A Novel Way for Personalized Music Recommendation Using Social Media Tags

Gunjan Advani[1] Neha Soni[2]
[1]Student [2]Assistant Professor
[1,2]Department of Computer Engineering
[1,2]Sardar Vallabhbhai Patel Institute of technology, Vasad, Gujarat, India

Abstract— Rapid growth of online music content has opened new opportunities for implementing new effective information access services – commonly known as music recommender systems – that helps users filter and discover songs according to their tastes. It supports music navigation, discovery, sharing, and formation of user communities. This survey illustrates various tools and techniques that can be used for addressing recommendation systems. Recommendation techniques can be classified in to four major categories: Collaborative Filtering, Content Based, Hybrid and Context based Recommendations. This survey elaborates these approaches and discusses their limitations and advantages. In the past decade, Social Tagging Systems have attracted increasing attention and tag-aware recommender systems (Collaborative tagging) have emerged. Besides the underlying structure of tagging systems, many efforts have been addressed to unify tagging information to reveal user behaviors and preferences, extract the latent semantic relations among items, make recommendations, and so on. Finally, this survey focuses on tag based recommendation approach.

Key words: Collaborative Filtering, Content Based, Social Tagging Systems

I. INTRODUCTION

Music has always played a major role in an individual's entertainment. With the popularity of digital music and Internet technologies, a huge amount of music content has become available to millions of users around the world. As there are millions of artists and songs in the market, it is becoming increasingly difficult for the users to search for music content—there is a lot of potentially interesting music that is difficult to discover. Furthermore, huge amounts of available music data have opened new opportunities for researchers working on music recommendation to create new viable services that support music navigation, discovery, sharing, and formation of user communities. The necessity for such services – commonly known as music recommender systems – is high due to the economic potential of online music content.

Also, music recommender is to help users filter and discover songs according to their tastes. Meanwhile, the development of recommender systems provides a great opportunity for industry to aggregate the users who are interested in music. More importantly, it raises challenges for us to better understand and model users' preferences in music [1].

Music recommender systems are decision support tools that reduce the information overload by retrieving only items that are reckoned as relevant for the user, based on the user's profile, i.e., a representation of the user's music preferences [2]. For example, Last.fm – a popular Internet radio and recommender system – allows a user to mark songs or artists as favorites. It also tracks the user's listening habits, and based on this information can identify and recommend music content that is more likely to be interesting to the user.

Presently, based on users' listening behaviour and historical ratings, collaborative filtering algorithm has been found to perform well [4]. Combined with the use of content-based model, the user can get a list of similar songs by low level acoustic features such as rhythm, pitch or high-level features like genre, instrument etc. [3].

However, most of the available music recommender systems recommend music without taking into consideration the user's context, e.g., mood, or current location and activity [5]. In fact, a study on users' musical information needs [6] showed that people often seek music for a certain occasion, event, or an emotional state. In response to these observations, in recent years a new research topic of contextual (or situational) music recommendation has emerged.

Collaborative tagging systems, also known as folksonomies are web-based systems that allow users to label the resources with arbitrary words, so-called tags. These systems are becoming more common among web users now-a-day. For example popular web services such as Flickr, del.icio.us, Last.fm, Gmail, etc, provide possibility for users to tag or label an item of interest. In general, tagging is associated to the Web 2.0 and is becoming the new trend enabling people to easily add metadata to content. Hence, these additional metadata can be used to improve search mechanisms or provide personalized recommendations fitting the users' interests. Content information used in attribute aware RS algorithms is typically attached to the items and is usually provided by domain experts. Therefore, an item always has the same attributes among all users. On the other hand, tags are provided by various users. Thus, tags are not only associated to the items but also to the users. Although attributes and tags are both metadata and could act as additional background knowledge to improve RS algorithms, they should be handled differently.

Thus, music recommendation systems can be classified into five categories such as illustrated in the figure.

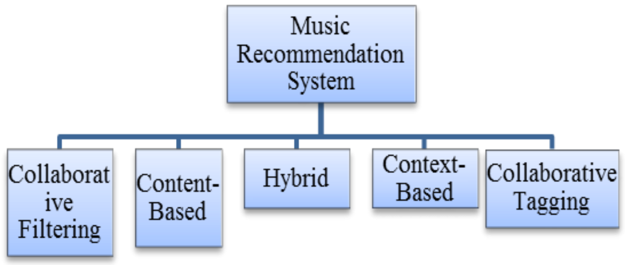

Fig. 1: Music Recommendation Systems

A review of previous work is given in Section 2. In Section 3, we describe the proposed method for predicting the user's preferences by mining social tags. Empirical evaluations of our proposed method on real data set are described in Section 4. Finally, conclusions and future work are stated in Section 5.

II. MUSIC RECOMMENDATION SYSTEMS

A. *Collaborative Filtering:*

Collaborative filtering (CF) is the most common approach for music recommendation systems. This technique relies on user generated content such as ratings or implicit feedback. It is based on the "word of mouth" approach to recommendations— items are recommended to a user if they were liked by similar users [7]. The conclusion is that, collaborative systems do not need to deal with the content, i.e., they do not base the decision whether to recommend an item or not on the description, or the physical properties of the item. In case of music recommendations it allows to avoid the task of analyzing and classifying music content.

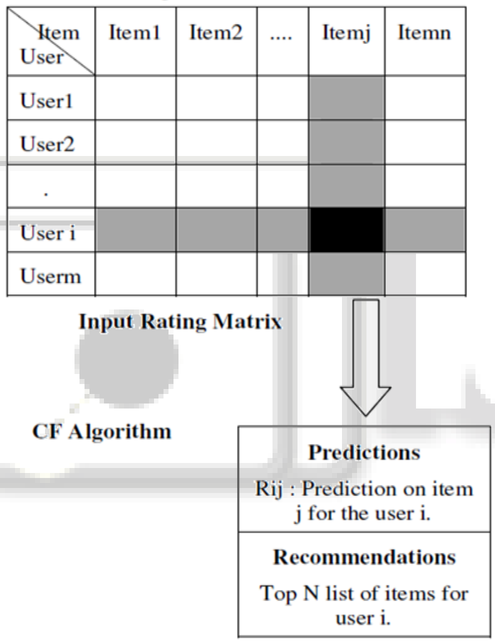

Fig. 2: Collaborative Filtering Process [30]

Recommendation is a list of top N items that the user will like the most. This output interface is known as Top-N Recommendation.

B. *Categories:*

Collaborative filtering algorithms can be classified into three general categories—memory based, model based and content based.

1) *Memory-Based CF:*

This is a traditional recommendation paradigm that infers the ratings by the user-to-item matrix. It is well known that user-based recommender systems [8][9] predict the item ratings by the most-relevant users on similar ratings, while item-based ones [10][11] predict the item ratings by the most-relevant items on similar ratings.

Memory based algorithms operate over the entire database to make predictions. Suppose U is the set of all users, and I the set of all items. Then the rating data is stored in a matrix R of dimensions |U| × |I|, where each element ru,i in a row u is equal to the rating the user u gave to item i, or is null if the rating for this item is not known. The task of CF is to predict the null ratings. An unknown rating of user u for item i can be predicted either by finding a set of users similar to u (user-based CF), or a set of items similar to i (item-based CF), and then aggregating the ratings of similar users/items. The formulas for user-based CF are given below. Given an active user u and an item i, the predicted rating for this item is:

$$\hat{r_{ui}} = r_u + K \sum_{v=1}^{n} w(u,v)(r_{vi} - r_v) \tag{2.1}$$

Where ru is the average rating of user u, n is the number of users in the database with known ratings for item i, w(u, v) is the similarity of users u and v, K is a normalization factor such that the sum of w(u, v) is 1 [12]. Different ways have been proposed to compute the user similarity score w. The two most common are Pearson correlation (1) and Cosine distance (2) measures:

$$w(u,v) = \frac{\sum_{j=1}^{k} (r_{uj} - r_u)(r_{vj} - r_v)}{\sqrt{\sum_{j=1}^{k} (r_{uj} - r_u)^2 \sum_{j=1}^{k} (r_{vj} - r_v)^2}} \tag{2.2}$$

$$w(u,v) = \frac{\sum_{j=1}^{k} r_{uj} r_{vj}}{\sqrt{\sum_{j=1}^{k} r_{uj}^2 \sum_{j=1}^{k} r_{vj}^2}} \tag{2.3}$$

Where k is the number of items both users u and v have rated.

2) *Model Based CF:*

Model based algorithms use the database of user ratings to learn a model which can be used for predicting unknown ratings. These algorithms take a probabilistic approach, and view the collaborative filtering task as computing the expected value of a user rating, given his ratings on other items. If user's ratings are integer values in the range [0,m], the predicted rating of a user u for an item i is:

$$\hat{r_{ui}} = \sum_{j=0}^{m} \Pr(r_{ui} = j | r_{uk}, k \in R_u) j \tag{2.4}$$

Where Ru is the set of ratings of the user u, and Pr(ru,i =j|ru,k, k ∈ Ru) is the probability that the active user u will give a rating j to the item i, given her previous ratings. The most used techniques for estimating this probability are Bayesian Network and Clustering approaches [12][13].

Recently, a new group of model-based techniques – known as matrix factorization models – has become popular in the recommender systems community [14][15]. These approaches are based on Singular Value Decomposition (SVD) techniques, used for identifying latent semantic factors in information retrieval.

Given the rating matrix R of dimensions |U| × |I|, matrix factorization approach discovers f latent factors by finding two matrices – P (of dimension |U| × f) and Q (of dimension |I| × f) – such that their product approximates the matrix R:

$$R \approx P \times QT = \hat{R} \tag{2.5}$$

As memory-based algorithms compute predictions by performing an online scan of the user-item ratings matrix to identify neighbor users of the target one, they do not scale well for large real-world datasets.

On the other hand, model-based algorithms use pre-computed models to make predictions. Therefore, most practical algorithms use either pure model-based techniques, or a mix of model- and memory-based approaches [13].

3) Content Based CF:

It takes advantage of additional content information such as low-level audio features [16], profiles, tags, etc. to enhance the recommendation.

Limitations faced by collaborative filtering approach are stated as follows:

a) Popularity Bias:

Generally, popular music can get more ratings. The music in long tail, however, can rarely get any. As a result, collaborative filtering mainly recommends the popular music to the listeners. Though giving popular items are reliable, it is still risky, since the user rarely get pleasantly surprised

b) Cold Start:

It is a problem associated with new users/ items. When a new item/user is added to the rating matrix, few ratings are provided. Due to the lack of these ratings, prediction results are poor.

c) Human Effort:

A perfect recommender system should not involve too much human efforts, since the users are not always willing to rate. The ratings may also grow towards those who do rate, but it may not be representative. Because of this absence of even distributed ratings, it can either give us false negative or false positive results.

d) Data Sparsity:

It is another common problem of CF. When the number of users and items is large; it is common to have very low rating coverage, since a single user typically rates only a few items. As a result, predictions can be fallacious when based on neighbors whose similarity is estimated on a small number of co-rated items.

e) Rating-Diversity Problem:

It stands for that the ratings are inconsistent among items or users [17]. For example, consider a table shown in Figure 1. It depicts a matrix containing the ratings between six users and six items. In this example, the most relevant item to item 2 is item 4 for traditional item-based recommender systems, while the most relevant user to user 1 is user 5 for traditional user-based recommender systems. If the target item for user 4 is item 4, the rating should be predicted as 1 by referring to item 2. Accordingly, the rating error is 4-1=3 as the ground truth is 4. Such error is too large to represent the real preference. Also, the big rating error, which is 5-1=4, occurs in user-based recommender systems, while the target item is item 5 for user 1.

	Item1	**Item2**	**Item3**	**Item4**	**Item5**	**Item6**
User1	1	0	0	5	1	0
User2	0	2	0	3	2	3
User3	0	2	0	2	0	0
User4	4	1	0	1	0	0
User5	1	0	0	5	1	3
User6	0	2	0	1	3	0

Fig. 3: Example of Traditional CF-based Predictions for Item Ratings

The most important advantage of Collaborative systems is that they do not need to deal with the content (i.e. description, or the physical properties) of the item.

C. Content Based CF:

Though collaborative filtering was one of the first approaches used for recommending music, content-based (CB) recommendations in music domain have been used substantially less. The reason for this might be that content-based techniques require knowledge about the data, and music is extremely difficult to describe and classify.

Content-based systems [18][19] store information describing the items, and retrieve items that are similar to those known to be liked by the user. Items are typically represented by *n*-dimensional feature vectors. The features describing items can be collected automatically (e.g., using Signal analysis in case of music tracks) or assigned to items manually (e.g., by field experts).

The vital step of content-based approach is learning the user model based on his preferences. This is a classification problem where the task is to learn a model that given a new item would predict whether the user would be interested in the item. A number of learning algorithms can be used for this. A few examples are the Nearest Neighbor and the Relevance Feedback approaches. The Nearest Neighbor algorithm simply stores all the training data, i.e., the items implicitly or explicitly evaluated by the user, in memory. In order to classify a new, unseen item, the algorithm compares it to all stored items using a similarity function (typically, Cosine or Euclidean distance between the feature vectors), and determines the nearest neighbor, or the *k*-nearest neighbors. The class label or a numeric score for a previously unseen item can then be derived from the class labels of the nearest neighbors. Relevance Feedback was introduced in information retrieval field by Rocchio [20]. It can be used for learning the user's profile vector. Initially, the profile vector is empty. It gets updated every time the user evaluates an item. After a sufficient number of iterations, the vector accurately represents the user's preferences.

$$q_m = \alpha q_0 + \left(\beta \frac{1}{|D_r|} \sum_{d_j \in D_r} d_j\right) - \left(\gamma \frac{1}{|D_{nr}|} \sum_{d_j \in D_{nr}} d_j\right) \qquad (2.6)$$

Here, q_m is the modified vector, q_0 is the original vector, D_r and D_{nr} are the set of relevant and non-relevant items, and α, β, and γ are weights that are shifting the modified vector in a direction closer, or farther away from the original vector.

A fundamental issue with content-based filtering is whether the system is able to learn user preferences from user's actions regarding one content source and use them across other content types. Content similarity cannot completely capture the preferences of a user. This causes semantic gap between the user's perception of music and the system's music representation. Next limitation is automatic feature extraction. The recommended tracks may also lack novelty, and this occurs because the system tends to recommend items too similar to those that contributed to define the user's profile.

D. Hybrid Recommender Systems:

Recent research has revealed that a hybrid approach, combining collaborative filtering and content-based filtering could be more effective in some cases. Hybrid approaches can be implemented in many ways: by making content-based and collaborative-based predictions separately and then combining them; by adding content-based capabilities to a collaborative-based approach (and vice versa); or by unifying the approaches into one model [21]. Several studies practically compare the performance of the hybrid with the pure collaborative and content-based methods and proved that the hybrid methods can provide more accurate recommendations than pure approaches. These methods also overcome some of the common problems in recommender systems such as cold start and the sparsity problem. Netflix is an example of hybrid systems.

E. Context Based Approach:

In the area of recommender systems, context can be the situation of the user when searching for recommendations (e.g., time, mood, current activity). Clearly, such information may influence the information need of the user and thus it could be taken into account, in addition to the more conventional knowledge of the user's long term preferences, when providing recommendations [22].

Classification of contextual information for area of music can be done as: environment-related context (information about the location of the user, the current time, weather, temperature, etc.), user-related context (information about the activity of the user, the user's demographic information, emotional state), and multimedia context (other types of information the user is exposed to besides music, e.g., text, images).

Limitations of this approach are:

- Labeling of music tracks with appropriate tags is difficult
- Requires a lot of user's effort and therefore might discourage some users from using the system

F. Collaborative Tagging Approach:

Social tagging, also known as collaborative tagging, was defined by Golder and Huberman [23] as "the process by which many users add metadata in the form of keywords to shared content". This phenomenon started with the growth of online content in the late 90's [24]. The authors discovered that the types of tags used by taggers follow a power law distribution, i.e., few kinds of tags are used most frequently, while the rest form the "long tail" of the tag cloud. This is partly explained by the fact that, while annotating resources, users are often influenced by existing tags. Furthermore, the users belonging to the same tagging community have certain knowledge in common, which helps them choose similar tags.

Tags can provide valuable knowledge about both items and users, and can be used to improve the performance of standard recommendation techniques. This is especially true for the music domain, as pure content analysis often fails to capture important aspects of human perception of music tracks. The major challenges encountered when acquiring tags are: the absence of common tag vocabulary; tag polysemy and noise; choosing between a small and clean set of labels, and a large and noisy set; popularity and tagger bias; spamming [25]. The difficulty of acquiring high quality tags is the main reason why the potential of tags in music retrieval and recommendation is not yet fully exploited.

The major challenge faced when collecting social tags is having the access to a sufficiently large online user community. With over 40 millions of users, Last.fm is the largest community website dedicated to music. Last.fm has millions of songs annotated with 960,000 individual tags [26]. This data is made publicly available via an API.

With the increasing popularity of the collaborative tagging systems, tags could be interesting and useful information to enhance RS algorithms. Unlike attributes which are "global" descriptions of items, tags are "local" descriptions of items given by the users [27].

This method alleviates the problem of rating diversity associated with Collaborative filtering method. For example [17], consider Item4 in Figure 1, it suffers from the rating-diversity problem as mentioned in Section 1.1. Assume the tag set in the database is {tag1, tag2, tag3, tag4, tag5} and the tag sets of item1, item2 and item4 are {tag1, tag2, tag5}, {tag3, tag4} and {tag1, tag2, tag5}, respectively. In this example, it is obvious that, the relevance between item1 and item4 is higher than that between item2 and item4 under considering the tag similarities. Consequently, the rating of item4 for user4 is 4 by referring to item1 and the error is 4-4=0, which is much smaller than that by referring to item2 using rating similarities.

G. Comparison of Different Music Recommendation Methods:

Methods	Advantages	Limitations
Collaborative Filtering	– Do not need to deal with the content of the item	– Rating diversity problem – Data Sparsity – Human Effort – Cold start problem – Popularity Bias

Content-based Filtering	– The "cold start" problem is only partly present – No popularity bias	– Modeling of user's preferences is a major problem – Automatic feature extraction is difficult – Recommended tracks may lack novelty
Hybrid	– It takes advantage and avoids the shortcomings of both techniques – More accurate recommendations than pure approaches	– They consider only direct similarity between items or users
Context-based	– Context information influences the information need of the user	– Requires a lot of user's effort – Labeling of tracks with tags is difficult
Collaborative tagging	– It alleviates the problem of rating diversity associated with CF – Can be shared among users – It provides information about various features, like the genre, style, mood, or instrumentation – It can be used to improve the performance of standard recommendation techniques	– Absence of common tag vocabulary – Choosing between a small and clean set of labels, or a large and noisy set is difficult – Popularity bias – Tags may have more than one meaning- a problem called polysemy – Difficulty of acquiring high quality tags

Table 1: Comparison of Different Music Recommendation Methods

III. COLLABORATIVE TAGGING ALGORITHM

Collaborative tagging algorithm can be divided into following steps according to Ja-Hwung et al.[17]

Step 1: Rating transformation

The play counts are divided into two ranges by a threshold *T*, which is defined as:

$$T = \mu - \tau * \sigma \quad (3.1)$$

Where μ indicates mean of play counts, σ indicates standard deviation of play counts for a user and τ is a weight.

The range lower than *T* is further divided into two equivalent sub-ranges with respect to the range number set {1, 2}, while that higher than *T* is divided into three equivalent sub-ranges with respect to the range number set {3, 4, 5}.

If a play count is in the specific range, it can be transformed into the referred range number.

Step 2: Construction of item similarity matrix

Calculate the item-tag-driven similarity between itm_a and itm_b, given *i* unique items IM=$\{itm_1, itm_2, \dots\dots, itm_i\}$, *j* unique tags TG=$\{tag_1, tag_2, \dots\dots, tag_j\}$ in the database and the tag feature vector for $itm_n \in$ IM as $iv_n=\{f_1^n, f_2^n, \dots\dots, f_j^n\}$ where f_j^n is the frequency of tag_j for itm_n, from given formula

$$ITsim_{a,b}=\frac{\sum_{0<q\le j} f_q^a * f_q^b}{\sqrt{\sum_{0<q\le j}(f_q^a)^2} * \sqrt{\sum_{0<q\le j}(f_q^b)^2}} \quad (3.2)$$

For example, let there be 10 unique items, 5 unique tags and 5 unique artists i.e. i=10, j=5. Therefore, tag feature vector for itm_1 and itm_2 are

$iv_1=\{f_1^1, f_2^1, f_3^1, f_4^1, f_5^1\}$
={5,10,7,0,1}
iv_2={2,4,6,8,1}

Thus, item similarity between itm_1 and itm_2 is given as

$$ITsim_{1,2}=\frac{(5*2)+(10*4)+(7*6)+(0*8)+(1*1)}{\sqrt{5^2+10^2+7^2+0^2+1^2}\sqrt{2^2+4^2+6^2+8^2+1^2}}$$
=77.33

Calculate the artist-tag driven similarity between art_a and art_b, given *m* unique artists AT=$\{art_1, art_2, \dots\dots, art_m\}$ in the database and the tag feature vector for art_m is defined as $av_m=\{d_1^m, d_2^m, \dots\dots, d_j^m\}$, from given formula

$$ATsim_{a,b}=\frac{\sum_{0<q\le j} d_q^a * d_q^b}{\sqrt{\sum_{0<q\le j}(d_q^a)^2} * \sqrt{\sum_{0<q\le j}(d_q^b)^2}} \quad (3.3)$$

where $art_a=\{itm_x | itm_x \in IM\}$ and $art_b=\{itm_y | itm_y \in IM\}$

$$d_q^a=\frac{\sum_{itm_x \in art_a} f_q^{itm_x}}{|art_a|} \quad (3.4)$$

and

$$d_q^b=\frac{\sum_{itm_y \in art_b} f_q^{itm_y}}{|art_b|} \quad (3.5)$$

Now, let $art_1=\{itm_1, itm_2\}$ and $art_2=\{itm_2, itm_3\}$. So,

$$d_1^1 = \frac{f_1^{itm_1}+f_1^{itm_2}}{|art_1|}=3.5$$

∴ av_1={3.5,7,6.5,4,1}

$$d_1^2 = \frac{f_1^{itm_2}+f_1^{itm_3}}{|art_2|}=1$$

∴ av_2={1,2.5,4.5,5,3.5}

Thus, artist similarity between art_1 and art_2 is given as

$$ATsim_{1,2} =\frac{(1*3.5)+(2.5*7)+(4.5*6.5)+(5*4)+(3.5*1)}{\sqrt{3.5^2+7^2+6.5^2+4^2+1^2}\sqrt{1^2+2.5^2+4.5^2+5^2+3.5^2}}$$
=0.834

Given an item-tag-driven similarity $ITsim_{a,b}$ and an artist-tag-driven similarity $ATsim_{a,b}$, the fusion similarity between itm_a and itm_b can be calculated as

$$FUsim_{a,b}= ITsim_{a,b} * ATsim_{a,b} \quad (3.6)$$

Step 3: Rating prediction

The goal of this step is to infer the unknown ratings for the active user.

It starts with an active user's *uz*, visit and then the unknown ratings for the active user are predicted one by one.

First, the prediction determines *k* most-relevant items to the target item *itmi* by the calculated item similarities which are given as $U_z= \cup\ itm_p$, where $itm_p \in IM$ e.g.: $U_1=\{itm_1, itm_3, itm_7\}$

Next, the rating v is calculated for an unknown target item , itm_i for u_z as:

$$v_i^z = \frac{\sum_{itm_p \in Uz} sim_{i,p} * v_p^z}{\sum_{itm_p \in Uz} sim_{i,p}} \qquad (3.7)$$

where

$$\begin{cases} sim_{i,p} = ITsim_{i,p}, & \text{if using item-tag-driven similarity} \\ sim_{i,p} = ATsim_{i,p}, & \text{if using artist-tag-driven similarity} \\ sim_{i,p} = FUsim_{i,p}, & \text{if using fusion similarity.} \end{cases}$$

Therefore, the rating v is calculated for itm_5 for u_1 as follows:

$$\therefore v_5^1 = \frac{(66*1)+(50*3)+(42*4)}{(66)+(50)+(42)}$$

$= 2.43 \approx 2$

IV. CONCLUSION

Till now many recommendation systems have been proposed that are based on collaborative filtering, content based filtering and hybrid recommendation methods and so far most of them have been able to solve the problems associated with providing better recommendations. Therefore this paper presents the review of the state-of-the-art research on music recommendation. This survey paper gives detailed classification of music recommendation systems along with their merits and demerits. Yet much amount of research needs to be carried out to explore and provide new methods that can provide recommendation in a wide range of applications while considering the quality and privacy aspects. Recently, Social Tagging Systems have attracted increasing attention and tag-aware recommender systems (Collaborative tagging) have emerged. Thus a novel study in the utilization of tags as supplementary source to predict item recommendations has been conducted. Therefore this survey also focuses on tag based recommendation approach.

REFERENCES

[1] Alexandra Uitdenbogerd and van Schyndel Ron. A Review of Factors Affecting Music Recommender. In 3rd International Conference on Music Information Retrieval (2002), 2002.
[2] F. Ricci, L. Rokach, B. Shapira, P.B. Kantor (Eds.), Recommender Systems Handbook, Springer, 2011.
[3] Dmitry Bogdanov and Perfecto Herrera. How Much Metadata Do We Need in Music Recommnendation? A Subjective Evaluation Using Preference Sets. In 12th International Society for Music Information Retrieval Conference, number ISMIR 2011, pages 97–102, 2011.
[4] Robin Burke. Hybrid Recommender Systems: Survey and Experiments. User Modeling and User-Adapted Interaction, 12(4):331–370, 2002.
[5] G. Adomavicius, B. Mobasher, F. Ricci, A. Tuzhilin, Contextaware recommender systems, AI Magazine 32 (3) (2011) 67–80.
[6] J.Y. Kim, N.J. Belkin, Categories of music description and search terms and phrases used by non-music experts, in: 3rd International Conference on Music Information Retrieval, Paris, France, 2002, pp. 209–214.
[7] U. Shardanand, P. Maes, Social information filtering: algorithms for automating "word of mouth", in: CHI'95: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, ACM Press/Addison-Wesley Publishing Co., New York, NY, USA, 1995, pp. 210–217.
[8] Herlocker J. L., Konstan J. A., Borchers A. and Riedl J. An algorithmic framework for performing collaborative filtering. Proc. of the international ACM SIGIR conference on Research and development in information retrieval, pp. 230-237, Berkeley, USA, 1999.
[9] Resnick P., Iacovou N., Suchak M., Bergstrom P. and Riedl J. GroupLens: An Open Architecture for Collaborative Filtering of Netnews. Proc. Int. Conf. on ACM Computer Supported Cooperative Work, pp.175-186, 1994.
[10] Deshpande M. and Karypis G. Item-based top-N recommendation algorithms. ACM Transactions on Information Systems. Vol. 22, No. 1, pp. 143-177, 2004.
[11] Sarwar B.M., Karypis G., Konstan J. and Riedl J. Item-Based Collaborative Filtering Recommendation Algorithms. Proc. Int. Conf. on World Wide Web, pp. 285-295, Hong Kong, 2001.
[12] J. Breese, D. Heckerman, C. Kadie, et al. Empirical analysis of predictive algorithms for collaborative filtering, in: Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence, Vol. 461, 1998, pp. 43–52.
[13] J. Schafer, D. Frankowski, J. Herlocker, S. Sen, Collaborative filtering recommender systems, Lecture Notes in Computer Science 4321 (2007) 291.
[14] Y. Koren, R. Bell, C. Volinsky, Matrix factorization techniques for recommender systems, IEEE Computer 42 (8) (2009) 30–37.
[15] Y. Koren, R. Bell, Advances in collaborative filtering, in: F. Ricci, L. Rokach, B. Shapira (Eds.), Recommender Systems Handbook, Springer Verlag, 2011, pp. 145–186.
[16] Ja-Hwung Su, Hsin-Ho Yeh, Philip S. Yu and Vincent S. Tseng. Music Recommendation Using Content and Context Information Mining. IEEE Intelligent Systems, Vol. 25, No. 1, pp.16-26, 2010.
[17] Ja-Hwung Su, Wei-Yi Chang and Vincent S. Tseng. "Personalized Music Recommendation by Mining Social Media Tags". 17th International Conference in Knowledge Based and Intelligent Information and Engineering Systems - KES2013, Elsevier, 2009.
[18] P. Lops, M. de Gemmis, G. Semeraro, Content-based recommender systems: state of the art and trends, in: F. Ricci, L. Rokach, B. Shapira, P. Kantor (Eds.), Recommender Systems Handbook, Springer Verlag, 2011, pp. 73–105.
[19] M. Pazzani, D. Billsus, Content-based recommendation systems, Lecture Notes in Computer Science 4321 (2007) 325.
[20] J. Rocchio, Relevance feedback in information retrieval, in: The SMART Retrieval System: Experiments in Automatic Document Processing, vol. 313, 1971, p. 323.
[21] http://en.wikipedia.org/wiki/Recommender_system

[22] Q.N. Nguyen, F. Ricci, "User preferences initialization and integration in critique-based mobile recommender systems", in: Proceedings of the 5th International Workshop on Artificial Intelligence in Mobile Systems, AIMS'04, Nottingham, UK, 2004, pp. 71–78.
[23] S. Golder, B. Huberman, "Usage patterns of collaborative tagging systems", Journal of Information Science 32 (2) (2006) 198.
[24] T. Hammond, T. Hannay, B. Lund, J. Scott, "Social bookmarking tools (I)", D-Lib Magazine 11 (4) (2005) 1082–9873.
[25] P. Lamere, "Social tagging and music information retrieval", Journal of New Music Research 37 (2) (2008) 101–114.
[26] F. Miller, M. Stiksel, R. Jones, Last.fm in numbers, Last.fm press material.
[27] Tso-Sutter K.H. L, Marinho L.B. and Schmidt-Thieme L. "Tag-aware recommender systems by fusion of collaborative filtering algorithms". Proc. Int. Conf. on the ACM symposium on Applied computing, Fortaleza, pp. 1995-1999, 2008.
[28] Marius Kaminskas and Francesco Ricci. "Contextual music information retrieval and recommendation: State of the art and challenges". Elsevier, 2012.
[29] Yading Song, Simon Dixon, and Marcus Pearce. "A Survey of Music Recommendation Systems and Future Perspectives". 9th International Symposium on CMMR,2012.
[30] Meenakshi Sharma and Sandeep Mann. "A Survey of Recommender Systems: Approaches and Limitations".