


Content and Popularity-Based Music Recommendation System

Mamata Garanayak, Centurion University of Technology and Management, Bhubaneswar, India*

Suvendu Kumar Nayak, Centurion University of Technology and Management, Bhubaneswar, India

Sangeetha K., KG Reddy College of Engineering and Technology, India

Tanupriya Choudhury, University of Petroleum and Energy Studies, Dehradun, India

 <https://orcid.org/0000-0002-9826-2759>

Shitharth S., Vardhaman College of Engineering, India

ABSTRACT

The future of many modern technologies includes machine learning and deep learning methodologies. One of the prominent applications of these technologies is the recommender system. Due to the rapid growth of the songs in digital formats, the searching and managing of songs has become a great problem. In this study, the authors developed a recommender system using popularity and rhythm content of the song. The studies compared various techniques to improve the robustness and minimal error of the system. The authors will mostly focus on content-based, popularity-based, and collaborative-based filtering algorithms and also try to combine them using a hybrid approach. The authors utilized MAE for comparing the several procedures implemented here for the recommendation. Out of all procedures used, SVD performed well with MAE of 1.60 while KNN didn't perform that well as the authors had fewer features of song with mean absolute error of 2.212. User-relied and item-relied prototypes performed the best with MAE of 0.931 and 0.629.

KEYWORDS

Collaborative Filtering, Evaluation Matrices, Million Song Database, SVD

INTRODUCTION

Every day peoples have to deal with many decisions like which type of cloth to buy to wear, which things to collect and what type of song to play to listen? Therefore we exponentially depend on recommender process to build options. As enormous amounts of information are present in internet sources, one person has billions of choices to pick out from. This is an extensive provocation to give suggestion to persons from the huge information accessible from the internet. Amazon, eBay furnish recommendations relied on personalization to customers relied on their flavour and the past. In addition, different houses such as Spotify (Ciocca, 2017), Pandora makes use of ML procedures to come up with relevant recommendations.

DOI: 10.4018/ijismd.315027

*Corresponding Author

This article published as an Open Access article distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/4.0/>) which permits unrestricted use, distribution, and production in any medium, provided the author of the original work and original publication source are properly credited.

Most of the people in the world considered the listening of music as a very likable aspect of their living style and they engaged in listen to song frequently as an activity. Listing the song is more often as compared to any other activities such as reading story books, watching cinema and watching TV. From customer to customer the likability of the songs are also different, thus the several approaches that are designed previously cannot able to reach the customer's requirements. The emotion relied prototype was built to solve this problem which is relied on mood of the customer as well as the context relied prototype was built which is relied on contextual data such as playlist, review of the songs, and the comments given by the customers which does not fully fulfill the customer's requirements. The development of hybrid recommendation system for songs are also it its beginning stage. Now a day, the most difficult part is to manage and organize the trillions of song's title produced by the musicians or the producers. Genre classification, identification of artist and recognition of the instruments problems can be solved by using the MIR technique. The MIREX, one of the annual evaluation events is conducted to provide the developments of MIR procedures. Based on the hearing behavior and previous ratings given by people, it has been found that the CF procedures perform well.

The feature of song is universal as well as subjective. The songs not only deliver emotion but also it can change the people's mood. The choices of songs are different from customer to customer, thus the procedure described above cannot meet the customer's requirements every time.

Here, the authors will focus on implementing a recommendation process relied on personalization by utilizing user's past. The authors have attempted out different procedures to create effectual recommendation procedure. Here first the authors' implemented prototype relied on popularity which is extremely easy and not personalized but followed up by CF and content relied filtering which give recommendations based on personalization relied on past which are most common ones. The authors will also implement a combined procedure in which the authors integrate both CF and content procedures to obtain correct accuracy and for getting the better of disadvantages of two categories.

LITERATURE SURVEY

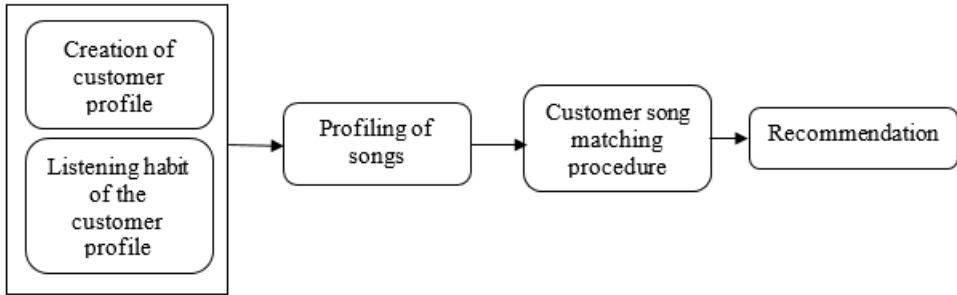
Constituents of Song Recommendation Prototype

The song recommendation or suggestion prototype normally consists of 3 parts – customers modeling, songs profiling and customer – song matching procedure or query type. Customer modeling deals with the creation of several customers' profile and creation of listening behavior of customer. The main aim of this step is by using the basic data and behavior of the customer, it differentiate the song tastes. The customer profiling can be divided into 3 categories; one is based on gender, age, married or unmarried (i.e. demographic feature), another is based on living city, country or location (i.e. geographic feature) and the last one is based on mood, lifestyle, likeliness etc. (i.e. psychographic feature). Depending on the level of song expertise, their song's expectation, the creation of listening behavior of customer can be achieved (Ciocca, 2017).

The song profiling describes the several types of metadata, which can be utilized in several recommendation procedures. It describes the several data used in MIR. The songs metadata can be categorized into 3 types: metadata acquired by a single expert or several experts (editorial data), metadata acquired from the internet (Cultural data) and the metadata acquired by the audio signal analysis (i.e. acoustic data). Editorial data are utilized in retrieval of information and the cultural data are utilized in context relied retrieval of information (Million song dataset triplet file, n.d.). Moreover, the acoustic data are utilized for discovering the context relied data retrieval.

The customer – song procedure deals with the several queries used in recommendation procedure and matching procedure. Assuming that the customer has knowledge on the data about the song, the fastest way to find the song is via editorial data based on key such as song's title, singer's name and the lyrics. The Figure 1 shows the components of the song recommendation prototype.

Figure 1. Constituents of song recommendation prototype



The song recommendation procedure still needs lots of customers' effort. In recommendation procedure, the more relevant way is to utilize past listening behavior or seed song as the query to find the customer's song preferences.

Model Based on Popularity

Popularity relied prototype is the very easy and intuitive prototype. In this prototype the authors recommend the customer the N top most music – the most likable music. Demands of music are calculated based on count of listen. This prototype has different obstacles as it is very naive one. The min demerit of this is it doesn't give personalization that is everyone is suggested with similar most likable music as well as several unlikable music are never recommended in this prototype (Ciocca, 2017).

Content Relied Filtering

Content relied techniques aims on the account of the customers in order to obtain recommendation to the customer. Accounts of the customers have all the data regarding Customer's likeness and the content relied technique utilizes this feature for suggestion of a particular item (Ciocca, 2017). Customer's past plays a vital role in the designed prototype. Here the authors attempt to get music same as to others in which customer has rated confidently in their past. All music in customer's past can be presented with a feature vector. This feature vector helps us in finding similar songs. Using cosine similarity the similarity between any two music can be found, which a very common procedure is. In cosine similarity as the name suggests the authors attempt to get angle among 2 characteristics vectors presenting the 2 music that is obtained by the dot product of 2 vectors divided by the norm of the 2 vectors. If the angle is lesser, closer the characteristics vectors are in that dimension and thus more same as the 2 music are (Asanov, 2011).

Table 1 shows an example which is list of movies seen by a user and ratings given by the user to each movie. Table 2 shows the feature vector of each movie; here the features are the genre of the movies. Based on these feature vectors we can predict which movies the user likes to watch.

Here every movie is represented by feature vector of 3 dimensions. The similarity among any of the two music represented by feature vectors \bar{w}_c and \bar{w}_s is given in equation 1 as:

Table 1. Movie list

Movie Name	Green lantern	Source code	American pie	Hangover
Ratings	7	8	7	9

Table 2. Feature vector

Movie Name	Comedy	Violence	Horror
American Pie	9	4	2
Scary Movie	7	7	5
Saw	4	10	9

$$u(c, s) = \cos(\bar{w}_c, \bar{w}_s) = \frac{\bar{w}_c \cdot \bar{w}_s}{\|\bar{w}_c\| \times \|\bar{w}_s\|} \quad (1)$$

Collaborative Relied Filtering

Filtering relied on collaborative techniques is most researched and frequently utilized procedure in recommendation techniques (Renick & Varian, 1997). This procedure is totally relied on the past characteristics but not relied upon the context. This assemble it one of the most often utilized procedure as this is not dependent on any supplementary data, which doesn't need any information or characteristics for the music for suggestion and therefore it permits us to recommend music without knowing actually, what is the type or category of the songs? Who sings the song and who compose the song, etc.? This utilizes only the ratings given by the customer for calculating recommendation for customer. Unlike the popularity based algorithm which focuses on popularity of a song collaborative filtering takes into account user's taste. Item relied collaborative method and user relied collaborative method is the two basic types of recommendation. Here, the authors try to find similar customers and items based on the rating provided by the customer.

User Relied CF

An important intuition behind this method is same customer listen to same music. In this procedure it is assumed that the customer having same past or having same rating figured have similar flavour and therefore it can suggest music from past of customers having same flavour to the present customer to whom it have to suggest the songs (Sun & Luo, 2010). Consider Table 3 which consists of ratings provided by 3 customers. It can be seen that all the customers have similar taste as they have rated the movies similarly. User A hasn't rated the movie Troy that means probably he hasn't watched that movie yet but the other two users have rated that movie positively. Now as all these users have similar taste and given that user B and C have rated the movie Troy positively we can say that user A will also like that movie and will rate it positively and so recommending him that movie would be a good choice.

Similarity among the customers can be measured by utilizing several procedures such as the similarity calculated by cosine formula or by calculating the correlation using the Pearson rule. The authors will be using Pearson correlation for computing similarity among the customers. Pearson

Table 3. Ratings provided by users

Movies/Users	Titanic	Gladiator	Black Swan	The Fighter	TRON Legacy
A	8	7	9	10	-
B	9	7	9	9	10
C	9	8	9	8	9

correlation is similar to cosine similarity with a little variation. The authors use mean normalization of ratings in R before applying cosine similarity. For example: One user can give a 3 for a song he likes while another user can give a 3 to the same song which he finds of medium quality. To overcome this problem the authors will subtract the mean rating of that user from every rating provided by the user before taking cosine similarity (Garg & Fangyan, 2014; Sarwar et al., 2001). Therefore, similarity among the customer using Pearson correlation is represented by equation 2:

$$s(u, v) = \frac{\sum_{i \in I_{u \cap I_v}} (r_{u,i} - \bar{r}_u)(r_{v,i} - \bar{r}_v)}{\sqrt{\sum_{i \in I_{u \cap I_v}} (r_{u,i} - \bar{r}_u)^2} \sqrt{\sum_{i \in I_{u \cap I_v}} (r_{v,i} - \bar{r}_v)^2}} \quad (2)$$

The below equation 3 is utilized to obtain the rating. Here u is the user whom we are recommending songs and u' is all other users in U:

$$P_{u,i} = \bar{r}_u + \frac{\sum_{u' \in u} s(u, u')(\bar{r}_{u',i} - \bar{r}_{u'})}{\sum_{u' \in u} |s(u, u')|} \quad (3)$$

Item Relied CF

The main aim in back of this procedure is items that are rated closely by the likewise customers. In item-relied prototype, it is take for granted that music which are listened together by particular customer tend to be similar and are more likely to be listened together in later also by some several customer. It is same as the customer relied other than in this case The authors perceive same songs by utilizing the rating matrix's column values for finding cosine similarity (Ekstrand et al., 2011; Herlocker et al., 2004; Niu et al., n.d.). User relied methods fails or doesn't achieve that much better when no.s of customers increase as if numbers of customers are much more than no. of music. To overcome this problem the authors use item relied collaborative method (Garanayak et al., 2019). The ratings in item relied method is performed by using the following equation 4 as:

$$P_{u,i} = \frac{\sum_{j \in S} s(i, j) r_{u,j}}{\sum_{j \in S} |s(i, j)|} \quad (4)$$

Table 4 shows the user relied and item relied procedures.

Cold Start Problem

Though CF procedures are very efficient but they have issues. Here a new customer the authors don't have any or very less amount of data regarding his/her past and thus it would not get better recommendation or would not be capable of getting recommendation (Garanayak et al., 2020; Ricci et al., 2011; Zhou et al., 2015). This difficulty is termed as cold start issue. But as long as the customer listens some of the music, past will be created and then the collaborative filtering procedure can be used for that user (Vozalis & Margaritis, 2005).

k-Nearest Neighbor Model

In this model, the authors make use of the metadata and form the feature vector for each song. These features include the artist, album, genre of the songs. These songs are represented in some N

Table 4. User and item relied procedure

User relied collaborative procedure(Estimates the similarity of nearest user)						Item relied collaborative procedure(Estimates the similarity of nearest item)					
U ₁	I ₁	I ₂	I ₃	I ₄	I ₅	U ₁	I ₁	I ₂	I ₃	I ₄ (Active item)	I ₅
U ₂ (Active User)	4	-	3	4	3	U ₂	4	-	3	4	3
U ₃	5	4	-	3	4	U ₃	5	4	-	3	4
U ₄	3	5	2	4	-	U ₄	3	5	2	-	-
U ₅	4	5	-	3	-	U ₅	4	5	-	3	-
U ₆	2	-	5	2	4	U ₆	2	-	5	2	4

dimension vector space and then the authors try to find k most songs which are closest to the songs from the history list of the given user in this N dimension vector space. K is a hyper parameter and so you need to find which value suits for your data the best. Different distance metric like Euclidean distance, Hamming distance, Manhattan distance and so on out of which Euclidean distance is most commonly used (Doke & Joshi, 2020; Garg & Fangyan, 2014; Kumar & Goyal, n.d.). Instead of using the feature vectors the authors can also use the rating matrix for KNN that is, it can be used in user relied and item relied collaborative techniques. Using KNN reduces the time complexity of these algorithms as the authors use only K similar users or items for calculating the ratings of unrated songs. So formula for calculating prediction is given by equation 5:

$$P_{u,i} = \bar{r}_u + \frac{\sum_{u' \in u} s(u, u') (\bar{r}_{u',i} - \bar{r}_{u'})}{\sum_{u' \in u} |s(u, u')|} \quad (5)$$

In this case N is the set of K likely customers. Same changes can be utilized for Item relied CF.

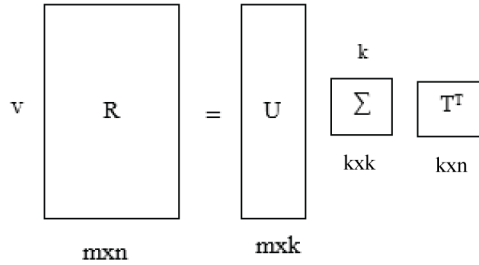
SVD (Singular Value Decomposition)

The authors work with feature vectors of songs or rows of rating matrix for recommending songs. Singular Value Decomposition is one of the algorithms used for reducing dimension and are extensively used for recommendation systems. Singular Value Decomposition is a linear algebra method which not only reduces the dimension but in addition does an extra interesting thing. It also obtains latent characteristics of the music. Some of the characteristics such as genre are problematic to obtain which affect the listening past of customers. (Ekstrand et al., 2011; Ricci et al., 2011; Vozalis & Margaritis, 2005; Zhou et al., 2015). The mathematical representation is given by equation 6:

$$R = U \Sigma T^T \quad (6)$$

The R, matrix of the sparse rating can be categorized into 3 types of matrices such as U, T & Σ . Here, the R is m x n matrix consisting m users and n songs. U is m x m matrix consisting of users and T is n x n matrix consisting of songs. U and T are orthogonal Matrices. Σ is a M x n diagonal matrix consisting of squares of unique values with only r nonzero entries such that $s_1 > 0$ and $s_1^2 \geq s_2^2 \geq \dots \geq s_r^2 > 0$ where r is the rank of R matrix as shown in Figure 2 (Ren & Gong, 2009; Vozalis & Margaritis, 2006; Vozalis & Margaritis, 2007).

Figure 2. SVD



The matrix R which is reconstructed is the closest approximation of the original R matrix. In this case k is no. of latent c and the rows of user matrix U_{mxk} matrix represents users interest in those k features while each row of song matrix $T_{n \times k}$ matrix represent the relevance of each features on that particular song (Guan et al., 2017). Hence recommendation of any music can be measured by assuming the normalization mean (Brand, 2003). Recommendation using SVD is represented as in equation 7:

$$P_{i,j} = \bar{r}_i + \left(U_k \sqrt{S_k^T(i)} \right) \cdot \left(\sqrt{S_k} \cdot V_k^T(j) \right) \quad (7)$$

where, $P_{i,j}$ is for the forecasting of i^{th} user and j^{th} product \bar{r}_i is the average of the rows. Since k is a constant but computing SVD is time expensive that is decomposing the rating matrix into the other 3 matrices takes too long and so can be done offline.

Chen and Chen (2005) designed recommendation system of music, which gives a personalized service of recommendation of music. They first analyzed the polyphonic music objects of MIDI format for getting the data for grouping the music by taking the six features of music. The customer access histories are analyzed to get the profiles of customer interests and behaviors for customer grouping. Content-relied, collaborative, and statistics-relied recommendation are implemented based on the favorite degrees of the customers to the music groups, and the customer groups they belong to.

Chang et al. (2018) proposed a personalized music recommender system (PMRS) relied on CNN technique which differentiate music based on beats of the music into several categories. They present a CF procedure to merge the result of CNN with the log files which contains the history of all customers to forecast music to the customers. They use MSD to evaluate the model and used the confidence score metrics for different music categories to check the performance of the PMRS.

Gunawan and Suhartono (2019) proposed a music recommender system that can give recommendations based on similarity of features on audio signal. They uses convolutional recurrent neural network (CRNN) for extracting the features and similarity distance to find the similarity between the features. The results of this study indicate that customers prefer recommendations that consider music genres compared to recommendations based solely on similarity.

PROPOSED MODEL

Data Set and Data Preprocessing

The dataset of million songs given by Kaggle (Ciocca, 2017; Million song dataset metadata file, n.d.) is used here (<https://www.kaggle.com/insiyeh/musicfeatures>). This dataset was given for million song challenges for forecasting the history of 12,039 customers by giving training to the other half and full listening past of another million users. Dataset is provided by Columbia university laboratory for recognition of speech and audio. The information set consist of 2 files; a file of triplet and a file

of metadata. The trio file is group of ids of the customer, ids of the songs and number of listen value while the data file consist of information with reference to the songs such as the id of song, artist, year and album which acts as characteristics of songs or the quality vector. The dataset doesn't have any information about the users like age or other demography and timestamp of listening event. The actual dataset is very large roughly 48 million records which is enormous for CPU processing and Memory expensive. Hence, the authors will be using 1 billion data from triad for the recommendation process that has over 41,045 customers with 9100 music.

After that the authors did preprocessing on the information. First we combined the triplet file and metadata on song id. Moreover, the triplet has only listen count but ratings are easier to work with as ratings are in fixed range. The ratings were generated in range 1 to 5 depending on the listen count. It was achieved by taking the highest value of listen of every customer 5 as the highest rating, for which the customer and the ratings of other music are estimated according to that.

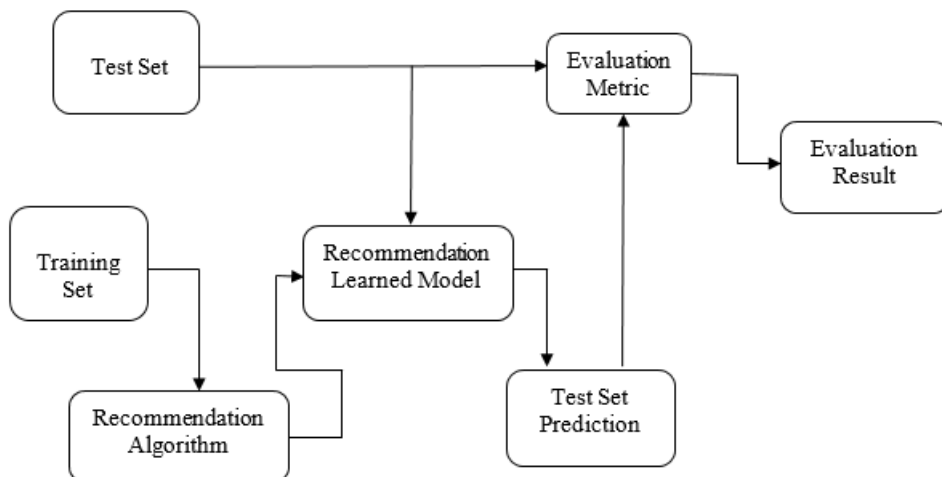
Let's look at all the notations the authors will be using throughout this paper. The authors will be focusing on triplets of users, songs and ratings. Ratings are numbers in range 1 – 5. The authors will be forming a rating matrix utilizing the triads where rows will speak for the customers while columns will speak for the item. That's why row will consist of ratings provided by customer to all possible items in their past and ratings of other items will be not known and for more clarity the authors will be filling them with zero values. Therefore, the authors have a totality comprising of a set of U of customers and set I of songs. I_u is the set of songs rated by customer u, and U_i is the set of customers who has rated i^{th} song. R is the matrix which represents rating, the rating user u provided for item I is denoted by $r_{u,i}$, the vectors of all ratings provided by user u is denoted by r_u , and the vector of all ratings given for item i is represented by r_i , the average of a customer u or an item i's ratings are represented by \bar{r}_u and \bar{r}_i , respectively. The rating matrix will be very sparse. The authors try to predict the ratings for users which were not available in R. The recommender's prediction of $r_{u,i}$ (which will be zero) is denoted by $p_{u,i}$.

Recommendation Model Workflow

The authors will be using a hybrid procedure which consists of both the collaborative and content procedure even popularity relied prototype. The authors will try to mitigate the drawbacks of collaborative filtering model by using content based and popularity-based model.

The workflow of our model is given in figure 3.

Figure 3. Recommendation model



At the beginning the authors divide the information into training set and test set. The authors can also keep separate validation for choosing hyper parameters and checking their generalization in test set. After the division, the train information is applied to learning procedure such as user relied algorithm or item relied collaborative procedure or SVD that learns to do predictions. To evaluate the prototype the authors utilize evaluation metrics where test information is given to the learned the procedure that in return creates prediction. With assistance of real rating of customer from test set and evaluation metric the authors can check how efficient our prototype is.

The cosine similarity formula is represented by equation 8:

$$u(c, s) = \cos(\bar{w}_c, \bar{w}_s) = \frac{\bar{w}_c \cdot \bar{w}_s}{\|\bar{w}_c\| \times \|\bar{w}_s\|^{1-\alpha}} \quad (8)$$

When the authors are creating suggestion or recommendation for the customer u, the rule of u remains same while calculating the similarity with all other customers. Therefore, to emphasize the affect of another customer the authors utilize this form of cosine similarity. Therefore here the authors attempt to minimize the alpha count. Range of alpha is (0,1).

Now let's look at the proposed algorithm in which the authors will be combining 2 or more prototypes by utilizing aggregating procedure. The authors will be aggregating customer relied and item relied CF procedure by merging the recommendation of all the prototypes. If the authors consider z% recommendation from one prototype then the authors will take 1-z% recommendation from other prototype.

Proposed Algorithm

```

Acquire id of the customer for recommendation if customer in database:
  if query is given:
    add query to the history list if no of music of customers > 10:
      apply user relied CF procedure to acquire z% recommendation
      and after that item relied CF to acquire another 1-z% recommendation.
    else:
      Use item relied CF procedure to acquire 80% of recommendation
      apply CB procedure to suggest or recommend another 20% recommendation.
  else:
    assign customer in database if query is given:
      use content based and popularity-based algorithm to get
      recommendation else:
        utilize popularity relied prototype to acquire recommendation
  
```

Metric for Evaluating Recommendation

Evaluation metric is a tool that tells us how accurate is the recommendation system. The mean absolute error is represented by equation 9:

$$|\bar{E}| = \frac{\sum_{i=1}^N |p_i - r_i|}{N} \quad (9)$$

The authors will be utilizing precision and recall along with MAE (Garanayak et al., 2019; Garanayak et al., 2020; Herlocker et al., 2004). But the authors have rated the items in range 1 to 5, so the authors required get the way to transform the numerical issue into binary issue. This is achieved by forming 2 categories - applicable and not applicable by dividing the rating into 2 groups. For ratings given equal to 3 and more, the authors will take that it is applicable otherwise it is not applicable.

The authors will be using precision mean average for evaluating the prototype. In this metric, at the beginning the authors will acquire top K recommendations from the procedure. After that at each and every rank k, it will calculate precision at that k if the music is relevant. Assume y be the list of predictions specified by the prototype such that $y(j)=i$ define i^{th} music is at the j rank in the list of prediction. The equation 10 will be as follows:

$$P_k = P_{k(u,y)} = \frac{\# \text{ of relevant songs is } Y_k}{\# \text{ of songs is } Y_k} \quad (10)$$

For each rank k the authors' average all values of precision using formula presented in equation 11:

$$AP(u, y) = \sum_{i=1}^k P_i \cdot (u, y) \quad (11)$$

At last the authors will average over all the customers in equation 12:

$$mAP = \frac{\sum_{u=1}^m AP(u, y)}{m} \quad (12)$$

RESULTS

The authors used MAE for comparing the different procedures the authors executed for the recommender model. Above is the pictorial representation or outcomes the authors acquired for our prototype. Singular vector decomposition performed well with mean absolute error of 1.60 whereas KNN didn't perform that well as the authors had very less features of song with mean absolute error of 2.212. User relied and Item relied prototype performed the best with mean absolute error of 0.931 and 0.629. Item relied prototype outperformed user relied as user relied prototype suffered scalability issue. Aggregate method for Item based and user based model. The authors have first taken the music that subsists in both the customers relied and item relied recommender lists. The overall result comparison is shown in Figure 4.

Then utilize z% recommendations from Item relied list, and (1-z) % from User- relied list for the rest of the final recommendation list which is shown in Table 5.

The authors chose different values of z and saw which performed best for us. Table 5 shows mAP (Mean Average Precision) for different values of z and the graph shown in Figure 5 helps us to visualize which value works best for us.

CONCLUSION AND FUTURE WORK

The authors have built a good song recommendation by studying and implementing various different algorithms of machine learning. The authors also built our own algorithm by combining two collaborative filtering algorithms in order to maximize our precision and lowering the error. We

Figure 4. Result analysis

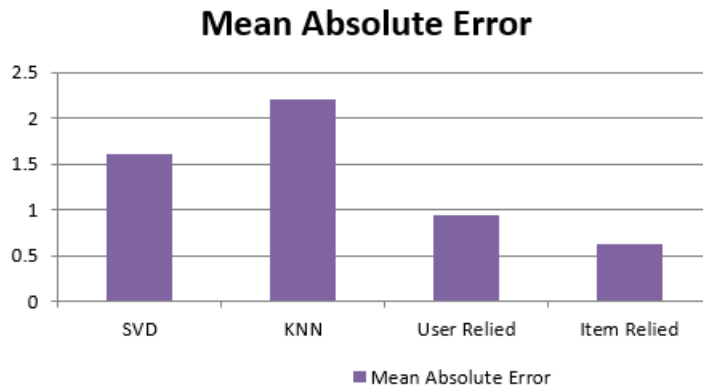
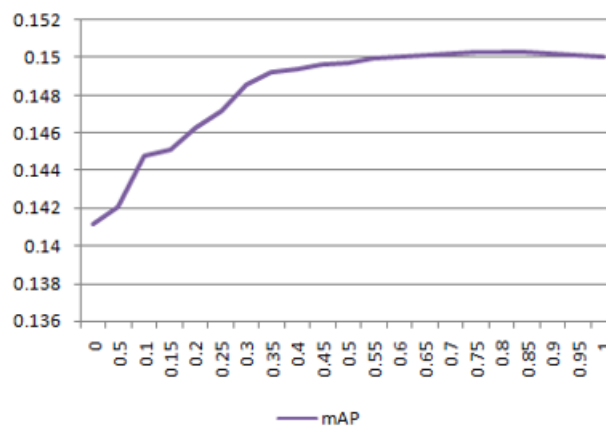


Table 5. Final recommendation result

Z%	mAP (Mean Average Precision)	Z%	mAP (Mean Average Precision)
0	0.141200	0.55	0.149930
0.5	0.142130	0.6	0.150010
0.1	0.144789	0.65	0.150110
0.15	0.145102	0.7	0.150200
0.2	0.146238	0.75	0.150250
0.25	0.147146	0.8	0.150280
0.3	0.148552	0.85	0.150250
0.35	0.149210	0.9	0.150210
0.4	0.149410	0.95	0.150100
0.45	0.149640	1	0.150070
0.5	0.149710		

Figure 5. mAP result



implemented Popularity relied prototype, SVD, K nearest Neighbor Algorithm (Relied on Content), user relied and item relied CF procedures. As the popularity centric prototype had no personalization it performed the worst. K-NN too didn't perform well as we didn't have many features of songs and there was no variation in recommendation. User relied and item relied model performed best for our model. With item relied prototype performing well as in customer relied prototype, we got the scalability difficulty where number of customers were more than the number of music. Singular value decomposition too did best though the matrix was too sparse to converge the objective function to global minimum.

Recommender systems are active field of research and the authors can still further improve our system by working more on them trying out several things and checking which things works best for you. To merge the item and user relied prototype by utilizing the linear merging and learning the each model's weights, build feature to automatically find features or metadata of songs, to evolve procedure for several aspects such as customer's mindset, time, day, etc., and to utilize deep learning for audio files process of each song in order to acquire characteristics of music for suggestion or recommendation may be included as the future work.

CONFLICT OF INTEREST

The authors of this publication declare that there is no conflict of interest.

FUNDING AGENCY

This research received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors.

REFERENCES

- Asanov, D. (2011). *Algorithms and methods in recommender systems*. Berlin Institute of Technology.
- Brand, M. (2003, May). Fast online SVD revisions for lightweight recommender systems. In *Proceedings of the 2003 SIAM international conference on data mining* (pp. 37-46). Society for Industrial and Applied Mathematics. doi:10.1137/1.9781611972733.4
- Chang, S. H., Abdul, A., Chen, J., & Liao, H. Y. (2018, April). A personalized music recommendation system using convolutional neural networks approach. In *2018 IEEE International Conference on Applied System Invention (ICASI)* (pp. 47-49). IEEE. doi:10.1109/ICASI.2018.8394293
- Chen, H. C., & Chen, A. L. (2005). A music recommendation system based on music and user grouping. *Journal of Intelligent Information Systems*, 24(2), 113–132. doi:10.1007/s10844-005-0319-3
- Ciocca, S. (2017). How does Spotify know you so well? *Medium*, 10. <https://medium.com/s/story/spotify-s-discover-weekly-how-machine-learning-finds-your-new-music-19a41ab76efe>
- Doke, N., & Joshi, D. (2020). Song Recommendation System Using Hybrid Approach. In *Proceeding of International Conference on Computational Science and Applications* (pp. 319-327). Springer. doi:10.1007/978-981-15-0790-8_31
- Ekstrand, M. D., Riedl, J. T., & Konstan, J. A. (2011). *Collaborative filtering recommender systems*. Now Publishers Inc. doi:10.1561/9781601984432
- Garanayak, M., Mohanty, S. N., Jagadev, A. K., & Sahoo, S. (2019). Recommender system using item based collaborative filtering (CF) and K-means. *International Journal of Knowledge-based and Intelligent Engineering Systems*, 23(2), 93–101. doi:10.3233/KES-190402
- Garanayak, M., Sahoo, S., Mohanty, S. N., & Jagadev, A. K. (2020). An automated recommender system for educational institute in India. *EAI Endorsed Transactions on Scalable Information Systems*, 7(26).
- Garg, S., & Fangyan, S. U. N. (2014). *Music Recommender System CS365: Artificial Intelligence*. Academic Press.
- Guan, X., Li, C. T., & Guan, Y. (2017). Matrix factorization with rating completion: An enhanced SVD model for collaborative filtering recommender systems. *IEEE Access: Practical Innovations, Open Solutions*, 5, 27668–27678. doi:10.1109/ACCESS.2017.2772226
- Gunawan, A. A., & Suhartono, D. (2019). Music recommender system based on genre using convolutional recurrent neural networks. *Procedia Computer Science*, 157, 99–109. doi:10.1016/j.procs.2019.08.146
- Herlocker, J. L., Konstan, J. A., Terveen, L. G., & Riedl, J. T. (2004). Evaluating collaborative filtering recommender systems. *ACM Transactions on Information Systems*, 22(1), 5–53. doi:10.1145/963770.963772
- Kumar, A., & Goyal, A. (n.d.). *CS365: Artificial Intelligence Project Proposal A multimodal approach for predicting the induced affect content in videos*. Academic Press.
- Million song dataset metadata file. (n.d.). https://static.turi.com/datasets/millionsong/song_data.csv
- Million song dataset triplet file. (n.d.). <https://static.turi.com/datasets/millionsong/10000.txt>
- Niu, F., Yin, M., & Zhang, C. T. (n.d.). *Million Song Dataset Challenge!* Academic Press.
- Ren, Y., & Gong, S. (2009, November). A collaborative filtering recommendation algorithm based on SVD smoothing. In *2009 Third International Symposium on Intelligent Information Technology Application* (Vol. 2, pp. 530-532). IEEE. doi:10.1109/IITA.2009.491
- Renick, P., & Varian, H. R. (1997). Recommender system. *Communications of the ACM*, 40(3), 56–58. doi:10.1145/245108.245121
- Ricci, F., Rokach, L., & Shapira, B. (2011). Introduction to recommender systems handbook. In *Recommender systems handbook* (pp. 1–35). Springer. doi:10.1007/978-0-387-85820-3_1
- Sarwar, B. M., Karypis, G., Konstan, J. A., & Riedl, J. (2001). et almbbox. In *Item-based collaborative filtering recommendation algorithms* (pp. 285-295). ACM. <https://doi.acm.org/10.1145/371920.372071>

Sun, Z., & Luo, N. (2010, August). A new user-based collaborative filtering algorithm combining data-distribution. In *2010 International Conference of Information Science and Management Engineering* (Vol. 2, pp. 19-23). IEEE. doi:10.1109/ISME.2010.48

Vozalis, M. G., & Margaritis, K. G. (2005, September). Applying SVD on item-based filtering. In *5th International Conference on Intelligent Systems Design and Applications (ISDA'05)* (pp. 464-469). IEEE. doi:10.1109/ISDA.2005.25

Vozalis, M. G., & Margaritis, K. G. (2006). Applying SVD on Generalized Item-based Filtering. *IJCSA*, 3(3), 27–51.

Vozalis, M. G., & Margaritis, K. G. (2007). Using SVD and demographic data for the enhancement of generalized collaborative filtering. *Information Sciences*, 177(15), 3017–3037. doi:10.1016/j.ins.2007.02.036

Zhou, X., He, J., Huang, G., & Zhang, Y. (2015). SVD-based incremental approaches for recommender systems. *Journal of Computer and System Sciences*, 81(4), 717–733. doi:10.1016/j.jcss.2014.11.016

Mamata Garanayak is working as an assistant professor in Centurion University of Technology and Management, Bhubaneswar, Odisha. She has an experience of 16 years in different institutes. She has completed her Ph.D. on recommendation system from KIIT University, Bhubaneswar, Odisha. Her interest research areas are machine learning, recommendation system.

Suvendu Kumar Nayak is working as an Assistant Professor in Centurion University of Technology and Management, Bhubaneswar, Odisha. he has an experience of 15 years in different institution. His research areas include cyber security.

K. Sangeetha received her B.E. degree in Computer Science from Ramakrishna College of Engineering, Coimbatore, India, in affiliation with Anna University, Chennai, India in 2012; and her M.E. degree in Computer Science & Engineering from SNS College of Engineering, Coimbatore, India, in affiliation with Anna University, Chennai, India in 2014. She has completed her Ph.D. degree in the Department of Computers Science & Engineering, Sri Satya Sai University of Technology of Medical Sciences. She is working as a Assistant Professor in KG Reddy college of engineering, Hyderabad. She has published six International Journals along with many International & National conferences. Her current research interests include Artificial Intelligence, Machine Learning, Cryptography.

Tanupriya Choudhury received his bachelor's degree in CSE from West Bengal University of Technology, Kolkata, India, and master's Degree in CSE from Dr. M.G.R University, Chennai, India. He has received his PhD degree in the year 2016. He has ten years of experience in teaching as well as in Research. Currently he is working as an Associate Professor in dept. of CSE at UPES Dehradun. Recently he has received Global Outreach Education Award for Excellence in best Young researcher Award in GOECA 2018 His areas of interests include Human computing, Soft computing, Cloud computing, Data Mining, etc. He has filed 14 patents till date and received 16 copyrights from MHRD for his own software. He has been associated with many conferences in India and abroad. He has authored more than 85 research papers till date. He has delivered invited talk and guest lecture in Jamia Millia Islamia University, Maharaja Agersen college of Delhi University, Duy Tan University Vietnam etc. He has been associated with many conferences throughout India as TPC member and session chair etc. He is a lifetime member of IETA, member of IEEE, and member of IET(UK) and other renowned technical societies. He is associated with Corporate and he is Technical Adviser of DeetyaSoft Pvt. Ltd. Noida, IVRGURU, and Mydigital360, etc. He is holding the post of Secretary in IETA (Indian Engineering Teacher's Association-India), He is also holding the Advisor Position in INDO-UK Confederation of Science, Technology and Research Ltd., London, UK and International Association of Professional and Fellow Engineers-Delaware-USA.

S. Shitharth received his B.Tech. degree in Information Technology from Kgisl Institute of Technology, Coimbatore, India, in affiliation with Anna University, Chennai, India in 2012; and his M.E. degree in Computer Science & Engineering from Thiagaraja College of Engineering, Madurai, India, in affiliation with Anna University, Chennai, India in 2014. He completed his Ph.D. degree in the Department of Computers Science & Engineering, Anna University. He is currently working as Assistant Professor in Vardhaman College of Engineering, Hyderabad. He has published more than 10 International Journals along with 12 International & National conferences. He has even published 3 patents in IPR. He is also an active member in IEEE Computer society and in 5 more professional bodies. His current research interests include Cyber Security, Critical Infrastructure & Systems, Network Security & Ethical Hacking. He is an active researcher, reviewer, and editor for many international journals.