Enhancing cultural recommendations through social and linked open data

DOI:10.1007/s11257-019-09225-8

Noname manuscript No.
(will be inserted by the editor)

Enhancing Cultural Recommendations through Social and Linked Open Data

Giuseppe Sansonetti · Fabio Gasparetti · Alessandro Micarelli · Federica Cena · Cristina Gena

Received: date / Accepted: date

Abstract In this article, we describe a hybrid recommender system (RS) in the artistic and cultural heritage area, which takes into account the activities on social media performed by the target user and her friends, and takes advantage of linked open data (LOD) sources. Concretely, the proposed RS *(i)* extracts information from Facebook by analyzing content generated by users and their friends; *(ii)* performs disambiguation tasks through LOD tools; *(iii)* profiles the active user as a social graph; *(iv)* provides her with personalized suggestions of artistic and cultural resources in the surroundings of the user's current location. The last point is performed by integrating collaborative filtering algorithms with semantic technologies in order to leverage LOD sources such as DBpedia and Europeana. Based on the recommended points of cultural interest, the proposed system is also able to suggest to the active user itineraries among them, which meet her preferences and needs and are sensitive to her physical and social contexts as well. Experimental results on real users showed the effectiveness of the different modules of the proposed recommender.

Keywords Cultural heritage · Recommender systems · Social network · Linked Open Data

1 Introduction

With the spread of mobile technologies, users can easily share online comments and pictures regarding their visited points of interest (POIs), thus generating a vast amount of social data. User modeling techniques based on activities on social networks are gaining increasing attention from the research community (Carmagnola et al., 2007). However, such services suffer from the absence of a standard for processing and managing social information. Hence, Semantic Web technologies may be adopted to structure and enrich such data in order to

G. Sansonetti, F. Gasparetti, A. Micarelli
Roma Tre University, Department of Engineering, Via della Vasca Navale, 79
00146 Rome, Italy
E-mail: gsansone, gaspare, micarel@dia.uniroma3.it

F. Cena, C. Gena
University of Torino, Department of Computer Science, Via Pessinetto, 12
10149 Torino, Italy
E-mail: federica.cena,cristina.gena@unito.it

integrate it, provide it with semantics, and make it available on the Web. Linked open data (LOD) represents an enormous repository of structured data so that they can be interlinked based on the semantic meaning (Bizer et al., 2009). This data can be freely queried by anyone, machine or human, thus representing a unique resource to draw upon. In this paper, we describe *Cicero*, a social recommender system (Biancalana et al., 2013; Ricci et al., 2015) realized for analyzing how data extracted from a user's activities on social networks may be enriched with the semantic knowledge provided by LOD. Specifically, this recommender is deployed in the artistic and cultural heritage domain.

The main contributions of this paper are the following ones:

- personalized POIs recommendation according to the user profile extracted from social networks by means of a semantic-based match;
- enrichment of information provided to users exploiting the LOD cloud according to the user profile;
- personalized itineraries recommendation according to the user profile and her physical and social contexts.

The rest of this article is organized as follows. Section 2 details Cicero, namely, the proposed approach to POI recommendation, the evaluation methodology and the experimental results. Section 3 describes the itinerary recommender and shows the results of the experimental evaluation performed on it. Section 4 presents some related works and their differences with our recommendation engine. Section 5 illustrates our conclusions and plans for future work.

2 The Cicero Recommender

Cicero is a social recommender system realized for analyzing how data extracted from users' activities on social networks may be enriched with the semantic knowledge provided by LOD. Specifically, this recommender is deployed in the artistic and cultural heritage domain (De Angelis et al., 2017).

The recommendation process occurs in three steps. In the first one, the social network is analyzed and relevant information about users and their activities is retrieved. Thereafter, this information is represented as a model of user's interests stored in a graph database (i.e., Neo4j[1]). The last step is the recommendation itself. The Facebook[2] social network has been chosen because of the large amount of geo-referenced user data available through the *Facebook Places* functionality.

2.1 Social Data Extraction

Activities on the users' timelines, that is, the space where the users can collect posts, *"like"*, stories, and multimedia, are the primary source of information for the recommendation. In addition to the content on timelines, users can also report their position by submitting a

[1] https://neo4j.com (Accessed: 26 November 2018). Actually, Neo4j can be used both for storing data in RDF format and as semantic query engine, thus datasets such as DBpedia and Europeana can be easily analyzed. Thus Neo4j allows us to overcome some of the limitations of the atomic/primitive RDF stores, such as the impossibility to uniquely identify instances of relationships of the same type, or the inability to qualify instances of relationships (Antoniou et al., 2012)

[2] http://www.facebook.com/ (Accessed: 29 November 2018).

check-in. The Facebook *Graph API* interface [3] allows developers to access this information once the user has granted the application associated with the recommender system.

Given, for example, the status “Rome is really amazing! - at Colosseum”, as shown in Figure 1, we can obtain the explicit Facebook tag *Colosseum*, along with its spatial coordinates associated with the check-in. The social network platform assigns to each place its category and city. The list of categories for places can be obtained via the graph API through the appropriate search query.[4] Such a list includes, for instance, concert venues, churches, and libraries. In our example, the Colosseum is a *Historical Place* in the city of *Rome*, and 27 user’s friends have already checked-in.

Fig. 1 A user’s post on Facebook.

The textual content of posts is considered each time the place annotation is missing. Because many Facebook pages related to POIs are created and managed by users, it is possible to have multiple pages associated with the same place. The textual content of pages is not always correct and some pages refer to places not relevant for the community, such as “My Home”. All these issues prevent the recommendation process from providing accurate suggestions. For such reasons, the places obtained from the target user’s posts are subjected to a disambiguation process, where multiple instances of the same place are merged, and irrelevant information is discarded. *GeoNames*[5] - a worldwide geographical LOD database - is considered for this step. Briefly, an active user’s post is analyzed for extracting references to relevant entities. Such entities are given as input to the GeoNames API that returns, if available, the spatial coordinates of each entity. If the geographic distance between those coordinates and the ones of the Facebook place is less than a given threshold (i.e., 500 meters), they are deemed correct. If multiple coordinates are present, the ones that minimize the distance are considered. The output of the disambiguation process is a unique URI on DBpedia[6]. For instance, two pages such as “Università Roma Tre” and “Roma Tre University”, or “Biberach Baden-Wurttemberg” and “Biberach (district)” are being considered as a single place by the recommender, respectively.

[3] https://developers.facebook.com/docs/graph-api (Accessed: 29 November 2018).
[4] The search query for getting from Facebook the list of categories is `search?type=place_topic&topic_filter=all`.
[5] https://www.geonames.org/ (Accessed: 29 November 2018).
[6] https://wiki.dbpedia.org/ (Accessed: 29 November 2018).

2.2 User Modeling

A direct and heterogeneous social graph represents a user model. Each vertex (or node) is associated with one of the following classes:

– `Person:` an active Facebook user.
– `Place:` the visited place recognized by the extraction step.
– `Location:` the coordinates of a place.
– `Category:` the category of a place obtained by the Facebook page.

The edges (or ties) can assume one of the following labels:

– `KNOWS:` the edge connecting two `Person` nodes sharing a social tie.
– `VISITED:` the edge connecting a `Person` node to a visited `Place`.
– `LOCATED_IN:` the edge connecting a `Place` node to a `Location` node at the time the post was evaluated.
– `HAS_CATEGORY:` the edge connecting a `Place` node to the corresponding `Category` node.

The `Person` node representing a user and the graph of users sharing a social tie with her form a Cicero's *social cluster* (or cluster, as introduced in (Raad et al., 2013)). An example of such a cluster can be found in Figure 2.

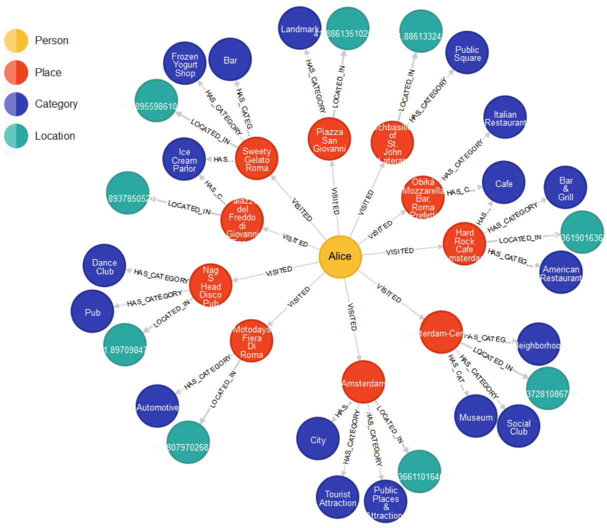

Fig. 2 Cicero cluster related to the user Alice.

In Figures 3 and 4, it is possible to note an excerpt of the social graph with Alice's friends, as well as the graph showing her friends that visited two given POIs.

2.3 Recommender Systems

In Cicero, different recommendation engines are implemented. They can be divided into two main groups: *social recommenders* and *semantic recommenders*. Cicero allows the active user to merge more recommenders in sequence in order to analyze their combined effects.

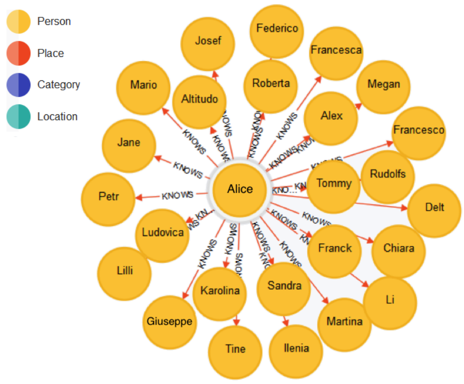

Fig. 3 An excerpt of the Alice's social graph.

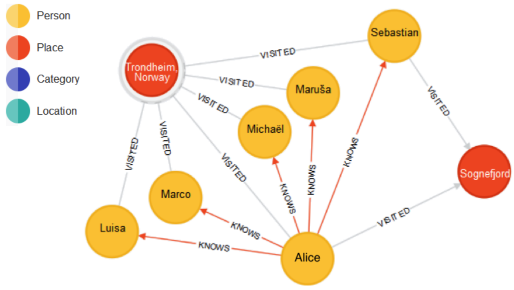

Fig. 4 Alice's friends that visited some POIs.

2.3.1 Social Recommenders

The *Baseline Social Recommender (BS)* randomly returns some of the places visited by the community of the active user, which are in the surroundings (i.e., 5 kilometers) of her current position. We used it as baseline in the experimental trials. Such recommender does not exploit any user profile, thus providing non-personalized suggestions. Conversely, the *Community-Based Social Recommender (CM)* and the *Collaborative Filtering Social Recommender (CF)* model the user's preferences in terms of categories of interesting places. Of all the `Place` nodes with a `VISITED` link to the `Person` node representing the user, the `Category` nodes linked to them are returned, ranked in descending order according to the score. This score is the number of `HAS_CATEGORY` links direct to them. The output is then filtered based on the following strategy: from the 238 Facebook categories, we selected the 37 categories related to the cultural heritage domain, such as *City*, *Monument*, *Museum*, *Historical Place*, *Touristic Attraction* and *Church*. If the category extracted from the user profile belongs to them, its initial score remains unchanged, otherwise it is reduced by a θ value. In order to compute the best value for θ, we performed a sensitivity evaluation of such parameter through a large-scale gradient descent algorithm (Zhang, 2004) with learning rate $\zeta = 0.1$. In our experimental setup, such value was 0.7. The rationale behind the *Community-Based Social Recommender* is that a friend is someone who can be trusted and that, therefore, belongs to her *trust network*. It is highly likely that the most popular places among friends are of interest for the target user as well. The algorithm queries the active user's social graph to retrieve all `Place` nodes belonging to the previously computed categories, ranked in descending order according to the number of incoming `VISITED` links, namely, the number of people who visited them. The preference score for a POI is then assigned as a weight normalized based on the number of occurrences. After filtering the returned items based on their coordinates, their number is limited to the top ten. Within the community of a user's friends, it is not necessarily that their interests correspond to her own. Conversely, the *Collaborative Filtering Social Recommender* includes filtering techniques peculiar to the collaborative approach. The goal is to select people with the most similar preferences to the user's ones, returning only the items related to them within the social graph. The output of the algorithm consists of all `Place` nodes belonging to the preferred categories, visited and tagged by friends with similar interests. The resulting list is filtered based on the geographic location and sorted by the number of incoming edges to `Place` nodes with `VISITED` label.

2.3.2 Semantic Recommenders

Semantic recommenders enrich user profiles with information from linked open data sources, in particular from the DBpedia and Europeana[7] projects. Europeana represents a multilingual online collection of millions of digitized items from European museums, libraries, archives and multi-media collections, with procedures for content providers. Figure 5 shows some of the items related to the *Colosseum* resource that are available in the Europeana collection. In Cicero, two different techniques for semantic recommendation are proposed:

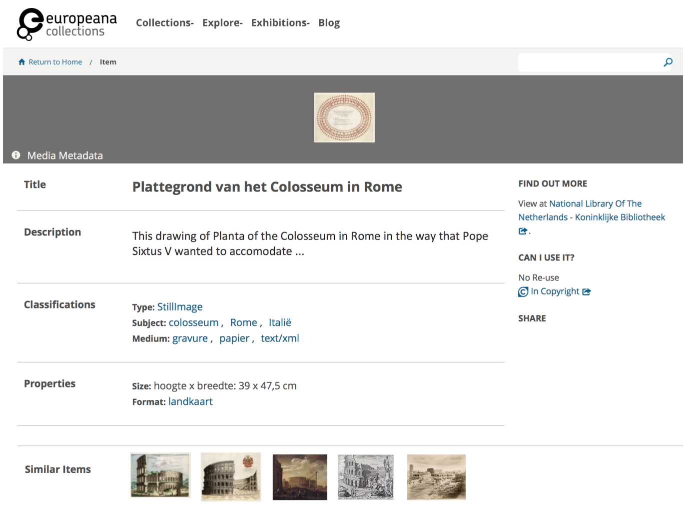

Fig. 5 Multimedia available in the Europeana collection for the *Colosseum* resource.

DBpedia Recommender (DB) and *Europeana Recommender (EU)*. The former takes advantages of the LOD knowledge base of DBpedia. From a list of URIs of artistic and cultural resources on DBpedia, this recommender suggests to the target user places sharing the same

[7] https://www.europeana.eu/ (Accessed: 29 November 2018).

semantic categories as the input resources. In the DBpedia Recommender, queries are formulated in SPARQL. To enhance the user experience, the system is able to provide her with not only the title of the resource, but also its representative image and an external link to the corresponding Wikipedia page. This metadata can be easily extracted from the RDF dataset. The *Europeana Recommender* explores the RDF graphs from the Europeana collection of resources (e.g., paintings, maps, audio, and video resources) for suggesting those related to the user's interests. Further metadata is extracted in addition to the title and the subject: the creator, the URL of the media resource (e.g., photos, audio, videos, or thumbnail in the case of documents, books or manuscripts), and an external link to the respective page of the Europeana website, thus enabling a more thorough contextualization of the item. Such semantic recommenders are enhanced as follows. From a URI of a DBpedia resource, corresponding to a place visited by the user, the algorithm navigates the DBpedia and Europeana datasets and returns the concepts having the highest level of *semantic similarity* with the input URI. In (Passant, 2010), the author proposes several theoretical methods for assessing the semantic distance between two entities, which can be seen as a measure of how closely related they are. A first mathematical estimate of the distance between two resources r_a and r_b, named *direct distance*, is calculated based on the number of direct links from r_a to r_b and vice versa, that is, $C_d(r_a, r_b)$ and $C_d(r_b, r_a)$ as follows:

$$D_d(r_a, r_b) = \frac{1}{1 + C_d(r_a, r_b) + C_d(r_b, r_a)} \tag{1}$$

The second estimate, called *indirect distance*, is computed based on the number of links from r_a and r_b to a third entity in common, and from a third entity in common towards r_a and r_b, respectively, $C_{io}(r_a, r_b)$ and $C_{ii}(r_a, r_b)$, as follows:

$$D_i(r_a, r_b) = \frac{1}{1 + C_{io}(r_a, r_b) + C_{ii}(r_b, r_a)} \tag{2}$$

The third estimate, called *combined distance*, takes into account direct and indirect links, as follows:

$$D_c(r_a, r_b) = \frac{1}{1 + C_d(r_a, r_b) + C_d(r_b, r_a) + C_{io}(r_a, r_b) + C_{ii}(r_b, r_a)} \tag{3}$$

Based on the previous definitions, let c_i be the concept (extracted from the user data and identified by an URI) in input, c_o the concept of which we want to assess the semantic similarity with c_i, and c_x any node of the RDF graph, connected to c_i, c_o, or both of them. Let p, p_1, and p_2 represent the possible properties that can exist between two resources. The graph patterns, expressed as RDF triples in the form of *subject-predicate-object*, which can therefore occur in the dataset and contribute to the degree of similarity, are the following ones:

- $c_i\ p\ c_o$
- $c_o\ p\ c_i$
- $c_i\ p_1\ c_x\ .\ c_x\ p_2\ c_o$
- $c_i\ p_1\ c_x\ .\ c_o\ p_2\ c_x$
- $c_x\ p_1\ c_i\ .\ c_o\ p_2\ c_x$
- $c_x\ p_1\ c_i\ .\ c_x\ p_2\ c_o$

The first two statements denote direct links, that is, patterns where c_i and c_o belong to the same RDF statement. The remaining statements are indirect: it is possible to reach c_o from c_i through a path with a higher depth level. In this way, our algorithms can infer that the user was also interested in concepts related to the concepts mentioned by her. In other terms, the user may be interested in a c_1 concept that is directly related to c_i or she may be interested in a c_2 concept that is indirectly related to c_i.

The following examples are related to concepts having the same semantic distance. The first one is as follows:

c_i = dbr:Colosseum
p_1 = dbp:builder
c_x = dbr:Titus
c_x = dbr:Titus
p_2 = dbp:builder
c_o = dbr:Arch_of_Titus

The example above shows that the Colosseum and the Arch of Titus are linked through the emperor Titus, who built both monuments. The second example is as follows:

c_i = dbr:Colosseum
p_1 = dct:subject
c_x = dbc:Amphitheatres_in_Rome
c_x = dbc:Amphitheatres_in_Rome
p_2 = dct:subject
c_o = dbr:Amphitheatrum_Castrense

In this example, the Colosseum and the Amphitheatrum Castrense are linked through the concept of Roman Amphitheatrum.

The output of semantic recommenders are a list of $\langle c_i, c_o \rangle$ pairs sorted in descending order according to their semantic similarity value. Figure 6 shows the prototype of the map-based user interface implemented in Cicero. Each pinpoint represents a cultural resource matching the active user's interests and located near her current location. The user can click on it to take advantage of all the resources made available by LOD sources. Figure 7 illustrates the case in which the user selected the pinpoint number four.

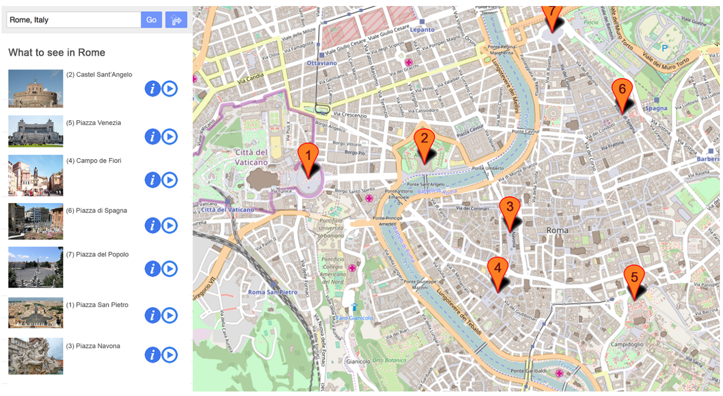

Fig. 6 The classic map-based UI of the recommender.

Fig. 7 One of the multimedia available in the Europeana collection.

2.4 Experimental Evaluation

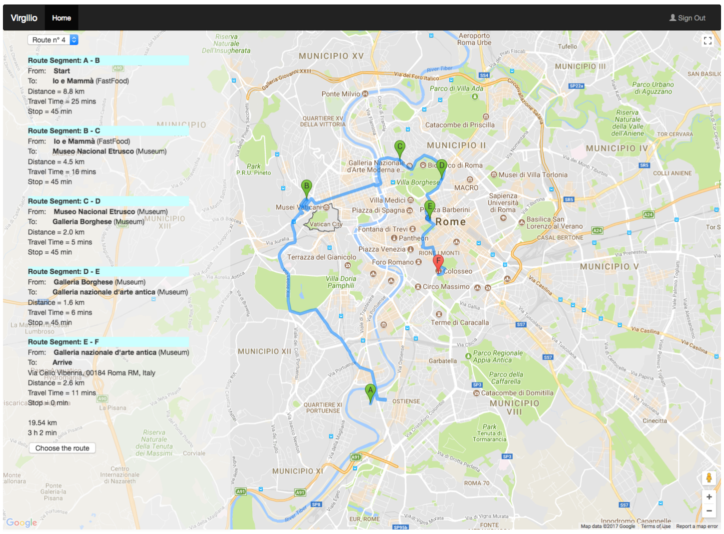

Fig. 8 A screenshot of the interface under heuristic evaluation.

In order to assess the performance of Cicero recommendations, we carried out an extensive user-centric experimental evaluation. The evaluation of a complex system such as a

Table 1 Statistical data related to number of posts and photos analyzed in the experimental evaluation. The numbers are average over the sample of users.

	User	User's friends
Number of published posts	534	252,582
Number of published photos	195	92,235
Number of published geolocalized posts	28	13,244
Number of published geolocalized photos	44	20,812
Number of friends for user	473	
Number of test users	50	

social hybrid recommender system requires the integration of evaluation approaches from different areas, such as adaptive and user modeling systems, recommender systems, etc.

The evaluation of a *user-adapted system* requires an evaluation of the different components that collaborate to produce the adaptation. The so-called *layered approaches* have been proposed for the separate evaluation of each adaptation feature. Such approaches identify at least two layers: the content layer (related to content recommendations), and the interface layer (related to interface adaptations). This idea originated from Totterdell and Boyle (1990), who first phrased the principle of layered evaluation. Karagiannidis and Sampson (2000) and Brusilovsky *et al.* (2001) also distinguished two levels in the adaptive process: the interaction assessment phase and the adaptation decision-making phase. Examples of layered evaluations can be found in (Brusilovsky et al., 2001; Paramythis et al., 2001; Weibelzahl, 2001; Weibelzahl and Lauer, 2001; Weibelzahl and Weber, 2001; Weibelzahl, 2003).

In the evaluation of Cicero we adopted a standard layered evaluation approach, distinguishing the evaluation of the interface layer (see Section 2.6) from that the content layer (recommendations), see Sections 2.5 and 3.3. Regarding the last point, several metrics are used in the area of recommender systems in order to evaluate the recommendation quality, and in particular we focused on: the predictive accuracy of the recommendation process, the user's selection of recommended contents (Sarwar et al., 2001; McLaughlin and Herlocker, 2004), and finally the perceived recommendation quality (Pu et al., 2011a).

2.5 Evaluation of content layer

2.5.1 User sampling

To this aim, we asked some users with an active Facebook account to participate in voluntary testing. The evaluation of the system was attended by a sample of 50 individuals aged between 20 and 65 years, mostly students and academics. Statistics related to the number of all the posts and photos analyzed during the experimental tests are reported in Table 1.

On average, each tester had 473 friends. The second column of the table shows statistics for the single user only, the third column for her friends. As a whole, we analyzed 12,629,100 posts and 4,611,750 images. From the third column we can also see that only the 5,24% and 22,56%, respectively, of the 252,582 posts and 92,235 photos extracted from the stream of activities produced on a single account of the social network, have a geolocalized tag. As a result, the collaborative filtering algorithms proposed in Cicero can exploit - on average per profile - a sample of 34,056 visited places (the 13,244 extracted from the posts plus the 20,812 photos published by the user and her community) for user modeling. This data shows a social behavior by users while geotagging their own content on location-based social networks. Generally speaking, users prefer to share their actual location when they publish photos (61,1%) in comparison with when they add new posts on their own social network profiles (38,9%).

2.5.2 *Data security and protection policy*

Before starting the experimental evaluation, we asked the participants - all of them were volunteers - their informed consent for collecting data according to the current Italian privacy law (D.L. 196/2003). We thoroughly informed them of the ways in which we would collect, use, and store data related to their activities on Facebook. Children or minors, people not able to understand the above information due to any reasons, were not recruited. Moreover, users were at any time able to withdraw from the system and request their data to be removed. During the experiments, effective security and privacy measures were taken to prevent any leakage of personally identifiable information belonging to users. The overall data storage was implemented in compliance with EU General Data Protection Regulation (GDPR).

2.5.3 *Evaluation in terms of nDCG*

Several techniques have been proposed for assessing the predictive accuracy of the recommendation process. We employed the *normalized* version of *Discounted Cumulative Gain (nDCG)* (Järvelin and Kekäläinen, 2002), which ensures important advantages compared to other evaluation metrics (Wang et al., 2013). It measures the utility (relevance) of an item based on its position in the returned list. The *nDCG* is usually truncated at a particular rank level p to emphasize the importance of the first retrieved documents. Its definition is as follows:

$$nDCG@p = \frac{DCG@p}{IDCG@p} \tag{4}$$

where the *Discounted Cumulative Gain (DCG)* is defined as follows:

$$DCG@p = rel_1 + \sum_{i=2}^{p} \frac{rel_i}{\log_2 i} \tag{5}$$

with rel_i being the graded relevance of the $i-$th result. The *Ideal DCG (IDCG)* for a query corresponds to the *DCG* measure where scores are resorted monotonically decreasing, that is, the maximum possible *DCG* value over that query. *nDCG* is often used to evaluate search algorithms and other techniques whose goal is to order a subset of items in such a way that highly relevant documents are placed on top of the list, while less important ones are moved lower. Basically, higher values of *nDCG* mean that the system output gets closer to the ideal ranked output.

The evaluation procedure of the system consisted of several steps. In the first phase, Cicero had to acquire enough data to infer and model the user's interests. After having access to their Facebook account through the Cicero interface, testers enabled the framework to extract and store their own information (such as name, city of residence, gender, work, education, visited places) and then data related to their friends, especially all those places tagged on a post or photo. The second phase was the actual evaluation. Cicero provided the target user with a list of ten items recommended through a particular pipeline of algorithms, with the user unaware of details. For each list of objects returned by the selected recommender, the user was asked to assign a rating to each item according to the degree of relevance to her interests. The rating was expressed in the range of a 5-point Likert scale. For each rank and algorithm the system averaged the ratings collected from users. Those values represented the gain rel_i (see Eq. 5) of each document and were used to obtain the results in terms of *nDCG@p* measures.

Table 2 Comparison of recommendation algorithms in terms of *accuracy* (where *RS: Recommender System*, *BS: Baseline Social*, *CF: Collaborative Filtering Social*, *CM: Community-Based Social*, *EU: Europeana*, and *DB: DBpedia*).

RS	*nDCG@1*	*nDCG@5*	*nDCG@10*
BS	0.22	0.31	0.36
DB	0.41	0.49	0.54
EU	0.43	0.50	0.56
DB-EU	0.47	0.54	0.62
CM	0.37	0.49	0.53
CM-DB	0.43	0.55	0.58
CM-EU	0.47	0.58	0.63
CM-DB-EU	0.52	0.63	0.69
CF	0.56	0.65	0.70
CF-DB	0.62	0.68	0.73
CF-EU	0.66	0.71	0.78
CF-DB-EU	**0.70**	**0.73**	**0.83**

Table 2 summarizes the obtained results. These results show that the Baseline Social Recommender (BS) was not effective. The accuracies of the Community-Based (CM) and Collaborative Filtering (CF) recommenders were higher, being able to personalize recommendations. Even the *pure* semantic recommenders, namely, algorithms that exploit the LOD knowledge bases of DBpedia (DB) and Europeana (EU) and are personalized through the profile of the target user without information concerning her friends and the users more similar to her, allow the system to obtain performances significantly higher than those of the baseline. However, the recommenders that exploit the strengths of both (semantic and social) techniques always enable Cicero to achieve the highest performance in terms of *nDCG@p* in the various scenarios, showing the significant benefits of the synergistic action of the two techniques.

2.5.4 Statistical significance analysis

A truly comprehensive and rigorous evaluation includes the statistical significance analysis of the experimental results. In other terms, it is needed to prove that the observed values are real and not due to chance. In our scenario, this analysis results in verifying that the collaborative algorithms are actually better than the baseline and the recommender systems based on the context information extracted from linked open data can provide more accurate recommendations than recommenders based only on the analysis of the user's social graph. Among the different tests proposed in the research literature, we chose the *ANalysis Of VAriance (ANOVA)* test (Fisher, 1925).

In order to assess the statistical significance of the results obtained for the perceived accuracy of the proposed recommendation algorithms, a single factor ANOVA test has been carried out on the achieved *nDCG@p* values. More specifically, four ANOVA tests were performed. The first three trials compared the results for each of the following three pipelines:

Table 3 Results of the ANOVA test on all the pipelines. The upper table shows the *Mean* and the *Variance* of the *nDCG@p* values (for $p = 1, ..., 10$) for each group; the lower table shows other significant ANOVA values according to different *Sources of Variation* (*Between-Groups*, *Within-Groups* e *Total*): *Sum of Squares (SS)*, *Degrees of Freedom (DF)*, *Mean Square (MS)*, F_{test}, *P-Value*, and F_{crit}.

Groups	*Mean*	*Variance*
BS	0.246667	0.074348
CM	0.453333	0.026531
CM-DB	0.539920	0.011027
CM-EU	0.567638	0.017169
CM-DB-EU	0.643427	0.023103
CF	0.617257	0.032665
CF-DB	0.684317	0.021217
CF-EU	0.723226	0.02044
CF-DB-EU	0.787413	0.039552

Source of variation	*SS*	*DF*	*MS*	F_{test}	*P-Value*	F_{crit}
Between-groups	0.326497	8	0.040812	19.28871	7 E-14	2.960351
Within-groups	0.124835	59	0.0002116			
Total	0.451332	67				

1. *Baseline Social Recommender (BS) - Community-Based Social Recommender (CM) - Collaborative Filtering Social Recommender (CF);*
2. *Collaborative Filtering Social Recommender (CF) - Collaborative Filtering Social Recommender + DBpedia (CF-DB) - Collaborative Filtering Social Recommender + Europeana (CF-EU) - Collaborative Filtering Social Recommender + DBpedia + Europeana (CF-DB-EU);*
3. *Community-Based Social Recommender (CM) - Community-Based Social Recommender + DBpedia (CM-DB) - Community-Based Social Recommender + Europeana (CM-EU) - Community-Based Social Recommender + DBpedia + Europeana (CM-DB-EU).*

The fourth ANOVA test was accomplished on all the nine possible pipelines in a single group. In all four cases, the null hypothesis could be rejected, because the computed F_{test} was significantly higher than the corresponding F_{crit} value. In order to not excessively weigh down the manuscript, we only report the results of the last test, which considered each of the nine possible pipelines as groups. The obtained results are shown in Table 3, which is arranged in two sections. The upper one shows all input data, namely, the *Group*, the *Mean*, and the *Variance* of the *nDCG@p* values for all ranks (i.e., for $p = 1, ..., 10$). The lower one reports the results of the ANOVA test, therefore, the *Variance between Groups (Between-groups)*, the *Variance within Groups (Within-groups)*, the *Sum of Squares (SS)*, the *Degrees of Freedom (DF)*, the *Mean of Squares (MS)*, the F_{test}, the *P-Value*, and the F_{crit}. It can be noted that the value of F_{test} was far greater than the value of the corresponding F_{crit}, so the null hypothesis could be rejected.

2.5.5 Evaluation in terms of novelty, serendipity, and diversity

In our evaluation tests we considered not only the perceived *accuracy* (i.e., the degree to which the user feels the recommendations match her interests and preferences) of predictions, but also their *novelty* (i.e., how unknown the recommendations are), *serendipity* (i.e., how surprising the relevant recommendations are), and *diversity* (i.e., how dissimilar the recommendations are) (Pu et al., 2011b). We decided to evaluate such metrics in the same way they are perceived by the users. Thus, we exploited a questionnaire, given to all the users involved in this evaluation phase, since we wanted to collect their real opinions and feelings.

To this aim, after evaluating the recommender in terms of predictive accuracy, the user had to express her agreement/disagreement, in a Likert 5-point scale, with the three following statements:
"This recommender helped me discover items I did not know before" (novelty);
"This recommender helped me discover surprisingly interesting items I might not have known in other ways" (serendipity);
"The recommended items are different to each other" (diversity).
Figure 9 summarizes the average values of such ratings on the different proposed pipelines of recommenders.

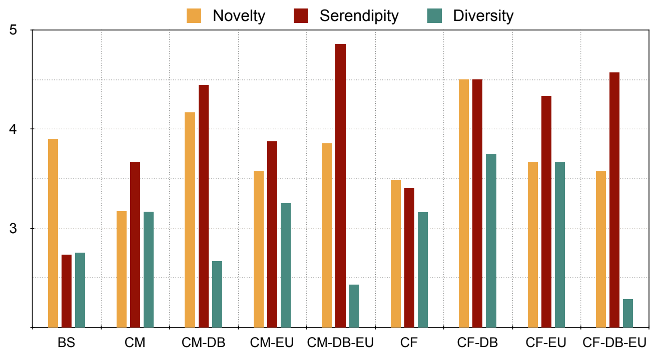

Fig. 9 Comparison of recommendation algorithms in terms of *novelty*, *serendipity*, and *diversity* (where *BS: Baseline Social*, *CF: Collaborative Filtering Social*, *CM: Community-Based Social*, *EU: Europeana*, and *DB: DBpedia*).

As expected, the Baseline Social recommender (BS) achieved lower serendipity values than the other social recommenders (i.e., CM and CB), which profile users as social graphs. The high average score (3.9/5) of the novelty metric related to the BS recommender is due to the fact that, randomly choosing items from those referenced by the active user's friends, the recommender is highly likely to recommend a cultural resource unknown to her. The corresponding low value (2.74/5) of serendipity denotes that testers did not consider the recommender so useful in discovering new relevant items, but they were convinced to be able to achieve the same goals by themselves. Comparing the results of social recommenders with those of social+semantic recommenders, it can be noted again that leveraging LOD knowledge bases leads to better results in terms of not only the perceived accuracy, as shown before, but also novelty and serendipity. The combined use of DBpedia and Europeana brings the highest values (4.84/5) of serendipity, even though the recommenders exploiting only one of these components achieved still satisfactory results. Diversity was the metric with the lowest values. The highest score (3.75/5) was obtained by the CF-DB recommender. Interestingly, the CF-DB-EU recommender obtains the lowest performance in terms of diversity. Evidently, the combined use of both LOD sources, on the one hand, allows the system to recommend to the target user POIs that positively surprise her, but, on the other hand, the categories of the suggested venues are rather similar to each other. This is probably due to the nature of the deployed techniques that rely on the concept of semantic similarity to identify resources of probable interest to the active user.

2.6 Evaluation of interface layer

During the preliminary phase of development of Cicero we carried out a heuristic evaluation of the first release of the interface.
The *heuristic evaluation* was carried out by two separate HCI experts on a set of static interface prototypes (see Fig. 8) with the aim of checking *i)* the usability of the interface and the conformance to general HCI principles; *ii)* the usability of the adaptive behavior of the system, based on the existing literature in adaptive systems evaluation (Jameson, 2006; Gena and Weibelzahl, 2007). In particular, the experts were asked to follow Jameson's five usability challenges (Jameson, 2006) for adaptive interfaces, in addition to the standard usability heuristics (Nielsen, 1999). In particular, the experts found problems of

- *transparency*, explaining to the user what is happening, and better explaining what the system is recommending and why, e.g. adding explanations as "Hello Giuseppe, here are our suggested routes, and below choose one of the paths that we recommend to you", etc.;
- *breadth of experience*, namely giving more alternatives and recommendations to the user, e.g. it might be useful to suggest her alternative routes (as Google Maps does), telling the user whether the itinerary is reachable by foot or by by bus), etc.;
- *feedback*, adding interactive feedback when the user focuses on an itinerary (e.g., for each route the user chooses reporting the title of the route and the total duration), adding the user presence on the screen (e.g., the user must perceive that she is logged in, with the right icon and under the user's name, which add a feeling of personalization), linking the stop points to some more information on them (if underlined, they immediately appear to be links), etc;
- *labeling* naming the routes, instead of numbering them, etc.

This evaluation led to a first re-design both of the user interface and of some aspects of the system functions. Moreover, thanks to the suggestions emerged during the expert evaluation, we decided to add, in a future re-design of the system, a 3D map-based navigation, in order to increase the user orientation among streets and monuments.

3 Itinerary Recommendation

3.1 The Recommender

The module of the system presented in Section 2 analyzes the activities performed by the target user and her friends on social media and provides her with personalized suggestions of artistic and cultural resources located nearby. However, these resources can be numerous and, moreover, not available or of little use at the time when the user is active on the system. The solution to those problems is supplied by an additional module of the system that is able to recommend personalized itineraries. Such itineraries take into account not only the active user's interests and information needs, but also her physical and social context. In the proposed itinerary recommender each user is profiled as a vector of weights whose values (between 0 and 1) express the user's interest in a certain category of points of interest (POIs) (see Table 4).

Such user profile is explicitly created and implicitly updated. More specifically, when registering on the system, the user is shown a form (see Fig. 10) and asked to indicate her main interests by clicking on a series of images related to the 37 different categories of POIs,

Table 4 Examples of user profiles.

	Arts	*Museums*	*Monuments*	*Churches*	...	*Theatres*	*Exhibitions*
User1	0.80	0.70	0.20	0.13	...	0.25	0.42
User2	0.23	0.15	0.71	0.56	...	0.50	0.83
User3	0.44	0.68	0.10	0.20	...	0.14	0.21
User4	0.15	0.19	0.20	0.77	...	0.69	0.40

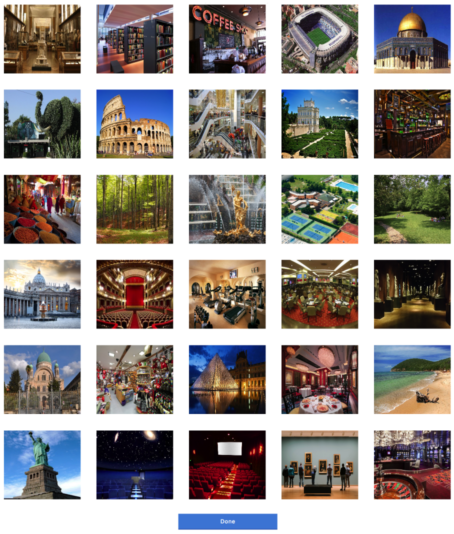

Fig. 10 The form shown to the user during the registration process.

in order to craft an initial representation of her profile. Then, the implicit feedback step takes place every time the user leaves a check-in on the online service.

Let us introduce the function f_{venues} in such a way:

$$f_{venues}(U) \rightarrow \mathcal{P}(V) \tag{6}$$

which returns the venues visited by a given user, and the function *cat* that returns the vector of categories associated with a venue, for instance:

$$cat(Rome\_Colosseum) = [0, 0, 1, 0, \cdots, 0, 0] \tag{7}$$

The vector w that represents the user profile is drawn at each new check-in as follows:

$$w_u = w_u^{(0)} + \frac{\sum_{v \in f_{venues}(u)} cat(v)}{|f_{venues}(u)|} \tag{8}$$

A crucial factor in the recommendation process is the current physical context, namely, any information that can be used to characterize the situation of an entity (Dey, 2001). In this case, *entities* are the active user and the POIs, whilst *information* concerns the current location, the time of the day, the day of the week, the weather conditions, the means of transport, and so on. Almost all this information can be determined without the user's involvement. The location is detected by the GPS sensor of the mobile device, as well as the means of transport is detected by the accelerometer. Moreover, the weather conditions are obtained by querying weather services, based on the current location.

Since the main goal is the recommendation of popular routes, the problem has been modeled as the search of a direct graph. Each node represents a POI extracted from LOD (see Sect. 3.2), each edge represents a direct link between two POIs, with a weight denoting their distance in terms of time (minutes). For the graph construction, we have to select the set of POIs, and then derive the set of edges among them. Therefore, the first step is to select the rectangular region containing those POIs, which is delimited by the latitudes and longitudes of the starting point (obtained from the GPS sensor of the device), and the end point (entered by the user). After defining the region boundaries, all the POIs included in the

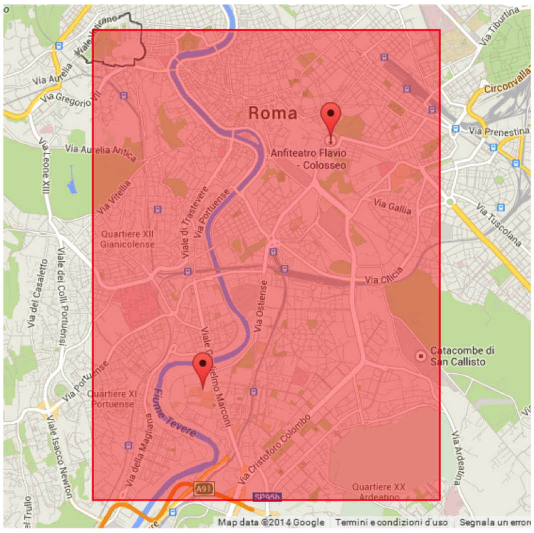

Fig. 11 Detection of the region boundaries for the POIs selection.

database that fall within this area, form the graph nodes. Such POIs are then filtered based on the contextual information. For instance, time and weather conditions can be used to rule out all the POIs that would be not valid for the current situation. The schema of the system is reported in Figure 13. The edge inference and the graph construction occur as follows. The information related to an edge comprises the shortest path to get from one node to another and the traveling time, taking into account the user's means of transport. Obviously, if the user is walking, the edge weight will be increased. Such information is obtained through the Google Maps API [8]: for each pair of nodes (e_i, e_j) the system asks Google for the traveling time from e_i to e_j and the traveling time to e_j to e_i, thus creating the edge. Starting and end nodes are slightly different from the other nodes: while the latter have both incoming and outgoing edges, the starting node has only outgoing edges, the end node only incoming edges. Once all the edges are inferred, a complete graph from the starting node to the end node is obtained. Then, a routing algorithm is executed on it. Clearly, our aim is to lead the user to the final node through the least expensive path (i.e., the shortest), which at the same time includes both the most popular and user-personalized venues. The basic idea is,

[8] https://developers.google.com/maps/ (Accessed: 29 November 2018).

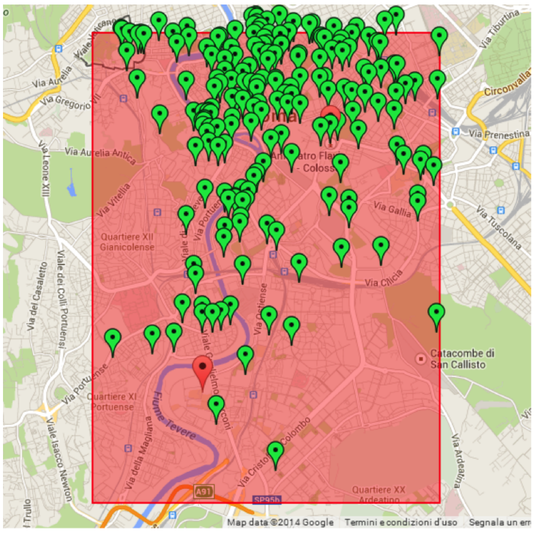

Fig. 12 Selected POIs in the detected region.

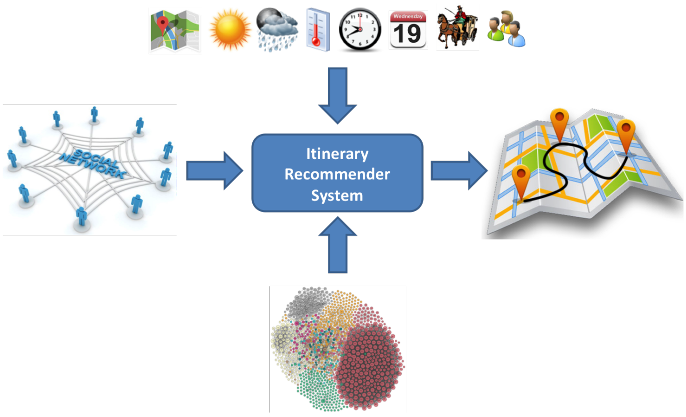

Fig. 13 Schema of the Itinerary Recommender.

therefore, to look for all the possible eligible routes (i.e, those that fall within the time frame set by the user) and, then, sort such routes according to a certain score function.

First, we thought to use the Dijkstra's algorithm (Dijkstra, 1959) for the path search: given a weighted graph (V, E) with non-negative costs, a source node s, and a destination node d, the Dijkstra's algorithm is able to solve the problem of the shortest path from the origin to all the other nodes in time $O(V^2)$ and, therefore, it seemed an excellent starting point on which to rely some subsequent reasoning. The pseudocode of the algorithm is shown in Algorithm 1.

However, it is not the ideal algorithm for our goal as it does not take advantage of all the time available to the user or maximize the number of places to visit. In fact, the Dijkstra's algorithm searches for the shortest path, taking into account only the cost, that is, the time that is taken to arrive from the source node to the destination node. Therefore, it does not consider user's preferences in any way, which is not insignificant for a recommender

Algorithm 1 Dijkstra's algorithm

```
Input:
    graph G=(V,E)
    start node s
    edge lengths l

    // initialize
s.dist ← 0
for each node v ∈ V − {s} do
    v.dist ← ∞
end for

insert all nodes in a priority queue pQ keyed by the dist field
while pQ is not empty do
    v ← remove node in pQ with smallest dist field among queue elements
    for each node w such that (v, w) ∈ E do
        if w.dist > v.dist + l(v, w) then
            w.dist ← v.dist + l(v, w)
            update pQ to reflect changed value of w.dist
            w.pred ← v
        end if
    end for
end while
```

system. Consequently, we looked for another way that could fill these gaps. Based on the system presented in (Hagen et al., 2005), an algorithm was devised to consider all this. More specifically, starting from the itinerary made up of only the starting and end points, further POIs are gradually inserted until all the available time has been spent. This insertion is not random, but occurs while sorting the remaining POIs based on several factors, such as popularity and distance. We gave the name of PVTour (PopularVenueTour) to this algorithm, whose pseudocode is shown in Algorithm 2.

The routing algorithm returns many itineraries from the starting node to the end node. In order to obtain the first k of them, which maximize the user's satisfaction, the following scoring function is used:

$$score(u, v_i) = \alpha \sum_{i=1}^{n} pop(v_i) - \beta \sum_{i=1}^{n-1} dist(v_i, v_{i+1}) + \gamma f(n) + \delta \sum_{i=1}^{n} sim(u, v_i) + \theta \sum_{i=1}^{n} soc(u, v_i) \quad (9)$$

where u is the user and v_i is th i-th venue, with i from 1 to n. This function is made up of several terms. All of them are normalized and weighed by constants whose values were set through gradient descent optimization performed on a dataset extracted from Foursquare. More specifically,

- The first term denotes the relevance/popularity level of individual POIs. The popularity level of an itinerary is calculated by summing all the users' check-ins in each POI;
- The second term represents the total distance of the itinerary, which is given by the sum of the traveling times of each single path. This term is the only negative one, in order to give greater relevance to the shorter routes than the longer ones;
- The third term takes into account the number of venues in the path;

Algorithm 2 PVTour

```
Input:
    graph G=(V,E)                                   ▷ A graph
    startNode                                       ▷ Start node
    endNode                                         ▷ End node
Output:
    routeList                                       ▷ A list of route

    // initialize
routeList ← ∅
route ← [startNode, endNode]
candidateList ← list of all venues in the graph

while candidateList is not empty do
    move v at the head of candidateList
    RE(route,candidateList,true)
end while
return routeList
```

Algorithm 3 Route Enhancement (RE)

```
Input:
    route                                           ▷ A route
    candidateList                                   ▷ A list of candidate venues

if firstCall = false then
    candidateList ← sort candidateList
end if
for each venue v ∈ candidateList do
    n ← remove first venue from candidateList
    newRoute ← try to insert n into route
    if exists newRoute then
        RE(newRoute,candidateList,true)
    end if
end for
add route to routeList
```

– The fourth term expresses the path affinity with the user's taste: for every POI, its affinity with respect to the user's interests is assessed. Such a value is computed through the cosine similarity function between the weight vector representing the user profile and the weight vector representing the POI category;
– The last term gives the social contribution, which includes information derived from social networks. The assumption behind this is that if some friends of the active user perform check-ins in a given POI, it receives a bonus value depending on the check-ins amount. More formally, given a user u, the set of her friends can be denoted by:

$$f_{friends}(U) \rightarrow \mathcal{P}(U) \qquad (10)$$

If we consider the subset of user's friends that visited the same venue v_i as the u-user did as follows:

$$f^{(v_i)}_{friends}(u) = u_i \in f_{friends}(u) | u_i \text{ and } u \text{ shares the check-in } v_i \tag{11}$$

We can define the $soc(u, v_i)$ function as follows:

$$soc(u, v_i) = \left[1 + e^{-k \frac{f^{(v)}_{friends}(u)}{f_{friends}(u)}} \right]^{-1} \tag{12}$$

which basically resembles a logistic function which behaves linearly when the number of friends that share a check-in with u is limited, and reduces the growth of this term when the number of shares moves towards the total number of friends of u.

3.2 Data Extraction from LOD

Each single point of interest that composes the final itinerary is created by extracting data available in the LinkedGeoData dataset[9] using appropriate SPARQL queries. An example of such queries is shown in Query1, which allows POIs to be filtered based on the current context of use (i.e., location and opening hours).

Query 1 Example of SPARQL query

```
PREFIX lgdo: <http://linkedgeodata.org/ontology/>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns♯>
PREFIX g: <http://www.w3.org/2003/01/geo/wgs84_pos♯>
PREFIX lgd-addr: <http://linkedgeodata.org/ontology/addr3A>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema♯>

SELECT ?obj (SAMPLE(?l) AS ?label) (SAMPLE(?lat) AS ?latitudine)
(SAMPLE(?long) AS ?longitudine) (SAMPLE(?openHours) AS ?open)
(SAMPLE(?tipo) AS ?category) (SAMPLE(?street) AS ?s) (SAMPLE(?number) AS ?numb)
WHERE {
?obj rdf:type ?tipo
FILTER regex(str(?tipo), "http://linkedgeodata.org/ontology/Museum")
?obj rdf:type lgdo:Museum ;
rdfs:label ?l ;
g:lat ?lat ;
g:long ?long
OPTIONAL
 ?obj lgdo:opening_hours ?openHours
OPTIONAL
 ?obj lgd-addr:street ?street
OPTIONAL
 ?obj lgd-addr:housenumber ?number
FILTER ( ( ( ( ?lat > 41.79 ) && ( ?lat <= 41.99 ) ) && ( ?long > 12.39 ) ) && ( ?long <= 12.59 ) )
}
GROUP BY ?obj
```

[9] http://linkedgeodata.org/ (Accessed: 29 November 2018).

The SPARQL endpoint for the dataset is `http://linkedgeodata.org/sparql`. Initially, the query searches for all the objects in the dataset (?obj) that have the field *rdf:type* equal to `http://linkedgeodata.org/ontology/Museum`. In this way, it is possible to retrieve all the elements that have been tagged as Museum. At this point, the characteristics of the POI are saved so that they can be used as a filter to allow us to retrieve the results. More specifically, the previous query retrieves the name (label) of the POI, its geographical coordinates (latitude and longitude) and, if available, the opening hours of the venue, the street in which it is located and the street number. Then, the results are filtered according to the location for obtaining the POIs available in the surrounding of the itinerary to be determined. Once all the information has been obtained, the venues are created and subsequently filtered based on the current context of use.

3.3 Evaluation of the itinerary recommendations

This section summarizes the findings of the experimental evaluation of the itinerary recommender. We focused here on the the user's selection of recommended contents (content layer). Tests were performed on a sample of 40 real users, whose characteristics are reported in Table 5. Also in this experimental evaluation, all the necessary security and privacy measures for guaranteeing participants the protection of their data were taken (see Sect. 2.5.2).

Table 5 Characteristics of testers.

	Item	Frequency	Percentage
Gender	Male	23	58%
	Female	17	43%
Age	18-30	34	85%
	31-50	3	8%
	51-70	3	8%
Profession	Student	16	40%
	Teacher	1	3%
	Employee	16	40%
	Freelancer	4	10%
	Unemployed	3	8%
Evaluated Contexts	Rome	11	28%
	London	8	20%
	Paris	5	13%
	New York	10	25%
	Florence	6	15%

All testers had an active account on one or more social networks. First of all, we created five different scenarios with different contexts, so as to have the most possible varied situations. An example is: "suppose yourself to be in Rome, in Termini Railway Station, it is raining, on Monday, at 2 pm, by car, traffic is heavy; you have six hours to get to Piazza Navona". Each tester was presented with one of these five scenarios randomly chosen. Based on the scenario and the user profile, the system returns the first ten itineraries, for each of which the user has to express her satisfaction through a five-point Likert scale. Itineraries are randomly returned to the target user, so as to preserve the rating fairness.

Also in this case, the performance of the recommender was assessed in terms of the *normalized* version of *Discounted Cumulative Gain (nDCG)*. As seen in Section 2.5.3, *nDCG* is

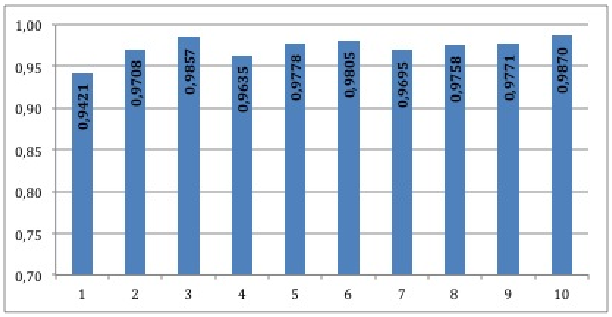

Fig. 14 Values of *nDCG* for different rank levels.

usually truncated at a particular rank level to emphasize the importance of the first retrieved documents. To focus on the top-ranked items, we considered the $nDCG@p$ by analyzing the ranking of the top p itineraries in the recommended list with p from 1 to 10.

The graph shown in Figure 14 illustrates the average values of *nDCG* for each position, based on the ratings of 40 users. The x-axis reports the rank (from 1 to 10), while the y-axis displays the respective value of *nDCG*. Analyzing the obtained results, we can notice how the system allowed us to achieve high *nDCG* values, showing high accuracy performance.

3.3.1 ANOVA Test

Also in this case a statistical significance analysis of the obtained experimental results has to be performed. The most important data for the ANOVA test is shown in Table 6.

Table 6 Data for ANOVA test.

	Dijkstra	PVTour	PVTour+Cont.	PVTour+Cont.+Soc.	Total
Size	40	40	40	40	160
Mean	2.125	3.225	3.725	3.775	3.213
Stand. Dev.	1.749	1.06	0.741	0.758	1.077
SSQ_a	47.306	0.006	10.506	12.656	70.475
SSQ_e	122.375	44.795	21.975	22.975	212.3

For each of the four algorithms we first calculated the sample size n_g (the number of the submitted ratings), the average of the ratings of each algorithm m_i with the relative standard deviation σ_i, and the total average m with the relative total standard deviation σ. Then, we determined the sum of the squared mean deviations of each algorithm m_i from the total mean m (SSQ_a, or intergroup squares) and the sum of the mean squared deviations of the single ratings v_{gj} compared to the mean m_i of the group to which they belong (SSQ_e, or intragroup squares), according to the following formulas:

$$SSQ_a = \sum_{g=1}^{G} n_g(m_g - m)^2 \tag{13}$$

$$SSQ_e = \sum_{g=1}^{G} \sum_{j=1}^{n_g} (v_{gj} - m_g)^2 \tag{14}$$

The F_{test} variable becomes:

$$F_{test} = \frac{SSQ_a/(G-1)}{SSQ_e/(n-G)} \tag{15}$$

where G is the number of algorithms (i.e., 4), n_g is the number of votes for each algorithm (i.e., 40), and n is the total number of observed ratings (i.e., 160). After performing all the calculations, we get (see Table 7)

$$SSQ_a = 70.475 \tag{16}$$

$$SSQ_e = 212.3 \tag{17}$$

and, hence,

$$F_{test} = \frac{70.475/(4-1)}{213.3/(160-4)} = 17.26 \tag{18}$$

This value is compared with the values of a random variable F_{crit} of Snedecor with 3 (i.e., $4-1$) and 156 (i.e.,$160-4$) degrees of freedom. The critical value is the number that the F variable must exceed to reject the null hypothesis. If we accept a false positive percentage of $5\% = 100\% - 95\%$ (probability level $p = 0.05$) then this critical value is:

$$F_{crit}(0,95;3;156) = 2.66 \text{ (vedi Fig. 15)} \tag{19}$$

Therefore, being $17.26 \gg 2.66$, namely, $F_{test} \gg F_{crit}$, we can reject the null hypothesis

Table 7 Data for the computation of the F_{test} value.

	Deviation	Degrees of freedom	Variance
Between groups	70.475	3	23.942
Within groups	212.3	156	1.361

which provided for the absence of effects and it is concluded that at least one of the two variables, that is, context and social, plays a significant role for the purpose of the recommendation. The complete algorithm is, hence, more effective than the one without context and social data, and even more effective than the Dijkstra's algorithm, as shown in Table 6.

4 Related Work

The research literature is rich in recommender systems in the cultural heritage area (Ardissono et al., 2012), but - as far as we know - few of them take advantage of both social and linked open data for providing users with items relevant to their actual interests and preferences.

g2 \ g1	1	2	3	4	5	10	20	30	100
1	161,45	199,50	215,71	224,58	230,16	241,88	248,01	250,10	253,04
2	18,51	19,00	19,16	19,25	19,30	19,40	19,45	19,46	19,49
3	10,13	9,55	9,28	9,12	9,01	8,79	8,66	8,62	8,55
4	7,71	6,94	6,59	6,39	6,26	5,96	5,80	5,75	5,66
5	6,61	5,79	5,41	5,19	5,05	4,74	4,56	4,50	4,41
6	5,99	5,14	4,76	4,53	4,39	4,06	3,87	3,81	3,71
7	5,59	4,74	4,35	4,12	3,97	3,64	3,44	3,38	3,27
8	5,32	4,46	4,07	3,84	3,69	3,35	3,15	3,08	2,97
9	5,12	4,26	3,86	3,63	3,48	3,14	2,94	2,86	2,76
10	4,96	4,10	3,71	3,48	3,33	2,98	2,77	2,70	2,59
20	4,35	3,49	3,10	2,87	2,71	2,35	2,12	2,04	1,91
30	4,17	3,32	2,92	2,69	2,53	2,16	1,93	1,84	1,70
100	3,94	3,09	2,70	2,46	2,31	1,93	1,68	1,57	1,39
150	3,90	3,06	2,66	2,43	2,27	1,89	1,64	1,54	1,34
156	3,90	3,05	**2,66**	2,43	2,27	1,89	1,64	1,53	1,34
200	3,89	3,04	2,65	2,42	2,26	1,88	1,62	1,52	1,32

Fig. 15 Snedecor table.

4.1 Semantic POI recommender systems in the cultural heritage domain

In the last years, some approaches for providing recommendations in CH based on semantic data have been presented, especially exploiting domain ontologies and vocabularies, such as (Ruotsalo et al., 2013), (Moreno et al., 2013), (Bartolini et al., 2013), (Albanese et al., 2011), (Wang et al., 2009).

Smartmuseum (Ruotsalo et al., 2013), is a mobile recommender system that exploits ontologies to provide the user with context-aware personalized access to digital CH (such as museums or buildings of architectural interest, and objects on those sites, such as sculptures or other works of art, and provides explanatory descriptions and multi-media content associated with individual objects). The Smartmuseum system utilizes Semantic Web languages (RDF, RDFS) as the form of data representation. Ontologies are used to bridge the semantic gap between heterogeneous content descriptions, sensor inputs, and user profiles.

SigTur/E-Destination (Moreno et al., 2013) is a Web-based system that provides personalized recommendations of touristic activities in the region of Tarragona. Such activities are classified using an ontology, which guides the reasoning process. The recommender takes into account many kinds of data: demographic information, travel motivations, the actions of the user on the system, the ratings provided by the user, the opinions of users with similar demographic characteristics or similar tastes, etc.

In (Bartolini et al., 2013) a recommender engine in the CH domain is described. Such a system can provide context-aware recommendations of heterogeneous multimedia data (i.e., images, videos/shots, documents) based on low level descriptors and semantic annotations of multimedia resources and a user model expressed as a list of tags.

Albanese et al. (2011) present a strategy for a semantic multimedia recommender system that computes customized recommendations using semantic contents and low-level features of multimedia objects, past behavior of individual users and behavior of the users community as a whole.

Wang et al. (2009) present a content-based recommender able to suggest art-related concepts on the basis of user ratings of artworks. To mitigate the semantic complexity in the datasets, the authors identify several semantic relations within one vocabulary and across different vocabularies (AAT, ULAN, SKOS).

More recently, few systems have stared to exploit the great opportunity provided by (semantic) Linked Open Data (Thalhammer, 2012), such as (Varfolomeyev et al., 2015) and (Lo Bue et al., 2015).

Varfolomeyev et al. (2015) provide smart personal assistants for the tourism recommendations, exploiting semantic relations over data on historical objects available as LOD, and thus augmenting POIs with historical facts. The user preferences are determined by a set of categories, which define the POIs that the user has already chosen. They exploited well-known mathematical methods to compute distance among categories on the smart space.

Lo Bue et al. (2015) propose an approach to provide users with context-aware personalized recommendations of CH resources based on their previous visit experiences. The *cultural objects* are represented through semantic and LOD models. The matching between those resources and the user's interests is performed using a graph similarity technique, reverse path length applied to a graph of ontology terms extracted from DBpedia. A first fundamental difference with our approach lies in the user model. The authors of both the approaches represent the user's interests as sets of words extracted from textual descriptions of resources the users appreciated/visited. On the contrary, we extract the interests from the social media activities of users and model users through a heterogeneous social graph. A further difference consists in the metric used to evaluate the semantic relatedness between user's interests and available resources. Varfolomeyev *et al.* exploit probabilistic distance, Lo Bue *et al.* analyze the reverse path length, while we consider the total number of occurrences in a set of graph patterns.

Table 8 summarizes the main features of the cited work with respect to Cicero.

4.2 Itineraries recommendations

Several approaches in the literature recommend not only single POIs but also complete itineraries with a set of POIs. This can be done taking into account several features of the path, beside traditional efficiency (length and speed (Ludwig et al., 2009; Chang et al., 2011), such as pleasantness (Quercia et al., 2014), accessibility (Comai et al., 2017), safety (Kim et al., 2014).

An idea behind some recent work is to use geo-referenced online content (e.g., Flickr[10] pictures) to learn and recommend popular trajectories such as (Baraglia et al., 2013), as we did in Cicero using Foursquare check-ins to infer popular paths. Others exploit them as sources for mining popular venues (Brilhante et al., 2013), travel sequences (Zheng and Xie, 2011) or, more in general, travel attractiveness (Waga et al., 2012).

More recently, given the popularity of location-based social networking applications, researchers have also been able to provide personalized paths. In fact, such large amount of geo-tagged photos shared on social media allow location-based services to mine also demographic information by detecting people attributes by means of image analysis techniques.

For example, Cheng et al. (2011) annotated historical data of traveled paths with demographic information (such as family, friends, couple, etc) and used a Baeysian learning model to generate personalized travel recommendations based on user profile.

Kurashima et al. (2010) propose a travel route recommendation method that makes use of the photographers' histories as held by Flickr. The authors profiled users according to

[10] https://www.flickr.com/ (Accessed: 29 November 2018).

Table 8 Comparison between Cicero and some state-of-the-art POI recommenders.

System	UM Representation	UM Source	CH Representation	Match CH-UM	Recommender Techniques	Recommender Output	Social-based
Cicero	graph with user interest	Facebook (like)	LOD	graph-based sim (total number of occurrence)	Mixed content-based collaborative filt.	POIs (CH and restaurants,..)	yes
(Ruotsalo et al., 2013)	ontology-based user profile	feedback on proposed object	ontologies and RDF data	ontology-based reasoning	context-aware rec.	historical buildings and objects on those sites	no
(Moreno et al., 2013)	ontology-based user profile	user ratings and user behaviour in the system	turist ontology	direct match and interest propagation	Mixed, content-based collaborative filt. .	touristic activities	no
(Albanese et al., 2011)	matrix of user preferences	past interactions	semantic metadata	metrics for semantic relatedness of concepts based on a vocabulary (Li-Bandar-McLean, Wu-Palmer, Rada, Leacock-Chodorow)	Mixed, content-based collaborative filt. .	digital collections	no
(Bartolini et al., 2013)	list of tags	past user interactions (objects watched) overall behaviour of the whole community of users to	semantic GIS repres. (CIDOC-CRM model)	co-occurrence-based distance function	context-aware rec.	multimedia content	no
(Wang et al., 2009)	preferences	ratings on artworks	semantic repres. ((AAT, ULAN, SKOS)	-	content-based	artworks	no
(Varfolomeyev et al., 2015)	list of categories	POIs already visited	LOD	probabilistic distance	content-based	POIs (historical)	no
(Lo Bue et al., 2015)	bag-of-words	past user interactions	LOD	graph-based sim (reverse path length)	implicit content-based	POIs (CH)	no

their past travel histories using geotagged photos. Recommendations are performed by a photographer behavior model, which estimates the probability of a photographer visiting a landmark, and incorporate user preference and present location information.

Takayuki et al. (2005) dynamically suggest new POIs according to the last visited ones, their characteristics and categories, personalizing the recommendation as the current context evolves. Multiple conflicting criteria and undesirable situations that may result in the modification of the current schedule can also be considered by monitoring the user's behavior.

Lim et al. (2015) propose an algorithm for recommending personalized tours using POI popularity and user interest preferences, which are automatically derived from real-life travel sequences based on geotagged photos. They consider user trip constraints such as time limits. In our work, we also reflect levels of user interest based on visit durations.

Di Bitonto et al. (2010) propose a method for generating tourist itineraries in knowledge-based recommender systems. The method is based on a theoretical model that defines space-time relations among items of intangible cultural heritage (called events) and on transitive closure computation of the relations, that is able to construct chains of events. The output is a sequence of attractions or spots to be visited, filtered according to the tourist's constraints (day of visit, cost, and so on) specified in the request.

Table 9 Comparison between Cicero and some state-of-the-art itinerary recommenders.

System	User Features	Friends Taste	Context	Popularity	Path's Features
Cicero	taste on categories and constraints	check-ins (Foursquare)	location, time, weather, means of transport	in Foursquare	shortest path
(Baraglia et al., 2013)	-	-	-	picture (Flickr)	-
(Yoon et al., 2012)	-	-	time	GPS trajectories	-
(Cheng et al., 2011)	demographic features	-	-	-	-
(Kurashima et al., 2010)	user preferences	-	location	-	-
(Takayuki et al., 2005)	user needs and constraints	-	time, user's schedule	-	resulting from conflict management
(Lim et al., 2015)	user interest and constraints	-	time	yes	-
(Di Bitonto et al., 2010)	user's constraints day of visit, cost,	-	time and place	-	-
(Hagen et al., 2005)	taste on categories and constraints	-	-	-	-

In Cicero we consider user preferences for POIs categories exploiting information provided in social networks, as well as information about the user's friends.

Time constraints are in general more sensitive in itinerary recommendation. Determining the proper visiting time of each place and the proper transit time from one place to another is fundamental for defining route goodness functions (Hsieh et al., 2014). Yoon et al. (2012) explicitly model both the available time of the user and the staying time for each POI included in the itinerary. Techniques based on signal processing are proposed for including time dimension in context-aware recommendation tasks (Biancalana et al., 2011; Arru et al., 2013; Sansonetti et al., 2017).

The recommender system in (Hagen et al., 2005) makes use of an ontology to semantically model the interests of the users and the itineraries to suggest. The user interests are explicitly stated at the beginning of the interaction by means of a multiple choice-paradigm implemented in the user interface by means of a list of POI categories. The recommender proposes one or more itineraries that maximize the number of POIs given the initial time frame. Non-accessible POIs (e.g., opening hours do not fall back into the time frame) are not included in the itineraries. The recommender makes also use of multiple online services that provide structured and semantic information, such as the list of POIs in a given range, and the category of each POI. The proposed approach is one of the first attempts to make use of structured and semantic information for recommending itineraries, although it does not use LOD and limits the user profile to the initial information expressed by the users.

Table 9 summarizes the features considered in path recommendations by some of the cited work with respect to Cicero.

5 Conclusions

In this article, we have presented a hybrid recommender system in the cultural and artistic area. Such a recommender takes into account the activities on Facebook of the target user and her friends. The system integrates collaborative filtering and community-based algorithms with semantic technologies to exploit linked open data sources in the recommendation process. Furthermore, the proposed recommender provides the target user with personalized and context-aware itineraries among cultural POIs.

Regarding the limitations of our approach, we must first refer to the similarity measure adopted. We decided to implement a classical graph-based (distance-based) similarity measure based on shortest-paths similar to (Rada et al., 1989), since it was simple but effective in finding closer concepts. The main advantage of this kind of measures is their unsupervised nature, low cost and lack of dependence on any external corpus (Harispe et al., 2015). We are aware that their main drawback is the absence of extensive control over the semantics which are not taken into account; this generates difficulties in justifying, explaining, and therefore analyzing the resulting scores. Thus, as future work, we will apply finer measures based on the graph property model, such as feature-based measures (Tversky (1977) and its variations, Jaccard, Dice, etc.), since considering and comparing the properties of concepts, allows us to better consider its meaning (Meymandpour and Davis, 2016).

Several interesting results have emerged, which pave the way for future developments. Among others, we intend to enrich the user profiles with additional information from social and LOD sources, as well as information related to their emotions and personality (Bologna et al., 2013; Onori et al., 2016). Furthermore, we would like to supplement our system with a sentiment analysis module that enables them to better capture the users' attitude towards the cultural resources referenced by them and their friends on social contexts (Feltoni Gurini et al., 2014, 2018). Moreover, we would like to enhance our system with a cross-domain recommendation engine in order to furnish the active user with multimedia and textual content related to the suggested itineraries. From this point of view, in fact, the LOD cloud guarantees an inexhaustible resource of varied information from which to draw heavily. Regarding the interaction design and the proposed user interfaces, we are designing, in a new release of the system, a 3D map-based navigation, in order to improve the user orientation among streets and monuments. We will test the new prototype with usability experts and target users.

References

Albanese M, d'Acierno A, Moscato V, Persia F, Picariello A (2011) A multimedia semantic recommender system for cultural heritage applications. In: Proceedings of the 2011 IEEE Fifth International Conference on Semantic Computing, IEEE Computer Society, Washington, DC, USA, ICSC '11, pp 403–410

Antoniou G, Groth P, Harmelen Fvv, Hoekstra R (2012) A Semantic Web Primer. The MIT Press

Ardissono L, Kuflik T, Petrelli D (2012) Personalization in cultural heritage: the road travelled and the one ahead. User Model User-Adapt Interact 22(1-2):73–99

Arru G, Feltoni Gurini D, Gasparetti F, Micarelli A, Sansonetti G (2013) Signal-based user recommendation on twitter. In: Proceedings of the 22nd International Conference on World Wide Web, ACM, New York, NY, USA, WWW '13 Companion, pp 941–944, DOI 10.1145/2487788.2488088

Baraglia R, Muntean CI, Nardini FM, Silvestri F (2013) Learnext: Learning to predict tourists movements. In: Proceedings of the 22nd ACM International Conference on Information & Knowledge Management, ACM, New York, NY, USA, CIKM '13, pp 751–756, DOI 10.1145/2505515.2505656

Bartolini I, Moscato V, Pensa RG, Penta A, Picariello A, Sansone C, Sapino ML (2013) Recommending multimedia objects in cultural heritage applications. In: Petrosino A, Maddalena L, Pala P (eds) New Trends in Image Analysis and Processing – ICIAP 2013: ICIAP 2013 International Workshops, Naples, Italy, September 9-13, 2013. Proceedings, Springer Berlin Heidelberg, Berlin, Heidelberg, pp 257–267

Biancalana C, Gasparetti F, Micarelli A, Miola A, Sansonetti G (2011) Context-aware movie recommendation based on signal processing and machine learning. In: Proceedings of the 2nd Challenge on Context-Aware Movie Recommendation, ACM, New York, NY, USA, CAMRa '11, pp 5–10, DOI 10.1145/2096112.2096114

Biancalana C, Gasparetti F, Micarelli A, Sansonetti G (2013) An approach to social recommendation for context-aware mobile services. ACM Trans Intell Syst Technol 4(1):10:1–10:31, DOI 10.1145/2414425.2414435

Bizer C, Heath T, Berners-Lee T (2009) Linked data - the story so far. Int J Semantic Web Inf Syst 5(3):1–22

Bologna C, De Rosa AC, De Vivo A, Gaeta M, Sansonetti G, Viserta V (2013) Personality-based recommendation in e-commerce. In: CEUR Workshop Proceedings, CEUR-WS.org, Aachen, Germany, CEUR Workshop Proceedings, vol 997

Brilhante I, Macedo JA, Nardini FM, Perego R, Renso C (2013) Where shall we go today?: Planning touristic tours with tripbuilder. In: Proc. of the 22nd ACM International Conference on Conference on Information & Knowledge Management, ACM, New York, NY, USA, CIKM '13, pp 757–762, DOI 10.1145/2505515.2505643

Brusilovsky P, Karagiannidis P, Sampson D (2001) The benefits of layered evaluation of adaptive applications and services. In: S Weibelzahl DNC (ed) Empirical Evaluation of Adaptive Systems, vol Proc. of Workshop At the Eighth International Conference on User Modeling, UM2001, pp 1–8

Carmagnola F, Cena F, Gena C (2007) User modeling in the social web. In: International Conference on Knowledge-Based and Intelligent Information and Engineering Systems, Springer, pp 745–752

Chang KP, Wei LY, Yeh MY, Peng WC (2011) Discovering personalized routes from trajectories. In: Proceedings of the 3rd ACM SIGSPATIAL International Workshop on Location-Based Social Networks, ACM, New York, NY, USA, LBSN '11, pp 33–40, DOI 10.1145/2063212.2063218

Cheng AJ, Chen YY, Huang YT, Hsu WH, Liao HYM (2011) Personalized travel recommendation by mining people attributes from community-contributed photos. In: Proceedings of the 19th ACM International Conference on Multimedia, ACM, New York, NY, USA, MM '11, pp 83–92, DOI 10.1145/2072298.2072311

Comai S, Bernardi ED, Matteucci M, Salice F (2017) Maps for easy paths (mep): Enriching maps with accessible paths using mep traces. In: Gaggi O, Manzoni P, Palazzi C, Bujari A, Marquez-Barja JM (eds) Smart Objects and Technologies for Social Good, Springer International Publishing, Cham, pp 254–263

De Angelis A, Gasparetti F, Micarelli A, Sansonetti G (2017) A social cultural recommender based on linked open data. In: Adjunct Publication of the 25th Conference on User Modeling, Adaptation and Personalization, ACM, New York, NY, USA, UMAP '17, pp 329–332, DOI 10.1145/3099023.3099092

Dey AK (2001) Understanding and using context. Personal Ubiquitous Comput 5(1):4–7, DOI 10.1007/s007790170019

Di Bitonto P, Di Tria F, Laterza M, Roselli T, Rossano V, Tangorra F (2010) Automated generation of itineraries in recommender systems for tourism. In: Proceedings of the 10th International Conference on Current Trends in Web Engineering, Springer-Verlag, Berlin, Heidelberg, ICWE'10, pp 498–508

Dijkstra EW (1959) A note on two problems in connexion with graphs. Numer Math 1(1):269–271, DOI 10.1007/BF01386390

Feltoni Gurini D, Gasparetti F, Micarelli A, Sansonetti G (2014) iSCUR: Interest and Sentiment-Based Community Detection for User Recommendation on Twitter. In: User Modeling, Adaptation, and Personalization: 22nd International Conference, UMAP 2014, Aalborg, Denmark, July 7-11, 2014. Proceedings, Springer International Publishing, Cham, pp 314–319, DOI 10.1007/978-3-319-08786-3_27

Feltoni Gurini D, Gasparetti F, Micarelli A, Sansonetti G (2018) Temporal people-to-people recommendation on social networks with sentiment-based matrix factorization. Future Generation Computer Systems 78(P1):430–439, DOI 10.1016/j.future.2017.03.020

Fisher R (1925) Statistical methods for research workers. Edinburgh Oliver & Boyd

Gena C, Weibelzahl S (2007) Usability engineering for the adaptive web. In: The Adaptive Web, pp 720–762

Hagen K, Kramer R, Hermkes M, Schumann B, Mueller P (2005) Proc. of the International Conference in Innsbruck, Austria, 2005, Springer Vienna, Vienna, chap Semantic Matching and Heuristic Search for a Dynamic Tour Guide, pp 149–159. DOI 10.1007/3-211-27283-6_14

Harispe S, Ranwez S, Janaqi S, Montmain J (2015) Semantic similarity from natural language and ontology analysis. Synthesis Lectures on Human Language Technologies 8(1):1–254

Hsieh HP, Li CT, Lin SD (2014) Measuring and recommending time-sensitive routes from location-based data. ACM Trans Intell Syst Technol 5(3), DOI 10.1145/2542668

Jameson A (2006) Adaptive interfaces and agents. In: Jacko JA, Sears A (eds) Human-computer interaction handbook (2nd ed.), Mahwah, NJ: Erlbaum, pp 305–330

Järvelin K, Kekäläinen J (2002) Cumulated gain-based evaluation of ir techniques. ACM Trans Inf Syst 20(4):422–446

Karagiannidis C, Sampson DG (2000) Layered evaluation of adaptive applications and services. In: Proc. of the International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems, AH00, Springer-Verlag, London, UK, pp 343–346

Kim J, Cha M, Sandholm T (2014) Socroutes: Safe routes based on tweet sentiments. In: Proceedings of the 23rd International Conference on World Wide Web, ACM, New York, NY, USA, WWW '14 Companion, pp 179–182, DOI 10.1145/2567948.2577023

Kurashima T, Iwata T, Irie G, Fujimura K (2010) Travel route recommendation using geotags in photo sharing sites. In: Proceedings of the 19th ACM International Conference on Information and Knowledge Management, ACM, New York, NY, USA, CIKM '10, pp 579–588, DOI 10.1145/1871437.1871513

Lim KH, Chan J, Leckie C, Karunasekera S (2015) Personalized tour recommendation based on user interests and points of interest visit durations. In: Proceedings of the 24th International Conference on Artificial Intelligence, AAAI Press, IJCAI'15, pp 1778–1784

Lo Bue A, Wecker AJ, Kuflik T, Machì A, Stock O (2015) Providing personalized cultural heritage information for the smart region - A proposed methodology. In: Cristea AI, Masthoff J, Said A, Tintarev N (eds) Posters, Demos, Late-breaking Results and Workshop Proceedings of the 23rd Conference on User Modeling, Adaptation, and Personalization

(UMAP 2015), Dublin, Ireland, June 29 - July 3, 2015., CEUR-WS.org, CEUR Workshop Proceedings, vol 1388, pp 1–7

Ludwig B, Zenker B, Schrader J (2009) Recommendation of personalized routes with public transport connections. In: Tavangarian D, Kirste T, Timmermann D, Lucke U, Versick D (eds) Intelligent Interactive Assistance and Mobile Multimedia Computing, Springer Berlin Heidelberg, Berlin, Heidelberg, pp 97–107

McLaughlin MR, Herlocker JL (2004) A collaborative filtering algorithm and evaluation metric that accurately model the user experience. In: Sanderson M, Jrvelin K, Allan J, Bruza P (eds) SIGIR 2004: Proc. of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Sheffield, UK, July 25-29, 2004, ACM, pp 329–336

Meymandpour R, Davis JG (2016) A semantic similarity measure for linked data: An information content-based approach. Knowledge-Based Systems 109:276–293

Moreno A, Valls A, Isern D, Marin L, Borrís J (2013) Sigtur/e-destination: Ontology-based personalized recommendation of tourism and leisure activities. Eng Appl Artif Intell 26(1):633–651

Nielsen J (1999) Designing Web Usability: The Practice of Simplicity. New Riders Press, Harlow, Essex

Onori M, Micarelli A, Sansonetti G (2016) A comparative analysis of personality-based music recommender systems. In: Proceedings of the 4th Workshop on Emotions and Personality in Personalized Systems co-located with ACM Conference on Recommender Systems (RecSys 2016), Boston, MA, USA, September 16, 2016., pp 55–59

Paramythis A, Totter P, Stephanidis P (2001) A modular approach to the evaluation of adaptive user interface. In: Weibelzahl S, Chin D, Weber G (eds) Proceedings of the First Workshop on Empirical Evaluation of Adaptive Systems, Sonthofen, Germany, pp 9–24

Passant A (2010) Measuring semantic distance on linking data and using it for resources recommendations. In: Linked Data Meets Artificial Intelligence, Papers from the 2010 AAAI Spring Symposium, Technical Report SS-10-07, Stanford, California, USA, March 22-24, 2010

Pu P, Chen L, Hu R (2011a) A user-centric evaluation framework for recommender systems. In: Proceedings of the Fifth ACM Conference on Recommender Systems, ACM, New York, NY, USA, RecSys ’11, pp 157–164

Pu P, Chen L, Hu R (2011b) A user-centric evaluation framework for recommender systems. In: Proceedings of the Fifth ACM Conference on Recommender Systems, ACM, New York, NY, USA, RecSys ’11, pp 157–164, DOI 10.1145/2043932.2043962

Quercia D, Schifanella R, Aiello LM (2014) The shortest path to happiness: Recommending beautiful, quiet, and happy routes in the city. In: Proceedings of the 25th ACM Conference on Hypertext and Social Media, ACM, New York, NY, USA, HT ’14, pp 116–125, DOI 10.1145/2631775.2631799

Raad E, Chbeir R, Dipanda A (2013) Discovering relationship types between users using profiles and shared photos in a social network. Multimedia Tools Appl 64(1):141–170, DOI 10.1007/s11042-011-0853-7

Rada R, Mili H, Bicknell E, Blettner M (1989) Development and application of a metric on semantic nets. IEEE transactions on systems, man, and cybernetics 19(1):17–30

Ricci F, Rokach L, Shapira B (2015) Recommender Systems Handbook, 2nd edn. Springer Publishing Company, Incorporated

Ruotsalo T, Haav K, Stoyanov A, Roche S, Fani E, Deliai R, Mäkelä E, Kauppinen T, Hyvönen E (2013) Smartmuseum: A mobile recommender system for the web of data. Web Semantics: Science, Services and Agents on the World Wide Web 20:50 – 67, DOI

http://dx.doi.org/10.1016/j.websem.2013.03.001
Sansonetti G, Gurini DF, Gasparetti F, Micarelli A (2017) Dynamic social recommendation. In: Proceedings of the 2017 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining 2017, ACM, New York, NY, USA, ASONAM '17, pp 943–947, DOI 10.1145/3110025.3110149
Sarwar B, Karypis G, Konstan J, Reidl J (2001) Item-based collaborative filtering recommendation algorithms. In: Proc. of the 10th international conference on World Wide Web, ACM Press New York, NY, USA, pp 285–295
Takayuki S, Munenobu N, Naoki S, Yoshihiro M, Keiichi Y, Minoru I (2005) A personal navigation system with functions to compose tour schedules based on multiple conflicting criteria. IPSJ Digital Courier 46(11):2590–2598
Thalhammer A (2012) Leveraging linked data analysis for semantic recommender systems. In: Proceedings of the 9th International Conference on The Semantic Web: Research and Applications, Springer-Verlag, Berlin, Heidelberg, ESWC'12, pp 823–827
Totterdell, Boyle (1990) Adaptive user interfaces. In: Browne D, Totterdell A, Norman D (eds) The Evaluation of Adaptive Systems, Academic Press, London, pp 161–194
Tversky A (1977) Features of similarity. Psychological Review 84(4):327–352
Varfolomeyev A, Korzun D, Ivanovs A, Soms H, Petrina O (2015) Smart space based recommendation service for historical tourism. Procedia Computer Science 77:85 – 91, DOI http://dx.doi.org/10.1016/j.procs.2015.12.363
Waga K, Tabarcea A, Franti P (2012) Recommendation of points of interest from user generated data collection. In: International Conference on Collaborative Computing: Networking, Applications and Worksharing, pp 550–555
Wang Y, Stash N, Aroyo L, Hollink L, Schreiber G (2009) Semantic relations for content-based recommendations. In: Proceedings of the Fifth International Conference on Knowledge Capture, ACM, New York, NY, USA, K-CAP '09, pp 209–210, DOI 10.1145/1597735.1597786
Wang Y, Wang L, Li Y, He D, Liu T, Chen W (2013) A theoretical analysis of NDCG type ranking measures. CoRR abs/1304.6480:1–30
Weibelzahl S (2001) Evaluation of adaptive systems. In: User Modeling, Springer, Lecture Notes in Computer Science, vol 2109, pp 292–294
Weibelzahl S (2003) Evaluation of Adaptive Systems. Dissertation. University of Trier, Germany, Trier
Weibelzahl S, Lauer C (2001) Framework for the evaluation of adaptive CBR-systems. In: Reimer U, Schmitt S, Vollrath I (eds) Proc. of the 9th German Workshop on Case-Based Reasoning (GWCBR01), Aachen: Shakerm, pp 254–263
Weibelzahl S, Weber G (2001) A database of empirical evaluations of adaptive systems. In: Proc. of Workshop Lernen - Lehren - Wissen - Adaptivity (LLWA 01), Research Report in Computer Science Nr. 763, University of Dortmund, pp 302–306
Yoon H, Zheng Y, Xie X, Woo W (2012) Social itinerary recommendation from user-generated digital trails. Personal Ubiquitous Comput 16(5):469–484, DOI 10.1007/s00779-011-0419-8
Zhang T (2004) Solving large scale linear prediction problems using stochastic gradient descent algorithms. In: Proceedings of the Twenty-First International Conference on Machine Learning, ACM, New York, NY, USA, ICML '04, pp 116–123, DOI 10.1145/1015330.1015332
Zheng Y, Xie X (2011) Learning travel recommendations from user-generated GPS traces. ACM Trans Intell Syst Technol 2(1):2:1–2:29, DOI 10.1145/1889681.1889683

Giuseppe Sansonetti is ...

Fabio Gasparetti is Assistant Professor at the Artificial Intelligence Lab of the Department of Engineering, Roma Tre University. His research interests include user modeling and user adapted interaction; recommender systems and context-awareness; adaptive focused crawling.

Alessandro Micarelli is a Full Professor of Artificial Intelligence at Roma Tre University, where he is in charge of the Artificial Intelligence Laboratory at the Department of Engineering. His research interests include adaptive web-based systems, personalized search, user modeling, Artificial Intelligence in Education.

Federica Cena has been an Assistant Professor with the Department of Computer Science, University of Torino, since 2011. In particular, she is involved in user modeling, personalization and ubiquitous computing. In recent years, she is mainly devoted in studying the implications of Internet of Things for user model and personalization. Her research interests include artificial intelligence and human computer interaction.

Cristina Gena carries on research since 2000. She is Associate Professor in Computer Science at the Department of Computer Science, University of Torino. She is teaching at the School of ICT (social Innovation Communication and new Technologies), of which she is also Vice-President, and at the Computer Science phd program. She published more than 80 papers in international journals, books and conferences. She is heading the smart HCI lab of the ICxT Innovation center of the University of Turin. She is UM Inc. Advisory Board member (um.org).