Knowledge-based Music Retrieval for Places of Interest

Marius Kaminskas[1], Ignacio Fernández-Tobías[2], Francesco Ricci[1], Iván Cantador[2]

[1]Faculty of Computer Science
Free University of Bozen-Bolzano
39100 Bolzano, Italy
{mkaminskas, fricci}@unibz.it

[2]Escuela Politécnica Superior
Universidad Autónoma de Madrid
28049 Madrid, Spain
{i.fernandez, ivan.cantador}@uam.es

ABSTRACT

In this paper we address a particular recommendation task: retrieving musicians suited for a place of interest (POI). We present a knowledge-based framework built upon the DBpedia ontology linking items from different domains. Graph-based algorithms are used for ranking and filtering items in a target domain (music) with respect to their relatedness to an input item in a source domain (POIs). By conducting user studies we found that users appreciate and judge more valuable the suggestions generated by the proposed approach when a novel weight spreading activation algorithm is used to compute the matching between musicians and POIs. Moreover, users perceive compositions of the suggested musicians as suited for the POIs.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval—*information filtering*; I.2.4 [**Artificial Intelligence**]: Knowledge Representation—*semantic networks*

General Terms

Algorithms, Experimentation, Human Factors

Keywords

Music recommender systems, context-aware recommendation, semantic networks, Linked Data, DBpedia

1. INTRODUCTION

Music recommender systems are decision support tools that reduce information overload by retrieving relevant music items based on a user's profile [5]. However, most of the available music recommenders suggest music without taking into consideration the user's context, e.g., her mood, current location, activity, or any other contextual condition that might influence the user's perception or evaluation of music [1]. In response to these observations, in recent years a new research topic of contextual music retrieval and recommendation has emerged [9]. In this work we are addressing a particular type of context-aware music recommendation – recommending music for places of interest (POIs). Finding music items that suit POIs can be exploited in a number of engaging information services. For instance, a mobile city guide providing an enhanced presentation of the place visited by the tourist, and playing music that is related to the place, i.e., music that is culturally or emotionally associated to the place (e.g. Mozart in Salzburg, or a Bach fugue in a Gothic Cathedral). Another example is a car entertainment and navigation system that adapts music to the place the car is passing by.

The main challenge that one must face when addressing the above mentioned goal is related to the fact that POIs and music are two rather different domains, and there is no obvious way to match such heterogeneous items. However, with the advent of the Semantic Web, and specifically with its reference implementation in the Linked Data initiative [4], new opportunities arise to face the above difficulties. In this paper we propose to exploit DBpedia [3] – the Linked Data version of Wikipedia – for building a framework in which focused semantic networks linking items belonging to some selected domains are automatically created. Over these networks we propose to use a graph-based spreading activation algorithm to rank and filter the items in the target domain (music) that are most related with certain given items in the source domain (POI).

In a previous work [8] we showed that there exist latent similarities between the two domains – POIs and music – based on the emotions that users perceive when visiting a place and listening to music. We used tags describing the emotional characteristics of POIs and music tracks to establish a match between these items such that users appreciated and perceived it as meaningful. From that result, and aiming to complement the tag-based recommendation model proposed in [8], the framework and recommendation models presented herein enable computing matching between places and musicians based on their semantic relatedness. We hypothesize that given certain musicians related to a particular place of interest, recommendations of compositions from such musicians will also fit that place. In this context, we specifically aim to address the following research questions.

RQ1: how to automatically identify (by using DBpedia) musicians semantically related to a given POI? **RQ2**: how to evaluate the semantic relatedness between POIs and musicians?

By conducting user studies we found that users appreciate and judge more valuable the suggestions generated by the proposed approach when a novel weight spreading activation algorithm is used to compute the matching between POIs and musicians. We also found that our approach is able to characterize differently relevant and non-relevant musicians with respectively large and small number of considered semantic relations, and moreover, that users perceive compositions of the suggested musicians as suited for the POIs.

2. RELATED WORK

Finding music that suits a POI can be viewed as a context-aware recommendation problem, the place being the context for consuming the recommendation (music) [1]. There have been several works on context-aware music recommendation. Ankolekar and Sandholm [2] presented a mobile audio application that plays audio content associated with a particular location with the goal of enhancing the sense of being in a place by creating its emotional atmosphere. Instead of establishing semantic relations between music and location, the presented approach relies on crowd-sourcing – users of the system are allowed to assign audio piece (either a music track or a sound clip) to a specific location (represented by the geographic coordinates). Stupar and Michel [14] described an approach to select music tracks for a given image. They rely on associations between the low-level music and image features mined from movies and their soundtracks. Aiming to provide contextualized recommendations of music compositions when the user is visiting tourist attractions, Kaminskas and Ricci [8] showed that emotional tags assigned to both types of items can be used to effectively select music content that fits a POI.

There have been attempts to establish semantic relations between items of different types. Loizou [10] proposed to identify explicit semantic relations between items, and exploit such relations for cross-domain recommendation. Specifically, items were annotated and linked by concepts and properties extracted from Wikipedia. Then, with such relations, users and items were incorporated into a graph, upon which a probabilistic recommendation model was built. Passant [12] developed dbrec – a system built upon DBpedia that computes semantic distances between concepts to recommend related music bands and solo artists.

As in [10], in this paper we exploit Wikipedia as a source of multi-domain information, but we use Semantic Web technologies and Linked Data repositories to automatically build semantic networks interconnecting concepts from various domains – namely Architecture, Art, History, and Music. In previous work [6] we proposed a semantic-based framework that aims at extracting and aggregating DBpedia concepts and relations between two different domains, but we did not evaluate such approach. In this paper we extend the work presented in [6] by a) developing a more efficient method to extract cross-domain information from DBpedia, b) finding richer semantic relations between Architecture and Music concepts, and c) conducting two user studies to evaluate the semantic relatedness between automatically matched POIs and musicians.

3. KNOWLEDGE-BASED FRAMEWORK

Our recommendation framework is built upon an ontology-based knowledge representation model in the form of a graph/network of semantic entities (concepts) in different domains, and interlinked by semantic relations (properties). Entities can be roughly categorized as classes and instances. Classes are types or categories of concepts, such as 'city', while instances are particular members of classes, such as 'Madrid'. Moreover, semantic relations can link classes (e.g. a city 'belongs to' a country), instances (e.g. Madrid 'is the capital of' Spain), or both types of entities (e.g. Madrid 'is' a city). Links can express hierarchical relationships, e.g. 'subclass of' and 'instance of', or have an arbitrary meaning.

Our final goal is to automatically find paths in this semantic graph between an instance in a source domain (a POI) to instances in a target domain (musicians), and select (recommend) some of the reached target instances according to certain criteria.

Our approach consists of three main stages. In the first stage we restrict the subspace of DBpedia by identifying classes belonging to the two domains of interest, and the relations existing between instances of such classes. We then build a network consisting of a directed acyclic graph (DAG) whose nodes represent the identified classes, and edges represent the selected relations. In this graph there is a target node that does not have out edges, and corresponds to the class whose instances will be recommended. This stage is detailed in Section 3.1. In a second stage we instantiate the built framework into an instance network for a particular source instance (e.g. 'Vienna State Opera' in Figure 1). In practice, we query DBpedia to obtain instances related to the source instance according to the classes and relations defined in the above-mentioned graph. The retrieved instances are then incorporated into the network, and are used to query DBpedia for additional related instances. This process is repeated iteratively until all the identifiable instances of the target class are reached. This stage is described in Section 3.2. Finally, in the third stage we assign weights to the nodes (instances) and edges of the instance network that was created in this process, and perform a graph-based ranking algorithm over the network to identify (and recommend) the target instances with the highest score values. This stage is explained in Section 3.3.

3.1 Class Network

As we mentioned above, the output of the first stage consists of a DAG that describes how the classes in two domains of interest – source and target – are linked by means of semantic relations available in the used knowledge repositories (DBpedia). In this graph, nodes represent classes, and edges – relations between classes. Moreover, there is a target node without exiting edges that corresponds to the class of the instances to recommend. The selection of relevant classes and relations is guided by experts of the domains of interest and knowledge repositories. The source and target domains are respectively characterized by POIs and musicians, so 'POI' and 'Musician' classes represent the starting and ending nodes in the class network of our framework. Analyzing DBpedia, we identify three potential types of semantic paths from POIs to musicians:

- *Location paths*. A particular POI may be linked to musicians who were born, lived, or died in the city of

the POI. For instance, Arnold Schoenberg *was born in* Vienna, which is the city where Vienna State Opera *is located.*

- *Time paths.* A POI may be linked to musicians who were born, lived, or died in the same time period (e.g. year, decade, century) when the POI was built or opened. For instance, Gustav Mahler *was born in* 1869, the same decade when Vienna State Opera *was built.*

- *Category paths.* A POI may have Architecture categories (e.g., architectural styles and eras, building types) that are related to Music categories (e.g. music genres and eras, musician types), through relations with History and Art categories. In this way, musicians with matched Music categories could be linked to the input POI. For instance, Wolfgang Amadeus Mozart is a *classical music composer*, and classical compositions are played in *Opera houses*, which is the building type of the Vienna State Opera.

In our framework we can assign relevance values to the considered semantic entities and relations, which may be used in the retrieval stage. These values may be assigned by the domain experts, or could come from the user's profile. For instance, a domain expert may assign higher relevance to the class 'City' than to the class 'Building type', since the former can be considered more informative to link a particular POI with related musicians. Similarly, specific concepts like 'Opera composers' and 'Classical music' may receive high relevance if the user has a clear preference for them, hence producing personalized associations. Moreover, we consider the case in which relations also receive relevance values, e.g. to measure the strength with which 'Art Deco' architectural style and 'Swing' music genre (both emerged in the 1930s) are related with respect to other more/less related categories.

Formally, let $\mathcal{E} = \mathcal{C} \cap \mathcal{I}$ be the set of class and instance entities. We assume that a function $rel_e : \mathcal{E} \to \mathbb{R}_+$, with $\mathbb{R}_+ = \{x \in \mathbb{R} : x \geq 0\}$ is defined, representing the relevance value assigned to entities $e \in \mathcal{E}$. We also assume that a function $rel_r : \mathcal{R} \to \mathbb{R}_+$ representing the relevance value assigned to relations $r \in \mathcal{R} \subseteq \mathcal{E} \times \mathcal{E}$ between pairs of entities is defined. Given these functions we are interested in computing a scoring function (see next sections) that measures to what extent an instance is related to another one when they are not directly connected by a relation. For simplicity, in the framework implementation described here we set the above relevance values to 1, and leave for future work the investigation of strategies to establish distinct relevance values.

3.2 Instance Network

By exploring the structured data repository (i.e., DBpedia) through the classes and relations established in the class network (Section 3.1), we build a new network that links a particular instance in the source domain with related instances in the target domain. Figure 1 shows part of a sub-network that links the POI 'Vienna State Opera' and the musician 'Gustav Mahler'. The full network of a POI is obtained by aggregating all the sub-networks linking the POI to semantically related musicians.

An instance network has weights assigned to the relations between pairs of instances. These weights are computed

from the relevance values of the relations linking the instance pairs and their classes. Specifically, we define the weight of a relation between two instances as a function $w : \mathcal{I} \times \mathcal{I} \to \mathbb{R}_+$ that depends on the relevance values of the connections between each pair of instances $I, I' \in \mathcal{I}$, and between their classes $C_I, C_{I'} \in \mathcal{C}$:

$$w(I, I') = f(rel_r(I, I'), rel_r(C_I, C_{I'}))$$

where $w(I, I') = 0$ if there is no link from I to I'.

In the experiments presented in this paper, f is a linear combination of the relevance values assigned to the semantic connections between I and I', and between C_I and $C_{I'}$:

$$w(I, I') = \alpha \cdot rel_r(I, I') + (1 - \alpha) \cdot rel_r(C_I, C_{I'}), \alpha \in [0, 1]$$

To test the approach, in the experiments we set the relevance values to 1 and α to 0.5, i.e., the weights were set to 1.
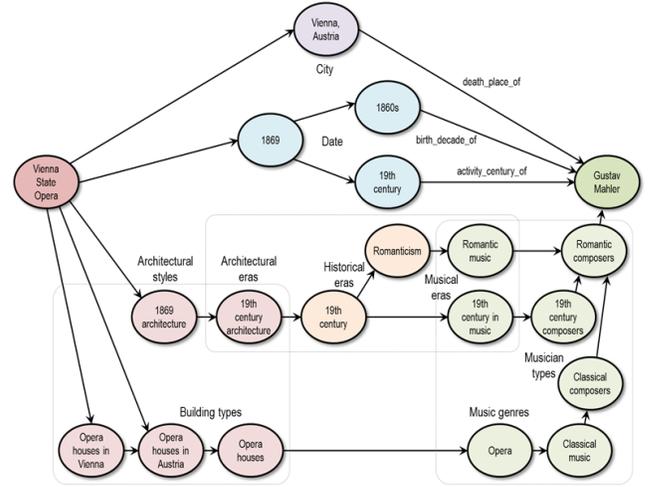

Figure 1: An instantiation of the semantic network linking Vienna State Opera with a related musician – Gustav Mahler.

3.3 Graph-based Ranking Algorithm

In the final stage we run a graph-based ranking algorithm on the built instance network. The algorithm computes a relevance score $s(I)$ for each entity node $I \in \mathcal{I}$ in the network by following a weight spreading strategy, and provides a scoring of the target nodes (i.e., those belonging to the class 'Musician') and the highest scored nodes are selected for recommendation. Hence, the score of an instance node I is a function $s : \mathcal{I} \to \mathbb{R}_+$ that not only depends on the relevance values of the instance and its class C_I, but also on the scores of connected instances $I_1, ..., I_k$ that are connected to I along some path in the semantic network, and the weights of their relations $w(I_1, I), ..., w(I_k, I), w(I, I_1), ..., w(I, I_k)$, that is:

$$s(I) = g(rel_e(I), rel_e(C_I); s(I_1), ..., s(I_k); w(I_1, I), ..., w(I, I_k))$$

The initial score values of the nodes are set based on the relevance values $rel_e(I)$ and $rel_e(C_I)$, which, as explained in Section 3.1, may be related to domain characteristics and/or user preferences. Then our Constrained Spreading Activation algorithm propagates the initial score of the source node through its weighted edges, updating the scores of its linked nodes. This is iteratively done for subsequent linked

nodes until reaching the target nodes, whose scores cannot be further propagated because they do not have out edges. Specifically, the graph nodes are sorted by graph topological ordering $(I_1, ..., I_K)$ to assure that every node receives the influence from final scores of its predecessors in the graph. Following this ordering, each instance $I = I_k \in \mathcal{I}$ is assigned a score based on the scores of predecessor connected instances in $\{I_1, ..., I_{k-1}\}$, the weights of the connections $w(I_1, I), ..., w(I_{k-1}, I)$, and the relevance of the instance's class C_I:

$$s(I) = s(I_k) = \beta \cdot rel_e(C_I) + (1-\beta) \cdot \sum_{i=1}^{k-1} s(I_i) w(I_i, I), \beta \in [0,1]$$

4. EXPERIMENTS

We have conducted two user studies aimed to evaluate how the users perceive and judge the musician recommendations provided by our approach. For the evaluation dataset we have extracted information on 25 POIs from 17 major European cities. For each POI we obtained the top 5 ranked musicians using the spreading activation algorithm and two additional baseline algorithms – HITS and PageRank [11]. Moreover, 5 musicians were randomly selected from the whole set of musicians belonging to the POI's semantic network, as an additional baseline method to compare with our approach. We have also downloaded a representative music composition for each musician. So in total, we had 4 methods to recommend a musician (and its music composition) for any given POI.

The first experiment was designed to evaluate how the users perceive the semantic relations between POIs and retrieved musicians, while the second experiment was performed to evaluate if the users perceive compositions performed by the retrieved musicians as relevant for the POIs.

4.1 Knowledge Evaluation Study

The interface for the first user study had to be carefully designed since assessing the quality of matching pairs of items from different domains is not easy, and requires a considerable user effort. Hence, we designed a tailored interface (Figure 2) that requires some considerable user effort, but can collect important and interesting information about the user-perceived quality of different musician recommendations for POIs.

During an evaluation session a user was presented with a sequence of 10 pairs of POIs and musicians, where the musicians were obtained using one of the four recommendation methods. The information describing each POI and musician pair was presented in a structured way, according to the representation in the graph model – location, date, and category were clearly separated. The user was asked to carefully check the presented information, and assess whether the musician was actually related to the POI, and if yes to specify which parts of the structured musician information were contributing, and in which degree, to the match (right part of the system's interface, in Figure 2). We aimed to understand which types of semantic paths linking a POI to a musician (i.e., location, date, or category paths) contributed more to the match and were better appreciated by the users.

4.2 Music Evaluation Study

For the second study we designed a simpler interface (Figure 3). During a single evaluation session a user was presented with a POI and a list of compositions of the musicians selected by each of the four methods. The order of the compositions was randomized, and the user was not aware of the algorithms that were used. The user was asked to read the description of the POI, listen to the compositions, and select those that in her opinion suited the POI.

Moreover, in this experiment we asked the users to enter their genre preferences prior to performing the evaluation. The genre taxonomy was selected based on the musicians in our dataset, and included Classical, Pop, Medieval, Opera, Rock, Ambient, Folk, Hip Hop, Metal, and Electronic music.
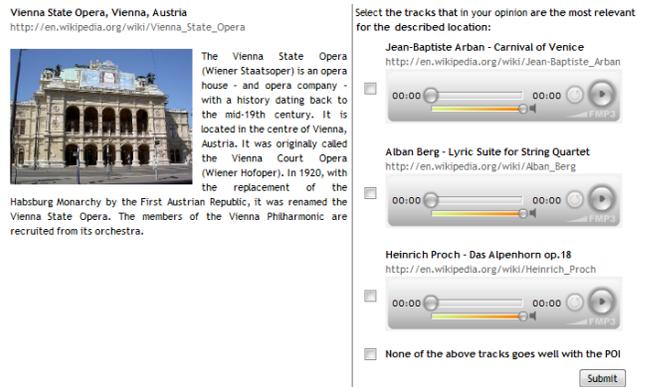

Figure 3: Screenshot of the system used in the Music Evaluation study.

We note that in this study, the users were not asked to evaluate the semantic relatedness of POIs and musicians. Instead, they were asked to judge whether the music composed/performed by these musicians would be fit to listen to when visiting the POIs. While the users had the possibility to follow the link to musicians' Wikipedia pages, we observed that in the majority of cases, the users did not click on the link, therefore basing their judgment solely on the audio track.

5. RESULTS

5.1 Knowledge Evaluation Results

A total of 97 users participated in the study. They were PhD students and academic staff recruited via email, and covered an ample spectrum of ages and nationalities. They provided 1155 assessments for 356 distinct POI-musician pairs (note that a musician may appear in the top ranked list of various algorithms). Among these 356 distinct pairs, 285 were assessed by 3 users, and the remaining pairs were assessed by 4 or more users. The Fleiss' Kappa correlation coefficient of the relatedness assessments per POI was 0.675, meaning a substantial agreement among users. In order to understand which semantic musician information contributed, and in which degree, to the matches, Figure 4 shows the average numbers of semantic paths in the instance network between the input POIs and the retrieved musicians. The higher these numbers, the more semantic relations between POIs and musicians, and thus the more semantically related the retrieved POI-musician pairs.

Based on the obtained results, we can highlight two aspects of the spreading algorithm. First, differently from
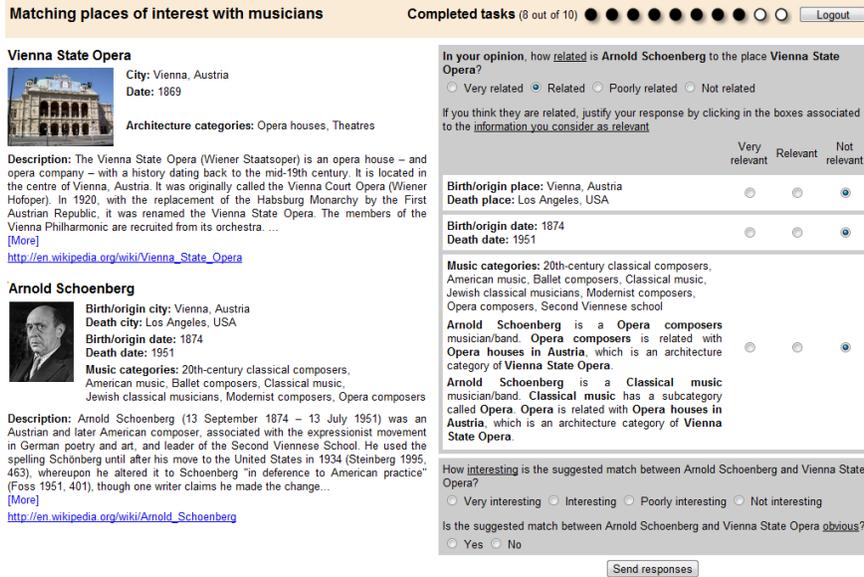
Figure 2: Screenshot of the system used in the Knowledge Evaluation study.

	P@1	P@2	P@3	P@4	P@5
Random	0.355*	0.391*	0.363*	0.435*	0.413*
HITS	0.688	0.706	0.711*	0.700*	0.694
PageRank	0.753	0.728	0.707*	0.660*	0.646*
Spreading	0.810	0.804	0.828	0.847	0.837

Table 1: Average precision values obtained for the top 5 ranked musicians for each POI. The values marked with * have differences statistically significant (Wilcoxon signed-rank test, p<0.05) with Spreading algorithm's.

the other two algorithms, it differentiates relevant and non-relevant musicians by finding a larger number of paths between a POI and a relevant musician compared with a non relevant one. Second, it uses in a balanced way all the different types of the considered relations.
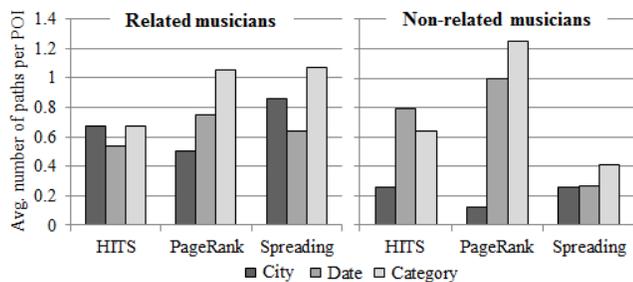

Figure 4: Avg. number of semantic paths per POI.

5.2 Music Evaluation Results

A total of 61 user participated in the second evaluation study. As in the first case, they were PhD students and academic staff recruited via email (some of these users also participated in the previous evaluation study). 1125 evalu-

ation sessions were performed (i.e., a POI shown to a user), and 1258 tracks were selected by the users as well-suited for a POI. Figure 5 shows the performance of the recommendation methods, computed as the ratio of the number of times a track produced by each method was considered as well-suited over the total number of evaluation sessions (1125). All methods perform significantly better than the random track selection ($p < 0.01$ in a two-proportion z-test). Moreover, the weight spreading activation outperforms the others ($p < 0.01$). These results are in accordance with the "Knowledge Evaluation" study.
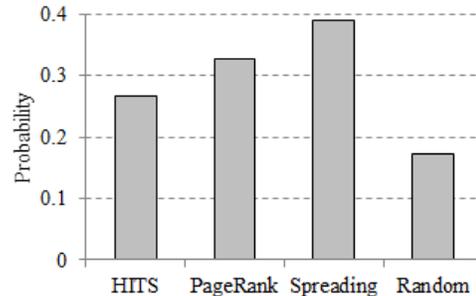

Figure 5: Selection probability of the recommendation approaches.

We have also analyzed the influence of users' genre preferences on the music tracks that they selected as well-suited for the POIs. In the following $TrkOK$ represents the condition that a track was marked by a user as well-suited for a POI, $GTrk = g$ – the condition that the genre of a track is g, $g \in uPref$ – the condition that a user has included the genre g in her genre preferences, and $g \notin uPref$ – the condition that a user has not included the genre in her preferences.

As a baseline for the analysis we computed, for each genre, the probability that a music track is of genre g, given that it was marked as suited for a POI – $P(GTrk = g|TrkOK)$ – as

the ratio of the number of tracks of genre g selected as well-suited for the POIs over the total number of tracks selected as suited for POIs (1258). Then, to check the deviation from this baseline produced by the users' preferences, we measured $P(GTrk = g|TrkOK, g \in uPref)$ and $P(GTrk = g|TrkOK, g \notin uPref)$.

We found that for Classical, Medieval and Opera music genres the deviations from the baseline are significant ($p < 0.01$ in a two-proportion z-test). For Classical music, the probability that a user who likes this genre will mark a classical track as well-suited for a POI is higher, and in the opposite case – lower. For other music genres the deviations from the baseline probabilities were not significant.

We have also measured the conditional probability for a track to be selected as suited for a POI, given that its genre is $g - P(TrkOK|GTrk = g)$ – as the ratio of the number of tracks of genre g selected as suited over the total number of times a track of genre g was displayed during the evaluation. The results showed that Medieval and Opera tracks are most often selected by the users (0.49 probability), followed by Ambient (0.36) and Classical (0.35) music. This can be explained by the fact that the POIs in our dataset are architecture objects (castles, monuments, cathedrals, etc.) which are easier associated with the above genres.

To further confirm the influence of genre preferences, we measured the probabilities $P(TrkOK|GTrk = g, g \in uPref)$ and $P(TrkOK|GTrk = g, g \notin uPref)$. From these probabilities the effect of user preferences is evident for Classical and Medieval music – the difference between the $g \in uPrefs$ and $g \notin uPrefs$ conditions are significant with $p < 0.05$.

We can thus confirm that for certain genres there are clear effects of user preferences on the decision for considering music to go well with a POI. Therefore, in the future we intend to take the users' music preferences into account.

6. CONCLUSIONS AND FUTURE WORK

The high precision values (over 80% for top 5 results in Table 1) obtained by the evaluated recommendations let us claim that our approach is able to automatically identify musicians semantically related to a given place of interest (**RQ1**). The user studies also showed that, in our approach, musicians evaluated as relevant tend to have relative high numbers of paths in a POI's semantic network for all the proposed types of relations (i.e., city, date and category), while these numbers are significantly lower for non-relevant musicians. This evidences that our approach, which is based on finding semantic paths between POIs and musicians, can be used effectively to define and evaluate the semantic relatedness between such instances (**RQ2**). Moreover, we have shown that users perceive music composed/performed by the recommended musicians as well-suited for the POIs.

We note that the presented framework fully relies on the data available in DBPedia repository – in order for a musician to be recommended by our approach, its record has to be present in DBPedia. Admittedly, this is a limitation, since less known musicians may not have DBPedia entries. In the future, this could be addressed by using additional Linked Data repositories.

An important next step in this research is adapting the proposed framework to incorporate user preferences into the recommendation process. As described in Section 3.1, this can be achieved using the weights of relations between classes and instances in the semantic network. We intend to ex-

plore personalization approaches that translate user's preferences into these weights. Furthermore, we intend to work on identifying and exploiting other arbitrary semantic relations between POIs and musicians, e.g. direct relations such as 'Gustav Mahler *was the director of* Vienna State Opera', and complex non-directed relations such as 'Ana Belén (a famous Spanish singer) *composed a song* whose lyrics *are about* La Puerta de Alcalá (a well-known POI in Madrid, Spain)'. We will search for these relations in different Linked Data repositories, and will use available tools such as RelFinder [7]. It is also important to better explore and evaluate strategies to initialize the importance weights of the relations and classes in the semantic network. In this context, we will explore other approaches to define and measure the semantic relatedness between concepts [13], and will adapt or extend them for cases in which the concepts belong to different domains.

7. REFERENCES

[1] G. Adomavicius, B. Mobasher, F. Ricci, and A. Tuzhilin. Context-aware recommender systems. *AI Magazine*, 32(3):67–80, 2011.

[2] A. Ankolekar and T. Sandholm. Foxtrot: a soundtrack for where you are. In *Interacting with Sound Workshop: Exploring Context-Aware, Local and Social Audio Applications*, pages 26–31, 2011.

[3] S. Auer, C. Bizer, G. Kobilarov, J. Lehmann, R. Cyganiak, and Z. Ives. DBpedia: A nucleus for a web of open data. In *ISWC'08*, pages 722–735, 2008.

[4] C. Bizer, T. Heath, and T. Berners-Lee. Linked data - the story so far. *International Journal on Semantic Web and Information Systems*, 5(3):1–22, 2009.

[5] Ò. Celma and P. Lamere. If you like radiohead, you might like this article. *AI Magazine*, 32(3):57–66, 2011.

[6] I. Fernández-Tobías, I. Cantador, M. Kaminskas, and F. Ricci. A generic semantic-based framework for cross-domain recommendation. In *HetRec'11*, pages 25–32, 2011.

[7] P. Heim, S. Hellmann, J. Lehmann, S. Lohmann, and T. Stegemann. Relfinder: Revealing relationships in rdf knowledge bases. *Semantic Multimedia*, pages 182–187, 2009.

[8] M. Kaminskas and F. Ricci. Location-adapted music recommendation using tags. In *UMAP'11*, 2011.

[9] M. Kaminskas and F. Ricci. Contextual Music Information Retrieval and Recommendation: State of the Art and Challenges. *Computer Science Review*, 6:89–119, 2012.

[10] A. Loizou. *How to recommend music to film buffs*. PhD thesis, University of Southampton, 2009.

[11] C. D. Manning, P. Raghavan, and H. Schtze. *Introduction to Information Retrieval*. Cambridge University Press, New York, NY, USA, 2008.

[12] A. Passant. dbrec - music recommendations using dbpedia. In *International Semantic Web Conference (2)*, pages 209–224, 2010.

[13] S. Ponzetto and M. Strube. Knowledge derived from wikipedia for computing semantic relatedness. *Journal of Artificial Intelligence Research*, 30(1):181–212, 2007.

[14] A. Stupar and S. Michel. Picasso - to sing, you must close your eyes and draw. In *ACM SIGIR'10*, pages 715–724, 2011.