A Framework for Recommending Multimedia Cultural Visiting Paths

(Discussion paper)

Ilaria Bartolini[1], Vincenzo Moscato[2], Ruggero G. Pensa[3], Antonio Penta[3], Antonio Picariello[2], Carlo Sansone[2], and Maria Luisa Sapino[3]

[1] University of Bologna, DISI, Viale Risorgimento 2, 40136, Bologna, Italy
i.bartolini@unibo.it,
[2] University of Naples Federico II, DIETI, via Clusio 21, 80125, Napoli, Italy
{vmoscato,antonio.picariello,carlo.sansone}@unina.it,
[3] University of Torino, DI, Corso Svizzera 185, I-10149, Torino, Italy
{pensa,penta,mlsapino}@di.unito.it.

Abstract. In this work, we present a general framework for Cultural Heritage applications able to uniformly manage heterogeneous multimedia data coming from several web repositories and to provide context-aware recommendation services in order to generate dynamic multimedia *visiting paths* useful for the users during the exploration of different kinds of cultural sites. A specific application of our system within the cultural heritage domain is proposed together with some experimental results.

1 Introduction

The promotion of worldwide Cultural Heritage by means of Information and Communication technologies represents nowadays an important research issue in the international scenario. This challenge is particularly perceived for the rich Italian artistic patrimony, capable of attracting millions of visitors every year to monuments, archaeological sites and museums. Within this context, it should be necessary to provide a cultural environment with several functionalities able to manage knowledge derived from current digital sources describing cultural heritage, such as text descriptions, pictures and videos, in order to allow a tourist visiting a site to enjoy *multimedia stories* in real time so as to enrich his/her cultural experience.

Our goal is to meet the discussed requirements "extending" classical recommendation techniques (*content-based*, *collaborative filtering* and *hybrid* strategies), usually exploited for facilitating the browsing of large web data repositories [9], to support useful *context-aware* services within a single framework. Such services must assist users when visiting cultural environments (indoor museums, archaeological sites, old town centers) containing cultural *Points Of Interest* - POIs - (e.g. paintings of museum rooms, buildings in ancient ruins or in an old town center, etc.) correlated with a large amount of multimedia data available in multiple web repositories.

In the area of Cultural Heritage, there are several multimedia systems designed and developed to help the user's exploration of available multimedia content [8]. Even if these systems have absorbed previous results coming from different multimedia research projects, they also pose new challenges in the recommendation process such as how different multimedia modules can be efficiently integrated, how conflicts coming from the management of heterogeneous data can be resolved or how the user with his/her preferences, habits and social relationships can be considered.

In this paper, we report results from our previous work [3,4], by generally describing a multimedia recommender system able to uniformly manage heterogeneous multimedia data and to provide context-aware recommendation techniques supporting intelligent multimedia services useful for the users.

In details, we address several drawbacks of state-of-the-art approaches: (i) analyzing in a separate way low and high level information, since both contribute to determine the utility of an object in the recommendation process; (ii) exploiting system logs to implicitly determine information about users and the related community, considering their browsing sessions as a sort of "ratings"; (iii) considering as relevant content for the recommendation the features of the object that a user is interested in (e.g. the item user is watching); (iv) exploiting user preferences and other context information (e.g. user location) to perform a pre-fitering of the candidate objects for recommendation; (v) arranging the obtained recommendations in dynamic visiting paths that take into account possible changes in user needs and in the surrounded environment.

2 System Overview

Figure 1 describes at a glance a functional overview of the proposed system in terms of its main components, that we are detailing in the following.

The *Multimedia Data Management Engine* (MDME) is responsible for: (*i*) accessing by the *Indexing and Access Manager* module to the media contents present in several data sources (*Multimedia Data Repositories*), (*ii*) extracting from multimedia data, by the *Feature Extraction* module, high and low level features useful both for indexing aims and to obtain a structured representation of the data (*Structural Description*). In particular, the *Repository Interface* provides a set of Restful API to communicate with the different multimedia repositories (e.g., *Wikipedia*, *Flickr*, *Europeana*, *Panoramio*, *Google Images*, *YouTube*, etc.). The multimedia data gathered from these sources are then stored in a *Multimedia Storage and Staging* area.

The *Sensor Management Middleware* is responsible for deriving, on the base of information accessible via physical sensors (e.g. GPS, WSN), Web-services/API or wrapping techniques, the "knowledge" related to the context in which the user is located. In particular, the *Knowledge Base* of our system consists of the *Contextual Data*, *User Preferences* (explicitly and implicitly captured), *Cultural POI Descriptions* and a *Support Cartography* useful to geo-localize users and visualize their positions with respect to POIs.

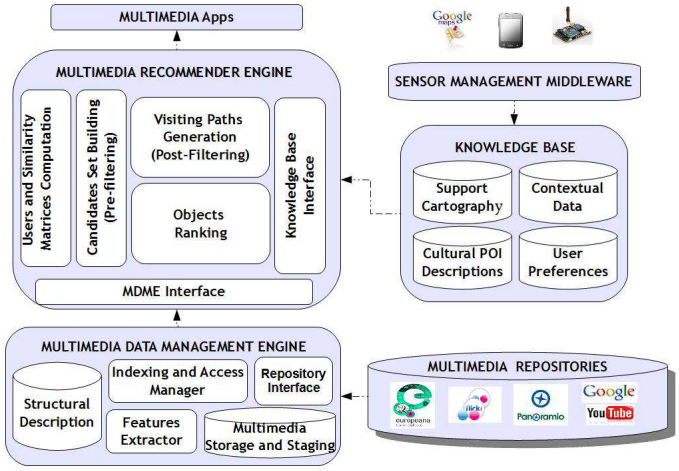

Fig. 1. System Overview.

The *MultiMedia Recommender Engine* provides a set of recommendation facilities for multi-dimensional and interactive browsing of multimedia data related to cultural POIs. In particular, exploiting context information about user location and preferences, the *Candidate Set Building* module selects a set of *candidate* objects for recommendation; successively, the *Objects Ranking* module performs a ranking of such candidates exploiting a proper strategy (that uses the *Users and Similarity Matrices Computation* module). Finally, the *Visiting Paths Generation* module dynamically selects a subset of candidates, on the base of the object that a user is currently watching and context information (e.g., environmental conditions), and eventually arrange them in *visiting paths* as in a touristic guide. All information about the context and multimedia data necessary for the recommendation aims are collected from the system Knowledge Base and Multimedia Data Management Engine using the primitives provided by *Knolwedge Base Interface* and *MDME Interface*, respectively.

Each user device is then equipped with a *Multimedia Guide App* that allows the fruition of multimedia contents and visualization of visiting paths.

3 Management of Multimedia Data

Our data and retrieval models are inspired by the WINDSURF ones [2] as follows.

We have a database $\mathcal{O}$ of M multimedia objects, $\mathcal{O} = \{O^1, \ldots, O^M\}$, such as images, videos, and documents, where each object O is composed of m_O *elements*, $O = \{o_1, \ldots, o_{m_O}\}$ representing regions of an image, shots of a video, and parts of a document, respectively.

Each element o is described by way of *low level features* F^l that represent, in an appropriate way, the content of o (e.g., the color distribution of image's regions or of a video keyframe, relevant terms for documents).[4] In order to enrich data representation, objects are also annotated by the *Features Extractor* module with high level (semantic) descriptors F^h (e.g., annotations concerning the history of a paint, experts' descriptions of an ancient manuscript, etc.). Semantic descriptors can be of two types: (*i*) meta-data, manually provided by users and/or visitors or automatically acquired by external multimedia repositories; (*ii*) (semi-)automatically provided annotations in the form of simple *semantic tags*.

With respect to the retrieval model, given a query object $Q = \{Q_1, \ldots, Q_m\}$ composed of m elements, and an element distance function δ, that measures the dissimilarity of a given pair of elements (using their features), we want to determine the top-k objects in $\mathcal{O}$ that are the most similar to Q.

Low-level similarity between objects is numerically assessed by way of an object distance function d_{F^l} that combines together the single element distances into an overall value. Consequently, object O^a is considered better than O^b for the query Q iff $d_{F^l}(Q, O^a) < d_{F^l}(Q, O^b)$ holds. Often, the overall object distance is computed by aggregating scores of the best possible matching, i.e., the one that minimizes the overall object distance; in this case, the computation of d_{F^l} also includes the resolution of an optimization problem in the space of possible matchings between elements of Q and elements of O. The efficient resolution of queries over low level features is ensured by the *Data Indexing and Access Manager* module which supports indexes built on top of elements (e.g., image regions, and video shots) based on the *M-tree* metric index [6].

With respect to high level features, following the well known *keyword-based paradigm*, given a user-provided set of keywords as query semantic concepts, objects are selected by the *Indexing and Access Manager* module by applying a $co-occurrence$-based distance function d_{F^h} on the feature space $\mathcal{F}$. The search provides the set of objects (i.e., images, videos/shots, documents) that share at least one keyword with the input. This can be carried out efficiently by exploiting the existence of indexing structure, e.g., inverted files.

Finally, both low level features and high level semantic descriptors concur to determine the *multimedia relatedness* $d(O^i, O^j)$ among two objects.

4 Context-Aware Multimedia Recommendation Services

The basic idea behind our proposal is that when a user is near to a cultural POI, the recommender system has to be able to: (i) determine a set of useful *candidate* objects for the recommendation, on the base of user location, needs and preferences (*pre-filtering stage*); (ii) opportunely rank these objects exploiting

[4] Note that, although we consider for an image/keyframe its regions and for each region its visual features, representing an image/keyframe as a set of local features, like SURF [5], is also easily achievable within the WINDSURF framework.

their intrinsic features and users' past behaviors (*ranking stage*); (iii) dynamically, when a user "selects" one or more of the candidate objects, determine the list of most suitable objects (*post-filtering stage*) and eventually arrange such items in appropriate *visiting paths* considering other context information.

4.1 Pre-filtering stage

Each object subject to recommendation may be represented in different and heterogeneous feature spaces. For instance, the picture of a monument may be described by annotations concerning history of the monument, the materials it has been built with, low-level image features, experts' descriptions, visitors' descriptions and reviews, and so on. Each of these sets of features contributes to the characterization of the objects to different extents.

The first step consists in clustering together "similar" objects, where the similarity should consider all (or subsets of) the different spaces of features. To this purpose, we employ high-order star-structured co-clustering techniques [7] to address the problem of heterogeneous data pre-filtering. In this context, the same set of objects is represented in different feature spaces. Such data represent objects of a certain type, connected to other types of data, the features, so that the overall data schema forms a star structure of inter-relationships. The co-clustering task consists in clustering simultaneously the set of objects and the set of values in the different feature spaces. In this way we obtain a partition of the objects influenced by each of the feature spaces and at the same time a partition of each feature space. The pre-filtering stage leverages the clustering results to select a set of candidate objects by using the user's profile, which is modeled as sets of descriptors in the same spaces as the objects' descriptors.

Let $\mathcal{O} = O^1, \ldots, O^M$ be a set of M multimedia objects and $\mathcal{F} = \{F^1, \ldots, F^N\}$ a set of N feature spaces. In our recommendation problem, a user is represented as a set of vectors $U = \{\boldsymbol{u}^1, \ldots, \boldsymbol{u}^N\}$ in the same N feature spaces describing the objects. To provide a first candidate list of objects to be recommended, we measure the *cosine distance* of each user vectors associated to the k-th space, with the centroids of each object clusters in the k-th space. For each space, the most similar object cluster is chosen leading to N clusters $\{X_1^c, \ldots, X_N^c\}$ of candidate objects. Then, two different strategies can be adopted to provide the pre-filtered list of candidate objects $\mathcal{O}^c$: (i) *set-union strategy* - the objects belonging to the union of all clusters are retained, i.e., $\mathcal{O}^c = \bigcup_k X_k^c$; (ii) *threshold strategy* - the objects that appears in at least ths clusters ($ths \in \{1 \ldots N\}$) are retained.

The first strategy is suitable when user's vectors are associated to very small clusters. In any other situation, the second strategy is the most appropriate. As a final step, objects already visited/liked/browsed by the user are filtered out. Notice that, thanks to this approach, users are not described by set of objects, but by sets of features that characterize the objects they visit, like or browse.

4.2 The ranking and post-filtering stages

We want to recommend to a user a subset of $\mathcal{O}^c$ on the base of one or more *target objects*, exploiting objects' intrinsic multimedia *features* and users past browsing *behaviors*. In particular, we use a novel technique that some of the authors have proposed in previous works, combining low and high level features of multimedia objects, possible past behavior of individual users and overall behavior of the whole "community" [1].

Our basic idea is to assume that when an object O_i is chosen after an object O_j in the same *browsing session*, this event means that O_i "is voting" for O_j. Similarly, the fact that an object O_i is very similar in terms of multimedia features to O_j can also be interpreted as O_j "recommending" O_i (and viceversa). Thus, we model a browsing system for the set of candidate objects $\mathcal{O}^c$ as a labeled graph (G, l), where: (i) $G = (\mathcal{O}^c, E)$ is a *directed graph*; (ii) $l : E \rightarrow \{pattern, sim\} \times R^+$ is a *labeling function* that associates each edge in $E \subseteq \mathcal{O}^c \times \mathcal{O}^c$ with a pair (t, w), where t is the type of the edge which can assume two enumerative values (*pattern* and *similarity*) and w is the weight of the edge. A *pattern label* for an edge (O_j, O_i) denotes the fact that an object O_i was accessed immediately after an object O_j and, in this case, the weight w_j^i is the number of times O_i was accessed immediately after O_j; a *similarity label* for an edge (O_j, O_i) denotes the fact that an object O_i is similar to O_j and, in this case, the weight w_j^i is the similarity between the two objects. Thus, a link from O_j to O_i indicates that part of the importance of O_j is transferred to O_i .

Given an object $O_i \in \mathcal{O}^c$, its *recommendation grade* $\rho(O_i)$ is defined as $\rho(O_i) = \sum_{O_j \in P_G(O_i)} \hat{w}_{ij} \cdot \rho(O_j)$, where $P_G(O_i) = \{O_j \in \mathcal{O}^c | (O_j, O_i) \in E\}$ is the set of predecessors of O_i in G, and $\hat{w}_{ij}$ is the normalized weight of the edge from o_j to o_i. In [1], it has been shown that the ranking vector $R = [\rho(O_1) \dots \rho(O_n)]^T$ of all the objects can be computed as the solution to the equation $R = C \cdot R$, where $C = \{\hat{w_{ij}}\}$ is an ad-hoc matrix that defines how the importance of each object is transferred to other objects. Such a matrix can be seen as a linear combination of the local and global *browsing* matrices and of a *multimedia similarity* matrix.

The successive step is to compute *customized* rankings for each individual user. In this case, we can rewrite previous equation considering the ranking for each user as $R_l = C \cdot R_l$, where R_l is the vector of preference grades, customized for a user u_l. We note that solving the discussed equation corresponds to finding the stationary vector of C and it can be solved using the *Power Method* algorithm [1].

The set of final candidates includes the objects that have been accessed by at least one user within k steps from O_j and the objects that are most similar to O_j according to the results of a *Nearest Neighbor Query* ($NNQ(O_j, \mathcal{O}^c)$) functionality. The ranked list can change on the base of weather and environmental situations and, finally, the list of K most important suggested items can be organized, according to the available POIs, into apposite *visiting paths* (considering distances from user location as in $\hat{\mathcal{O}}^c$). The visiting paths will be automatically updated when the set of target objects O_j is modified.

5 A Case Study

We consider as case study the archaeological site of *Paestum*, one of the major Graeco-Roman cities in the South of Italy. Here, the main cultural attractions for a tourist are represented by a set of ancient buildings: three main temples of Doric style. All the buildings are surrounded by the remains of the city's walls. In addition, there is a museum near the ancient city containing many evidences of the Graeco-Roman life (e.g. amphorae, paintings and other objects). Thus, the cited buildings will constitute in such a context the set of cultural POIs for our case study. Users visiting ruins could be happy of having a useful multimedia guide able to describe the main cultural attractions and to suggest automatically *visiting paths* containing multimedia objects of interest.

For instance, when a user is approaching a particular cultural POI (e.g. Temple of Neptune), the related multimedia description and the set of candidate objects (i.e. multimedia data of several kinds as audio, images, video and texts related to the different POIs) are delivered on the user's mobile device (pre-filtering stage). The list of proposed objects depends on the user's preferences (e.g. the majority of items will be images or texts if a user prefers to see such kinds of data and will reveal effective user needs), is initially ordered according to effective user location (i.e. the closest items will appear at the top of list) and contains data grouped by the related cultural POI. Successively, after the user has selected one or more objects as "of interest" (he/she has to select each time at least one target object, for example the item he is currently watching), the recommendation services first perform a final ranking (ranking stage) of all the candidate objects (e.g. images of Temple of Neptune, of other Temples and of Roman Forum) according to their *recommendation grades* and then filters the recommendation list considering only the most similar items to target objects (post-filtering stage). The *Top-K* objects from the obtained recommendations are finally arranged in visiting paths, shown on a proper map together with user's location with respect to POIs. [5]

We evaluated how a visiting path can effectively support browsing tasks of different complexity when multimedia items of interest can come from different cultural POIs placed in not close areas (e.g. buildings in an archaeological site). We decided to implement a web-based application that allows users to browse the entire multimedia collection (about 10,000 items) related to Paestum ruins. In this way, we were able to capture the browsing sessions of about 50 users among graduate students (that used the system for several weeks) and to build a consistent browsing matrices for the described collection. We then asked a different group of 10 profiled people (this group consisted of 5 not-expert users on graeco-roman art, 3 medium expert users and 2 expert users) to complete by the same application several browsing tasks of different complexity within the Paestum ruins collection (15 per user - 5 for each degree of complexity) and without any recommendation facility.

[5] Implementation details concerning the customization of developed prototype for Paestum ruins are reported in [3,4].

Table 1. Comparison between our system and no facilities

	Experts		Medium Exp.		Not Experts	
TLX factor	**With rec.**	**Without**	**With rec.**	**Without**	**With rec.**	**Without**
Mental	29.2	30.1	34.5	36.2	38	45
Physical	29	35	32	39	34.1	48
Temporal	31	35.2	31	39	33	38
Effort	29.4	36	38	45	40	55
Perfomances	75	72	76	75.3	78.5	78.7
Frustation	28	38	29.9	35.2	30	35

After this test, we asked them to browse once again the same collection with the assistance of our recommender system (by facilities provided by visiting paths generated obligating users to choose at least one target object for each suggested POI) and complete other tasks of the same complexity. The strategy we used to evaluate the results of this experiment is based on NASA TLX (*Task Load Index factor*). Thus, we obtained the average results scores for each of three categories of users reported in Table 1 (the lower the TLX score — in the range $[0 - 100]$ — the better the user satisfaction). Not-expert users find our system more effective because they consider very helpful the provided suggestions. Instead, in expert and medium expert users' opinion, our system outperforms a classical touristic guide in every sub-scale except for *mental demand and performances*: this happens because expert users consider sometimes not useful the automatic suggestions, just because they know what they are looking for.

References

1. M. Albanese, A. d'Acierno, V. Moscato, F. Persia, and A. Picariello. A multmimedia recommender system. In *ACM Trans. on Internet Technology*, 13(1), 2013.
2. I. Bartolini, P. Ciaccia, and M. Patella. Query processing issues in region-based image databases. In *Knowl. Inf. Syst.*, 25(2):389–420, Springer, 2010.
3. I. Bartolini, V. Moscato, R.G. Pensa, A. Penta, A. Picariello, C. Sansone, and M.L. Sapino. Recommending multimedia objects in cultural heritage applications. In *Proc. of ICIAP 2013, Workshops*, pages 257–267, 2013.
4. I. Bartolini, V. Moscato, R.G. Pensa, A. Penta, A. Picariello, C. Sansone, and M.L. Sapino. Recommending Multimedia Visiting Paths in Cultural Heritage Applications. To appear in *Multimedia Tools and Applications* Journal, 2014.
5. H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool. Speeded-up robust features (SURF). In *Comput. Vis. Image Und.*, 110(3):346–359, Elsevier, 2008.
6. P. Ciaccia, M. Patella and P. Zezula. M-tree: An efficient access method for similarity search in metric spaces. In *Proc. of VLDB'97*, pages 426–435, 1997.
7. D. Ienco, C. Robardet, R.G. Pensa, and R. Meo. Parameter-less co-clustering for star-structured heterogeneous data. In *Data Min. Knowl. Disc.*, 26(2):217–254, Springer, 2013.
8. K. Kabassi. Personalisation systems for cultural tourism. In *Multimedia Services in Intelligent Environments*, volume 25 of *Smart Innovation, Systems and Technologies*, pages 101–111, Springer, 2013.
9. F. Ricci, L. Rokach, and B. Shapira. *Recommender Systems Handbook*. Springer, 2011.