

Music Recommendation System based on Unsupervised Discretization

M. Sunitha
Vasavi College of Engineering
Department of CSE
Ibrahimbagh, Hyderabad-31,
India

T. Adilakshmi, PhD
Vasavi College of Engineering
Department of CSE
Ibrahimbagh, Hyderabad-31,
India

ABSTRACT

Because of the revolution in the field of Internet and E-commerce, users are overwhelmed by choices either it may be a book or movie or Music etc. Recommendations systems are serving as one of the important tool to handle information overloading by providing recommendations to users. In this paper we proposed a method to handle music recommendation problem. Unsupervised discretization is used to cluster the items which are similar in their frequency distribution. The proposed method is evaluated by using a benchmark dataset Last.fm. the results depict the fact that the proposed method performs better than the traditional popular recommendation approach.

Keywords

Internet, E-commerce, information overloading, Recommendations systems, Unsupervised discretization

1. INTRODUCTION

With the evolution of Internet, we are provided with millions of options for a song on Internet Radio, watch a movie, buy a book or see different accessories. We generally form an opinion about these things say some of them we like and some we do not like. But if we observe the likes and dislikes of a person, implicitly it follows a pattern. The goal of a recommendation system is to find the underlying pattern which describes the user taste. Once we know the taste of a person, recommendation system can recommend similar items. For example, if someone likes bacon-lettuce- tomato sandwiches, then there are good chances that that person will also like a club sandwich because they are very similar only with turkey replacing the bacon.

Music is omnipresent. Millions of songs are present at a click away from everyone. With the number of songs, bands, and artists coming up, music listeners are overwhelmed by choices. Users will always try to find the music that will match their taste. This is the motivational fact for the field of music recommendations. In the recent years, there have been many services like Pandora, Spotify, and Last.fm [1] that have come up in order to find a perfect solution, but haven't been completely successful. The Choice for music is influenced by taste, trust, and liking towards any particular artist. It is very difficult to quantify all these factors for a machine or software. Hence, it has been a very difficult experience for these service providers to find the music really interesting and satisfying the taste of a person. Every music

recommendations system works on a given set of assumptions in order to provide effective recommendations.

There are two fundamental styles of music recommendations: Collaborative filtering and Content-based Filtering.

Collaborative filtering is an approach in which information is gathered about the users' preferences for any particular item (books, videos, news articles, songs, etc.) [2][7][8]. The knowledge captured is then structured and used against all the unknown items and make intelligent predictions that a user might enjoy. In collaborative filtering, the interaction between users and items is important. The system relies on the past history to derive a suitable model for an entity. The historical data acts as an input to the system[3].

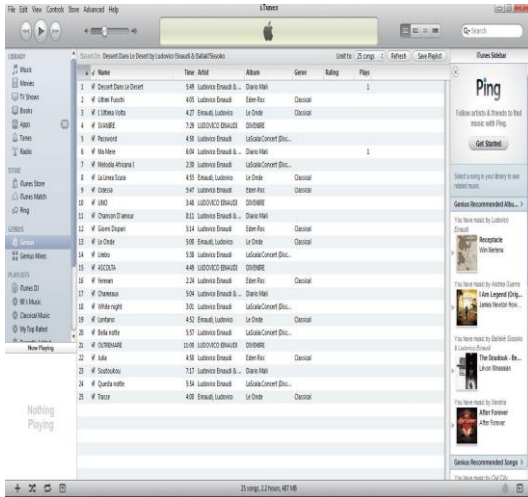
In Content-Based filtering, we analyze the attributes or the content of a song in order to make recommendations [7][8]. In the case of a song, we analyze the kind of instruments used, tempo, pitch of the song, and store all those information in a structured format. Now, when a user listens to a particular song, the system analyses that song and finds the neighboring similar songs to make active recommendations. This approach is a content dependent approach because the methodology that is used to analyze or recommend songs would not work for analyzing books or videos since those items has different sets of attributes. Therefore, they should be approached differently. Pandora is one of the music services that uses content based filtering for their music recommendations.

The rest of the paper is organized as section 2 describes about related work carried out, section 3 discuss about the proposed method, section 4 shows the results obtained for the proposed method and section 5 describe conclusion and future directions of research.

2. RELATED WORK

2.1 Automatic playlist generation

Automatic playlist generation works based on the seeds given by a user. It recommends songs similar to the chosen seeds to generate a new playlist. This approach will not consider the feedback of the user into consideration while generating playlist. One important issue about this approach is, it may tend to recommend same kind of songs as the seeds. Genius of iTunes [9] employs similar methods to generate a playlist from a seed.



2.1.1. Genius by iTunes

2.2 Dynamic music recommendation

Dynamic music recommendation improves automatic play list generation by considering the user's feedback. It starts with an arbitrary song and adjusts the playlist to accommodate the feedback given by the user. Pandora [4] follows this approach in recommending music.

2.3 Hybrid Approaches

Hybrid approach combine collaborative filtering and content based method. Use both content features and user access pattern to recommend music.

3. PROPOSED SYSTEM

This section describe the steps in the proposed approach i.e. Music recommendation system based on unsupervised discretization.

3.1 Unsupervised Discretization

Discretization is converting a continuous attribute to categorical attribute. This conversion involves two subtasks: deciding how many categories to have and determining how to map the values of the continuous attribute to these categories. Discretization can be supervised or unsupervised. A basic difference between these two methods is whether class information is used or not. If the class information is used then it is known as Supervised Discretization otherwise known as Unsupervised Discretization.

We have three types of approaches for Unsupervised Discretization [6] as shown in the fig.3.1.1.

Equal width approach divides the range of the attribute into a user-specified number of intervals each having the same width.

Equal frequency approach tries to put same number of objects into each interval.

K-Means clustering method to discretize the objects into clusters.

In the proposed approach for music recommendation system based on unsupervised discretization equal width binning has been used. Songs are discretized into equal width bins by considering the minimum and maximum frequency of songs as boundaries for bins.

Steps in the proposed approach are shown in the fig.3.2.1.

3.2 Flow Chart

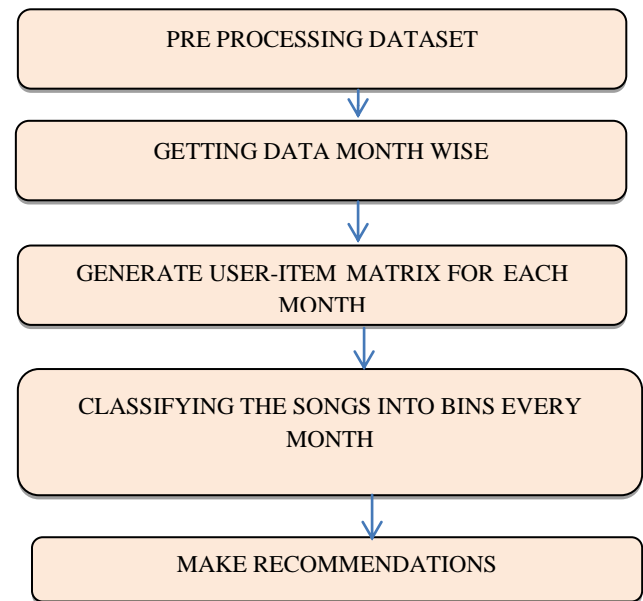


Fig.3.2.1. Flow of steps in the proposed approach

3.2.1. Pre-Processing

The dataset for this research work contains activities of users whose listening history has been captured anonymously for the period of 1 year from Last.fm. For every song that a user listens to, its activity is recorded in the following format:

userid (User_000004) – Since the data is captured anonymously, we assigned each user, a user-id of the format user_000004.

Date-Time (2009-04-09T12:49:50Z) – Time of activity is recorded which will be used in our algorithm to determine the session in which it will belong.

Album Id (078a9376-3c04-4280-b7d7-b20e158f345d) – A unique identifier is attributed to each Album.

Album name (A Perfect Circle) – An album to which that song belongs to

Song id (5ca13249-26da-47bd-bba7-80c2efeb9cd) – A unique identifier is attributed to each track / song.

Song name (People are People) – The songs which the user listened to.

3.2.2. Building Data

Once pre-processing of data is done, we get userid, song id and count from the database which is used to build user-item matrix as shown in the fig 3.2.2. month wise and store those in corresponding array lists respectively.

3.2.3. Generation of training set and test set

From given number of users in dataset we divide certain number of users into training set and remaining set as test set. This can be done as per our requirement.

3.2.4. Generation of User Song Matrix

Once we get the pairs of songs for each user i , we compose a user vector which consists of all the songs that are played in the user's history. We construct a user-item matrix for $nU(m)$ users \times $nS(m)$ songs; Where $nU(m)$ is number of users in particular month(m) and $nS(m)$ is no. of songs in that month(m).so that the value in each cell a_{ij} in the matrix is

directly proportional to the number of times a user i has listened to song j . We call this matrix as Matrix M , which is a sparse matrix.

U/S	S1	S2	S3	S4	S5	S6
U1	10	0	0	4	0	0
U2	5	2	0	0	0	0
U3	3	4	6	2	2	4
U4	2	6	0	0	4	7
Sum	20	12	6	6	6	11

Fig.3.2.2. Sample User-item Matrix Obtained for a Month

3.2.5 Pseudocode to form equal width bins of items or songs

1. Generated User-Item matrix for every month looks similar to the figure shown in fig.3.2.2.
2. Calculating the max of sums, which gives us 20 from the above example
3. Making Bins based on the max count and classifying the songs into corresponding bins
4. Example bins looks like this

BIN0:song7,song1

BIN1:song6,song9

BIN2:song3,song4,song5,song8

3.2.6. Recommendations

At recommendation stage test set users and songs objects are retrieved and test user song matrix is generated. From this each user vector is considered one by one and their songs are compared with generated bins. Depending on comparison result songs of corresponding bin are recommended.

3.3 Evaluation measures

For the analysis purpose, we withheld some of the data (users' listening history) of 15 users to perform the evaluation of song recommendations and check if it matched the benchmarks. Rather than caring about the degree to which a user would like to a recommended song, we sometimes are more interested if a user will add that song to his listening queue. This is an important aspect of music discovery. Therefore, we hold back partial data of 15 users and try to complete that list using the recommendations made by our algorithm. For every test user, depending upon the criteria, each song that is recommended can be categorized into one of four the groups. They are as shown below in Table 3.5.1.

Table.3.5.1. Confusion Matrix

	Not Recommended	Recommended
Used	False Negative (fn)	True Positive (tp)
Not	True Negative (tn)	False Positive (fp)

Here, we describe each of the term as:

True Negative (TN): A song which is uninteresting to the user is not recommended to them.

False Positive (FP): A song is recommended by the algorithm which a user is not interested in.

True Positive (TP): An algorithm recommends a song to the user which they are interested in.

False Negative (FN): The algorithm does not recommend a song to the user which they are interested in.

Based on obtained positives and negatives Precision, Recall and F-measure [6] are calculated in order to evaluate efficiency of system. We can define them as follows:

Precision: It is defined as the ratio of relevant items selected to number of items selected, shown in Eq. Precision represents the probability that a selected item is relevant.

$$Precision = \frac{TP}{TP + FP}$$

Recall: It is defined as the ratio of relevant items selected to total number of relevant items available. Recall represents the probability that a relevant item will be selected.

$$Recall = \frac{TP}{TP + FN}$$

F-measure: It is defined as a measure that combines both precision and recall in the harmonic mean of precision and recall. It is also called as balanced f-score or f-measure.

$$F - measure = 2 * \frac{(Precision * Recall)}{(Precision + Recall)}$$

4 RESULTS

The experiment is carried out on a benchmark dataset obtained from Last.fm. The results obtained are shown in the fig.4.1.

Table.4.1. Precision, Recall, f-Score for Sample Test Users with Discretization

Test users	Precision	Recall	F-score
4	0.046	0.696	0.083
5	0.037	0.579	0.067
6	0.032	0.516	0.057
7	0.074	0.555	0.116
8	0.067	0.556	0.104

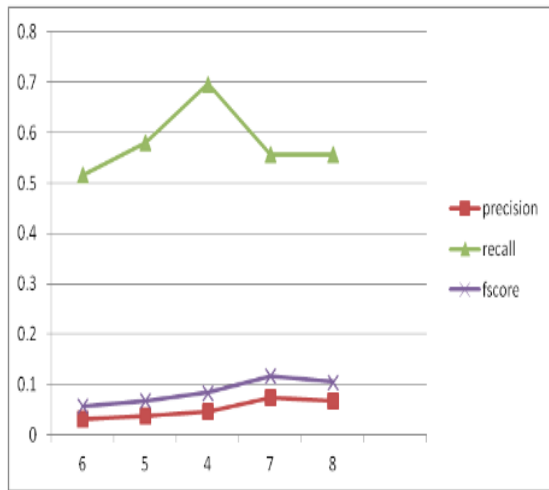


Fig.4.1. precision, recall, f-score for sample test users with discretization for the month January 2008

5 CONCLUSION AND FUTURE SCOPE

We proposed and implemented unsupervised discretization method on the benchmark dataset. The results obtained depicts that the proposed method performs better compared to the standard most popular baseline recommendations.

The proposed work can be extended to include other features of songs while performing discretization. Entropy can also be used to perform supervised discretization.

6 REFERENCES

- [1] Last. FM – A popular music web portal <http://www.last.fm>
- [2] Kaji Katsuhiko, Keiji Hirata, and Nagao Katashi. A music recommendation system based on annotations about listener's preferences and situations. *Axmedis*, 0:231-234, 2005.
- [3] G.Karypis, "Evaluation of item-based top-N recommendation algorithms," *CIKM 2001*, pp. 247–254, 2001.
- [4] Pandora – A free internet radio. <http://www.pandora.com>
- [5] C. Anderson. The long Tail. *Wired Magazine*, 12(10): 170-177, 2004.
- [6] Textbook: Pang-ning Tan, Vipin Kumar, Michael Steinbach, *Introduction to Data Mining*, Pearson
- [7] Adomavicius G, Tuzhilin A (2005) Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions. *IEEE Trans Knowl Data Eng* 17(6):734–749
- [8] Abbassi Z, Amer-Yahia S, Lakshmanan LVS, Vassilvitskii S, Yu C (2009) Getting recommender systems to think outside the box. In: *Proceedings of the third ACM conference on recommender systems RecSys 09*, ACM Press, pp 285–288
- [9] iTunes