
Design and test music recommendation system for online music websites using collaborative filtering approach

Teddy Oswari*

Faculty of Economic,
Gunadarma University,
Depok, Indonesia
Email: toswari@staff.gunadarma.ac.id
*Corresponding author

Tristyanti Yusnitasari

Faculty of Computer Science and Information Technology,
Gunadarma University,
Depok, Indonesia
Email: tyusnita@staff.gunadarma.ac.id

Reni Diah Kusumawati

Faculty of Economic,
Gunadarma University,
Depok, Indonesia
Email: reni_dk@staff.gunadarma.ac.id

Saurabh Mittal

Faculty of Information Technology,
GL Bajaj Institute of Management and Research,
Greater Noida, UP, India
Email: sau275@gmail.com

Abstract: The music industry is currently growing very rapidly. The limited time possessed by consumers to consume products conventionally is the main reason for online purchases by consumers. This change also occurred in the music industry, so that online music sites that provide convenience in service and this convenience made consumers decide to buy music products online and abandoned the old habit of consuming conventional music products on CDs. The ease of consuming songs online does not make the user easy to determine the songs they want to hear. The difficulty of users in determining what songs to listen to in a web music can be overcome by the existence of a music recommendation system, where music sites will provide recommendations on songs based on user preferences. This study aims to create a music recommendation system on online music sites to provide convenience for consumers in choosing music products according to their preferences. This

study explains the problem of users of online music sites who do not rank a song that has been heard, and overcome it with a recommendation system using user-based collaborative filtering methods.

Keywords: collaborative filtering; music recommendation system; music website; online system; rating prediction.

Reference to this paper should be made as follows: Oswari, T., Yusrinasari, T., Kusumawati, R.D. and Mittal, S. (2020) 'Design and test music recommendation system for online music websites using collaborative filtering approach', *Int. J. Digital Signals and Smart Systems*, Vol. 4, Nos. 1/2/3, pp.64–79.

Biographical notes: Teddy Oswari registered as lecturer since 2002 at Gunadarma University. He is active as a reviewer and have more than 90 scientific articles for national and international. He was a stakeholder of PT Carana Indonesia (holding company) as a President Commissioner, and as President Director of PT, Carana Lentera Prima on 2006–2009. He is also a Research Collaborator at the Department of AMBD in Indonesia, University of Cumbria, UK. He also registered in a Financial Business Process team and Insurance Information System Auditor at PT, Asuransi Jiwasraya. He is a member of The Future Defence Leader from Defence Ministry of Republic of Indonesia and member of International Global Management Association (IGMA), Canada. He is at the supervision team of excellent university student, Qingdo-China.

Tristyanti Yusrinasari is a graduate of Information Technology Doctoral Program, Gunadarma University. She is the Head of Information System Laboratory, Gunadarma University. Her activities are developing some applications and instructor for some courses at Information System Laboratory, Gunadarma University. She has written some research papers published in national journals and attended some seminars and presented her papers. She has organised some seminars and workshops in area of information technology.

Reni Diah Kusumawati has educational background in Management for Bachelor degree, Accounting Information System for Master degree, and PhD degree at Gunadarma University, Indonesia. She is a staff of secretariat at Gunadarma University. Her current research focuses on customer behavior and marketing. Besides that, she is interested with research about information system and trying to combine information system and economic sciences.

Saurabh Mittal has a rich experience of 14+ years in academics and industry. Currently, he is serving as an Area Chairperson – IT at GL Bajaj Institute of Management and Research, Greater Noida, UP (India). Prior to this, he was associated with the Asia-Pacific Institute of Management, New Delhi, JK Padampat Singhania Institute of Management and Technology, Gurgaon (member JK Organization) and Aggarwal College Faridabad as a Faculty in IT/Computer Science. He has research papers published in international journals, listed/indexed in SCOPUS, Web of Science and ABDC. He has 18 publications in the domain of information technology and corporate social responsibility in the international journals/book chapters/ national journals and magazines.

This paper is a revised and expanded version of a paper entitled 'Music recommendation system design for online platform with the concept of unified modelling language (UML)', presented at IEEE International Conference on Sustainable Energy, Electronics & coMputing Systems, ITS Engineering College, Greater Noida (UP) India, 26–27 October 2018.

1 Introduction

The development of data and information that is so large causes problems in conducting information searches. Search engines that exist today can solve these problems, but are still limited to the questions that are asked, while the search engine still cannot provide personalised information. Problems will arise when consumers only know the desired product but do not know the name of the product or what represents the product. This problem also occurs in music products, where consumers do not know the song title or singer of the desired song, this is like exploring a large store and people have to surround the store to find the desired product but do not find the product.

Various online music sites that exist today have a large collection of songs with various categories, but users sometimes find it difficult to choose songs for various reasons and one of them is because users do not know what songs are interesting to hear. The availability of features and convenience in using a music site is very necessary so that users feel comfortable. Everyone may have different song tastes, so a site is needed that can also understand consumer tastes that are user friendly. The availability of comment fields on a music site about a song will greatly help users in choosing songs, users can first read the testimony given by the previous user about a song before determining what song will eventually be consumed.

Various music sites that exist today have provided ratings for a song by only giving stars to a song according to the user's assessment of the song being heard. Some online music sites like iTunes Radio, Grooveshark, Pandora and Spotify provide features to provide recommendations about a song. Some music sites have similar services such as song search features based on tags or labels that are stored as additional information on each song and focus on the search feature. The convenience provided by online music sites makes music consumers begin to switch to consuming music online and leave old habits in consuming music.

This study aims to develop a recommendation system for music products. The recommendation system that will be developed will automatically provide recommendations from a large database of songs, and provide song recommendations that match the user's music preferences. The quality of this recommendation is influenced by many factors related to users and songs. Among the various related factors, the content of a song such as vocals, melodies, rhythms, genres, instruments or lyrics has a very important influence on the user. Users who don't listen to content will find it difficult to determine what songs they want to consume, so a recommendation system with a good level of prediction will be needed to help listeners or music lovers choose a song.

One method in the recommendation system that will be used in this study is collaborative filtering, collaborative filtering methods recommend songs by considering the preferences of other like-minded users. This method will recommend songs by considering the preferences of other users who have the same musical taste, so that the preferences of the previous user can be used to provide song recommendations to the next listener who has the same musical taste. Another method is the content base recommender system where the system will automatically recommend music that is liked by users.

This music site with a recommendation system feature in addition to providing convenience for music consumers will also benefit companies in the music field, because with this recommendation system companies can automatically know the music that is

being liked by consumers, so the music industry can produce more types – the type of music that is preferred by consumers in this digital era. The advantage of this recommendation system for musicians is that they can find out what type of music they are interested in, so that musicians can create more works. This study developed a system design recommendation for online music products with the rapid application development (RAD) concept with a machine recommendation system in it.

2 Literature review

2.1 Recommendation system

The recommendation system helps users to identify products that are in line with the needs, pleasures and desires of user. The recommendation system will guide users to find relevant and useful products from the number of products available.

The recommendation system is a system that works to search and get recommendations of personal information, products, or services (Melville and Sindhawani, 2010). The way to search for items that will be recommended can be done based on similarity, such as the resemblance to an item with other items based on content or similarity of tastes of a user with other users based on the rating given to the item.

The recommendation system is an application results model of observations on the circumstances and desires of users, in this case can be said as readers, customers and others. The recommendation system utilises a person's opinion on an item in a particular domain or category, to help someone in choosing a product. The recommendation system requires the right recommendation model, so the things that is recommended is in accordance with the customer's wishes, and makes it easier for customers to make the right decisions in determining the product to be purchased (Robi, 2007).

The recommendation system is a program that predicts an item, such as recommendations for films, music, books, news and others that are of interest to the user. This system runs by collecting data directly or indirectly from users (Scafer et al., 2016). Direct data collection can be done as follows:

- 1 ask the user to rate an item
- 2 asking the user to rank on favourite items at least choose one favourite item
- 3 give the user a choice of items and ask him to choose the best
- 4 ask the user to make a list of the items they like most or items they do not like.

Data collection by not directly relating to a user is done in the following ways:

- 1 observe items seen by a user on a web
- 2 collect transaction data at an online store.

The recommendation system functions to provide recommendations, usually in the form of goods or services to people, for example: Which book is good to read? What music is good to hear? Which Facebook account should I follow? Which school is right for me? The usefulness of the recommendation system is to narrow information overload, narrow down choices, and find new things so that information is useful for the user. In addition, the recommendation system can provide benefits to the provider in terms of increasing

user loyalty, because it is able to provide more personal recommendations to its users, increase purchases, get opportunities for promotion and gain knowledge about its users.

2.2 Traditional recommendation system

The form of a recommendation system that is said to be traditional is a system of recommendations that is common and often done before that usually recommends items by predicting rankings or providing a list of ranking items for each user (Tuzhilin and Adomavicius, 2005). Various traditional techniques that have existed in making recommendations, including content-based recommendation, collaborative recommendation and hybrid recommendation.

2.2.1 Content-based recommender system

Content-based recommender system is a form of recommendation using user profiles or item descriptions for recommendations. Content-based filtering: 'Recommend a book that matches the type of book I like'.

In order to provide relevant information to users, user profiles must be created using web mining methods or information retrieval methods based on the attributes and features of the items (Pereira and Varma, 2016). In content-based recommendation system, it filters the items based on the similarity of content that users are interested in. Some of the advantages of content-based recommender system are giving users freedom through exclusive assessments used by active users to create their own profiles. In addition, a content-based recommendation system also provides transparency to active users by providing an explanation of how the recommender system works and also allows for recommend items that have not been placed by any user. The content-based recommender system has several disadvantages, including that it is very difficult to get feedback from users in content-based filtering because users usually do not rank items and therefore it is not possible to determine whether the recommendations are correct so that it is difficult to generate attributes of items (Thorat et al., 2015).

2.2.2 Collaborative filtering system

Collaborative filtering system is a system that works by collecting feed-back in the form of a rating of items with a certain scope and looking for similarities in user behaviour to determine recommendations for an item (Pazzani and Billsus, 2007). Collaborative filtering: 'Recommend books that my friends like'. Use community preferences. The form of collaborative filtering can be in the form of memory-based collaborative filtering or model-based.

a Memory-based collaborative filtering

It is a form of technique that tries to find users who are similar to active users and use their preferences to predict the rankings of active users. Some of the advantages of memory-based collaborative filtering techniques, which are very measurable for large data sizes, easy to apply to any condition, make it easy to update the database so that new data can be handled easily and provide feed-back to explain how the recommender work system. Explicitly include content features or descriptions that cause items to appear in the list of recommendations (Thorat et al., 2015). This

memory-based collaborative filtering technique also has drawbacks, including being very slow, because it uses the entire database every time it makes predictions, making predictions that are unreliable and inaccurate if the active user does not have the same item as all users who have rated the item recommended. Neighbour-based collaborative filtering is one of the most representative memory-based collaborative filtering models. Neighbour-based collaborative filtering involves two steps: calculation of similarity and prediction. In the equation calculation step, the similarity value can be measured between the user or item.

b Model-based collaborative filtering

Is a model of machine learning or data mining that finds complex assessment patterns in training data then makes intelligent predictions or recommendations for users based on the learning model (Thorat et al., 2015). There are several advantages of a model-based collaborative filtering algorithm that is, improving predictive performance, can be measured by actual datasets and easily recognisable from the fittings. In addition, there are some limitations in model-based collaborative filtering techniques such as sparsity problems so that they cannot produce recommendations that make sense for users who do not rank and find it difficult to add data to a model-based system because it is not flexible. Model-based collaborative filtering methods such as Bayesian models, clustering models, and network dependencies, have been explored to overcome the shortcomings of model-based collaborative filtering algorithms. Usually, classification algorithms are unsupervised learning algorithms and are designed to combine objects into different categories. The clustering algorithm can be used as a collaborative filtering model if the user rating is categorical, and the regression model and singular value decomposition (SVD) method and are used for numerical assessment. Clustering algorithms, such as K-means, are used to group users or items in groups. Then, the probability of conditional ratings for items can be calculated based on their group information (Thorat et al., 2015; She et al., 2016). The most successful model-based collaborative filtering technique is matrix factorisation (MF) (Scafer et al., 2016). In the previous research found general factors that could be the basic reason for the assessment given by the user (Scafer et al., 2016). For example, in the film recommender system, this factor can be in the form of a genre, actor, or film director that can influence the behaviour of user ratings. MF techniques not only find this hidden factor, but also give importance to each user and how each item fulfils each factor.

2.2.3 Hybrid recommender system

Hybrid recommender systems are methods that incorporate content-based and collaborative filtering techniques to get better predictions or performance recommendations (Gedimina and Youngok, 2012). Some methods that can be implemented for hybrid methods are by applying collaborative and content-based methods individually and combining the predictions obtained, integrating several characteristics of content based into collaborative approaches, as well as collaborative approaches that incorporate some of their characteristics into content-based, then build a general consolidation model that integrates content-based and collaborative characteristics. Cold start and sparsity problems in the recommendation system can be solved using the hybrid method (Thorat et al., 2015).

2.2.4 Rapid application development

RAD or rapid prototyping is a model of software development processes belonging to a multilevel technique. RAD emphasises on short, short, and fast development cycles (Daud et al., 2010). Short time is an important limitation for this model. RAD uses an iterative (repetitive) method in developing a system where the working model (working model) of the system is constructed at the beginning of the development stage with the aim of determining the user's requirements. The work model is used only occasionally as the basis for the final system design and implementation. RAD is a software development methodology that focuses on building applications in a very short time. This term is a common marketing keyword that describes applications that can be designed and developed within 6,090 days, but it was originally intended to describe a development process involving application prototyping and iterative development.

3 Method

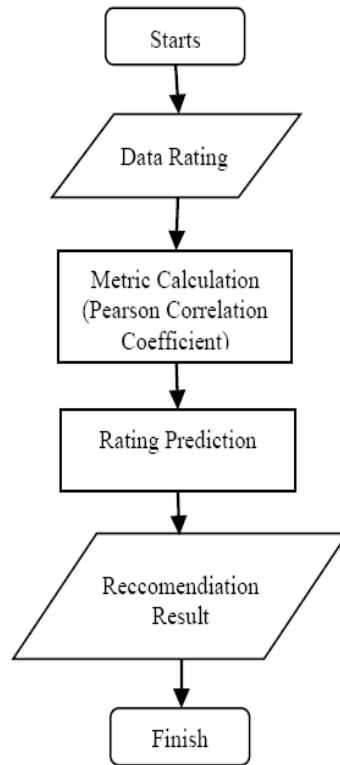
The method used in this study is the RAD method with four existing phases, and at the assessment stage involves the analyst and user (Kendall and Kendall, 2010). RAD is appropriate for producing a software system with urgent needs and a short time in its completion. The four phases described by Kendall consist of the requirements planning, design workshop, instruction, and implementation phases. Requirement planning, the stages of planning the conditions carried out on the film recommendation system include analysis of several. Namely system requirements analysis, analysis of input data, analysis of calculation metrics, and analysis of rating prediction. The design of the workshop or the design stage carried out on this music recommendation system consists of the flow of the music recommendation system algorithm.

4 Result

The rating process carried out by a user on an item, can provide important information to the system in providing a recommendation to other users. When the system presents so much data from many items and the user is faced with a variety of choices, the existence of a rating can be used as help to help a user in making choices and narrowing down searches. Without a rating system, it will be difficult to provide assistance to a user because there is no standardisation and information that is explore regarding the user's preferences. The user-based collaborative filtering recommendation system has two-stage namely the distance measurement step (metric calculation) and the step of providing recommendation through rating prediction.

At the distance measurement stage, rating data is to measure the distance from an active user to another user. The distance calculation is done one by one using the Pearson correlation distance coefficient calculation method. After the Pearson value is obtained, specify the user closest to active users sorted by their Pearson value. After getting a rating prediction value, the system then provides music recommendation based on the rating prediction with the highest value, and display according to the value of n .

Figure 1 User-based collaborative filtering recommendation system



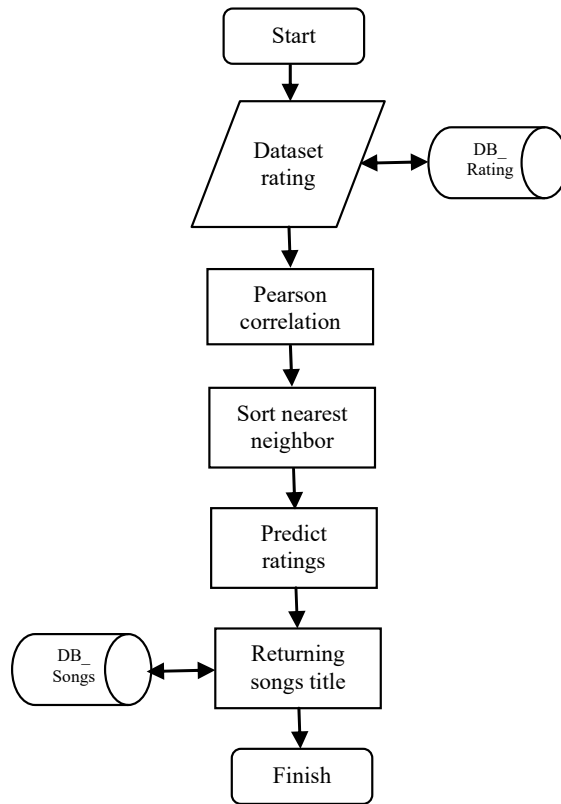
The dataset used in the form of 250 songs title and more than 50 million rating data from 5 million users. The data obtained includes title data and song information, user data and rating data given by user to a song title.

Data input on the recommendation system based on collaborative filtering is a collection of rating data given by a number of users to several songs title and has been categorised as a dataset. The song rating dataset consist of approximately 400 rating data given by 25 users to 50 songs title as a dataset. The data used has the following detail in Table 1.

Table 1 Detail data on input data

No.	User	Rating	No.	User	Rating
1	5 users	5 songs	4	5 users	25 songs
2	5 users	10 songs	5	3 users	50 songs
3	5 users	20 songs	6	2 users	0 songs

In collaborative filtering recommendation system, the dataset is the data that will be used in the recommendation process. Dataset are taken based on the attribute values that meet the needs in the recommendation process.

Figure 2 Dataset

The explanations of Figure 2 are:

- 1 The dataset that already contains the values needed in the recommendation process, is stored in database ratings.
- 2 Data in the database is extracted and stored in a data folder.
- 3 Data that has been extracted will be carried out distance measurement process using the Pearson correlation coefficient method.
- 4 After obtaining Pearson values between active users and other users in the database, the system then sorts the user with the closest value to the active user as a neighbour/neighbour consisting of one or several closest users.
- 5 The rating value prediction process is carried out next by using the closest user data (neighbour) obtained with Pearson value data obtained in the previous process.
- 6 Rating prediction data obtained is then carried out sorting/sorting with the number n, before then displayed as a recommendation for active users.

5 Discussion

The metric calculation stage is the stage where the system will measure the value of distance and compare between an active user and another user. The purpose of this stage is to classify some users who have closeness and similarity preferences with active users, commonly referred to as neighbours or neighbours. At this stage the distance measurement (metric) algorithm is used with the Pearson correlation coefficient method.

Pearson correlation coefficient algorithm is a measurement method (metric) that is used to determine the rating correlation rating given by an active user to another user. Pearson values obtained can also be used as a value of the comparison of proximity between an active user and other users so that they can be grouped into neighbour users. Correlation values are also commonly displayed as lines that connect between rating values.

Pearson correlation coefficient has the following formula/calculation formula:

$$r = \frac{\sum_{i=1}^n x_i y_i - \frac{\sum_{i=1}^n x_i \sum_{i=1}^n y_i}{n}}{\sqrt{\sum_{i=1}^n x_i^2 - \frac{(\sum_{i=1}^n x_i)^2}{n}} \sqrt{\sum_{i=1}^n y_i^2 - \frac{(\sum_{i=1}^n y_i)^2}{n}}}$$

For example, taken from the dataset, there is a rating data that compares the rating given by user 1 with the rating given by user 2 to several song titles as found in Table 2.

Table 2 Rating user

	<i>Song title 1</i>	<i>Song title 2</i>	<i>Song title 3</i>	<i>Song title 4</i>	<i>Song title 5</i>
User 1	4.75	4.5	5	4.25	4
User 2	4	3	5	2	1

5.1 Calculating the numerator

The first step is to calculate the value from the beginning of the numerator, which is the value of $\sum_{i=1}^n x_i y_i$ which is the sum of the variables x (user rating 1) with the value y (user rating 2). For each song title, the rating value of user 1 and user 2 is multiplied and the results are added up.

$$\begin{aligned}
 &= (4.75 \times 4) + (4.5 \times 3) + (5 \times 5) + (4.25 \times 2) + (4 \times 1) \\
 &= 19 + 13.5 + 25 + 8.5 + 4 \\
 &= 70
 \end{aligned}$$

Then, the remainder of the numerator is calculated and broken down according to the variable. Where the value of $\sum_{i=1}^n x_i$ is the sum of the rating given by Clara as the variable x :

$$\begin{aligned}
&= 4.75 + 4.5 + 5 + 4.25 + 4 \\
&= 22.5
\end{aligned}$$

And the value of $\sum_{i=1}^n y_i$ which is the sum of the rating given by user 2 as the variable y :

$$\begin{aligned}
&= 4 + 3 + 5 + 2 + 1 \\
&= 15
\end{aligned}$$

While the value of n or the number of song title given a rating of 5. So that the calculation process is as follows:

$$\begin{aligned}
&\sum_{i=1}^n x_i y_i - \frac{\sum_{i=1}^n x_i \sum_{i=1}^n y_i}{n} \\
&= 70 - 67.5 \\
&= 2.5
\end{aligned}$$

The obtained value from numerator or divider number that is equal to 2.5.

5.2 Calculating the denominator

The next stage of calculating Pearson's value is to calculate the value of the denominator. First the numerator values are broken down and counted separately as follows.

The first part is the calculation for user 1 or variable x . The formula used is

$\sum_{i=1}^n x_i^2 - \frac{\left(\sum_{i=1}^n x_i\right)^2}{n}$ where the value $\sum_{i=1}^n x_i^2$ of each rating user 1 gives to each music and then squared and summed:

$$\begin{aligned}
&= (4.75)^2 + (4.5)^2 + (5)^2 + (4.25)^2 + (4)^2 \\
&= 22.5625 + 20.25 + 25 + 18.0625 + 16 \\
&= 101.875
\end{aligned}$$

Next is the value of $\left(\sum_{i=1}^n x_i\right)$ which is the sum/sum of the rating given by user.

- 1 This value has been obtained in the process of calculating the previous numerator, which is worth 22.5. Then the value is squared and divided by the value of n , namely 5:

$$\begin{aligned}
&\frac{\left(\sum_{i=1}^n x_i\right)^2}{n} \rightarrow \frac{(22.5)^2}{5} \\
&= (22.5)^2 / 5 \\
&= 506.26 / 5 \\
&= 101.25
\end{aligned}$$

After obtaining the value from the whole, then the values are reduced and accentuated as follows:

$$\begin{aligned}
&= \sqrt{101.875 - 101.25} \\
&= 0.79057
\end{aligned}$$

The second part is the calculation using y variable or rating data from user.

- 2 The formula used is the root of $\sum_{i=1}^n y_i^2 - \frac{\left(\sum_{i=1}^n y_i\right)^2}{n}$, where the value of $\sum_{i=1}^n y_i^2$ is the value of each rating that user 2 gives to each song and then squared and summed the results:

$$\begin{aligned}
&= (4)^2 + (3)^2 + (5)^2 + (2)^2 + (1)^2 \\
&= 16 + 9 + 25 + 4 + 1 \\
&= 55
\end{aligned}$$

Then the value of $\left(\sum_{i=1}^n y_i\right)$ is the number of ratings given by user 2 for all songs that have been obtained by the process of calculating the numerator by 15. Then the value is squared and divided by the number n .

$$\begin{aligned}
&\frac{\left(\sum_{i=1}^n x_i\right)^2}{n} \rightarrow \frac{(15)^2}{5} \\
&= (15)^2 : 5 \\
&= 225 : 5 \\
&= 45
\end{aligned}$$

After getting the first and second values, then the value is reduced and rooted as follows:

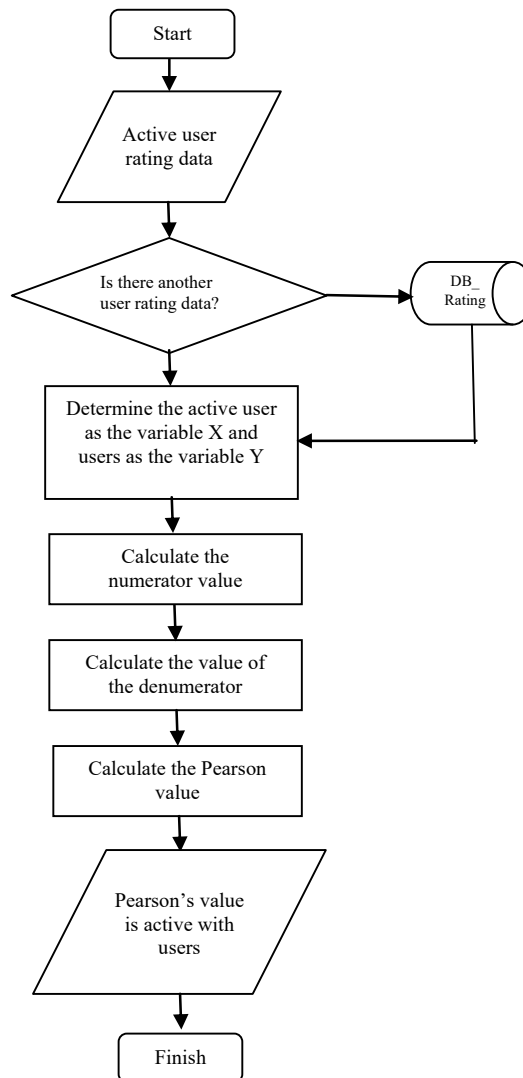
$$\begin{aligned}
&= \sqrt{55 - 45} \\
&= \sqrt{10} \\
&= 3.162277
\end{aligned}$$

5.3 Combining numerator and denominator

After obtaining each value from the numerator and denominator, the next is to do the division process as follows:

$$r = \frac{2.5}{0.79057(3.162277)} = \frac{2.5}{2.5} = 1.0$$

From the calculation of the Pearson correlation coefficient above, the Pearson value of 1.0 is obtained which indicates that the correlation between user 1 and user 2 is perfect. The process of calculating Pearson values can be described in the flowchart diagram as in Figure 3.

Figure 3 Pearson correlation coefficient

5.4 Analysis of the rating prediction process

Using the K-nearest neighbour (KNN) method in a recommendation system based on collaborative filtering, the amount of k is used in determining the number of users who have similar preferences with active users. The value of k depends on the specific dataset, and this value will affect the result of the rating prediction given by the system.

In the tests carried out, in providing recommendations to active users, namely user 1, using the KNN method with the amount of $k = 3$. The value of k determined by 3 means that the system will take three neighbours or three users with Pearson values. The highest or closest to value 1 is a perfect correlation. The value of the neighbour's Pearson from user 1 is as shown in Table 3.

Table 3 Pearson grades from each neighbour

<i>Users</i>	<i>Pearson</i>
User 2	1.0
User 3	0.6
User 4	0.4

The first step in determining rating predictions using the KNN method is to calculate the value of the user's influence on the active user (user 1). The value of influence is obtained by summing all the values of proximity k then divided by the value k divided by the number of values.

$$Influence = \frac{k}{\sum k, i}$$

Pearson values from the three closest users are known:

$$\begin{aligned} &= 1.0 + 0.6 + 0.4 \\ &= 2.0 \end{aligned}$$

After obtaining the total number of values from the nearest user Pearson value, then the effect of each user is calculated on the active user as in Table 4.

Table 4 Pearson value and influence

<i>Users</i>	<i>Pearson</i>	<i>Influence $(k/\sum k)$</i>
User 2	1.0	50%
User 3	0.6	30%
User 4	0.4	20%

Then show in the pie diagram as in Figure 4 according to the amount of influence between users on the active user. Percentage of influence/influence of user 2 on user 1 is described as a blue diagram while user 3 diagram is orange and user 4 is gray.

Figure 4 Pie diagram effect of users on active users (see online version for colours)



Next is to predict the rating that will be given by user 1 as an active user of the movie title that he has not given a rating, but has been given a rating by the neighbours based on the influence he has. For example, the system will predict the rating value that will be given by user 1 to the song title 2. The table of rating values given by the neighbours is as in Table 5:

Table 5 Value song title 2

<i>Users</i>	<i>Song title 2</i>	<i>Influence</i>
User 2	3.5	50%
User 3	5.0	30%
User 4	4.5	20%

The rating prediction value is:

$$\begin{aligned}
 &= (3.5 \times 0.5) + (5 \times 0.30) + (4.5 \times 0.20) \\
 &= 1.75 + 1.5 + 0.9 \\
 &= 4.15
 \end{aligned}$$

By obtaining a prediction value of 4.15, the system has succeeded in getting a rating prediction value for one song title that has never been rated by an active user.

Mathematical calculations used in this study to calculate the habits of consumers in listening to music based on consumer preferences, the results of these calculations can be used to provide further recommendations to these consumers or other consumers with the same preference.

6 Conclusions

The system has managed to get a rating prediction value for song titles that have never been rated by active users. The trial was conducted on 400 rating data provided by 25 users for 50 songs as datasets. By active users but has been given a rating by a number of their closest neighbours. Rating predictions for each song title are sorted from the largest to the smallest value, and are displayed as a music recommendation for active users. The results showed that user preferences in listening to songs on a music site can be used to provide music recommendations to other users who have the same preferences.

References

- Daud, N.M.N., Bakar, A.A.A. and Rusli, H.M. (2010) *Implementing Rapid Application Development (RAD) Methodology in Developing Practical Training Application System*, Institute of Electrical and Electronics.
- Gedimina, A. and Youngok, K. (2012) 'Improving aggregate recommendation diversity using ranking-based techniques', *IEEE Transactions on Knowledge & Data Engineering*, Vol. 24, No. 5, pp.896–911, IEEE.
- Kendall, J.E. and Kendall, K.E. (2010) *Analisis dan Perancangan Sistem*, Indeks, Jakarta.
- Melville, P. and Sindhawani, V. (2010) 'Recommender system', *Encyclopedia of Machine Learning*, Springer, USA.

- Pazzani, M.J. and Billsus, D. (2007) 'Content-based recommendation systems', in Brusilovsky, P., Kobsa, A. and Nejdl, W. (Eds.): *The Adaptive Web, Lecture Notes in Computer Science*, Vol. 4321, Chap. 10, pp.325–341, Springer-Verlag, Berlin, Germany.
- Pereira, N. and Varma, S.K. (2016) *Survey on Content Based Recommendation System*, Department of Information Technology, University of Mumbai PIIT.
- Robin, B. (2007) 'Hybrid web recommender systems', *The Adaptive Web, LNCS*, pp.377–408, Springer-Verlag, Berlin Heidelberg.
- Scafer, J.B., Konsta, J.A. and Riedl, J. (2016) Item-base collaborative filtering recommender algorithm', *Proceeding ACM*, 1-58113-348-0/01/0005, pp.285–295.
- She, C., Wen, W., Lin, Z. and Zheng, K. (2016) 'Detection of application-layer DDoS by clustering algorithm', *Proceedings of the 2016 2nd International Conference on Artificial Intelligence and Industrial Engineering*.
- Thorat, P.B., Goudar, R.M. and Barve, S. (2015) 'Survey on collaborative filtering, content-based filtering and hybrid recommendation system', *International Journal of Computer Applications*, Vol. 110, No. 4, pp.0975–8887.
- Tuzhilin, A. and Adomavicius, G. (2005) 'Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions', *IEEE Transactions on Knowledge and Data Engineering*, Vol. 17, No. 6.