Matching Places of Interest With Music

Marius Kaminskas
Free University of Bozen-Bolzano
Bolzano, 39100, Italy
Email: mkaminskas@unibz.it

Francesco Ricci
Free University of Bozen-Bolzano
Bolzano, 39100, Italy
Email: fricci@unibz.it

***Abstract*—In this paper we address a particular kind of cross domain personalization task consisting of selecting simultaneously two items in two different domains and recommending them together because they fit the user preferences and also they fit well together. We show that given some personalized recommendations for places of interests (POIs), the user satisfaction for these POIs can be increased by enriching their presentation with music tracks that match the user's profile and are also matching the POIs. We present the results of an online experiment where alternative approaches for matching POIs and music, based on tagging and text matching, have been tested with users.**

I. Introduction

Recommender systems are personalized information search and decision support tools that are generally designed to providing recommendations for just one type of information items, e.g., either movies, or CDs, or travels. Recently, there have been some attempts to combine information from different domains, introducing the notion of cross-domain recommender systems. These systems can reuse knowledge about the users, which is derived in one domain, e.g., CDs, to provide recommendations in another different domain, e.g., movies [1].

In this paper we focus on a particular kind of cross-domain recommendation task, not yet well studied in the literature, where we try to select simultaneously two items in two different domains and to recommend them together because they fit the user preferences and they fit well together. Consider for instance a person who is sightseeing a city. The tourist may use such a recommender system running on her mobile device. This system can offer to the user a walking itinerary and, while the user is visiting the suggested places of interest (POIs), it could play a soundtrack which matches the visited POIs and enhances the experience. For example, a user visiting a Baroque cathedral might hear a classical composition by J.S. Bach, or during a visit to a lively market square the user might be offered an amusing folk tune.

In this paper we show that the user satisfaction for an itinerary can be increased by providing a musical soundtrack matching to the places of interest included in the itinerary. Furthermore, we show that simultaneously searching for pairs of matching POIs and music tracks can provide as good results as independently selecting a personalized set of POIs and then matching music tracks to the POIs. In other words, not only the music can be matched to a pre-selected itinerary, but also the discovered relations between music and a place can influence the itinerary selection process and, to a certain extent, substitute the POI personalization.

The main research challenge we have faced pertains with the complex relations between the two domains: locations and music. Some previous attempts of relating music to physical surroundings of the user include research on context-aware systems that use the information about the user's surroundings (time, weather etc.) to infer user's music preference [2], [3], [4]. Other works tried to project different domain representations to a common space, e.g. projecting text and music to a mood taxonomy. However, it is still unclear how (and which) properties of one domain relate to the properties of the other [5], [6]. Here, we have explored the combined usage of a limited number of well selected tags and textual descriptions as a way for matching items from different domains.

The rest of the paper is structured as follows. In order to give immediately a concrete example of the potential application area for the technology we are researching, we present the application used to conduct a case study (Section II). In Section III we describe the used music and POI matching techniques. In Section IV the evaluation procedure and results are presented, and finally in Section V we draw conclusions and point out future work directions.

II. Case Study Application

We start with presenting the architecture of the recommender system designed for evaluating possible matching techniques of music and POIs. Figure 1 shows the logical architecture of the system. The system uses profile information for standard item personalization as well as music-to-POI similarity for combining music and POI in a joined recommendation. The main components are: the user profile, the POI profile, the music profile and the recommendation algorithm that consists of POI ranking, music filtering and music-to-POI similarity computation. We will now describe these components in detail.

The user profile contains the basic music and sightseeing preferences of the user. The sightseeing preferences are used for ranking the POIs. The user preferences can be any combination of: art, architecture, history and nature. These four categories roughly correspond to the types of POIs stored in our database. The objects include castles, museums, monuments, churches and nature objects. Another part of the user profile contains the preferred music genre; our database contains music tracks belonging to classical and rock music

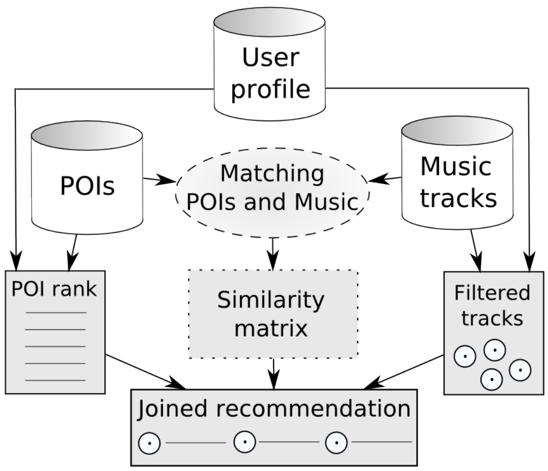

Fig. 1. Model of the system

genres. A user preferring classical (rock) music is offered only recommendations with soundtracks from the classical (rock) repertoire. In order to minimize the possibility that a user will not like a certain track, we selected some of the most famous compositions from both classical and rock music to be used in the system.

The music profile contains the title of the track, the description of the track, the genre of the track and a set of tags describing the music item. Table I shows an example of such a music profile. The music database, which was used in the experiments, consists of a set of manually selected music tracks (around 20 tracks per genre). The descriptions of classical music tracks have been taken from a dedicated web site (Classical Notes [7]), while the descriptions of rock music tracks were taken from Wikipedia [8]. The tags describing each music item have been assigned to them manually (see later).

Finally, the POIs are described by a POI profile containing: the name of the place of interest, the description the POI, a set of tags describing the POI, and the categories of the POI: art, architecture, history, nature. Table I gives an example of POI profile. 57 POIs in the city of Bolzano and its surroundings have been manually selected. The descriptions of the POIs have been taken from the local tourism web-sites [9], [10], [11].

The tags describing both music tracks and POIs were chosen from the ”List of Adjectives in American English” [12] (see Table II) and assigned to them manually through a separate tagging application. 38 adjectives were selected in such a way that each adjective could describe both a location and a music track.

In order to efficiently tag all the items in our database we have implemented a web application that allowed the users to tag POIs and music tracks. The application was used by around 30 people. On average, 11 tags were assigned to each POI and 13 tags to each music track. Some additional information on the tags' statistics is given in Section IV.

TABLE I
EXAMPLE OF MUSIC AND POI PROFILES

Music profile	**POI profile**
Name: Wagner - Tristan und Isolde	*Name*: Roncolo Castle
Genre: Classical	*Type*: Art, Architecture, History
Description: Tristan und Isolde (Tristan and Isolde) is an opera in three acts by Richard Wagner to a German libretto by the composer, based largely on the romance by Gottfried von Strazburg...	*Description*: First mentioned in 1237, Runkelstein castle is situated in a picturesque position on a massive mound of porphyry rock at the entrance to the Sarntal valley. It has become famous for its superb secular fresco cycles...
Tags: Beautiful(2), Big(1), Bright(1), Calm(1), Clear(1), Cold(1), Colorful(1), Dark(2), Fast(1), Gentle(1),...	*Tags*: Beautiful(1), Big(2), Calm(1), Cold(1), Dark(1), Dull(1), Gentle(1), Happy(1), Mysterious(5), Narrow(1),...

TABLE II
THE SET OF TAGS FOR DESCRIBING MUSIC AND POI

Sad	Happy	Wide	Narrow
Scary	Amusing	Flat	Uneven
Angry	Calm	Dark	Bright
Embarrassing	Pleasant	Dull	Colorful
Mysterious	Clear	Powerful	Feeble
Heavy	Light	Ugly	Beautiful
Fast	Slow	Cold	Warm
Big	Small	Harsh	Gentle
Loud	Silent	Noisy	Melodic
Old	Modern		

The tags assigned to POIs and music tracks by the users have not been post-processed. As a result, profiles of some items may contain contradicting tags. In the example in Table I we see tags ”Bright” and ”Dark” assigned to the same item. This is natural, since different users often have different perception of the same music track or POI. The tagging process was conducted some months before the itinerary recommendation experiment. Each user was offered a randomly selected POI item from our database of POIs. After tagging the POI, the user was given a randomly selected music track from or database which was followed by another random POI. The sequence continued until the user decided to logout or tagged all the items in our database. The users could choose the genre of the music tracks they want to tag. This was done to avoid forcing the users to tag music they do not like.

III. The Recommendation Process

In this work we evaluated three approaches for combining music and POIs that will be describe later. Here we start with describing music filtering and POI ranking since they are used in all the approaches. The goal of both music filtering and POI ranking steps is to reduce (filter) the set of POIs and music tracks before the two types of items can be paired by one of the POI and music recommendation algorithms. In fact, in all the three algorithms the recommendation process starts with the input of the user's ephemeral preferences, which are described by the tags entered by the user. Based on these ephemeral preferences and the user's profile data the ranking of the POIs and the filtering of music tracks is initially done separately.

Since we are not interested in the best quality single-domain recommendations, the music tracks are simply filtered by the user's preferred genre. POI ranking is more complex, and consists of measuring the overlap between the categories of POI and the user's interests as well as the tags entered by the user and the tags stored in the POI profile. The POI ranking algorithm takes as an input the user's profile data, the user's ephemeral preferences, the set of all POIs and the number of POIs to return (N). The result of the algorithm is the top N ranked POIs. The user's sightseeing preferences are defined in the user profile (a subset of {art, architecture, history, nature}) and the user's ephemeral preferences, $userTags$, are given as input by the user when requesting an itinerary recommendation. Only POIs that have at least one common category with the user's sightseeing preferences are considered for recommendation. The tags describing a POI, $POItags$, is a bag of words where tags may be repeated (if assigned to a POI by different users). The algorithm then computes the intersection of the bags $userTags$ and $POItags$. The cardinality of the intersection is stored as the score of POI.

Note that we take into account the repeating tags in POI profile, i.e. an intersection of the POI tags {sad, sad, sad, cold} with user's ephemeral preferences {sad, cold, dark} would give an intersection of 4 elements: {sad, sad, sad, cold}. This is done under the assumption that repeating tags are more important since more than one user associated a POI with these tags. Therefore, the presence of such important tags should be reflected in the POI score. After looping through all the POIs in the database, the algorithm sorts the ranked list according to the score of POIs and returns the top N POIs.

The outputs of both music filtering and POI ranking processes, i.e., a set of music tracks and a ranked list of POIs, are then combined in three possible ways:

- **POI rank.** In this approach, we assign to each of the top N POIs (from the ranked list of POIs) one music track chosen randomly from the set of filtered tracks. In this way the selected music is liked by the user but is not adapted to the POI. The order of the recommended POIs is determined by POI rank.
- **POI rank + music matching.** In this approach, after obtaining the ranked list of POIs, we assign the best-matching music track (from the filtered tracks) to each of the top N POIs. The music-to-POI similarity computation is described in Section III-A. Here, the order of recommended POIs remains identical to the first approach. Only the recommended music is different.
- **Music similarity rank.** In this approach, we do not simply assign music to the existing ranking of POIs, but rather combine the POI rank with the music-to-POI similarity score to produce a new ranking for the pairs of POIs and music tracks. We then recommend the top N pairs. The POI-music pair score is computed using the following formula:

$$score = (0.7 * POIpers) + (0.3 * similarity) \quad (1)$$

Here $POIpers$ is the POI score obtained from the POI ranking algorithm and $similarity$ is the music-to-POI similarity (see Section III-A).

The weights in formula 1 have been set after running experiments with different parameter values and analyzing the obtained itinerary recommendations. The aim of parameter tuning was to obtain itineraries that still reflected the user's ephemeral preferences (represented by $POIpers$), but also contain some of the highly matching music-POI pairs (represented by $similarity$). As assigning more weight to the $similarity$ parameter resulted in recommendations that did not reflect the user's preferences for the POIs, more weight was gradually shifted towards $POIpers$.

A. Computing Music-to-POI Similarity

The music-to-POI similarity scores are stored in a matrix that is computed off-line since it depends only on static item profiles. The similarity matrix S is a P x M matrix where P is the number of POIs in the database and M is the number of music tracks in the database. Each element of the matrix S_{ij} represents the similarity of music track j to POI i . Each similarity value is computed based on the profiles of the POI and the music track as follows:

$$similarity = (0.2 * tagOverlap) + (0.8 * textSc) \quad (2)$$

where $tagOverlap$ is the cardinality of the intersection of the tags in the POI profile with the tags in the music profile and $textSc$ is computed as follows:

$$textSc = \begin{cases} ldaSc, & \text{if } tfSc < 0.1 \\ tfSc, & \text{if } tfSc \geqslant 0.1 \end{cases} \quad (3)$$

In the above formula $ldaSc$ is the similarity of text descriptions in item profiles based on Latent Dirichlet Allocation *LDA* [13]. Using *LDA* we represented each document as a 100-dimensional vector of topic probabilities $d = (p_1, p_2, ..., p_{100})$, where p_i is the probability that the i^{th} topic is found in document d ($p_i \in [0, 1]$). The similarity between two text documents is then computed as the cosine distance between the two document vectors $d_1 = (p^1_1, ..., p^1_{100})$ and $d_2 = (p^2_1, ..., p^2_{100})$.

Likewise, $tfSc$ is the text similarity of the item profiles texts computed using classical *TF-IDF* representations [14]. Each document is represented by a term vector $d = (t_1, t_2, ..., t_n)$, where t_i is the weight of the i^{th} term in the document d and n is the number of distinct significant terms found in the document corpus (5595 in our case). The similarity between two text documents is then computed as the cosine distance between the two vectors $d_1 = (t^1_1, ..., t^1_n)$ and $d_2 = (t^2_1, ..., t^2_n)$.

Experiments have shown that the two text processing algorithms (*LDA* and *TF-IDF*) produce different results depending on the type of text documents. *TF-IDF* can give good results when two documents are linked by highly weighted keywords. However, such situation is not common in the dataset used in this research due to different nature of the described objects. POIs and music descriptions typically lack direct links between themselves. In such cases *LDA* algorithm performs better than *TF-IDF*. The $tfSc$ threshold in formula 3 was therefore set by comparing *TF-IDF* similarity values against human judgment of document relevance for our data corpus.

The weights in formula 2 as well as the number of topics in *LDA* computation have been established empirically. Due to the high cost of live-users experiments these parameters have not been optimized. More extensive testing and parameter tuning is a topic of future work.

IV. Evaluation and Results

In order to validate our approach we implemented a web application that offered the user itinerary recommendations for the city of Bolzano and its surroundings. An itinerary was composed of three places of interest inside and around the city (monuments, buildings, nature parks etc.) and a soundtrack (one music track per POI).

The human/computer interaction with the evaluation application consisted of three main parts: filling out the profile information (registration), viewing two recommended itineraries and providing a feedback, i.e., choosing the preferred itinerary.

Before using the system the users registered providing their age, gender, country, preferred music genre, sightseeing interests and the level of knowledge about Bolzano city. In the starting page, the user was asked to enter the ephemeral sightseeing preferences - a few tags describing the POIs that she would like to see. The tags were chosen from a finite set of tags that was previously used to tag both POIs and music tracks. After specifying the preferences for POIs, the user was given a first recommended itinerary. The suggested sequence of POIs was displayed, one POI after the other, and the accompanying music tracks were played. Having viewed the first itinerary, the user was offered a second itinerary that satisfied the same ephemeral preferences but was computed with a different version of the recommendation algorithm (see Section III).

Finally, after viewing the two itineraries, the user was asked to choose the itinerary that she liked the best. The users were not aware of the different modes used in the recommendation process. Therefore, recording the user choices and analyzing the data allowed us to compare the user satisfaction with different recommendation techniques and to analyze the effect of music matching on the quality of recommendations.

The users participated in the evaluation on a voluntary basis. Invitations to participate have been sent to research colleagues as well as readers of the user-modeling mailing list (um@unito.it). A total of 53 users used the system. 13 users used the system 3 times and more. 40 users used the system once or twice. In total, 115 evaluation sessions were performed.

In our study we compared the number of times each recommendation algorithm was preferred by the users in the evaluations. Figure 2 shows the selection probability of the three algorithms considering all session data together, and separately sessions where classical or rock music were the preferred genre of the user. Selection probability is estimated as the ratio of times an itinerary with a soundtrack generated by one method was selected over the total number of times it was offered in one of the two suggested itineraries in a recommendation session. In this figure, *POI rank*, *POI rank + matching*, and *music similarity rank* are the different versions of recommendation algorithm.

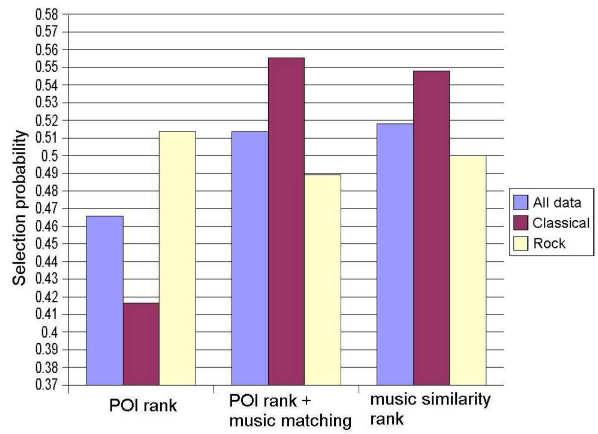

Fig. 2. Selection probability of different algorithms

Considering all session data we see that both *POI rank + matching* and *music similarity rank* approaches are preferred more often than the recommendations where music is assigned to POI without matching. This data supports our main hypothesis that the process of music matching to POI is beneficial and produces more satisfying recommendations. We note that even in the simpler algorithm, *POI rank*, the played music was selected among tracks of the user's preferred genre. So the played music was among the preferred type in all the three cases.

The results do not show a clear preference of the users for one of the two approaches: *music similarity rank* and *POI rank + matching*. In fact, these approaches have similar selection frequency. While in the *POI rank + matching* approach the recommended itineraries are primarily composed by ranking the POIs and only then assigning music track to each POI, the *music similarity rank* approach focuses rather on recommending pairs where POI and music are highly matching to each other. Therefore, we conclude that selecting highly matching pairs of POI and music may, to some extent, compensate the lack of a precise POI personalization.

The results for session data collected separately for each music genre show that while classical music listeners clearly preferred the advanced recommendation techniques, rock music lovers did not show any preference among the different approaches. This could mean that even the simpler approach, i.e., assigning a generic track in the user's preferred genre can be suitable for certain music genres. The results suggest that the effectiveness of the two proposed approaches for matching music with POI depends on the genre of music preferred by the system users. Genre dependencies in music information retrieval have been identified in previous works. For instance, when classifying music into emotional categories Li and Ogihara [15] obtained better results within certain music genres (classical and fusion music).

The dependence of the algorithm performance on music genre could be explained by the fact that classical music is much more diverse (in tone, melody, rhythm, mood and tempo changes) compared to rock music. Therefore, classical tracks can be easier for the users to tag with a wider variety of tags which makes the matching procedure more effective. This is backed up by the fact that out of more than 2000 possible pairs of music and POIs in our database the top 20 pairs contain only classical music. In order to investigate the possible reasons for this, we have checked the tag distribution for music items in our database. Figure 3 shows the distribution of tags assigned to music of different genres by the users. We computed the probability of each tag usage as $\frac{tags}{totalTags}$, where $tags$ is the number of times a tag was assigned to an item and $totalTags$ is the total number of tags assigned to the items in a collection (either classical or rock). The figure 3 shows that the tag distribution of the classical music tracks is more uniform (flat). In other words, the users used more diverse tags while tagging classical music tracks.

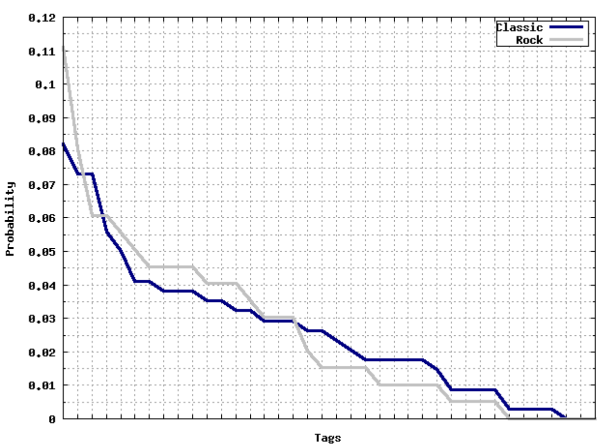

Fig. 3. Tags usage probability among the different music genres

Interestingly, during the tagging process rock music received less tags compared to classical music tracks (although similar number of users were tagging both types of music). We have collected on average 16.24 tags for each classical track and 9.9 tags for each rock track in our database. Moreover, during a single tagging session, a user tagging rock music assigned to a track an average of 4.3 tags. Meanwhile, a classical music listener assigned to a track an average of 5.68 tags. Conversely, we have collected on average 11.18 tags for each POI in our database. A user tagging POIs assigned an average of 3.79 tags to a POI.

We have also checked which types of tags were used when tagging POIs and music. We have identified 3 general categories for the 38 adjectives used in the tagging process: emotional, sound quality and physical quality adjectives. We compared the frequency of tags of each type in POIs and music respectively. The frequency of each tag type was computed as a ratio of tags of this type in POI/music profiles over the total number of tags of this type in our database. Figure 4 shows that POIs are mostly described by physical quality adjectives while music is most often described by sound quality adjectives, as expected. Conversely, emotional tags are almost equally distributed between POIs and music. Therefore, we conjecture that emotional tags could allow a more efficient music-to-POI similarity computation. This is an hypothesis that we want to test in the future.

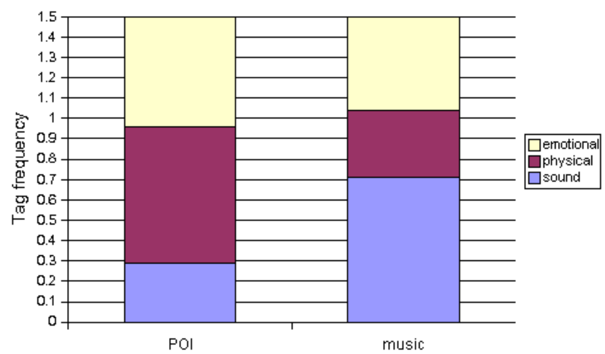

Fig. 4. Tags usage frequency in POIs and music

In fact, there have been already some attempts to use emotional labels for music classification [16]. Several works have studied and selected a restricted set of emotional labels that can describe music. Hevner [17] suggested 8 clusters of descriptive adjectives that were revised and expanded by Farnsworth [18]. A more recent and elaborated research on emotions evoked by music was carried out by Zentner et al. [19]. The proposed Geneva Emotional Music Scale (GEMS) consists of 9 emotional clusters with 4-5 adjectives in each group. We plan to use this model for re-tagging the items before further experiments.

V. Conclusions and Future Work

In this paper we have presented a matching technology for building multimedia and cross-domain recommender systems. We have validated two hypotheses: a) user satisfaction for a POIs recommendations can be increased by accompanying POIs with music tracks that are matching the user's profile and the POIs; and b) simultaneously selecting best-matching pairs of POIs and music tracks can provide as satisfying recommendations as those computed when first personalized POIs recommendations are selected and then the music is matched to each of these POIs. In fact, the general result of our live users experiments is that users more often prefer the itineraries with soundtrack music matching to POIs.

In particular, we have shown that the approach based on simultaneously selecting music and POI provides similar results to the approach where the music is matched to pre-selected

POIs. This could mean that the high similarity between a music track and a POI can compensate the lack of POI personalization. However, such results can also be caused by the relatively small set of music tracks, which resulted in similar music tracks recommended in the two approaches. Hence, this is an issue that deserves some more study and experimental tests.

Furthermore, the evaluation results showed that the performance of the implemented algorithms depends on the music genre preferred by the users. Our system offered the users two musical genres: classical and rock music. Recommendation sessions where classical music was chosen tend to give better results with the two music matching techniques that we proposed. While such results could have been determined by the relatively small size of the used data set, we believe that this is influenced by the nature of music genres; rock music being more homogeneous and difficult to tag.

This research is part of an ongoing work on tag-based similarity techniques for heterogeneous objects. Future work includes fixing some of the limitations of the current approach and performing a more extensive user study with more users, revised POIs and more music tracks. We observe that user studies in recommender systems are rare and most of the validations have been performed off-line; cross validating the accuracy of the recommendation in a given data set [20], [21]. So, our work is still among the few attempts to evaluate the recommendations with real users' feedback rather than trying to reproduce the past user evaluations on recommended items.

Before conducting new user studies we intend to revise the used data set. We will focus on classical music, since rock music did not perform well in the experiment described in this paper. Another potentially useful music genre could be movie soundtracks, since these tracks typically carry a strong emotional message and could be easier to tag. We also intend to group the POIs in a small number of categories, and to tag the categories rather than the instances. This should make the categories of POIs more unique and distinctive, and we conjecture that different categories will receive different tags. Most importantly, we are revising the set of adjectives used for tagging music and POIs. We are considering using the emotional adjectives from the GEMS model [19].

With respect to the algorithms used in the proposed approach, we note that the computation of the matching of two items as the intersection of the tags assigned to them should perform better when it is normalized by the number of tags present in the items' profiles. Furthermore, it is important to perform additional testing (cross validation) for tuning the different weights used in the matching formulas. We are also considering some newly proposed similarity functions for tagged objects that have been already validated in other applications [22].

We conclude by observing that the described technique could be used in many applications that requires the personalized presentation of POIs, e.g. in tourism web-sites, and electronic guides. They can also be applied in mobile applications for personalizing the travel experience with ad hoc music adapted to the location of the user. In particular, we believe that the proposed techniques can be very useful to filter and personalize stream audio content.

References

[1] S. Berkovsky, T. Kuflik, and F. Ricci, "Mediation of user models for enhanced personalization in recommender systems," *User Modeling and User-Adapted Interaction*, vol. 18, no. 3, pp. 245–286, 2008.

[2] H.-S. Park, J.-O. Yoo, and S.-B. Cho, "S.b.: A context-aware music recommendation system using fuzzy bayesian networks with utility theory," in *FSKD 2006. LNCS (LNAI)*. Springer, 2006, pp. 970–979.

[3] A. Ranganathan and R. H. Campbell, "A middleware for context-aware agents in ubiquitous computing environments," in *Middleware*, ser. Lecture Notes in Computer Science, M. Endler and D. C. Schmidt, Eds., vol. 2672. Springer, 2003, pp. 143–161.

[4] S. Reddy and J. Mascia, "Lifetrak: music in tune with your life," in *HCM '06: Proceedings of the 1st ACM international workshop on Human-centered multimedia*. New York, NY, USA: ACM, 2006, pp. 25–34.

[5] L. Lu, D. Liu, and H.-J. Zhang, "Automatic mood detection and tracking of music audio signals," *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 14, no. 1, pp. 5–18, Jan. 2006.

[6] R. Cai, C. Zhang, C. Wang, L. Zhang, and W.-Y. Ma, "Musicsense: contextual music recommendation using emotional allocation modeling," in *MULTIMEDIA '07: Proceedings of the 15th international conference on Multimedia*. New York, NY, USA: ACM, 2007, pp. 553–556.

[7] P. Gutmann, "Classical notes," http://www.classicalnotes.net/.

[8] "Wikipedia," http://en.wikipedia.org/.

[9] "The official website of south tyrol," http://www.suedtirol.info/.

[10] "Eppan tourism website," http://www.eppan.com/.

[11] "Ritten tourism website," http://www.ritten.com/.

[12] P. Noll and B. Noll, "List of adjectives in american english," http://www.paulnoll.com/Books/Clear-English/English-adjectives-1.html.

[13] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation," *J. Mach. Learn. Res.*, vol. 3, pp. 993–1022, 2003.

[14] G. Salton, *Automatic Text Porcessing: The transformation, analysis, and retrieval of information by computer*. Addison-Wesley, 1989.

[15] T. Li and M. Ogihara, "Detecting emotion in music," in *Proceedings of the fourth international conference on music information retrieval (ISMIR, 2003)*, Baltimore, USA, October 2003. [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.10.8122

[16] D. Baum and A. Rauber, "Emotional descriptors for map-based access to music libraries," in *Proceedings of the 9th International Conference on Asian Digital Libraries*, November 2006, pp. 370–379.

[17] K. Hevner, "Experimental studies of the elements of expression in music," *The American Journal of Psychology*, vol. 48, no. 2, pp. 246–268, 1936. [Online]. Available: http://www.jstor.org/stable/1415746

[18] P. R. Farnsworth, *The social psychology of music*. Iowa State University Press, 1958.

[19] M. Zentner, D. Grandjean, and K. R. Scherer, "Emotions evoked by the sound of music: Characterization, classification, and measurement," *Emotion*, vol. 8, no. 4, pp. 494–521, August 2008. [Online]. Available: http://dx.doi.org/10.1037/1528-3542.8.4.494

[20] G. Adomavicius and A. Tuzhilin, "Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions," *IEEE Transactions on Knowledge and Data Engineering*, vol. 17, no. 6, pp. 734–749, 2005.

[21] J. L. Herlocker, J. A. Konstan, L. G. Terveen, and J. T. Riedl, "Evaluating collaborative filtering recommender systems," *ACM Transaction on Information Systems*, vol. 22, no. 1, pp. 5–53, January 2004.

[22] B. Markines, C. Cattuto, F. Menczer, D. Benz, A. Hotho, and G. Stumme, "Evaluating similarity measures for emergent semantics of social tagging," in *WWW '09: Proc. of the 18th international conference on World wide web*. ACM, 2009, pp. 641–650.