A Web Recommender System for Recommending, Predicting and Personalizing Music Playlists

Zeina Chedrawy and Syed Sibte Raza Abidi

Faculty of Computer Science, Dalhousie University, Halifax, Canada
{chedrawy,sraza}@cs.dal.ca

Abstract. In this paper, we present a Web recommender system for recommending, predicting and personalizing music playlists based on a user model. We have developed a hybrid similarity matching method that combines collaborative filtering with ontology-based semantic distance measurements. We dynamically generate a personalized music playlist, from a selection of recommended playlists, which comprises the most relevant tracks to the user. Our Web recommender system features three functionalities: (1) *predict* the likability of a user towards a specific music playlist, (2) *recommend* a set of music playlists, and (3) *compose* a new personalized music playlist. Our experimental results will show the efficacy of our hybrid similarity matching approach and the information personalization method.

Keywords: Web personalization, Web recommender systems, music recommendation, semantic similarity matching.

1 Introduction

Access to and consumption of relevant information is paramount to Web users, especially given the sheer volume of information now available over the Web. A key approach to overcome cognitive overload faced by users is the development of *user-centric systems*—termed as Web recommender systems, adaptive or personalized information retrieval systems—that retrieve/recommend Web-based information artifacts such as documents and Websites based on the user's preferences and goals. The idea is that a *one size fits all* model for Web information retrieval is non-optimal, rather the individualistic nature of each user should be taken into account to provide the user with a personalized Web experience [12,13]. This brings to relief the need to pursue *intelligent information personalization* by working with the 'semantics' of the information through the use of Semantic Web technologies [15].

In this paper, we present a Web recommender system for recommending, predicting and personalizing music playlists based on a user model. We have developed an item and user matching approach that combines the Web 2.0 notion of peer wisdom and Web 3.0 concept of semantic relationships between items/users. Our similarity matching approach is a hybrid of collaborative filtering (CF) and semantic distance measurement methods. Our music recommender system offers the functionality to dynamically compose a personalized music playlist by selecting the most relevant individual tracks from a list of recommended playlists, and then

G. Vossen, D.D.E. Long, and J.X. Yu (Eds.): WISE 2009, LNCS 5802, pp. 335–342, 2009.

aggregating them to generate a personalized playlist. Our Web recommender system features three functionalities: (1) predict the likability of a user towards a specific music playlist, (2) recommend a set of music playlists that are potentially of interest to a user, and (3) compose a new personalized music playlist. Our experimental results will show that (1) the use of semantic descriptions of information items combined with the multi-attribute CF improves the accuracy of predictions and the quality of recommendations; and (2) the application of our compositional adaptation method allows fine-tuning of existing information items to make them more personalized vis-à-vis the user model. Fig. 1 shows the schematic of our Web-based music recommender system.

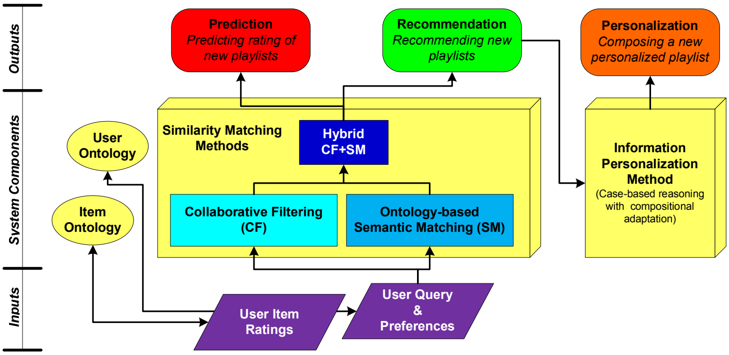
Outputs
System Components
Inputs
Prediction
Predicting rating of new playlists
Recommendation
Recommending new playlists
Personalization
Composing a new personalized playlist
User Ontology
Item Ontology
Similarity Matching Methods
Hybrid CF+SM
Collaborative Filtering (CF)
Ontology-based Semantic Matching (SM)
Information Personalization Method
(Case-based reasoning with compositional adaptation)
User Item Ratings
User Query & Preferences

Fig. 1. A schematic of our Web-based recommender system

2 Web Recommender Systems for Information Personalization

Web recommender systems can be categorized into the following three categories (a) Standard Content/Collaboration Based Systems; (b) Hybrid Systems; (c) Semantics Based Systems. Recommender Systems employing the *content-based filtering* approach recommend information artifacts based on their relevance to existing user profiles and on the descriptions of the artifacts [1,2]. The limitation of the content-based approach is that recommendation is limited to the description within the user profile, thus other interesting and/or related information artifacts that do not match the user profile are not recommended. Recommender Systems using *collaborative-filtering* (CF) approach take a community-based approach by computing the similarities between the profiles of multiple users and then recommend information artifacts that are preferred by the community closest to a given user [3,10]. CF based recommender systems on the one hand may recommend surprising information artifacts, whereas on the other hand, they may miss out on information artifacts that are uniquely of interest to a specific user but not necessarily to his/her community. To address the limitations of content and CF based recommender systems, the next generation of recommender systems employed a hybrid of both content and collaborative approaches [4,8,9]. To improve the information search and the matching of information items and users, the recent trend is to exploit the semantic information associated with user and item descriptions. Typically, ontologies are created to represent the user model and the information item by incorporating salient concepts and their relationships. A semantic matching algorithm can look beyond simple keyword search by traversing an RDF graph to find both specializations and generalizations of a given concept. Examples of such systems are [5,14].

From the literature review it is clear that there is an abundance of Web recommender systems that are apt at recommending information artifacts based on user and/or community model. However, these systems are not able to further adapt the available information artifacts to compose a *personalized information artifact*—the recommended artifact may have some elements that are of no use to the user. Our work is an extension of the abovementioned systems as we pursue to dynamically compose a personalized music playlist by selecting music objects from multiple Web sources and then aggregating them in a meaningful manner to yield a composite information artifact that is more pertinent to the user's interest.

3 Our Hybrid Item-Based Similarity Matching Method

For the purposes of predication, recommendation and personalization of playlists, the first step is to establish similarities between existing items/users. We have extended existing methods to develop a hybrid item-based similarity matching method.

Item-Based Collaborative Filtering Multi-Attribute Similarity: We have extended traditional single-attribute CF approach [6] by developing a multi-attribute rating scheme that allows users to rate a music playlist along five attributes (*lyrics*, *rhythm*, *tunes*, *performance*, and *overall likability*). The algorithm is described below:

<u>Step 1-</u> *Specify user preferences.* The user assigns the weight values (W_A) to each attribute along which similarities between information items are to be computed.
<u>Step 2-</u> *Compute the similarity between items with respect to every attribute.* For every attribute A, the similarity between information items I and J as given by [6]:

$$Sim_A(I,J) = \frac{\sum_{\forall U \in users}(R_A(U,I)-\bar{R}_A(U))(R_A(U,J)-\bar{R}_A(U))}{\sqrt{\sum_{\forall U \in users}(R_A(U,I)-\bar{R}_A(U))^2}\sqrt{\sum_{\forall U \in users}(R_A(U,J)-\bar{R}_A(U))^2}}. \tag{1}$$

Where $R_A(U,I)$ denotes the rating of user U on item I with respect to attribute A; $\bar{R}_A(U)$ is the average rating of user U as per attribute A.
<u>Step 3-</u> *Compute the CF multi-attribute similarity between items*

$$MultSim(I,J) = \frac{\sum_{A=1}^{5} W_A * Sim_A(I,J)}{\sum_{A=1}^{5} W_A}. \tag{2}$$

Item-Based Semantic Similarity: In this method, we calculate the similarities between two items based on their semantic descriptions given in an ontology. The similarity between items I and J is based on the ratio of the common/shared RDF descriptions between I and J (*count_common_desc(I,J)*) to their total descriptions (*count_total_desc(I,J)*) as proposed by [7] and is given by:

$$SemSim(I,J) = \frac{count\_common\_desc(I,J)}{count\_total\_desc(I,J)}. \tag{3}$$

We argue that our semantic similarity approach helps to bootstrap Web recommender systems in case not enough ratings are available on a particular item (cold-start problem), and also provides explanations about why a particular information artifact has been recommended or not.

Hybrid Item-Based Semantic-CF Similarity: Using (Eq.2 and Eq.3) we calculate the hybrid Semantic-CF similarity using a linear weighted approach as:

$$Sim(I,J) = W_M * MultSim(I,J) + W_S * SemSim(I,J) \,. \tag{4}$$

Where W_M and W_S are the weights assigned to CF multi-attribute and semantic similarities respectively.

4 Prediction, Recommendation and Personalization of Playlists

Prediction. Given a user *U*, an instance in the user ontology containing the ratings on music playlists, and a playlist *P* (not yet rated by *U*), our recommender system predicts the rating of *U* on *P* using the method provided in [6], where the similarity between items is replaced by our hybrid similarity (Eq.4).

Recommendation. The recommendation of a list of music playlists to a user is based on his/her past ratings and the playlists' ratings of other peers. The algorithm for recommending a list of playlist most similar to the user is the standard CF algorithm by [6] with the extension of using our hybrid Semantic-CF similarity (Eq.4).

Personalization. Typically, Web recommender systems recommend the complete information artifact even if it comprises multiple components—for instance a book comprises chapters, and a music playlist comprises individual songs. We have developed a component-level recommendation approach that allows the dynamic selection of components from artifacts, as per their relevance to the user model, and aggregating them to compose a personalized information artifact. It may be noted that, our approach is only applicable when the multiple independent components do not have any inter-relationships between them and are simply part of a larger artifact [12].

Given S_N as the set of the *N* playlists recommended to the user, we recommend a personalized playlist as follows:

Step 1- Compute the similarity of the individual tracks within S_N with the user model. Let $Sim(P_i, U)$ $(i=1$ *to* $N)$ be the similarity of playlist P_i to user *U* (i.e. similarity of playlist P_i to the set of playlists preferred by *U* as derived in [6]). Let $Sim_1(T,U)$ be the similarity of track *T* to *U*. Because the same track may belong to multiple playlists, therefore we define the similarity of a track to the user model over all the playlists in S_N that have the track *T* as follows:

```
For every Track $T \in P_i$,
        For every Playlist $P \in S_N$,
If $T \in P$ then $Sim_1(T,U) = Sim_1(T,U) + Sim(P_i,U)$.        (5)
```

Step 2: Compute the genre-based similarity between individual tracks and the user model. Each playlist and track has a list of music genres assigned to it (i.e. Rock). Each user is also assigned a list of genres that represent the genres of the playlists he has listened to. We represent users and tracks as vectors in the *g*-dimensional genre space and use the cosine between these vectors as a measure of their similarity. Let $\vec{U}_U$ and $\vec{R}_T$ be the vectors of *g* dimensions for user *U* and track *T* respectively, where *g* corresponds to the number of genres available. Then the cosine similarity *cos_sim(T,U)* between *U* and *T* is:

$$cos\_sim(T,U) = cos(\vec{R}_T,\vec{U}_U) = \frac{\vec{R}_T.\vec{U}_U}{\|\vec{R}_T\|_2\,\|\vec{U}_U\|_2} = \frac{\sum_g (m_g n_g)}{\sqrt{\sum_g m_g^2}\,\sqrt{\sum_g n_g^2}} \,. \tag{6}$$

Where n_g is the number of playlists of genre *g* that user *U* has listened to; m_g is the number of times the track *T* has been assigned genre *g*. We apply the sigmoid function to the cosine similarity to scale it within the range [-1,1]. Genre similarity between a user *U* and a track *T* is given as:

$$Sim_2(T,U) = \frac{1}{1+e^{cos\_sim(T,U)}} \,. \tag{7}$$

We compute the overall similarity of each track with the user model as follows:

$$Sim(T,U) = Sim_1(T,U) * Sim_2(T,U) \,. \tag{8}$$

Step 3: Select tracks that are most similar to the user model by sorting them based on their similarity values and selecting the top *M* tracks (*M=15*).

5 Experimental Results and Evaluation

In this Section, we evaluate the performance of our Web based music recommendation system in terms of (1) the impact of semantic similarity towards improving the accuracy and quality of predictions and recommendations, (2) the appropriateness of the personalized playlist towards the user model.

The music data is taken from the Website *Lastfm* (http://www.last.fm) that is a radio station which uses CF to recommend a radio stream to its listeners. The data set contains 215 users having 46850 album ratings distributed across five attributes of an album. In total we have 4426 albums, where each album contains a number of tracks. For testing purposes, we divide the user ratings dataset into *training* (80%) and *testing* sets (20%). We perform a 5-fold cross validation and results are averaged over the 5 cycles of execution. We conducted a set of experiments using four different scenarios. *Scenario 1* uses the CF method only, *Scenario 2* uses the semantic similarity method only, *Scenario 3* uses a hybrid of CF and semantic similarities methods with equal contribution (both have a weight of 0.5), and *Scenario 4* uses a hybrid of CF and semantic similarities with different weights (W_S=0.7; W_M=0.3).

Evaluating Prediction. We evaluate prediction based on the rating along the *overall likability* attribute only; the same evaluation can be applied to the other attributes. This experiment takes as input test users and their items for which a predicted rating value is desired, and the respective weights assigned to the rating

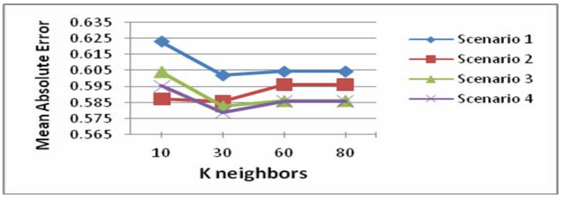

Fig. 2. MAE values for predictions with varying neighborhood sizes

attributes of these users. We use the *Mean Absolute Error (MAE)* [6] to measure the accuracy of our predictions. We initially set all test users' weights to $W_{lyrics} = 0.2$; $W_{performance} = 0.3$; $W_{tunes} = 0.4$; $W_{rhythm} = 0.1$; $W_{overall\ likability} = 0.9$ for all four scenarios. Fig. 2 shows the values of MAE obtained with various neighborhood sizes. It may be noted that $K=30$ is the optimal neighborhood size for all scenarios.

Table 1 shows the MAE values for the four scenarios with neighborhood size $K=30$. It may be noted that *Scenario 4* produces the least MAE.

Table 1. Prediction MAE values for the four scenarios *(K=30)*

	Scenario 1	Scenario 2	Scenario 3	Scenario 4
MAE	*0.60192*	*0.58563*	*0.58257*	***0.57878***

Next, we aim to understand the contribution of the semantic similarity method to the overall hybrid similarity matching approach. We modulate the contribution of the semantic similarity method towards the calculation of the overall similarity with $K=30$ (Fig. 3). We note that when $W_S=0$ (*Scenario 1*), MAE = 0.60192; when $W_S=1$ (*Scenario 2*), MAE = 0.58563. Fig. 3 shows that MAE reaches its minimum when the contributions are $W_M=0.3$ and $W_S=0.7$, thus highlighting the significant impact of our semantic similarity approach to the overall similarity value.

Evaluating Recommendation. We evaluate the recommendation accuracy of our music recommender system using the *HITS' number* as in [11]. We set N = 15 top recommendations, M = 15 tracks/playlist. We set K=30, W_S= 0.7, W_M= 0.3.

From Table 2 it may be noted that there is a clear advantage (i.e. number of HITS) when we combine the semantic and CF similarities thus vindicating the efficacy of our hybrid similarity matching approach.

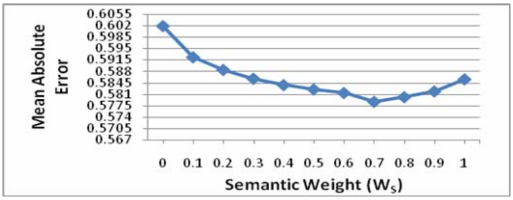

Fig. 3. Impact of the semantic similarity approach on the overall prediction

Table 2. HITS' number for CF and Semantic-CF recommendations for all test sets

Test Set		1	2	3	4	5
Semantic-CF Recommendation	*Hits*	13	10	20	17	12
CF Recommendation		2	7	4	6	2

Evaluating Personalized Playlists. We measure the quality of our personalized playlists in terms of (1) the similarity between the personalized playlist and the user model with respect to an *Appropriateness Factor (AF)*; and (2) the *Matching Genre Ratio (MGR)* of the personalized playlist to the preferred genre(s) of the user model. For each track in the personalized playlist, we compute the *MGR* as:

$$MGR_{U} = \frac{MGcount(U,T)}{Gcount(T)}, \tag{9}$$

Where *MGcount(U,T)* is the number of matching genres between user *U* and track *T* of the personalized playlist *P*; *Gcount(T)* is the number of genres for track *T*. We test our personalization approach by taking 20% of test users from every test dataset.

For all test users in each test dataset, we generated personalized playlists based on the *N* playlists that were earlier recommended to them. We compared the *AF* of the personalized playlist with the corresponding user model. Table 3 shows the percent increase in *AF* for all test users for every test dataset. On average, the increase in the *AF* over all 5 test datasets is 71.55%.

Our results show that by averaging the *MGR* over all test users in the test datasets, 74.43% of the genres associated with the personalized tracks match the user models (see Table 3). Therefore, we conclude that the personalized playlist is closer to the user's interests as compared to the original *N* recommended playlists.

Table 3. Impact of the personalization on the recommendations in terms of *AF* and *MGR*

	Test Set 1	Test Set 2	Test Set 3	Test Set 4	Test Set 5
AF % Increase	88.55 %	57.31%	81.38 %	66.59 %	63.93 %
Average	*71.55 %*				
MGR	68.69 %	78.34 %	72.63 %	75.32 %	77.18 %
Average	*74.43 %*				

6 Discussion and Concluding Remarks

Our work has demonstrated the potential of using semantic relationships to match items and individuals improves the search results, as opposed to the use of traditional collaborative filtering methods. Our information personalization approach generates improved web experiences for users in terms of providing them personalized information as opposed to the entire artifact.

As part of our future work, we plan to further extend our semantic similarity matching technique beyond using just the semantic attributes to compare items and we plan to develop a new semantic similarity measure that takes into account explicit relationships between information items that can be reasoned over to infer information items that are better correlated with the user model.

References

1. Billsus, D., Pazzani, M., Chen, J.: A Learning Agent for Wireless News Access. In: Proc. of the Intl. Conf. on Intelligent User Interfaces, pp. 33–36 (2002)
2. Cano, P., Koppenberger, M., Wack, N.: An industrial-Strength content-based Music Recommendation System. In: Proc. 28th Intl. ACM SIGIR Conf. on Research and Development in Information Retrieval, Salvador, Brazil (2005)
3. Shani, G., Chickering, M., Meek, C.: Mining Recommendations from the Web. In: Proceedings of the 2nd Intl. Recommender Systems Conference, RecSys (2008)
4. Debnath, S., Ganguly, N., Mitra, P.: Feature weighting in content based recommendation system using social network analysis. In: Proc. 17th Intl. Conf. on World Wide Web, Beijing, China, pp. 1041–1042 (2008)
5. Katakis, I., Tsoumakas, G., Banos, E., Bassiliades, N., Vlahavas, J.: An adaptive personalized news dissemination system. Journal of Intelligent Information Systems 32, 191–212 (2009)
6. Sarwar, B., Karypis, G., Konstan, J., Riedl, J.: Item-Based Collaborative Filtering Recommendation Algorithms. In: Proc. of the 10th Intl. WWW Conference (2001)
7. Hau, J., Lee, W., Darlington, J.: A Semantic Similarity Measure for Semantic Web Services. In: Web Service Semantics: Towards Dynamic Business Integration (2005)
8. Resnick, P., et al.: GroupLens: An Open Architecture for Collaborative Filtering of Netnews. In: Proc. of ACM Conf. on Computer Supported Cooperative Work, pp. 175–186. Chapel Hill, NC (1994)
9. Celma, O., Ramrez, M.H.P.: Foafing the music: A Music Recommendation System Based on RSS Feeds and User Preferences. In: Proc. 6th Intl. Conf. on Music Information Retrieval (2005)
10. Cotter, P., Smyth, B.: PTV: Intelligent Personalized TV Guides. In: Proc. 12th Conf. Innovative Applications of Artificial Intelligence, pp. 957–964. MIT Press, Cambridge (2000)
11. Sarwar, B., Karypis, G., Konstan, J., Riedl, J.: Analysis of Recommendation Algorithms for e-commerce. In: 2nd ACM conf. on Electronic Commerce, pp. 158–167 (2000)
12. Abidi, S.S.R.: Designing Adaptive Hypermedia for Internet Portals: A Personalization Strategy Featuring Case Base Reasoning With Compositional Adaptation. In: Garijo, F.J., Riquelme, J.-C., Toro, M. (eds.) IBERAMIA 2002. LNCS (LNAI), vol. 2527, Springer, Heidelberg (2002)
13. Abidi, S.S.R.: Intelligent Information Personalization; From Issues to Strategies. In: Germanakos, P. (ed.) Intelligent User Interfaces: Adaptation and Personalization Systems and Technologies. IGI Global Press (2008)
14. Mobasher, B., Jin, X., Zhou, W.: Semantically Enhanced Collaborative Filtering on the Web. In: Berendt, B., Hotho, A., Mladenič, D., van Someren, M., Spiliopoulou, M., Stumme, G. (eds.) EWMF 2003. LNCS (LNAI), vol. 3209, pp. 57–76. Springer, Heidelberg (2004)
15. Spivack, N.: The Third-Generation Web is Coming, `http://www.KurzweilAI.net/`
16. Schafer, J., Frankowski, D., Herlocker, J., Sen, S.: Collaborative Filtering Recommender Systems. In: Brusilovsky, P., Kobsa, A., Nejdl, W. (eds.) Adaptive Web 2007. LNCS, vol. 4321, pp. 291–324. Springer, Heidelberg (2007)