Recommender Systems for Museums: Evaluation on a Real Dataset

Ivan Keller, Emmanuel Viennet

L2TI Institut Galilée
Université Paris 13, Sorbonne Paris Cité
F-93430, Villetaneuse, France
Emails: (ivan.keller, emmanuel.viennet)@univ-paris13.fr

***Abstract*—This paper discusses the evaluation of several recommendation methods used to suggest relevant contents to museum visitors. We employed traditional recommender systems along with our versatile Social Filtering formalism to test different strategies on a genuine dataset, which was collected during a recent cultural exhibition that received significant interest in Paris, France. The results show the promising potential of recommendation techniques in the not so well explored application domain of museum visit. This work is part of the AMMICO ongoing research project that aims to develop "smart" audio guides for museums.**

Keywords–Recommender Systems; Social Networks; Museum.

I. INTRODUCTION

Museum visitors are often offered a wide selection of artworks. Most frequently, curators design exhibitions with a linear narrative, and wearable audio guides are optionally available to provide information to the visitors. This setting is generally very static with almost no interaction with the user. Gradually, some museums have developed devices offering predefined suggested visit paths with adapted contents according to the type of the audience, e.g., children, families or school groups. The AMMICO project [1] takes one step further: it aims to provide an audio guide prototype with several novel functions exploiting advanced digital information techniques to enhance the visitor's experience.

The most important functionality on this audio guide is the online recommender system, based on the analysis of the visitor behavior: trajectory (measurement of the accurate position of the user, time dedicated to each artwork), interaction with the device ("likes", search for complementary informations) [2]. In the present study, we will focus on the "likes": the visitor explicitly tells the audio guide that he is interested by the current *Point of Interest* (POI) he is viewing. Building such a recommender system faces the well-known challenges: cold start, data sparsity, over-specialization [3]. Recently, we developed a generic formalism that integrates various classic Recommender Systems (RSs) while providing additional novel ways to implement recommendation [4]: the Social Filtering framework (SF). This versatile tool provides an efficient way to test the performances of many different recommendation strategies. We used SF beside other RSs methods in the museum context. This paper analyses the results we obtained on a real dataset collected during a five-month exhibition held by an AMMICO museum partner. Our contribution lies in revealing the promising potential of recommendation techniques in the not so well explored application domain of museum visit.

The paper is organized as follows: Section II summarizes the concepts and notations of SF while Section III briefly explains the operation mode of traditional RSs we also tested; Section IV recalls the evaluation indicators we used to assess the performances of the tested RSs; Section V describes the dataset on which we ran our experiments; the results are displayed and commented in Section VI. Lastly, our conclusion identifies issues and perspectives.

II. SOCIAL FILTERING

This section outlines the concepts behind the Social Filtering formalism. We limited ourselves to the definitions we used in the RSs we tested. For a comprehensive description of this theoretical framework, please refer to [4].

In the RS domain, widely exploited in the marketing industry, it is usual to refer to *users* "consuming" *items*. Here we will employ the vocabulary associated to the museum context: *visitors* "interact" with *POIs* (any object liable to be exposed in a museum). In our experiments (see section V) we will consider POIs "liked" by visitors.

A. Bipartite Graph Visitors×POIs

The SF recommending approach is based on Social Network Analysis. More precisely, it relies on a *bipartite graph* (or *network*) and its *projections*. The bipartite graph we consider is defined over two separate set of nodes: visitors and POIs. A link can only exist between two nodes in different sets. For instance, links connect a visitor to the POIs he has viewed or liked depending on the semantic meaning we choose for the links. Such data structure can be represented by a binary *interaction* (or *preferences*) *matrix* R with L rows corresponding to the visitors and C columns corresponding to the POIs. Matrix R is thus of dimensions $L \times C$. The value r_{ui} at row u and column i is one if visitor u is connected to POI i, and zero otherwise. We denote:

- $r_{u\cdot}$ the line vector of matrix R corresponding to visitor u and $\overline{r_{u\cdot}} = \frac{1}{C}\sum_{i=1}^{C} r_{ui}$ the average number of POIs liked by u;
- $r_{\cdot i}$ the column vector of matrix R corresponding to POI i and $\overline{r_{\cdot i}} = \frac{1}{L}\sum_{u=1}^{L} r_{ui}$ the average number of visitors who liked i.

B. Graph Projections

The bipartite graph is then projected into two (unipartite) graphs, one for each set of nodes: a Visitors Graph and a POIs

 ISBN: 978-1-61208-415-2

Graph. In the projections (see Figure 1 for a toy example), two nodes are connected if they had common neighbors in the bipartite graph. The link weight can be used to indicate the number of shared neighbors. For example, two visitors are connected if they have liked at least one same POI (we usually impose a more stringent condition: at least K POIs). The projected networks can thus be viewed as the network of visitors that liked at least K same POIs (visitors having the same preferences) and the network of POIs liked by at least K' same visitors. Thus, such projected networks are *implicit social networks*: they do not follow from a deliberate social connection like in usual explicit social networks (e.g., friendship networks on Facebook). Instead, they reflect relations derived from similar behaviors of the visitors.

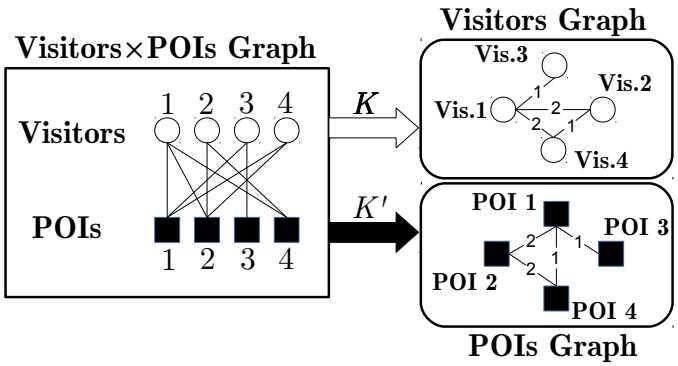

Figure 1. Bipartite Visitors×POIs graph and its projections (K=K'=1).

The general idea of Social Filtering (SF) is to leverage the concepts and methods of network analysis by exploiting the central hypothesis of social recommendation: connected entities (visitors or POIs) are *similar* in some way and thus share tastes or attributes. This property is known as *homophily* [5].

Network structures allow to define similarity between instances, neighborhoods and communities that can be relevant, as we will see, to suggest content to museum visitors. We apply these techniques to the visitors (user-based recommendation) or the POIs projected graphs (item-based recommendation).

C. Similarity Measures

We consider an *active visitor* a for whom we seek recommendations, u being any other visitor. *Asymmetric cosine similarity* [6] is a flexible way for defining the similarity between them:

$$\text{Sim}_{\text{asymcos}}(a,u) = \frac{r_{a\cdot} \cdot r_{u\cdot}}{\|r_{a\cdot}\|^{2\alpha}\|r_{u\cdot}\|^{2(1-\alpha)}} \quad (1)$$

where $r_{a\cdot} \cdot r_{u\cdot} = \sum_{i=1}^{C} r_{ai} r_{ui}$ denotes the dot product of vectors $r_{a\cdot}$ and $r_{u\cdot}$, $\|\cdot\|$ is the associated euclidian norm and α is a real number in $[0,1]$. Note that for $\alpha = 0.5$ we obtain the classic cosine similarity.

The similarity between two POIs i and j (not displayed here for space-saving purposes) can be defined in the same fashion simply by replacing visitors by POIs or, equivalently, rows by columns.

Many more similarity measures are implemented in the SF framework that are not described here because there were not used in the experiments.

D. Neighborhoods

Given a network and a similarity measure we can now define the neighborhood $K(a)$ for an active visitor a (we only give the definition for visitors; neighborhood $V(i)$ for a POI i is defined in a similar manner):

- $K(a)$ is the first circle of neighbors of a in the Visitors Graph, where they can be rank-ordered by their similarity to a.
- $K(a)$ is the *local community* of a in the Visitors Graph, where local communities are defined as in [7].
- $K(a)$ is the *community* of a in the Visitors Graph where communities are defined, for example, by maximizing modularity [8].

As stated in [4], these last two cases are novel ways to define neighborhoods for recommendation systems: visitors in $K(a)$ might not be directly connected to the active visitor a. These definitions thus embody some notion of paths linking visitors through common behavior patterns.

E. Scoring Functions

The last step in the RS pipeline consists in providing a ranked list of recommended POIs to the active visitor a. This is done through a *scoring function* that aggregates the preferences concerning a given POI i. The user-based approach considers the preferences of the neighbors in $K(a)$ about i:

$$\text{Score}_{\text{U}}(a,i) = \sum_{u \in K(a)} f\big(\text{Sim}(a,u)\big)\ r_{ui} \quad (2)$$

Alternatively, the item-based approach takes into account the preferences of a on POIs in the neighborhood $V(i)$ of i:

$$\text{Score}_{\text{I}}(a,i) = \sum_{j \in V(i)} r_{aj}\ g\big(\text{Sim}(i,j)\big) \quad (3)$$

Various functions f and g can be used [9]. For the sake of brevity, we only mention the scoring functions we applied (alternatives are thoroughly described in the reference paper on SF [4]):

- *weighted average popularity* for a of POI neighbors of i in $V(i)$ weighted by their similarity to i:

$$\text{Score}_{\text{I}}(a,i) = \frac{1}{\sum_{j \in V(i) \cap I(a)} |\text{Sim}(i,j)|} \sum_{j \in V(i)} r_{aj}\, \text{Sim}(i,j) \quad (4)$$

  where $I(a)$ is the set of POIs liked by a.
- *scoring function "with locality"*: Aiolli [6] proposed another mechanism to produce locality without having to explicitly define neighborhoods. Function g is defined so as to put more emphasis on high similarities (with high q'):

$$\text{Score}_{\text{I}}(a,i) = \sum_{j \in V(i)} r_{aj}\ \big(\text{Sim}(i,j)\big)^{q'} \quad (5)$$

Finally, POIs are rank-ordered by decreasing scores and the top k POIs $(i_1^a, i_2^a, ..., i_k^a)$ are recommended to a, where $\text{Score}(a, i_1^a) \geq \text{Score}(a, i_2^a) \geq ... \geq \text{Score}(a, i_k^a)$

 ISBN: 978-1-61208-415-2

III. Classical Recommender Systems

Many recommendation techniques are commonly used in industry. We now briefly describe those we used as baseline or for comparison purposes. Some of these methods can be expressed as special cases of the SF formalism.

A. Popularity

Perhaps the simplest recommendation method: we rank POIs by decreasing popularity (the sum of ones in each column of matrix R) and suggest the list of top k most popular POIs to the active visitor. This method is used as a baseline.

B. Collaborative Filtering (CF)

CF is a widely used technique to implement RSs. There exist two main groups of CF techniques: *memory-based* (or neighborhood methods) [9] and *model-based* (or latent factor models) [10]. CF methods use the opinion of a group of similar visitors to recommend POIs to the active visitor.

1) Memory-based Methods: as SF, these techniques rely on the notion of similarity between visitors or POIs to build neighborhood. Unlike content-based methods, that we did not implement, similarity is not computed on the basis of the attributes of the instances (visitors or POIs). Instead, it is based on the shared preferences between two visitors (user-based CF) or the number of common visitors who liked two given POIs (item-based CF). The ways for computing the similarity are the same as described in the Section II-C. In fact, it is easy to observe that CF can be obtained with the SF framework by choosing $K = K' = 1$ as parameters of the projected graphs, cosine similarity (eq. 1 with $\alpha = 0.5$) and a score function as in Section II-E.

2) Model-based Methods: Model-based RSs estimate a global model, through machine-learning techniques, to predict ratings. This generally leads to models that neatly fit data and therefore to good quality RSs. However, learning a model may require a great amount of training data which could be a problem in some applications. Many model-based CF systems have been proposed [11]. One of the most efficient and used model-based methods is *matrix factorization* [12] in which visitors and POIs are represented in a low-dimensional latent factors space. This technique is more suited to feedback with ratings (e.g., zero to five "stars").

C. Association Rules

Association rules mining [13] is a popular technique widely used in marketing in order to find regularities in large databases like products often purchased together. Association rules of length two can be used for recommendation [14]. They are equivalent to the item-based SF choosing asymmetric confidence-based similarity with the suitable parameters, but we preferred to use the classic Apriori algorithm [15] to implement this method.

IV. Evaluation of Recommender Systems

This section recalls the evaluation methods listed in the corresponding part of [4].

For a given active visitor, a RS produces a list of ranked POIs. We want to evaluate whether they are adequate for him. Two scenarios may be considered to evaluate RSs:

- *online evaluation*: if live interactions between visitors and POIs are available we can build RSs on past behaviors and measure the reaction of visitors to the suggestions: does the visitor take them into consideration, like them, etc.? Several groups of control can be considered in order to test different recommendation strategies. This approach is used by merchant websites, for example.
- *offline evaluation* relies on a static dataset of interactions between visitors and POIs on which we simulate recommendation. We underline the fact that this dataset corresponds to visits without recommendation. As it is usual in the evaluation of machine-learning algorithms, the original dataset is split into a training set and a test set. For each visitor of the test set, considered as an active visitor, recommendations are computed based on the data from the training set and from part of the interactions between the visitor and the POIs, taking into account time stamps if available. Evaluation is then computed by comparing the recommended POIs with the remaining real interactions the active visitor had with the POIs not taken into consideration for the recommendation computation.

One may argue that this last approach is flawed since the active visitor would probably have behaved differently if he had been actually recommended with POIs. Moreover, one might also question the relevance of evaluating RSs on the basis of the accuracy to predict the POIs the active visitor liked without being recommended, since the recommendation principle is precisely to suggest contents that the visitor would not have been likely to discover without being recommended. Nevertheless, although these arguments are valid when there exists a vast choice of items like in most marketing situations, in a museum exhibition it is reasonable to assume that the visitors interacted with almost all of the available POIs. Thus, predicting his appreciation on part of the POIs is valuable to evaluate the performance of a RS. Naturally, offline evaluation is unable to take into account the influence of being recommended: there is a psychological bias that is beyond the scope of this study.

A. Performance Metrics

In both situations, for each active visitor of the test set we have a *target set* T_a that represents the set of POIs he liked after being recommended. Let $R_a = (i_1^a, i_2^a, ..., i_k^a)$ be the set of k POIs recommended to a. The metrics classically used in this context are:

- Precision@$k = \frac{1}{L} \sum_a \frac{|R_a \cap T_a|}{k}$
- Recall@$k = \frac{1}{L} \sum_a \frac{|R_a \cap T_a|}{|T_a|}$
- Mean Average Precision:
  MAP@$k = \frac{1}{L} \sum_{a=1}^{L} \frac{1}{k} \sum_{i=1}^{k} \frac{C_{ai}}{i} 1_{ai}$

where C_{ai} is the number of correct recommendations to visitor a in the first i recommendations (Precision@i for visitor a) and $1_{ai} = 1$ if POI at rank i is correct (for visitor a), 0 otherwise.

B. Qualitative indicators

Additionally, more "qualitative" metrics indicate whether all visitors (resp. POIs) receives recommendations (resp. are recommended) or which of the more or less popular POIs are recommended: as we will see, some RSs might be better on

 ISBN: 978-1-61208-415-2

performance metrics and poorer on these qualitative indicators. Let U_{test} denote the set of visitors in the test set and $L_{\text{test}} = |U_{test}|$ the number of visitors in it, then:

- VisitorsCoverage@$k = \frac{\text{nb visitors in } U_{test} \text{ with } k \text{ reco}}{L_{\text{test}}}$

  is the proportion of visitors who get recommendations.

- Average number of recommendations: when visitors coverage is not 100% , i.e, not all visitors got k recommended POIs, we may want to know the average number of POIs recommended for the visitors with partial lists:

  $\text{AvNbRec@}k = \sum_{K=0}^{k-1} K \frac{\text{nb visitors in } U_{test} \text{ with } K \text{ reco}}{L_{\text{test}} - \text{nb visitors in } U_{test} \text{ with } k \text{ reco}}$

- POIs coverage: a high diversity of suggested POIs should result in more attractive recommendations. We thus seek a high proportion of POIs that are recommended:

  $\text{POIsCoverage@}k = \frac{\text{nb distinct POIs in reco lists}}{C}$

- Head/Tail coverage: if we rank POIs by decreasing popularity (number of visitors who liked each POI), we call *Head* the 20% of POIs with highest popularity and *Tail* the remaining 80%. Recommending only most popular POIs will result in relatively poor performances and low diversity. We thus define the rate of recommended POIs in the Head and in the Tail:

  $\text{RateHead@}k = \frac{1}{L_{\text{test}}} \sum_{u \in U_{test}} \frac{\text{nb reco for } u \text{ in Head}}{\text{nb reco for } u}$

  $\text{RateTail@}k = 1 - \text{RateHead@}k$

C. Accuracy vs.Originality

Ideally we would like to produce accurate recommendations that are not too popular, providing the visitors with "pleasant discoveries". This amounts to maximize both MAP@k and RateTail@k. Furthermore, it could be interesting to give more or less emphasis to each of these two metrics depending on the objectives of the recommendation: accuracy vs. originality (or novelty). We propose the following (not normalized) combined indicator:

$$\text{Perf}_e = e\ \text{MAP@}k + (1-e)\ \text{RateTail@}k \qquad (6)$$

where e is chosen in $[0, 1]$ depending on the relative importance we want to give to each aspect of the performance.

Now that we have described how we build RSs and evaluate their performances it is time to expose our experimental results on a dataset extracted from a real museum exhibition.

V. DATASET

This section describes the origin and principal features of the dataset we used to experiment on RSs for museums.

A. General description

From March 11 to August 24, 2014, the Great Black Music exhibition (GBM) took place at the Cité de la Musique in Paris [16]. Both Cité de la Musique and the curator M. Benaïche (director of the digital art factory l'Atelier 144) are members of the AMMICO project consortium. The exhibition showcased the variety and story of the black music around the world by means of numerous multimedia installations. It has been successful with around 76 000 unique visitors.

At the entrance, the visitor got a stereo headset connected to a "smartguide" device which was an Android smartphone running a specifically developed application. Several technological solutions are explored nowadays in order to have direct and accurate information about which exhibition items is viewed by a specific visitor. In the GBM exhibition, visitors simply introduced manually the POI identification number displayed on it in the exhibition space. Since a significant amount of the content was only available through the device (e.g. musical content), visitors were highly motivated to use it. With this equipment, the visitor was able to interact with numerous audiovisual material (11 hours of available recordings in total). Among other features, the device allowed him to create his personal playlist by "liking" (or bookmarking) his favorite contents (POIs) that he could later retrieve online by logging into a dedicated personal webpage [17]. This possibility was a fairly good incentive for visitors to bookmark POIs. On the example displayed on Figure 2 the visitor can add POI n°23 (artist: Tumi & The Volume; song: Asinamali) on his favorites playlist. .

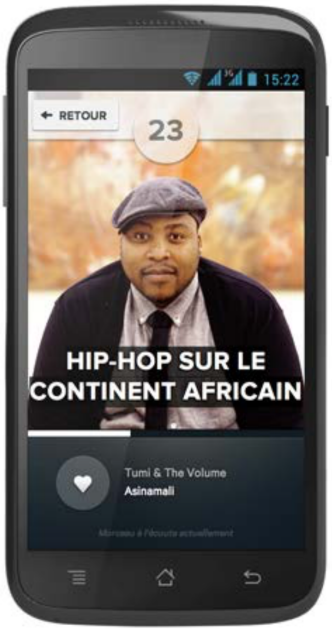

Figure 2. Example of the user interface used at GBM exhibition.

On the museum side, this setting allowed to collect a large amount of data on visitors' behaviors: each time they interacted with a content by the means of the device, which was indispensable given the very nature of the exhibition, the details of the action (visitorId, POIId, time, duration, liked or not) was recorded in the exhibition database. In total, more than 20 million interactions were recorded concerning all the 75 774 visitors.

From this raw database we constructed a dataset focusing on the bookmarked POIs: we considered the bipartite graph consisting of the two sets U and I of the visitors and the liked POIs respectively, where a link connecting a visitor u with a POI i means that "u liked i". We ended up with $|U| = 67\,883$ users, $|I| = 600$ liked POIs (among 608 possible POIs to bookmark) and $|E| = 1\,681\,534$ links (bookmark notifications) between the two sets. Visibly, around 10% of the visitors ($75\,774 - 67\,883$) did not make use of the bookmarking functionality. For them, other strategies might be implemented

TABLE I. GBM DATASET STATISTICS

number of visitors $\|U\|$	67 883
number of POIs $\|I\|$	600
number of "likes" $\|E\|$	1 681 534
min and max visitor degree	1...447
mean and std visitor degree	24.8 ± 34.4
min and max POI degree	1...13 005
mean and std POI degree	2802.6 ± 2481.4

in order to provide recommendation, for example taking into account the time they spent viewing a POI as a measure of their interest. Since this study aims at evaluating different recommendation methods and not at providing explicit live recommendation to visitors we simply excluded them from the dataset.

B. Degree distribution

A vast majority of visitors bookmarked a relatively small amount of POIs and as the number of bookmarked items per visitor increases less visitors are concerned. We thus get a typical *power-law* distribution of visitors nodes' degrees.

POIs are also unevenly popular: 19 of them received a single "like" from the visitors, while others were bookmarked by a large amount of them. The three top favorite POIs are the ones corresponding to the songs "Why I Sing the Blues" by B.B. King, "Sodade" by Cesaria Evora and "Respect" by Aretha Franklin, liked respectively by 13 005 visitors (19.2% of the visitors and 0.77% of the "likes"), 11 801 visitors (17.4% of the visitors, 0.70% of the "likes") and 11 552 visitors (17.0% of the visitors, 0.69% of the "likes").

C. Mega-hubs

In a network, *hubs* are nodes with the highest degree. They are common in social networks as a consequence of the power-law degree distribution. *Mega-hubs* are nodes connected to all or almost all the other nodes of the graph. We generally consider them as not informative in the recommendation context. Moreover, they could undermine the performance of RSs by not allowing the meaningful communities to emerge. Also, from a technical point of view, the presence of mega-hubs causes increased loads of computation and memory. These considerations often lead practitioners to remove mega-hubs from the networks.

What about our dataset? The highest POI degree in the bipartite graph is 13 005 out of a maximum of 67 883 possible links to visitor nodes. The corresponding node in the POIs Graph is a hub connected to less than 20% of the nodes. With respect to visitors, the highest degree is 447 out of a maximum of 600. This relative high value ($\approx 75\%$ of the possible links) indicates the presence of potential mega-hubs in the Visitors Graph. There are several ways to define mega-hubs, but we will not enter into details here: in this work-in-progress study we first used the entire original dataset without eliminating the potential mega-hubs.

Table I summarizes some statistics of the GBM dataset.

VI. EXPERIMENTS

We performed an offline evaluation of several RSs on the GBM dataset. As explained before, it consists in simulating a recommendation scheme and comparing suggested POIs to some visitors with the POIs they actually liked.

A. Experimental Setting

We randomly split the data into 90% visitors for training and the remaining 10% for testing. For training, we used all the interactions (likes) of the 90% visitors. For testing we input 50% of the interactions of each test visitor and compared the obtained recommendation list to the remaining 50% liked POIs.

The RSs methods we chose to evaluate are (see Sections II and III):

- Popularity: used as a baseline;
- Bigrams: association rules of length two implemented using Apriori algorithm [15] with thresholds on support and confidence at 1%. POIs are ranked in decreasing confidence of the rule generating them;
- NMF (non-negative matrix factorization): we used the code associated to [18], with maximal rank 10 and maximum number of iterations 50. These parameters were chosen after several attempts at maximizing the performances, but without a systematic exploration of their value space.
- CF_UB: user-based collaborative filtering implemented as a special case of SF with cosine similarity (eq. 1 with $\alpha = 0.5$) and weighted average popularity (eq. 4 adapted to visitors) as scoring function;
- CF_IB: item-based collaborative filtering implemented as a special case of SF with cosine similarity (eq. 1 with $\alpha = 0.5$) and weighted average popularity (eq. 4) as scoring function;
- SF_IB: item-based social filtering with asymmetric cosine similarity (eq. 1). The neighborhood is defined as the top 10 most similar neighbors in the first circle of neighbors of the POIs graph and we used the scoring function with locality (eq. 5). We explored several combinations for the parameters α (of similarity) and q' (of the scoring function) and reported the most interesting results.
- We tried user-based SF with different parameters α and q, but it shed poor results that we will not report here.

We produced suggested POI lists of length $k = 10$ and evaluated the RSs using all the indicators described in Section IV. In order to obtain more accurate measures we repeated the process on 30 different randomly split training/test sets (90%-10%) and computed the mean value for each indicator.

B. Results

Members of the L2TI laboratory implemented the SF formalism in a Python library released under an open source license [19]. A flexible processing pipeline and the versatility of our SF formalism provided an efficient way to assemble the various elements for experimenting on several methods.

The performances are shown in Table II. Values in **bold** and *italic* indicate respectively the best and second-best performances for the corresponding indicator (except for computation time, "higher is better" for all the performance indicators). We ran our simulations on an Intel Xeon E7-4850 2,00 GHz (10 cores, 512 GB RAM), shared with members of the team so that concurrent usage may have happened in some of the experiments, with impact on reported time. Computing time

 ISBN: 978-1-61208-415-2

TABLE II. PERFORMANCES OF RECOMMENDATION SYSTEMS ON GBM DATASET.

	Popularity	Bigrams	NMF	CF_UB	CF_IB	SF_IB $\alpha = 0.1$ $q' = 3$	SF_IB $\alpha = 0.1$ $q' = 2$	SF_IB $\alpha = 0.9$ $q' = 1$	SF_IB $\alpha = 1$ $q' = 3$
MAP@10	0.035	0.080	0.004	0.005	0.077	**0.087**	*0.086*	0.049	0.015
Precision@10	0.059	0.124	0.078	0.018	0.119	**0.132**	*0.131*	0.092	0.036
Recall@10	0.080	**0.158**	0.105	0.023	*0.152*	**0.158**	**0.158**	0.113	0.067
VisitorsCoverage@10	62.60%	**100%**	*96.93%*	47.22%	85.44%	74.00%	74.23%	84.77%	83.77%
AvNbRec@10	8.52	-	4.67	3.26	6.46	5.63	5.62	5.24	4.88
POIsCoverage@10	1.69%	71.81%	20.41%	93.96%	**99.17%**	93.88%	*95.63%*	**99.17%**	94.61%
RateTail@10	0%	24.12%	6.35%	35.23%	44.83%	35.09%	36.92%	*73.64%*	**91.87%**
$\text{Perf}_{0.0}$	0.000	0.241	0.063	0.352	0.448	0.351	0.369	*0.736*	**0.919**
$\text{Perf}_{0.1}$	0.003	0.225	0.058	0.317	0.411	0.325	0.341	*0.668*	**0.828**
$\text{Perf}_{0.5}$	0.017	0.161	0.034	0.178	0.263	0.219	0.228	*0.393*	**0.467**
$\text{Perf}_{0.9}$	0.031	0.096	0.010	0.039	0.114	0.113	*0.114*	**0.118**	0.106
$\text{Perf}_{1.0}$	0.035	0.080	0.004	0.005	0.077	**0.087**	*0.086*	0.049	0.015
Computation time	**0:00:10**	0:30:00	0:30:00	*0:00:30*	*0:00:30*	*0:00:30*	*0:00:30*	*0:00:30*	*0:00:30*

is thus indicative only (0:00:10 is 10 seconds, 0:30:00 is 30 minutes).

C. Discussion

The first observation one might be inclined to make is that the performance measurements of MAP, Precision and Recall seem to be low in relation to their possible values in $[0, 1]$. However, compared to similar experiments on other datasets, we note that these apparently low values are common in the RS evaluation context (see the results on four publicly available datasets presented in [4]).

Supporting the observations reported in [6] where the author carried out the same kind of experiments but with a different dataset, the results for SF_IB show that asymmetric cosine similarity and the scoring function with locality bring enhanced performances to classic methods, provided a suitable choice for the parameters α and q'. It outperforms in all indicators except for VisitorCoverage@10: it is able to provide a full list of ten recommended POIs for a maximum of around 85% of the visitors. The remaining visitors received an average of five suggested items. It gives a significant improvement on traditional item-based CF (CF_IB) from which it is derived.

Within the variants of SF_IB when changing the parameters, we note the necessary trade-off between accuracy and originality of the recommendation: when performance metrics increase (MAP, Precision and Recall) it is at the expense of qualitative indicators, especially RateTail. This is well captured by our combined indicator Perf_e.

Following SF_IB, Bigrams presents fairly good relative performances, particularly on VisitorCoverage which is 100%. However, it has low RateTail and the computation time is 60 times longer.

User-based CF does not give good results on this dataset, similarly to all the user-based methods we tried (SF_UB). This could be caused by the presence of mega-hubs in the Visitors Graph. This point would be worth exploring.

NMF performs particularly bad on this dataset, only slightly better than the baseline Popularity. This may be due to the insufficient amount of data necessary to build an accurate model and on the fact that we consider simple binary feedback (liked or not) instead of an explicit rating with which this method is known to perform better.

Finally, the baseline Popularity behaves as expected: by recommending the 10 most popular POIs without taking into account the similarities of visitors' behaviors, its POIsCoverage is dramatically low and the RateTail is null, by definition.

VII. CONCLUSION

We have presented in this paper an evaluation study of several recommender systems applied to the museum visit context. We compared and discussed the performances of different recommendation strategies by evaluating them on a genuine dataset concerning visitor behaviors in a real exhibition. Beside classic recommendation methods, we used a versatile Social Filtering formalism developed and implemented in our laboratory. The results show that promising improvements can be achieved with efficient algorithms provided that parameters are properly adjusted.

We are currently conducting experiments regarding the neighborhoods we consider in the projected graphs: better recommendations could be obtained by taking into account the graph community or local community of the active user as described in Section II-D instead of the simple first circle of neighbors since interesting suggestions of POIs could come from related but not directly connected visitors. In parallel, we are carrying out a set of experiments on modified versions of the present dataset in order to observe the influence of mega-hubs on the recommendation quality. Combining several recommendation methods in what are commonly denominated *ensemble methods* in statistical learning is another direction that could somehow enhance performances.

This application domain raises other issues that may be interesting to investigate: how is recommending content perceived and accepted by museum visitors? Beyond the quality and relevance of the suggested content, what is the influence of the presentation and editorialization in its receptivity?

ACKNOWLEDGMENT

This work was supported by the French AMMICO project funded by Banque Publique d'Investissement (BPI) in the FUI 13 program.

REFERENCES

[1] http://ammico.fr.

[2] R. Fournier, E. Viennet, S. Sean, F. Soulié-Fogelman, and M. Bénaïche, "Ammico: social recommendations for museums," in Proceedings of Digital Intelligence (DI2014), Nantes, France, 2014.

[3] L. Ardissono, T. Kuflik, and D. Petrelli, "Personalization in cultural heritage: the road travelled and the one ahead," User modeling and user-adapted interaction, vol. 22, no. 1-2, 2012, pp. 73–99.

 ISBN: 978-1-61208-415-2

[4] D. Bernardes, M. Diaby, R. Fournier, F. Fogelman-Soulié, and E. Viennet, "A social formalism & survey for recommender systems," SIGKDD Explorations, vol. 16, no. 2, Dec. 2014.

[5] M. McPherson, L. Smith-Lovin, and J. M. Cook, "Birds of a feather: Homophily in social networks," Annual review of sociology, 2001, pp. 415–444.

[6] F. Aiolli, "Efficient top-n recommendation for very large scale binary rated datasets," in Proceedings of the 7th ACM Conference on Recommender Systems, ser. RecSys '13. New York, NY, USA: ACM, 2013, pp. 273–280. [Online]. Available: http://doi.acm.org/10.1145/2507157.2507189

[7] B. Ngonmang, M. Tchuente, and E. Viennet, "Local community identification in social networks," Parallel Processing Letters, vol. 22, no. 01, 2012.

[8] V. D. Blondel, J.-L. Guillaume, R. Lambiotte, and E. Lefebvre, "Fast unfolding of communities in large networks," Journal of Statistical Mechanics: Theory and Experiment, vol. 2008, no. 10, 2008, p. P10008.

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

[10] M. Richardson and P. Domingos, "Mining knowledge-sharing sites for viral marketing," in Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2002, pp. 61–70.

[11] D. Jannach, M. Zanker, A. Felfernig, and G. Friedrich, Recommender systems: an introduction. Cambridge University Press, 2010.

[12] P. Victor, M. De Cock, and C. Cornelis, "Trust and recommendations," in Recommender systems handbook. Springer, 2011, pp. 645–675.

[13] R. Agrawal, T. Imieliński, and A. Swami, "Mining association rules between sets of items in large databases," in ACM SIGMOD Record, vol. 22, no. 2. ACM, 1993, pp. 207–216.

[14] M. J. Pazzani and D. Billsus, "Content-based recommendation systems," in The adaptive web. Springer, 2007, pp. 325–341.

[15] C. Borgelt, "Frequent item set mining," Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, vol. 2, no. 6, 2012, pp. 437–456.

[16] http://www.greatblackmusic.fr.

[17] http://www.greatblackmusic.fr/fr/mon-expo.

[18] J. Kim and H. Park, "Fast nonnegative matrix factorization: An active-set-like method and comparisons," SIAM Journal on Scientific Computing, vol. 33, no. 6, 2011, pp. 3261–3281.

[19] https://bitbucket.org/danielbernardes/socialfiltering.

 ISBN: 978-1-61208-415-2