The International Journal of Engineering and Information Technology
journal homepage:www.ijeit.misuratau.edu.ly

Movie Recommendation Engine Based on Cosine Similarity and KNN

Fairouz Youssef Debbek
Libyan Academy – Misurata
Computer science Department
Fairouz.debbek@it.lam.edu.ly

Farij Omar Ehtiba
Libyan Academy – Misurata
Computer science Department
f.ehtiba@lam.edu.ly

Haitham Saleh Ben Abdelmula
Libyan Center for Electronic Systems ، Programming, and Aviation Research
hsaa8383@gmail.com

Abstract-- **Recommendation systems are becoming increasingly important as a means of managing information overload due to the proliferation of large volumes of material on the internet. In this paper, the hybrid method that uses the cosine similarity technique has been employed to find the similarity between the user preferences and the Netflix movie dataset. K-Nearest Neighbor method was emerged to identify films that closely correspond with the user's interests. This strategy aims to increase customer acceptability and utilization of the Netflix service by offering individualized recommendations to consumers based on their preferences. Although the examined papers presented performance for the model, they did not provide precise information about accuracy. The experimental results were evaluated using key metrics such as Accuracy, Precision at k, Mean Score, and Cross-Validation Scores. The empirical investigation demonstrates that the suggested approach gives customers precise recommendations based on their preferences where the accuracy of the proposed system scores around 80%.**

Index Terms—**recommendation system, cosine similarity, Machine Learning, KNN, Netflix.**

I. INTRODUCTION

In daily lives, we rely on recommendations provided by others through word of mouth or reviews from general surveys. the Internet has grown rapidly and continues to grow each day. The abundance of information available online, has made it a strenuous task to access the right information quickly and easily [1]. Fortunately, recommendation systems can assist in solving this issue. People frequently utilize recommender systems on the internet to assist them in making decisions regarding items that align with their preferences. One approach to AI is the use of recommender systems. Ecommerce companies have started using them as a solution to this problem. Systematic recommendations have been produced for users without the necessity for a specific search query [2]. Recommendation systems are software tools and techniques designed to offer valuable and relevant suggestions to a group of users for products or items that may be of interest to them.

Received 02 May , 2024; revised 11 May, 2024; accepted 15 Mar 2024. Available online 08 Aug, 2024.

is a major area which is very popular and useful for people to take proper

automated decisions [3]. In the era of information overload, recommendation systems play a crucial role in assisting users in discovering relevant content from a vast pool of data. The newest technology and the simplicity of use of the internet have made sharing videos simple. A single click on any smartphone allows for the creation and sharing of videos on numerous social media sites, including Facebook, YouTube, WhatsApp, Instagram, and many more, as well as with the entire world. There is a great deal of duplication as a result of this. Determining what kind of videos yo

u want to watch is an extremely challenging undertaking. This kind of laborious and time-consuming task has to be automated [4]". This paper discusses a movie recommendation engine based on cosine similarity and K-Nearest Neighbors (KNN) in the field of movies. Effective movie recommendation poses a significant challenge in our digital age, which is saturated with vast amounts of information and available content. Users' need for guidance and recommendations for new and exciting movies is a complex task. The cosine similarity and KNN-based recommendation engine relies on a fundamental idea that users with similar movie preferences are likely to have similar recommendations. The engine utilizes cosine similarity as a measure of similarity between user preferences and relies on KNN to find the nearest users in terms of similarity.

II. RECOMMENDER SYSTEMS

Recommender systems are commonly employed to furnish individuals with suggestions predicated on their individual interests. Recommender systems have proven to be a helpful aid in overcoming information overload due to the constantly expanding amount of information available online [5]. Recommender systems leverage sophisticated algorithms and machine learning techniques to provide accurate and tailored recommendations. As a result, recommender systems have become indispensable tools for businesses and platforms seeking to deliver personalized experiences and maximize user engagement and conversions. The recommendation system works by

making product recommendations based on information to obtain desired products, past history, and user preferences [6]. Interestingly, services like Netflix use recommendation algorithms to make movie recommendations based on users' ratings, watching preferences, and interests [6]. This can facilitate users in easily discovering content that aligns with their interests, thus enhancing their satisfaction in their film-watching experience on these platforms. Recommendation systems can be broadly classified into 3 types show on Figure 1.

2.1 Content-Based Filtering

A content-based recommender system receives data from users, either knowingly or unknowingly (ranking or evaluating a link). After gathering this information, the RS creates a profile for each user. The user profile was used to create the recommendation. By looking at only one user's profile, content-based filtering generates recommendations [7]. The characteristics of a film are displayed; these are primarily taken from its metadata, which include details on the genre, star, director, and other features. These components give the movies their similarity [8].

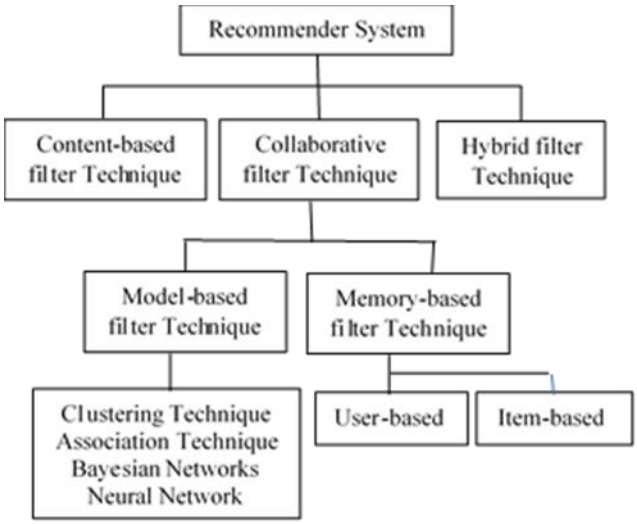

Figure 1: Types of recommendation system

2.2 COLLABORATIVE FILTERING

One technique for filtering data and giving the user pertinent information is collaborative filtering. Collaborative filtering is one of the most used methods for item recommendations. Unlike content-based filtering, collaborative filtering identifies users whose preferences align with a given user's. since of this, it suggests a product or other item based on the possibility that the particular user would continue to appreciate it, which other users enjoy since their interests are similar [9]. Collaborative Filtering (CF) is a technique that sorts data flow so that it can be offered to a target user based on his preferences and areas of interest. The profile of the target user is constructed by comparing him to other users. Because of this, the CF approach is highly sensitive to the similarity measure that is employed to determine how dependent two users (or two objects) are on one another [10].

2.3 HYBRID RECOMMENDATION SYSTEMS

Hybrid filtering may be utilized in a variety of ways. It addresses cold start, data sparsity, and scalability issues. The system can employ content-based filtering initially and thus transfer the results to collaborative filtering (and vice versa), or it could integrate both filters into a single system for better outcomes [9]. " Hybrid systems such as IMDB combine the previous two groups. In order to create relevant representations of movies and to evaluate their similarity, these systems rely on man-made information about films. Content-based methods rely on movie metadata and build a database of this information for classification purposes. In other words, these systems do not take the raw content of the movie itself into account, but are based solely on user-generated annotations [11].

III. RELATED WORKS

Authors in [12] focuses on the development of a movie recommender system using K-Means Clustering and K-Nearest Neighbor algorithms. The existing technique is compared to the proposed technique, highlighting the optimization of the recommendation process by gathering data to reduce the number of clusters. The recommender system predicts user preferences based on various parameters and the common preferences among users. The study aims to enhance the recommendation quality through collaborative predictions and fuzzy similarity measures. Additionally, references to related works and conferences in the field of machine learning and data science are provided throughout the paper.

While the study in [13] explores a movie recommendation system by illustrating the modeling of a movie recommendation system using content-based filtering, Cosine Similarity, and the KNN algorithm. It highlights that the use of Cosine Similarity provides more accuracy compared to other distance metrics, with lower complexity. Cosine Similarity is employed to measure the similarity between movies based on the angle between them, with higher similarity when the angle is closer to 1 and lower when closer to 0. This similarity is used to recommend movies to users based on their similarity. Recommendation systems are emphasized as crucial sources of relevant and reliable information on the internet, with simpler systems considering fewer parameters and more complex ones utilizing multiple parameters to enhance user-friendliness.

In [14], authors discuss a Netflix Recommendation System based on TF-IDF and Cosine Similarity Algorithms. It includes an exploratory analysis of data from Flexible, a search engine listing Netflix content. The study analyzed 7,787 unique records and implemented a recommendation system using TF-IDF and Cosine similarity algorithms. The TF-IDF algorithm evaluates word importance in a corpus, while Cosine Similarity measures similarity between vectors. The analysis revealed insights into current content trends on Netflix. The system shows promise but may benefit from additional features for enhanced performance.

The study in [15] presents a Movie Recommendation System using Machine Learning techniques, specifically incorporating cosine similarity and sentiment analysis. The cosine similarity algorithm was found to be suitable for the movie recommendation system due to its speed

and accuracy. Additionally, the Support Vector Machine (SVM) classifier outperformed the Naive Bayes (NB) classifier in analyzing movie reviews.

An improved a Movie Recommendation System Design by deploying Association Rules Mining and Classification Techniques was proposed in [16]. The design utilizes a portion of the KNN algorithm to create an initial candidate list based on the Movie Lens dataset obtained from Netflix. Additionally, a portion of the Prior algorithm is employed to analyze the same dataset and generate a secondary list. The proposed system then combines these two lists to create a final recommendation list. The results demonstrate that the proposed system outperforms existing systems in terms of importance degree.

Finally, the authors in [17] is basically focusing on Movies Rating Predictions and Recommendations. The efficacy of all models was tested with Movie Lens 100k and 1M dataset. KNN, SVM, and Random Forest are three supervised machine learning models that have been used to predict movie ratings. The primary goal of comparing these three models is to determine which one provides the most accurate predictions of movie ratings using Supervised machine learning. Finally, the experimental study revealed that KNN has a 69% accuracy score, SVM has a 54% accuracy score and the Random Forest model has a 92% accuracy value, indicating that Random Forest model achieves a higher accuracy score as compared to KNN and SVM Models.

IV. METHODOLOGY

In this section, the proposed system, which consists of three steps, was explained in details. Figure 2 shows the component of the proposed system.

4.1 PREPROCESSING DATA

In order to increase the efficiency of the preprocess stage applied on dataset. the ineffective elements are removed. It presence leads to decrease the efficiency of the system.

4.2 COSINE SIMILARITY

Cosine similarity is a metric used to measure the similarity between two vectors in a multi-dimensional space. The cosine of the angle formed by the vectors is calculated, signifying their similarity.

Cosine similarity is frequently used in recommendation systems to determine how similar user preferences or item attributes are to one another. Cosine similarity calculates the cosine value between vectors to facilitate the comparison of similarity among attributes in a dataset [18].

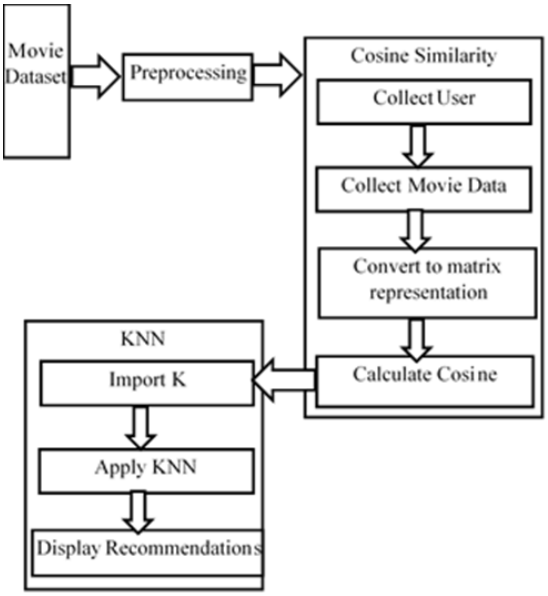

Figure 2: Proposed Movie Recommendation System

By representing users or items as vectors, where each dimension corresponds to a specific attribute or feature, cosine similarity can be calculated to determine how similar two users or items are to each other. The formula for calculating cosine similarity between two vectors A and B is as shown in question (1).

$$\boldsymbol{cosine\ similarity} = \frac{(\boldsymbol{A}.\boldsymbol{B})}{(||\boldsymbol{A}|| * ||\boldsymbol{B}||)} \qquad (\mathbf{1})$$

Where:

(A.B) represents the dot product of vectors A and B,

||A||*||B|| represent the magnitudes (or norms) of vectors A and B, respectively.

The cosine similarity value ranges (-1 to 1). A value of 1 indicates that the vectors are identical. In contrast, a value of -1 indicates to completely dissimilar, and a value of 0 indicates the vectors are orthogonal (not similar). In recommendation systems, cosine similarity can be used to find similar users or items based on their preferences or features. It helps in identifying users with similar tastes or items with similar characteristics, enabling personalized recommendations or item-based recommendations. show on Figure 3.

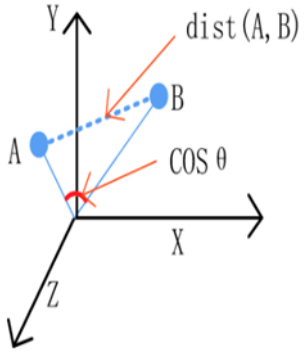

Figure 3: cosine similarity

4.3 K-NEAREST NEIGHBORS (KNN) ALGORITHM

KNN is a non-parametric technique that uses a similarity measure to identify the KNN to a new data point. It then classifies the new point according to the majority class of its neighbors. The KNN algorithm's

success depends on the similarity metric selection since it establishes the degree of similarity between two data points. Based on a similarity metric, the algorithm finds the k closest neighbors and uses the ratings of these neighbors to infer the user's preference for a specific movie. A key idea in the creation of recommendation systems is similarity, which is comparing the characteristics of several objects to determine which are most similar [19]. Distance Calculation: When a new unlabeled data point is provided, the algorithm calculates the distance between this data point and all the labeled data points in the feature space. Common distance metrics used include Euclidean distance. Neighbor Selection: The algorithm selects the k nearest neighbors based on the calculated distances, question (2). The value of k is a user-defined parameter that determines how many neighbors influence the prediction.

$$\boldsymbol{Euclidean\ Distance(k)} = \sqrt{|\boldsymbol{X} - \boldsymbol{Y}|} = \sqrt{\sum_{i=1}^{i=n} (\boldsymbol{x_i} - \boldsymbol{y_i})^2} \qquad (2)$$

To begin, data is collected from a wide range of users' ratings on different movies. Cosine similarity is used to calculate the degree of similarity between user preferences. The cosine similarity value between two users is closer to 1 if their preferences are identical and closer to 0 if their preferences are completely different.

Next, the KNN algorithm is used to determine the nearest users in terms of similarity. A value of K is determined, which specifies the number of nearest users whose preferences will be relied upon for movie recommendations. The preferences of these nearest users are used to suggest a list of potential movies for recommendation. By using the cosine similarity and KNN-based recommendation engine, the user experience in discovering new and exciting movies can be enhanced, providing personalized recommendations that suit their individual tastes.

V. IMPLEMENTATION

The MovieLens Dataset was used to evaluate the proposed system [20], which obtained from Kaggle website. It has two files, ratings and movies. The data is based on 105339 ratings given to 10329 movies. There are 668 user who has given their ratings for 149532 movies, as well as data about users who watch movies and comprehensive movie data. Likewise, Python programming language was used to implement the proposed system.

Every time, a random movie is chosen, cosine similarity is found to calculate the degree of similarity =0.5 between the random movie and the rest of the movies in the dataset. the nearest neighbor is found using the KNN algorithm. is calculated, the Euclidean distance, the accuracy is determined, the average accuracy as well. In addition, the mean average is calculated for each random movie.

VI. RESULTS AND DISCUSSION

An experimental analysis of the suggested system and the experiment's findings are presented in this section. The dataset's performance was assessed using the significant degree term.

First, a random movie called Black Book (Zwartboek) (2006) is chosen. Next, the degree of similarity between a randomly selected movie and the other movies in the collection is determined using a cosine similarity metric. 63 movies were the outcome that was attained. Subsequently, the KNN technique is employed to identify the closest neighbor, yielding 51 (K=51) movies that are cosine similarity scores away from the randomly chosen film.

TABLE 1: THE RESULT OF THE PROPOSED SYSTEM

Accuracy	80%
Precision at K	81%
Mean score	80.44%
Cross validation	80.28%
F1-Score	89.85%
Error Rate	18.43%

Table 1 displays the outcomes of the proposed Movie Recommendation System. The accuracy of the proposed system, which measures how frequently the recommendation system successfully identifies related movies for the randomly chosen movie, is 81.57%, indicating good performance.

Furthermore, the precision at K value is 81.57%, emphasizing the highest K recommendations. In this instance, K=51 indicates that the measure prioritizes the top 51 suggested films. Further, 81% of the suggested films have an average similarity score. This score is determined by dividing the total number of recommended movies by the sum of the similarity scores of all the recommended movies.

The process of cross-validation is performed to assess a model's performance on hypothetical data. The recommendation system in this instance appears to function well on data that was not used to train the model, based on the cross-validation score of 81%. This is a crucial indicator because it helps make sure the model can generalize effectively to new, untested data and isn't overfit to the training set. an error rate of 18.43% and an F1-score of 89.85%.

To summarize, these metrics offer a thorough assessment of the recommendation system's effectiveness, encompassing both the precision and caliber of the recommendations. Based on the significant degree term, the proposed recommendation system is successful in finding related films, as seen by the high scores obtained for each of these criteria.

The confusion matrix is a crucial assessment metric for assessing the effectiveness of a classification model. It provides a comprehensive examination of the relationship between the model's predictions and the actual labels on the data.

Because they show which predictions the recommendation system made correctly, the values in the major diagonal are significant. While a high value in the True Negatives (TN) field indicates that the system is also successfully identifying the movies that are not similar, a high value in the True Positives (TP) cell suggests that the system is effectively detecting the actually related movies. These diagonal values play a critical role in determining other performance indicators

and the recommendation system's overall correctness. generally, indicates that the main diagonal values are likewise high, demonstrating the system's capacity for accurate prediction, as seen in Figure 4.

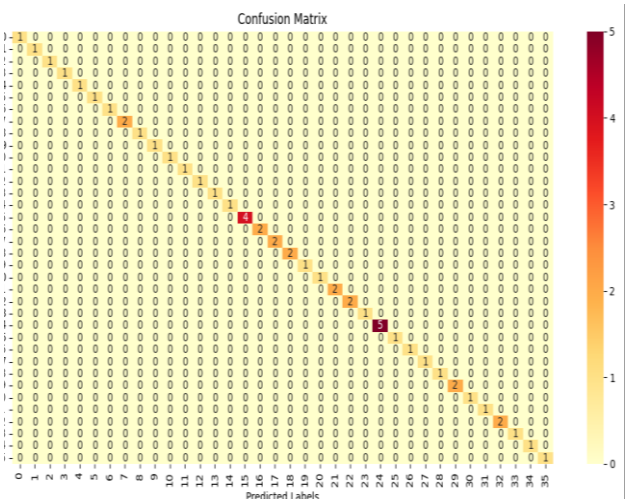

Figure 4: Confusion Matrix

VII. CONCLUSIONS

High similarity scores were obtained between randomly chosen movies and other movies in the collection, signaling a successful study's conclusion. A list of possible movies was suggested to customers based on these findings. The proposed system makes recommendations that are tailored to each user's interests and improves their experience finding new and intriguing movies by employing cosine similarity and KNN-based algorithms.

Based on the results, it appears that the proposed system works well and produces good movie suggestions, providing consumers with efficient and customized recommendations. To improve user experience, boost content discovery, and encourage variety in the offered content, streaming apps and entertainment platforms can make use of this method.

Future recommendation: Test the system with more datasets. To verify the effectiveness and scalability of the suggested solution, carry out tests and assessments on bigger and more varied datasets. This may provide light on how well the system performs in practical situations.

REFERENCES

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

[2] Delimayanti, M. K., Laya, M., Warsuta, B., Faydhurrahman, M. B., Khairuddin, M. A., Ghoyati, H., ... & Naryanto, R. F. (2022, August). Web-Based Movie Recommendation System using Content-Based Filtering and KNN Algorithm. In 2022 9th International Conference on Information Technology, Computer, and Electrical Engineering (ICITACEE) (pp. 314-318). IEEE.

[3] Goyani, M., & Chaurasiya, N. (2020). A review of movie recommendation system: Limitations, Survey and Challenges. ELCVIA: electronic letters on computer vision and image analysis, 19(3), 0018-37.

[4] Ibrahim, Z. A. A., Haidar, S., & Sbeity, I. (2019). Large-scale text-based video classification using contextual features. European Journal of Electrical Engineering and Computer Science, 3(2).

[5] Fayyaz, Z., Ebrahimian, M., Nawara, D., Ibrahim, A., & Kashef, R. (2020). Recommendation systems: Algorithms, challenges, metrics, and business opportunities. applied sciences, 10(21), 7748.

[6] Permana, A. H. J. P. J., & Wibowo, A. T. (2023). Movie Recommendation System Based on Synopsis Using Content-Based Filtering with TF-IDF and Cosine Similarity. International Journal on Information and Communication Technology (IJoICT), 9(2), 1-14.

[7] Sridhar, S., Dhanasekaran, D., & Latha, G. (2023). Content-Based Movie Recommendation System Using MBO with DBN. Intelligent Automation & Soft Computing, 35(3).

[8] Reddy, S. R. S., Nalluri, S., Kunisetti, S., Ashok, S., & Venkatesh, B. (2019). Content-based movie recommendation system using genre correlation. In Smart Intelligent Computing and Applications: Proceedings of the Second International Conference on SCI 2018, Volume 2 (pp. 391-397). Springer Singapore.

[9] Sharma, R. S., Shaikh, A. A., & Li, E. (2021). Designing Recommendation or Suggestion Systems: looking to the future. Electronic Markets, 31, 243-252.

[10] Fkih, F. (2022). Similarity measures for Collaborative Filtering-based Recommender Systems: Review and experimental comparison. Journal of King Saud University-Computer and Information Sciences, 34(9), 7645-7669.

[11] Kordabadi, M., Nazari, A., & Mansoorizadeh, M. (2022). A movie recommender system based on topic modeling using machine learning methods. International Journal of Web Research, 5(2), 19-28.

[12] Ahuja, R., Solanki, A., & Nayyar, A. (2019, January). Movie recommender system using k-means clustering and k-nearest neighbor. In 2019 9th International Conference on Cloud Computing, Data Science & Engineering (Confluence) (pp. 263-268). IEEE.

[13] 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.

[14] Chiny, M., Chihab, M., Bencharef, O., & Chihab, Y. (2022). Netflix recommendation system based on TF-IDF and cosine similarity algorithms. no. Bml, 15-20.

[15] Marappan, R., & Bhaskaran, S. (2022). Movie recommendation system modeling using machine learning. International Journal of Mathematical, Engineering, Biological and Applied Computing, 12-16.

[16] Zubi, Z. S., ELROWAYATI, A., & Fanas, I. S. A. (2022). A movie recommendation system design using association rules mining and classification techniques. WSEAS Transactions on Computers, 20, 189-199.

[17] Siddique, A., Abid, M. K., Fuzail, M., & Aslam, N. (2024). Movies Rating Prediction using Supervised Machine Learning Techniques. International Journal of Information Systems and Computer Technologies, 3(1), 40-56.

[18] Jain, S., & Sarkar, M. (2023). Efficient Hybrid Movie Recommendation System Framework Based on A Sequential Model. International Journal of Intelligent Systems and Applications in Engineering, 11(9s), 145-155.

[19] Ezeh, A. (2023). Developing Machine Learning-based Recommender System on Movie Genres Using KNN.

[20] Harper, F. M., & Konstan, J. A. (2015). The movielens datasets: History and context. Acm transactions on interactive intelligent systems (tiis), 5(4), 1-19