It is especially well-suited for classification tasks in the field of machine learning. Despite its straightforwardness, it excels in a wide range of applications including text classification, spam detection, sentiment analysis, and recommendation systems. Bayes theorem is a fundamental concept that forms the basis of Naive Bayes. It allows us to calculate the probability of a hypothesis based on the evidence we observe. Bayes theorem can be expressed as:

$$P(C\P D) = \frac{P(D|C)\cdot P(C)}{P(D)} \quad (12)$$

The expression P(C|D) represents the posterior probability of class C given feature D. The expression P(D|C) represents the conditional probability of feature D given class C. The term P(C) refers to the initial probability of class C. The term P(D) refers to the initial probability of feature D. The" naive" component of Naive Bayes stems from the assumption of independence. It presupposes that all characteristics are unrelated to one another, provided the category is known. Here is the streamlined model:

$$P(E|Q_1, Q_2, \ldots, Q_n) \propto P(E)\cdot\prod_{i=1}^{k}P(Q_i|C) \quad (13)$$

In this context, P(E) represents the initial probability of class E, whereas P(Qi|E) represents the probability of feature Qi given class E. Three primary categories of Naive Bayes classifiers exist, each designed to handle distinct data types: Gaussian Naive Bayes is designed for continuous data, it assumes that the features adhere to a normal (Gaussian) distribution. It calculates the mean and variance for each feature and class, and then uses these parameters to determine the likelihood. The probability density function for a Gaussian distribution is

$$P(q_i \mid r) = \frac{1}{\sqrt{2\pi\sigma_r^2}}\exp\left(-\frac{(q_i-\mu_r)^2}{2\sigma_r^2}\right) \quad (14)$$

Multinomial Naive Bayes is well-suited for analyzing discrete data, particularly word counts in text classification. The likelihood is modeled using a multinomial distribution, which proves to be highly effective for document classification tasks where the features represent word frequencies. The probability of a feature vector given a class is

$$P(\mathrm{x} \mid y) = \frac{N_y!}{x_1!x_2!\cdots x_n!}\left(\frac{\theta_{y1}^{x_1}\theta_{y2}^{x_2}\cdots\theta_{yn}^{x_n}}{(\sum_i x_i)!}\right) \quad (15)$$

Bernoulli's contribution Naive Bayes is specifically designed to handle binary or boolean features. The assumption is that every feature conforms to a Bernoulli distribution, which means it can be used effectively for tasks such as binary text classification, where features indicate whether words are present or absent. The probability of a feature vector given a class is

$$P(\mathrm{x} \mid y) = \prod_{i:x_i=1}\theta_{yi}\prod_{i:x_i=0}(1-\theta_{yi}) \quad (16)$$

III. RESULTS

Dataset is divided two datasets based on genre into good mood genre which consists of comedy, romance, drama, animation, fantasy, sci-fi, & music and bad mood genre consisting of drama, romance, documentaries, fantasy & music shown in Figure 3.

```
Good Mood Genre
 ['Comedy', 'Romance', 'Drama', 'Animation', 'Fantasy', 'Sci-Fi', 'Music']
Bad Mood Genre
 ['Drama', 'Romance', 'Documentaries', 'Fantasy', 'Music']
```

Fig. 3 Good Genre and Bad Genre

After the user has responded to all the questions in the questionnaire, positive effect means and negative effect mean is calculated. Depending on the score good mood genre dataset or bad mood genre dataset is selected and thereafter recommendation model based on cosine similarity starts its execution based on the random selection of genre from the good mood or bad mood depending on the score generated shown in Figure 4.

	genres	movie_id	title	tags
0	[action, crime, drama]	49026	The Dark Knight Rises	dccomics crimefighter terrorist christianbale ...
1	[adventure, drama, action]	254	King Kong	filmbusiness screenplay showbusiness naomiwatt...
2	[drama, romance, thriller]	597	Titanic	shipwreck iceberg ship katewinslet leonardodic...
3	[action, drama, horror]	72190	World War Z	dystopia apocalypse zombie bradpitt mireilleen...
4	[drama, romance]	64682	The Great Gatsby	basedonnovel infidelity obsession leonardodica...
...	...	...	...	...
2251	[drama, foreign]	13898	The Circle	nargessmamizadeh maryiampalvinalmani mojganfa...
2252	[drama]	182291	On The Downlow	confession hazing gangmember tonysancho michae...
2253	[drama]	124606	Bang	gang audition policefake darlingnarita petergr...
2254	[sciencefiction, drama, thriller]	14337	Primer	distrust garage identitycrisis shanecarruth da...
2255	[comedy, drama, romance]	231617	Signed, Sealed, Delivered	date loveatfirstsight narration ericmabius kri...

2256 rows × 5 columns

Fig. 4 Movies data based on Drama genre

Based on input, a movie title is randomly selected from the list of good mood genre dataset or bad mood genre dataset. After selecting a random movie, 10 similar movies are recommended to the user shown in Figure 5.

```
Recommendation for movie: End of Watch
Recommended movie: 835          Street Kings
1903           Harsh Times
201         Gangster Squad
930     Observe and Report
496               Sabotage
1137       The Lucky Ones
1785              Stranded
199                Shooter
332               Prisoners
799         Moonlight Mile
Name: title, dtype: object
```

Fig. 5 Recommended movies list

To test the accuracy of Recommendation system, three classification algorithms Naive Bayes (Gaussian NB, Multinomial NB, Bernoulli NB), Support Vector Machine (SVM) using linear kernel with value of regularization parameter C = 10 and radial basis kernel (rbf) with the value of C = 10 and gamma = 0.05 and Decision Tree using criterion log loss, gini and entropy have been applied on 70%, 75% and 80% data as training dataset and the result has been formulated in Table 1. It can be seen that Decision tree with criterion gini scored highest accuracy i.e. 95.56% and 98.93% for 70% and 75% training data respectively and for 80% training data. Naive Bayes has the accuracy range from 47.42% to 84.40%, SVM has the accuracy range from 87.44% to 96.80% and Decision Tree has the accuracy range from 95.27% to 98.93%. It can be stated that Decision Tree performed approximately same and is quite consistent for all three training data while Naive Bayes being the worst among three.

Table 1 Comparison of Algorithms

Algorithms	Training Size		
	70%	75%	80%
Gaussian NB	47.42%	84.40%	81.86%
Multinomial NB	56.87%	71.99%	70.80%
Bernoulli NB	55.98%	67.91%	65.93%
SVM (linear)	88.92%	96.80%	94.69%
SVM (rbf)	87.44%	96.09%	94.69%
Decision Tree (log_loss)	94.97%	98.40%	97.34%
Decision Tree (gini)	95.56%	98.93%	97.56%
Decision Tree (entropy)	95.27%	98.58%	97.78%

IV. CONCLUSION

The objective of the paper was to develop a recommendation system that classifies movies into two unique genres, namely" good mood" and" bad mood," based on their ability to impact a user's emotional state. The happy mood genre comprises comedy, romance, drama, animation, fantasy, sci-fi, and music, whereas the bad mood genre consists of drama, romance, documentaries, fantasy, and music. This categorization guarantees that users are provided with film suggestions that are in line with their present emotional condition. Upon completion of a questionnaire, the responses provided by users are utilized to compute the average positive and negative impacts, so assessing their overall emotional state. Using these scores, the recommendation algorithm
chooses either the dataset for the positive mood genre or the dataset for the negative mood genre. From the given dataset, a genre is selected at random, and then a movie title is chosen randomly from that genre. Afterwards, the system employs a recommendation algorithm based on cosine similarity to provide ten movies that are comparable to the chosen title. In order to evaluate the precision and efficiency of the recommendation system, several classification algorithms were utilized, including Naive Bayes (specifically Gaussian NB, Multinomial NB, and Bernoulli NB), Support Vector Machine (SVM) with both linear and radial basis function (rbf) kernels, and Decision Tree with criteria such as log loss, gini, and entropy. The algorithms underwent testing on training datasets that consisted of 70%, 75%, and 80% of the entire data. The outcomes of these tests were then gathered and presented in a thorough
table. The results showed that the Decision Tree method, specifically using the gini criterion, attained the maximum level of accuracy. It scored 95.56% when trained with 70% of the data and 98.93% when trained with 75% of the data. The Decision Tree model, using the entropy criteria, attained an accuracy of 97.98% on 80% of the training data. On the other hand, the Naive Bayes algorithm showed the least accurate results, with accuracy ranging from 47.42% to 84.40%, indicating notable discrepancy. The SVM method demonstrated satisfactory performance, with accuracy ranging from 87.44% to 96.80%. The Decision Tree algorithm's consistent and strong performance on all training datasets indicates that it is the most dependable approach for this recommendation system. The fact that it can consistently achieve accuracy levels ranging from 95.27% to 98.93% demonstrates its resilience and appropriateness for the given task. However, the performance of Naive Bayes suggests that it may not be suitable for this particular application, as it exhibits a broad range of accuracy and lower total scores.

REFERENCES

[1] Shani, G., Gunawardana, A. (2011). Evaluating Recommendation Systems. In: Ricci, F., Rokach, L., Shapira, B., Kantor, P. (eds) Recommender Systems Handbook. Springer, Boston, MA. https://doi.org/10.1007/978-0-387-85820-3 8

[2] Rajarajeswari, S., Naik, S., Srikant, S., Sai Prakash, M.K., Uday, P. (2019). Movie Recommendation System. In: Shetty, N., Patnaik, L., Nagaraj, H., Hamsavath, P., Nalini, N. (eds) Emerging Research in Computing, Information, Communication and Applications. Advances in Intelligent Systems and Computing, vol 882. Springer, Singapore. https://doi.org/10.1007/978-981-13-5953-8 28.

[3] C. -S. M. Wu, D. Garg and U. Bhandary, Movie Recommendation System Using Collaborative Filtering," 2018 IEEE 9th International Conference on Software Engineering and Service Science (ICSESS), Beijing, China, 2018, pp. 11-15, doi: 10.1109/ICSESS.2018.8663822.

[4] N. Pradeep; K. K. Rao Mangalore; B. Rajpal; N. Prasad; R. Shastri. "Content based movie recommendation system". International Journal of Research in Industrial Engineering, 9, 4, 2020, 337-348. doi: 10.22105/riej.2020.259302.1156

[5] Reddy, S., Nalluri, S., Kunisetti, S., Ashok, S., Venkatesh, B. (2019). Content- Based Movie Recommendation System Using Genre Correlation. In: Satapathy, S., Bhateja, V., Das, S. (eds) Smart Intelligent Computing and Applications. Smart Innovation, Systems and Technologies, vol 105. Springer, Singapore. https://doi.org/10.1007/978-981-13-1927-3 42

[6] Goyani, Mahesh; Chaurasiya, Neha. A Review of Movie Recommendation System: Limitations, Survey and Challenges. ELCVIA. Electronic letters on computer vision and image analysis, Vol. 19 Num. 3 (2020), p. 18-37. DOI 10.5565/rev/elcvia.1232! https://ddd.uab.cat/record/232276.

[7] S. Kumar, K. De and P. P. Roy," Movie Recommendation System Using Sentiment Analysis from Microblogging Data," in IEEE Transactions on Computational Social Systems, vol. 7, no. 4, pp. 915-923, Aug. 2020, doi: 10.1109/TCSS.2020.2993585.

[8] T. Zhou, L. Chen and J. Shen," Movie Recommendation System Employing the User-Based CF in Cloud Computing," 2017 IEEE International Conference on Computational Science and Engineering (CSE) and IEEE International Conference on Embedded and Ubiquitous Computing (EUC), Guangzhou, China, 2017, pp. 46-50, doi: 10.1109/CSE-EUC.2017.194.

[9] Robbins, Stephen, Timothy A. Judge, Bruce Millett, and Maree Boyle. Organisational behaviour. Pearson Higher Education AU, 2013.

[10] Arrow, Holly, Joseph E. McGrath, B. M. Staw, and L. L. Cummings. "Membership dynamics in groups at work: A theoretical framework." Research in organizational behavior 17 (1995): 373-373.

[11] Watson, David, Lee Anna Clark, and Auke Tellegen. "Development and validation of brief measures of positive and negative affect: the PANAS scales." Journal of personality and social psychology 54, no. 6 (1988): 1063.

[12] Raghunathan, Rajagopal, and Michel Tuan Pham. "All negative moods are not equal: Motivational influences of anxiety and sadness on decision making." Organizational behavior and human decision processes 79, no. 1 (1999): 56-77.

[13] https://www.kaggle.com/datasets/tmdb/tmdb-movie-metadata

[14] Merz, Erin L., et al." Psychometric properties of Positive and Negative Affect Schedule (PANAS) original and short forms in an African American community sample." Journal of affective disorders 151.3 (2013): 942-949.

[15] Magyar-Moe, Jeana L. Therapist's guide to positive psychological interventions. Academic press, 2009.

[16] Wen, H., Ding, G., Liu, C., Wang, J. Matrix Factorization Meets Cosine Similarity: Addressing Sparsity Problem in Collaborative Filtering Recommender System. Web Technologies and Applications. APWeb 2014. Lecture Notes in Computer Science, vol 8709. Springer, Cham. https://doi.org/10.1007/978-3-319-11116-2 27

[17] E. Bigdeli and Z. Bahmani," Comparing accuracy of cosine-based similarity and correlation-based similarity algorithms in tourism recommender systems," 2008 4th IEEE International Conference on Management of Innovation and Technology, Bangkok, Thailand, 2008, pp. 469-474, doi: 10.1109/ICMIT.2008.4654410.

[18] Saunders, Craig, Mark O. Stitson, Jason Weston, Leon Bottou, and A. Smola." Support vector machine-reference manual." (1998).

[19] Schoellkopf, Bernhard, and Alex Smola. "Support vector machines." Encyclopedia of Biostatistics (1998): 978-0471975762.

[20] Tu, P-L., and J-Y. Chung. "A new decision-tree classification algorithm for machine learning." TAI' 92-proceedings fourth international conference on tools with artificial intelligence. IEEE Computer Society, 1992.

[21] Patel, Harsh H., and Purvi Prajapati. "Study and analysis of decision tree based classification algorithms." International Journal of Computer Sciences and Engineering 6.10 (2018): 74-78.

[22] Anyanwu, Matthew N., and Sajjan G. Shiva. "Comparative analysis of serial decision tree classification algorithms." International Journal of Computer Science and Security 3.3 (2009): 230-240.

[23] Tangirala, Suryakanthi. "Evaluating the impact of GINI index and information gain on classification using decision tree classifier algorithm." International Journal of Advanced Computer Science and Applications 11.2 (2020): 612-619.

[24] Yang, Hang, and Simon Fong. "Improving the accuracy of incremental decision tree learning algorithm via loss function." 2013 IEEE 16th International Conference on Computational Science and Engineering. IEEE, 2013.

[25] Reddy, E. M. K., Gurrala, A., Hasitha, V. B., & Kumar, K. V. R. (2022). Introduction to Naive Bayes and a review on its subtypes with applications. Bayesian reasoning and gaussian processes for machine learning applications, 1-14.

[26] Kamel, Hajer, Dhahir Abdulah, and Jamal M. Al-Tuwaijari. ”Cancer classification using gaussian naive bayes algorithm.” 2019 international engineering conference (IEC). IEEE, 2019.

[27] Qiang, Guo. ”An effective algorithm for improving the performance of Naïve Bayes for text classification.” 2010 Second international conference on computer research and development. IEEE, 2010.

[28] Ashari, Ahmad, Iman Paryudi, and A. Min Tjoa. ”Performance comparison between Naïve Bayes, decision tree and k-nearest neighbor in searching alternative design in an energy simulation tool.” International Journal of Advanced Computer Science and Applications (IJACSA) 4.11 (2013).