Solving Cold Start Problem for Recommendation System Using Content-Based Filtering

Ei Ji Chia
Department of Internet Engineering
and Computer Science
Universiti Tunku Abdul Rahman
Kuala Lumpur, Malaysia
lwchia1999@1utar.my

Maryam Khanian Najafabadi
Department of Internet Engineering
and Computer Science
Universiti Tunku Abdul Rahman
Kuala Lumpur, Malaysia
maryamkn@utar.edu.my

2022 International Conference on Computer Technologies (ICCTech) | 978-1-6654-9918-7/22/$31.00 ©2022 IEEE | DOI: 10.1109/ICCTECH55650.2022.00015

***Abstract*—A recommendation system is an effective supporting tool for a user to obtain the relevant result and avoid information overloading. However, the user cold-start problem has occurred when the system does not have adequate information about the user. In this project, we propose a content-based filtering method that applies the movie genres for finding the similarities among movies. We use the Term Frequency-Inverse Document Frequency (TF-IDF) as a text mining technique for retrieving the frequent movie genres and produce the TF-IDF into a matrix. Then, we use the matrix to generate recommendation models that apply with different similarity techniques which are Cosine similarity, Pearson correlation coefficient, and Euclidean distance. As a result, the average of similarities has reached the highest of 84 percent using Cosine similarity and at most 20 percent of a not satisfied movie recommended among 30 recommendation items. The experimental results on MovieLens-20M data that apply to our approach have achieved high similarity results among movies recommended, low frequency of not-satisfied movie ratings, and it consistently improved the performance in addressing the user cold-start problem.**

Keywords-content-based filtering; recommendation system, cold-start problem; similarity measurement technique

I. INTRODUCTION

The recommendation system is a subclass of information filtering systems that use algorithms to suggest the relevant items like movies, music, products, services, etc. to users [1]. The system is invented to solve the data and information overload problem by filtering the information fragment from a large amount of data and generate information regarding user's preferences, observed behavior about the items, and predict the user pattern and behavior for generate the relevant information to the user [2]. There are two basic approaches to building a recommendation system, collaborative filtering method, and content-based filtering method.

The cold start problems occurred when the recommendation engine initialized in a new area. This is because the system does not have any form of historical data or reference from the new users or the new items. Hence, the system does not have adequate information for prediction by matching the users' preferences with the item [2].

In this paper, we introduce content-based filtering method to solve the cold-start problem because this method only methods rely on the analysis of the item's attributes and generate the prediction instead of historical data and interactions from users. there are different types of algorithm models such as vector space model, probabilistic modal, decision tree, or neural networks to find out the similarity between the items for generating the most meaningful recommendations [2]. However, in this paper, we propose Cosine Similarity, Pearson Correlation, and Euclidean Distance as algorithms of similarity measurement to generate a recommendation model.

We evaluate our model based on a dataset named MovieLens-20M from Kaggle, a data science community, and compare results with 10, 20, 30 recommendations generated with similarity measurement algorithms. The experimental result shows the average of similarity of selected movie with recommendation results.

The remainder of this paper is coordinated as 5 sections, where the Section 2 outlines the research and study of recommendation system, Section 3 explains the proposed solution in detail, Section 4 evaluate the experimental results with several scenarios. Lastly, we conclude the paper and suggest a future work.

II. RELATED WORKS

Recent studies related to our proposed method is reviewed in this section, including Section A where it outlines how previous work applies filtering method to a recommendation system. Section B outlines the previous work of the similarity measurement techniques for generating the recommendation models.

A. Filtering Method of Recommendation System

There is an approach named demographic filtering method for recommendation system, which is used to address the cold-start problem as the similarity among new item or a new user with exiting item or user will be calculated based on the demographic attributes. The author in [4] has explained about the combination of personality traits with the user's demographic attributes can be used to obtain the K-nearest neighbour for new users to address the cold-start problem.

The author in [4] have proposed the user-based collaborative filtering method to develop a recommendation system. The recommendation to the new user is based on the

978-1-6654-9918-7/22/$31.00 ©2022 IEEE
DOI 10.1109/ICCTech55650.2022.00015

similarity between a new user and an existing user with the combination of personality traits and demographic attributes.

There are two common recommendation algorithms where are collaborative filtering (CF) and content-based filtering method (CBF), it is adopted as an integrative approach for item clustering, user clustering, and reducing the cold-start problem for the music recommendation [5]. The author in [5] has explained the CBF is based on the matching cluster centroid with the item attributes, when there is no user playlist, the track music is recommended based on its popularity with an existing user. The CBF method has also being used by [7] for dealing with the new user and new item cold-start problem in the recommendation engine for an e-learning platform.

The author in [8] has addressed the cold-start problem by using a sematic CBF method for the recommendation system. This method is built the user profile regarding the similarity of user preferences with semantic of item attributes, so when a new profile is built, the attributes of items are matched with existing items attribute and generates the recommendation for solving the cold-start problem.

B. Similarity Measurement Technique

Similarity measurement is one of the techniques that optimizes the solution to resolve the cold-start problem for the recommendation system. The author in [3] has explained the concepts of the Jaccard correlation coefficient, the optimized similarity techniques such as Cosine similarity index, Pearson formula, and Matrix Factorization algorithm to measure the different aspects of similarities such as the similarity among movies and similarity among directors and actors.

To measure the similarity between selected item and the recommended item, the author in [4] has proposed 3 algorithms for finding the similarity among demographic attributes with personality traits where are the Pearson correlation coefficient for finding similarity based on users' interest change, similarity among user's personalities, and similarity among user's demographic attributes.

Jaccard correlation coefficient is the similarity measurement technique for collaborative method filtering, it measures the similarity of 2 sets of combined measures and getting a combined similarity range [9]. According to [6] findings, the Jaccard correlation coefficient has been used to to calculate the similarity between the sample sets for improving the accuracy of movie recommendations. However, it lacks accuracy, because it calculates the similarity based on the appearance of the movie actor or directors. Hence, the author in [3] has introduced the Cosine similarity, Pearson formula, and Matrix Factorization algorithm to optimize the similarity measures. The statement was supported by [10], which apply the Matrix Factorization algorithm by evaluating the performance of user-rating interaction that builds a multi-attention deep neural network model.

The Cosine similarity index and Pearson correlation coefficient formula were being used by the author in [9] for conducting the comparison of similarity measurement and to enhance the performance of the recommendation system by improving the accuracy of recommendation to users. The author in [11] has explained the Cosine similarity in calculating the similarity between unit items (movie) with the genre of the movie so it can be refined as a preliminary recommendation.

The author in [13] has applied uniform Euclidean distance (ED) for enhancing the recommendation system with the collaborative filtering method. By generating the similar results from selected item, uniform ED improves the accuracy compared to standard Euclidean distance as the standard ED only focus on the counting the number of co-rated items among user while uniform ED concerns on the union of the users' rating [13].

III. PROPOSED RECOMMENDATION MODEL

This paper is proposed to resolve the cold-start problem for the new user in the recommendation system by using the content-based filtering method with similarity measurement techniques. In this paper, we apply a text mining technique named Term Frequency-Inverse Document Frequency (TF-IDF) for data filtering and information retrieval. Based on the TF-IDF matrix, we build a content-based filtering recommendation model with applying similarity techniques, which are Cosine similarity (CS), Pearson correlation coefficient (PPMCC), and Euclidean distance (ED). The performance of recommendation models is evaluated using the average of similarity among movies recommended and identified the not-satisfied movies from different selected movies.

A. Data Filtering Process

Before building a recommendation system, data pre-processing has been implemented to reduces the noisy data and increase the accuracy of the output. In this paper, the proposed machine learning algorithm where is the Term Frequency-Inverse Document Frequency (TF-IDF) is to be used to determine the relevance among the items. TF is used for calculating the frequency term in an item set while IDF is used for calculating the universe of the frequency term among itemsets [12]. Hence, the TF-IDF matrix is used for indicates the similarity between itemsets. Starting with TF, it is to measure the frequency of a term in a current document, the working principle has shown as in (1).

$$tf_{x,y} = \frac{t}{N} \tag{1}$$

While IDF is used to measure the significance of a term in the whole document. The working principle of IDF has been shown as in (2).

$$idf = \log(\frac{N}{df_x}) \tag{2}$$

Hence, the TF-IDF is calculated as in (3), to retrieve the information by measuring the significance of a term in a set of documents to prepare for recommendation models where the higher the score indicates the higher relevance of the term in the particular document [12].

$$tfidf = tf_{x,y} \times idf_x \quad (3)$$

B. Similarity Measurement Technique

Since a TF-IDF matrix has been formed for frequent data retrieval, a recommendation is now built based on the similarities among the selected movie and recommended movies through the matrix. Three similarity measurement techniques have been used to compute the similarity among movies where are Cosine similarity, Pearson correlation coefficient, and Euclidean distance.

Cosine similarity (CS) is an algorithm that can be used to compute the similarities among the movies based on their attributes. In the mathematical perspective, the smaller the angle between two vectors, the higher the similarity [15]. To generate a recommendation based on the formula of CS, it consists of two main parts where are generate a matrix of movie genres and calculate the distances regarding the given movie title so that (4) could be used in finding the similarity among items. After the matrix value has been obtained based on (3), we apply Cosine similarity to compute the similarity between the selected movie and recommended movies.

$$similarity = \cos(\theta) = \frac{A \cdot B}{||A||\,||B||} = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} A_i^2}\sqrt{\sum_{i=1}^{n} B_i^2}} \quad (4)$$

Pearson correlation coefficient (PPMCC) is the alternative similarity measurement technique that is used to compare the performance of recommendation models with other models that applied a different similarity technique where are CS and Euclidean distance. To initialize the recommendation model, the TF-IDF matrix has been used to generate a PPMCC matrix among movies based on (5), then the matrix has been used to compute the movie recommendations and defined the similarities regarding the PPMCC matrix.

$$r = \frac{\Sigma(x-m_x)(y-m_y)}{\sqrt{\Sigma(x-m_x)^2\ \Sigma(y-m_y)^2}} \quad (5)$$

In the recommendation system, PPMCC has the same mathematical formula as an adjusted CS which the mean values are done subtracting with the given sample values before it computing to CS formula. In other words, CS finding similarity between items based on vector while PPMCC finding similarity based on level of degree, either positive or negative [14]. Therefore, PPMCC is considered as the centered Cosine similarity.

The Euclidean distance (ED) as a similarity technique is used to calculate the distance between the movies that having the most similar genres which have similar functionality with CS and PPMCC recommendation models. The ED is illustrated in (6). This technique is implemented to compare with CS and PPMCC in terms of similarities among movies recommended. To initialize ED, the TF-IDF matrix has been built which is similar with the CS and PPMCC, ED computed the movie recommendations regarding the TF-IDF matrix.

$$d(x,y) = \sqrt{\sum_{i=1}^{n}(y_i - x_i)^2} \quad (6)$$

However, the working principles of ED differ from CS. CS similarity score is calculated based on angles between two movies while the ED is the distance between the points (selected movie and movie might be recommended), so the result of similarities or distance might be greater than 1 which is considered as an abnormal result because the similarity has to be in the range of zero to one. Therefore, normalization has been implemented to the ED similarity technique to range the similarity of 0 to 1.

IV. Experimental Results

To understand the recommendation result whether it has an accurate result, we propose two metrics where are average of similarity among movies recommended (7), and not satisfied rating and similarity among movies recommended (8) to measure the performance of the recommendation process. The average of similarity has been used to evaluate the overall similarities or how much closest distance between the selected movie and the nearby movies.

$$Average = \frac{\sum Similarity\ between\ movies}{No.of\ movies\ recommended} \quad (7)$$

The range of similarity has been limited to 0 to 1 across all the similarity measurement techniques. This is to ensure all the similarity techniques are evaluated under the same fixed variables.

Throughout the experiments, some items are considered not satisfying results. We define the mean ratings of a movie recommended that is under 2.5 over 5 is considered not satisfied rating, and if the movie similarity is under 0.5 or 50% of 100, then the item recommended is considered a not satisfying recommendation.

$$Rate\ of\ not\ satisfied\ ratings = \frac{n-x}{n} \times 100\% \quad (8)$$

To evaluate the performance of recommendation models, we have observed the recommendation results using different movie titles. There are three evaluation recommendation models built with similarity techniques using Cosine similarity (CS), Pearson correlation coefficient (PPMCC), and Euclidean distance (ED) for the evaluation. This experiment computes 10, 20, and 30 movies recommended to each recommendation model to observe the different average similarities and the rate of the not satisfied movies recommended.

Fig. 1 has shown that the Cosine Similarity model has obtained the highest average of similarity of 81%, 73%, and 68% for 10, 20, and 30 movies recommended by selecting the movie Kung Fu Panda (2008), the Euclidean distance model has reached the lowest average similarities of only 62%, 58%, and 56% for 10, 20, 30 movies recommended. The average similarity score of Pearson Correlation model has only 1% lesser than the Cosine similarity model.

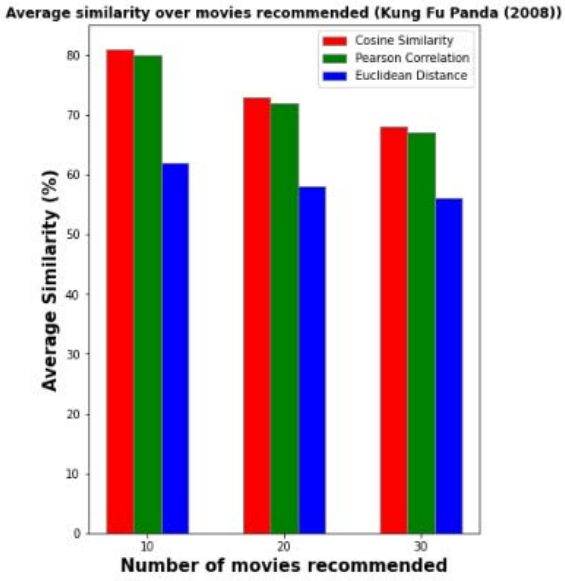

Figure 1. Average similarity over movies recommended (Kung Fu Panda (2008)).

Fig. 2 has shown the result of overall average similarities, it is lower than movie one because the movie titles consist of different features (genres) that are affected the performance of recommendation. Similar to the recommendation results from movie 1, the CS recommendation model has reached the highest average of similarities among 10, 20, 30 movies recommended, ED recommendation model has obtained the lowest average similarities compared to CS and PPMCC.

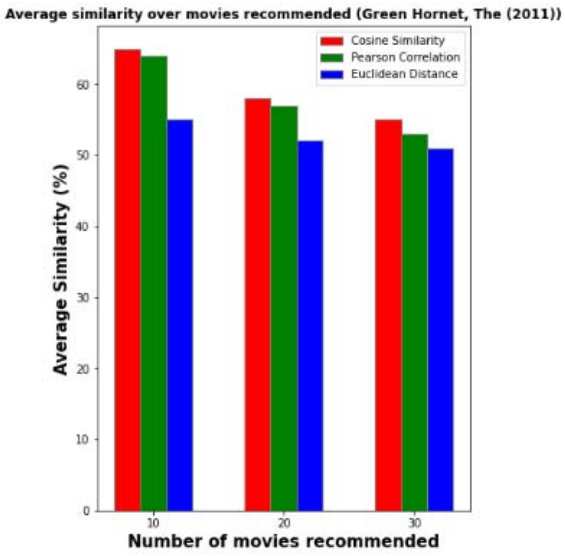

Figure 2. Average similarity over movies recommended (The Green Hornet (2011)).

Fig. 3 has shown the CS has also reached the highest average similarities among movies recommended compared to PPMCC and ED models. PC has only a minor difference from the CS model.

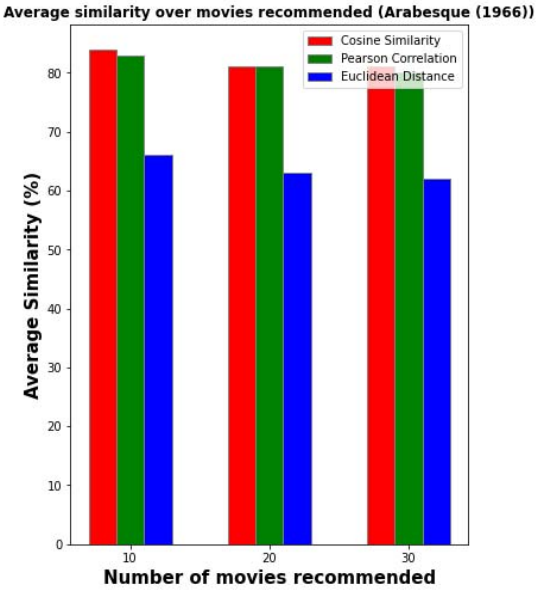

Figure 3. Average similarity over movies recommended (Arabesque (1966)).

Throughout the recommendation results, we observed the not-satisfied rating among movies recommended. The result shown frequency of not-satisfied movies are similar among CS, PPMCC, and ED recommendation models. However, the result is different when the other existing movie title is selected to compute the recommendation models.

TABLE I. FREQUENCY OF NOT-SATISFIED RATINGS FROM MOVIES RECOMMENDED

Movie Selected	Rating not satisfied		
	10 movies	*20 movies*	*30 movies*
Movie 1	1	2	4
Movie 2	2	5	6
Movie 3	1	1	3

Movie 1 is Kung Fu Panda (2008), Movie 2 is Green Hornet, The (2011), and Movie 3 is Arabesque (1966).

Table 1 has shown that Movie 2 has the highest frequency of not-satisfied ratings from movies recommended while Movie 3 has the least frequency ratings. The result indicates the overall performance of the recommendation models has reached at least 80% and above the accuracy of recommendation result that consists of relevant items. However, the accuracy of recommendation results is affected by different movies because of the characteristic of CBF in which only filters the itemset that having similar features (genres) but also filtered with the highest mean rating of movies. Therefore, if the rate of not-satisfied ratings from movies recommended is not exceeding 50%, we considered it as an accurate result.

V. CONCLUSION

Over the last decades, a content-based filtering method recommendation system emerged as a solution to address the cold-start problem for new users because it does not require collecting the user records to generate a recommendation. In this research, we have proposed six recommendation models using three similarity measurement techniques following by mining content features from the datasets. To evaluate the recommendation models, we have defined the average of similarities among movies recommended and observed the performance of recommendation results based on similarities. Besides, we have identified the rate of not satisfied rating from the movie recommended to evaluate the accuracy of recommendation results.

For this paper, we practiced with the movies with detailed profiles like the genres and the ratings from existing users. However, when a new item (new movie's profile) is introduced to the current recommendation models, the system will not recommend any item to the new users since the recommendation models do not have adequate information to generate recommendations. Whether our work can address the cold-start problem for a new item is the subject of a future project.

ACKNOWLEDGMENT

The author would like to thank the Research Management Center of UTAR University through UTARRF Grant Scheme project IPSR/RMC/UTARRF/2020-C1/M01.

REFERENCES

[1] Y. Zhou, D. Wilkinson, R. Schreiber, and R. Pan, "Large-Scale Parallel Collaborative Filtering for the Netflix Prize", Proc. International Conference on Algorithm Applications in Management, AAIM 2008, pp. 337-348, doi:10.1007/978-3-540-68880-8_32.

[2] F. O. Isinkaye, Y. O. Folajimi, and B. A. Ojokoh, "Recommendation systems: Principle methods and evaluation", vol 16, Nov 2015, pp. 261-273, doi:10.1016/j.eij.2015.06.005.

[3] P.Yi, C. Yang, X. Zhou, and C. Li, "A movie cold-start recommendation method optimized similarity measure", Proc. ISCIT International Symposium and Communications andd Information Technologies, Sep 2016, pp. 231-234, doi: 10.1109/ISCIT.2016.7751627.

[4] Z. Tilahun, H. D. Jun, and A. Oad, "Solving Cold-Start Problem by Combining Personality Traits and Demographic Attributes in a User Based Recommender System", Proc. IJARCSSE International Journal Advanced Research in Computer Science and Software Engineering, vol 7, May 2017, pp. 231-239, doi:10.23956/ijarcsse/V7I4/01420.

[5] P. Darshna, "Music recommendation based on content and collaborative approach & reducing cold start problem", Proc. 2nd International Conference on Inventive Systems and Control (ICSIC 2018), Jan 2018, pp. 1033-1037, doi:10.1109/ICISC.2018.8398959.

[6] S. Bag, S. K. Kumar, and M. K. Tiwari, "An efficient recommendation generation using relevant Jaccard similarity", vol 483, May 2019, pp. 53-64, doi: 10.1016/j.ins.2019.01.023.

[7] M. Špilka, A. Posoldova, G. Rozinaj, and P. Podhradský, "Importance of recommendation system in modern forms of learning", Proc. 23rd International Conference on System, Signals, and Image Processing (IWSSIP), May 2016, pp. 1-4, doi:10.1109/IWSSIP.2016.7502744.

[8] F. B. Moghaddam, and M. Elahi, "Cold Start Solutions For Recommendation Systems", Apr 2019, pp. 1-25, doi:10.1049/PBPC035G_ch3.

[9] A. A. Amer, H. I. Abdalla, and L. Nguyen, "Enhancing recommendation system performance using highly-effective similarity meassures", vol 217, Apr 2021, pp. 1-19, doi: 10.1016/j.knosys.2021.106842.

[10] J. Wang, and L. Liu, "A multi-attention deep neutral network model base on embedding and matrix factorization for recommendation", vol 1, Jun 2020, pp. 70-77, doi: 10.1016/j.ijcce.2020.11.002.

[11] O. Shahmirzadi, A. Lugowski, and K. Younge, "Text Similarity in Vector Space Models: A Comparative Study", Proc. 18th IEEE International Conference On Machine Learning And Applications (ICMLA), Dec 2019, pp. 659-666, doi: 10.1109/ICMLA.2019.00120.

[12] T. Badriyah, S. Azvy, W. Yuwono, and I. Syarif, "Recommendation system for property search using content-based filtering method", Proc. 2018 International Conference on Information and Communications Technology (ICOIACT), Mar 2018, pp. 25-29, doi:10.1109/ICOIACT.2018.8350801.

[13] T. Gao, B. Cheng, J. Chen, and M. Chen, "Enchancing Collaborative Filtering via Topic Model Integrated Uniform Euclidean Distance", China Communication, vol 14, Nov 2017, pp. 48-58, doi:10.1109/CC.2017.8233650.

[14] N. Dharaneeshwaran, S. Nithya, A. Srinivasan, and M. Senthilkumar, "Calculating the user-item similarity using Pearson's and cosine correlation", 2017 International Conference on Trends in Electronics and Informatics (ICEI), May 2017, pp. 1000-1004, doi:10.1109/ICOEI.2017.8300858.

[15] N. Bhalse, and R. Thakur, "Algorithm for movie recommendation system using collaborative filtering", Jan 2021, pp. 1-6, doi:10.1016/j.matpr.2021.01.235.