# Hybrid Event Recommendation using Linked Data and User Diversity

Houda Khrouf
Eurecom
Biot, France
houda.khrouf@eurecom.fr

Raphaël Troncy
Eurecom
Biot, France
raphael.troncy@eurecom.fr

## ABSTRACT

An ever increasing number of social services offer thousands of diverse events per day. Users tend to be overwhelmed by the massive amount of information available, especially with limited browsing options perceived in many event web services. To alleviate this information overload, a recommender system becomes a vital component for assisting users selecting relevant events. However, such system faces a number of challenges owed to the the inherent complex nature of an event. In this paper, we propose a novel hybrid approach built on top of Semantic Web. On the one hand, we use a content-based system enriched with Linked Data to overcome the data sparsity, a problem induced by the transiency of events. On the other hand, we incorporate a collaborative filtering to involve the social aspect, an influential feature in decision making. This hybrid system is enhanced by the integration of a user diversity model designed to detect user propensity towards specific topics. We show how the hybridization of CB+CF systems and the integration of interest diversity features are important to improve predictions. Experimental results demonstrate the effectiveness of our approach using precision and recall measures.

## Categories and Subject Descriptors

H.3.3 [**Information Search and Retrieval**]: Information filtering

## General Terms

Algorithms, Experimentation

## Keywords

Event recommendation, Linked Data, LODE ontology, user diversity

## 1. INTRODUCTION

Recommendation in online social services have gained momentum over the past years as a key factor to deliver person-

alized content and enhance the user experience. Reducing the information overload and assisting customers to make decision become part of primary concerns in the e-service area. In this sense, recommender systems attempt to provide efficient filters that decode the users' interests, and optimize accordingly the information perceived. To help these systems predict items of interest, various clues are available, ranging from a user profile, explicit ratings, to past activities and social interactions. More specifically, integrating such systems in an event-based web service is a key advantage to attract people for attending diverse events, and to promote face-to-face social interactions. Indeed, the event recommendation can draw on different information sources such as the user preferences (ratings, likes, etc.), the attended events (visited places, involved artists), or even the social co-participation. Broadly speaking, the decision making upon attending events depends on some restrictions such as time, location, category, popularity and which friend will attend. However, the existing collaborative and content-based techniques cannot cope at all with this complex inherent nature of such decision. In addition, attended events are often topically diverse inducing a significant variety of topics in the user profile. This diversity will obviously affect the similarity between the new events and the prior ones, which can decrease the accuracy of content-based recommendation.

To tackle these issues, we propose a hybrid event recommender system by means of Semantic Web technologies. Our belief is that a structured event model presents one solution to cope with the complexity of event-specific characteristics. This modeling will ensure a more straightforward way to explore and reason over the data. It makes possible to ask complex queries, for example, to retrieve events involving the same artist within a specific geographical area. In addition, the semantic model empowers the enrichment of event descriptions with additional information from Linked Data [1], a set of connected and distributed RDF datasets. Such enrichment could potentially provide valuable inputs for the content-based recommender system [5].

In the second step of our approach, we propose to quantify the user interests based on topic modeling technique. This model detects the user propensity towards specific topics. It will be integrated in the recommender system to control the impact caused by the diversity of a user profile. Finally, we exploit the collaborative participation assuming that the social information about "which friend will attend an event" plays an important role in decision making. In this paper, we mainly investigate the influence degree that could have the data enrichment, the social information and the user

diversity to improve the system performance.

The remainder of this paper is organized as follows: Section 2 describes our event model and details how to build a recommender system using semantic metadata from Linked Data. Section 3 outlines our hybrid method for event recommendation. In Section 4 and 5, we describe some based techniques to train the system and we evaluate the proposed approach. We review related work in Section 6, and we conclude in Section 7.

## 2. LINKED DATA FOR CONTENT-BASED RECOMMENDATION

The principle of the content-based recommender system is to suggest new items similar to those a user liked in the past. The similarity between items is computed using the internal structure of the item profile by means of a variety of distance measures such as Cosine similarity, Pearson correlation or Latent Semantic Analysis. The most common representation of the item profile is the keyword-based model, in which attributes are represented by weighted vectors of keywords usually computed using TF-IDF scheme (term frequency/inverse frequency). To build such a profile from unstructured data, feature extraction techniques are needed to shift the item description from the original representation to a structured form suitable for next processing (e.g. keyword vectors). This task becomes straightforward by the use of Semantic Web technologies thanks to data structuring into ontologies known as machine understandable format. In fact, one principle of Semantic Web is to use common vocabularies for the integration of fragmentary information into a logically coherent knowledge. This huge and distributed knowledge, also called Linked Data, contains growing set of public and linked RDF datasets, covering diverse domains such as government, media, geography or more generally encyclopedic data. A content-based recommender system can greatly benefit from the ease of ontology-enabled feature extraction, and the availability of connected data in different domains to enrich the item profile. In order to fully leverage the assets of Linked data, we published in a previous work a RDF dataset of events and linked it with relevant open datasets [10]. In the following, we provide an overview of our RDF dataset of events and we explain how to compute the similarities between items based on Linked Data.

### 2.1 RDF Dataset of Events

In their daily life, people naturally organize their personal data according to occurring events: wedding, birthday party, concert, etc. In this paper, the events considered are a natural way for referring to any observable occurrence grouping persons, places, times and activities. To represent the events as structured data, we use LODE ontology[1], a minimal model that encapsulates the most useful properties. The goal of this ontology is to enable an interoperable modeling of the "factual" aspects of events, where these can be characterized in terms of the four Ws: What happened, Where did it happen, When did it happen, and Who was involved. Figure 1 depicts the metadata attached to the event identified by 1380633 on Last.FM[2]. More precisely, it indicates that this concert has been given on the 21th of May 2012 at 12:45 PM in the The Paramount Theatre featuring the
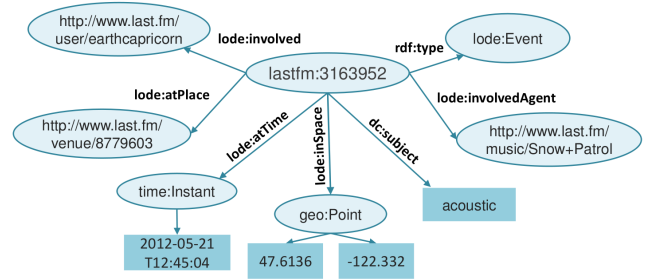


**Figure 1: The *Snow Patrol Concert* described with LODE ontology**

Snow Patrol rock band, and attended by the user earthcapricor.

The dataset was collected from three large event web directories (Eventful, Last.FM and Upcoming) and published in the Linked Data Cloud[3]. Using our SPARQL (RDF query language) endpoint[4], we can easily retrieve meaningful and semantic relationships about users. For instance, the following query extracts the list of attendees co-participating with a specific user along with the number of common events. A set of filters can simply be incorporated to target a specific geographical area or a period of time.

```
PREFIX UserURI:<http://data.linkedevents.org/user/6e150c44-
d5f3-4dfc-97f1-1c593d9cc4a2>
SELECT ?coattendee COUNT(DISTINCT ?event) as ?NB
WHERE {
    ?event   lode:involved UserURI: .
    ?event   lode:involved ?coattendee .
    ?event   lode:atPlace ?place .
    ?place   vcard:adr ?address .
    ?address vcard:locality "London" .
    FILTER(?coattendee!=UserURI:)
} ORDER BY DESC (?NB)
```

### 2.2 Linked Data as Tensor Space Model

In order to compute the similarity between items from Linked Data, we decided to apply the approach proposed by Di Noia et al [5]. The inceptive idea is that semantically similar items from RDF graph are the subject of two RDF triples having the same property and the same object (where a triple=<subject,property,object>). This approach is based on an adaptation of the classic Vector Space Model (VSM) [16], a well known technique in information retrieval. In this model, similarity between documents and queries is computed between their representative t-dimensional weighted vectors assigned to index discriminating terms. The application of VSM to RDF graph projects the Linked Data to 3-dimensional tensor, where each of its slices represents an adjacency matrix for one ontology property. In fact, the Semantic Web graph can be defined as a graph $G = (V, E)$, where $V$ is a set of RDF resources and $E$ is the set of properties between resources in $V$. For each existing property $p$ in $E$, the adjacency matrix models the linkage between subjects (on the rows) and objects (on the columns) from $V$ by this property $p$. Hence, a non null weight is assigned to each entry
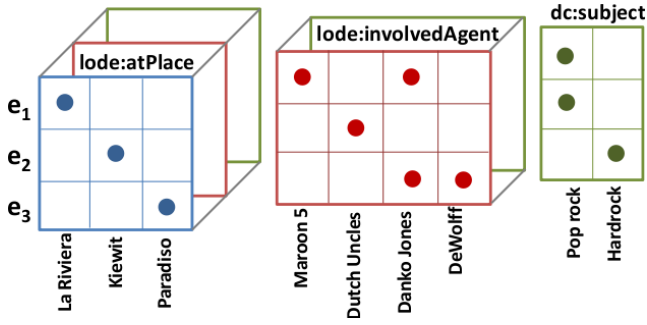
---

[1] http://linkedevents.org/ontology/
[2] http://www.last.fm

[3] http://linkeddata.org/
[4] http://eventmedia.eurecom.fr/sparql

**Figure 2: Tensor slices of some event properties**

$X_{ijp}$ in the tensor for each existing triple `<i-th subject, p-th property, j-th object>`. Figure 2 shows a small example of tensor slices related to some properties, namely: `lode:atPlace`, `lode:involvedAgent` and `dc:subject`.

Considering the data sparsity and the independent properties, we would be able to apply the approach above and compute the similarity between events with respect to each property separately. The representation of an event $e_i$ according to the property $p$ is a t-dimensional vector indexing the terms/resources related to $e_i$ via $p$. The TF-IDF weight of each resource $r$ is computed as:

$$w_{r,i,p} = f_{r,i,p} \; * \; log \left( \frac{N}{m_{r,p}} \right) \qquad (1)$$

where $f_{r,i,p} = 1$ if a link exists between the node $e_i$ to the resource $r$ via the property $p$, otherwise $f_{r,i,p} = 0$. $N$ is the total number of events in the dataset. $m_{r,p}$ is the number of events linked to the resource $r$ via the property $p$. Then, the similarity between two events $e_i$ and $e_j$ with respect to the property $p$ is calculated using Cosine distance between their representative vectors as following:

$$sim^p(e_i, e_j) = \frac{\sum_{r=1}^{t} w_{r,i,p} * w_{r,j,p}}{\sqrt{\sum_{r=1}^{t} w_{r,i,p}^2} * \sqrt{\sum_{r=1}^{t} w_{r,j,p}^2}} \qquad (2)$$

The approach described above can be simply applied to detect similarity between subjects or objects of RDF triples. It has been successfully used to recommend movies and to improve the quality of content-based system [5]. Nevertheless, it is still limited when the adjacency matrix is very sparse such as the case of matrices associated with `lode:atPlace` and `lode:involvedAgent` properties. In fact, such predicates are considered as discriminant properties in RDF, which are characterized by the diversity of its object values. For instance, the t-dimensional vector related to `lode:atPlace` property has only one non null weight since an event always takes place in one venue.

## 2.3 Similarity-based Interpolation

In order to mitigate the sparsity of the adjacency matrix, we decided to interpolate fictitious values based on the similarity between objects. First, we introduce a discriminability metric [17] to gain insights into the properties concerned by the similarity-based interpolation. The metric is defined via the following formula:

$$Discriminability(p) = \frac{| \{o \mid t = <s, p, o> \in G\} |}{| \; t = <s, p, o> \in G |} \qquad (3)$$

where G is the RDF graph, $t$ is the triple representing the link between the subject $s$ and the object $o$ via the property $p$. From this formula, it is trivial to see that having high discriminability means that many instances have different object values on this property. For instance, from a set of 1700 events (related to 10,323 agents, 627 places and 5,758 subjects), we found a discriminability score of 0.64 for the `lode:involvedAgent`, and 0.45 for the `lode:atPlace`, while it is only 0.10 for the `dc:subject` predicate. This is also reflected by the sparsity measures of the related adjacency matrices. Moreover, we intuitively know that similar events are not necessary occurred in the same venue or featuring the same performers. As a solution for such typical items, we propose to interpolate the similarity scores between objects in the adjacency matrix. More precisely, if the resource $r_k$ is similar to the resource $r_h$, and both $f_{r_h,i,p} = 1$ and $f_{r_k,i,p} = 0$, then $f_{r_k,i,p} = sim(r_k, r_h)$. Note that $f_{r_k,i,p}$ reflects the strength of the fictitious link which associates event $e_i$ with the resource $r_k$ via the property $p$. If $r_k$ is similar to more than one resource originally linked with $e_i$ via the property $p$, the weight $f_{r_k,i,p}$ will be equal to the highest similarity score. For each resource $r_k$, the equation 1 becomes more generally:

$$w_{r_k,i,p} = \max_{r_h \in H} sim(r_k, r_h) \; * \; log \left( \frac{N}{m_{r_k,p}} \right) \qquad (4)$$

where $H$ is the set of objects originally linked with the event $e_i$. In this context, we do not pay attention to how similarity between objects is computed. In fact, this measure depends on the structure of the object itself and there exist several existing distances that can be used. In our case, we have been based on the similarity scores between agents provided by third party services such as Last.FM, and we computed the normalized geographical distance between venues. A key advantage of the similarity-based interpolation is that the discriminability and the sparsity rates are 3 times reduced. Finally, the similarity between two events $e_i$ and $e_j$ is a normalized combination of their similarities according to each property $p$.

## 3. EVENT RECOMMENDATION

Different from a classic item, events occur at a specific place and during a period of time to become worthless for recommendation. Moreover, while the classic items continuously receive useful feedbacks, the user ratings about events are very sparse due to their transiency. In our dataset, these ratings are represented by the binary user-event attendance matrix with 98% sparsity rate (a set of users only attend a very limited number of all the available events). As a solution, one can address event recommendation using content-based algorithm relating event metadata to user preferences. This perfectly complies with the constraints a person considers to decide whether or not to attend an event. Metadata such as distance, time, topics and performers are important and influential factors in such decision. Nevertheless, the content-based recommendation might suggest items with a limited diversity, and it does not consider the social information that answers to the question: "which friend is going?".

To fill this gap, we first analyze the pure content-based recommendation where a weight value is assigned to each of metadata fields. We attempt to improve its performance by enriching descriptions from Linked Data. Then, we incorporate the social dimension (collaborative-filtering) and the topical user diversity by means of weighted hybrid recommendation.

## 3.1 Content-based Recommendation

The underlying principle of content-based system is to recommend future events similar to ones a user has attended in the past. We assume that there is a sufficiently number of past attended events to avoid the cold-start problem, which is out of the scope of the present work. Similarity values between events are then used to obtain a ranked list of recommended items, as following:

$$rank_{cb}(u, e_i) = \frac{\sum_{e_j \in E_u} \sum_{p \in P} \alpha_p \ sim^p(e_i, e_j)}{\mid P \mid \ * \mid E_u \mid} \quad (5)$$

where $e_i$ is the event to recommend, $P$ is the set of properties shared between two events, $E_u$ is the set of past events attended by the user $u$, and $\alpha_p$ is the weight assigned to each property reflecting its contribution to respect the user profile. In the next section we describe some methods how to learn these weights $\alpha$. The selected properties to compute the similarity between events are those which are related to the location, subjects (tags) and involved agents (artists). Note that we do not deal with the temporal property at the moment. We mainly focus our analysis on the topic enrichment, the social factor and the user diversity. We believe that temporal property could be harnessed to index only recent and future events in a specific period of time. This will reduce the computational processing time and increase the scalability. However, as this method reduces the knowledge about the user past ratings, there is a need to investigate its impact on the recommendation accuracy, an analysis left for further studies.

**Geographic Closeness.** Roughly speaking, users tend to attend nearby social events, thus the location is always considered as a valuable feature in event recommendation [15]. In our approach, to measure the similarity between events according to the `lode:atPlace` property, we need to normalize it according to a specific threshold $\theta$. Since the location of the user home is missing in our data, we analyzed the distance between attended events for each user and we reported results for all users in Figure 3. Note that the attendance rate becomes extremely low from $\theta = 80$ Km. This value is considered as the threshold from which the similarity between events equals to zero according to the location property.

**Enrichment with Linked Data.** One method to enrich the item profile from Linked Data lies in the exploitation of DBpedia, the RDF version of Wikipedia. The key advantage of this dataset is the availability of semantically rich information in various domains. In a previous study [9], we have automatically interlinked our dataset with DBpedia. As a result, a significant number of high quality mappings has been obtained between the artists of both datasets. Consequently, we decided to enrich the topics of an event using the DBpedia topics of the involved artists. More precisely,
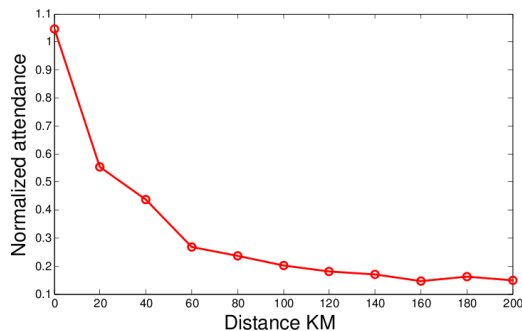


**Figure 3: Normalized average attendance rates per distance KM**

we retrieve the categories associated with the property `dc-terms:subject` of artists by simply querying the DBpedia SPARQL endpoint[5]. The reason behind our interest in DBpedia categories is their classification into a well defined hierarchical structure with highly consistent labeling.

## 3.2 Topical User Diversity

In the recommender system, one fundamental goal is to suggest items in compliance with the user interests. This aim is particularly difficult to achieve in event recommendation due to the presence of topically diverse events. In fact, the real-world social events can be classified into large set of categories, ranging from large festivals and conferences to small concerts and social gatherings. When attending an event, the user might be interested in a specific show or artist, or might have broad interests. In consequence, relying on events similarity in content-based recommendation can be influenced by the topical diversity of past attended events. To alleviate this impact, we decided to put an emphasis on the role of interesting events to predict the unknown ratings. This means that we assign different weights to the events included in the peaks of interest, and to those which are out of these peaks (equation 8). These weights $\beta$ are then estimated using training methods. The formula 5 becomes:

$$rank_{cb++}(u, e_i) = \frac{\sum_{e_j \in E_u} \sum_{p \in P} \alpha_p \ \beta_p \ sim^p(e_i, e_j)}{\mid P \mid \ * \mid E_u \mid} \quad (6)$$

where $\beta_p = 1$ if the property $p$ is different from `dc:subject`, otherwise the $\beta_{subject}$ is an estimated value depending on whether the event $e_j$ corresponds to a peak of interest or not. In order to detect these peaks of interest, we propose to quantify the interest diversity using Latent Dirichlet Allocation (LDA) [2], a topic modeling technique based on the co-occurrence of terms. Inspired by the method in [19], we consider that the interest diversity of each user is reflected by the previously attended events, where each event is considered as a document represented by a set of terms. For each event $e_i$, LDA generates a T-dimensional vector of topic proportions $\Theta_i = [\theta_i^1, \theta_i^2, ...\theta_i^T]$, where T is the number of topics. Then, we calculate the variance of each topic dimension $t$ of all the events $E$ attended by a user $\Theta^t = [\theta_1^t, \theta_2^t, ...\theta_E^t]$. The diversity score of a user is the mean of the variances of all topics dimensions (mean of $\Theta^1, \Theta^2...\Theta^T$). The normalized
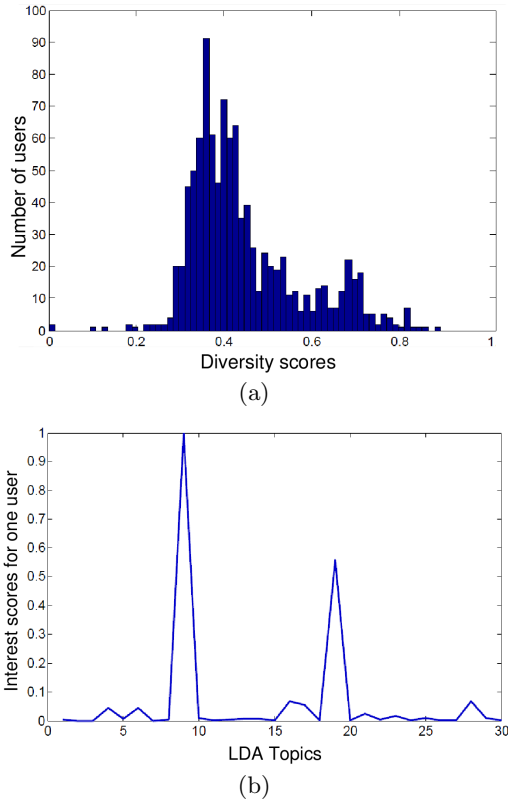
---

[5]`http://dbpedia.org/sparql`

**Figure 4: Distribution of topical diversity scores with T = 30: (a) for all the users; (b) for one specific user.**

scores obtained from a sample of 1,000 Last.fm users are depicted in Figure 4. From the histogram, it is shown that most of diversity scores range from 0.3 to 0.5 indicating users with relatively high interest in specific topics. The diversity scores near to 1 represent a very strong interest in some topics such as the user plotted in Figure 4(b) having a strong bias towards topic 9. Finally, the diversity score of zero represents the users associated with few attended events (the cold-start problem).

### 3.3 Collaborative filtering

A form of social interaction is the collaborative participation such as co-authoring a paper or co-attending an event. In [12], Liu et al. highlight the existence of an offline social network built by means of co-attending the social events. Such network seems cohesive but still less denser than an online social network. Accordingly, we consider that two users involved in the same event can potentially have a stronger tie than others users. The involvement could be offline by directly attending the event, or online by sharing media and comments about it. This let us assume that the more events in which the users involve, the more strongly their tie is. Thus, the sufficiently similar consumptions could probably provide information at first glance about which "friends" will attend an event. Since our dataset contains users intentions upon attending upcoming events, we exploit them to predict unknown ratings. However, unlike the traditional user-based collaborative filtering (CF), we would like in our approach to not only consider the similarity between users, but also

the contribution of a group of friends. We define the following formula as the prediction that a user $u_i$ will attend an event $e$ based on the intentions of his co-attendees (users who have attended past events with the user $u_i$):

$$rank_{cf}(u_i, e) = \frac{\sum_{j \in C} a_{i,j}}{\mid C \mid} \ * \ \frac{\mid E_i \cap (\cup_{j \in C} E_j) \mid}{\mid E_i \mid} \quad (7)$$

where $C$ is the set of co-attendees of the user $u_i$ that will attend the event $e$, $E_i$ is the set of attended events by the user $u_i$, and $a_{i,j}$ represents the fraction of common events between the users $u_i$ and $u_j$ by the cardinality of $E_j$. Note that the weight $a_{i,j}$ reflects whether the most of events which are attended by the user $u_j$ are also attended by the user $u_i$. The rationale behind this formula is two-fold: (1) in the first part, the contribution of each co-attendee is considered individually; (2) in the second part, we consider the co-attendees as a group of friends, and we assume that the more events they attended together with the user $u_i$, the more strongly is their relationship.

### 3.4 Hybrid Recommendation

To combine the features of both content-based and collaborative filtering algorithms, we propose a weighted hybridization using a linear combination of recommendation scores. Taking into account the user diversity and combining the equations (6) and (7), we propose the following rank function:

$$rank(u, e) = rank_{cb++}(u, e) + \ \alpha_{cf} \ rank_{cf}(u, e) \quad (8)$$

where $\alpha_{cf}$ is the weight of collaborative filtering feature, estimated in conjunction with the weights of content-based features using different approaches to train the system.

## 4. LEARNING RANK WEIGHTS

To learn the weights of our prediction function, we first test the linear regression with gradient descent that minimizes the least-squares cost function. Then, we use two evolutionary computation methods motivated by their success in a wide range of tasks. Results are reported in Section 5.

### 4.1 Genetic Algorithm

Genetic Algorithms (GAs) [20] are stochastic methods inspired from the mechanism of natural evolution and genetic inheritance. They have been applied to various domains including information retrieval, image processing and machine learning. In GAs, a population is a set of chromosomes (candidate solutions) and each chromosome denotes a set of genes. A fitness criterion is a function which provides an estimation of the quality of each solution. An initial population of candidate solutions (chromosomes) are evaluated using the fitness function. The solutions having higher fitness values than others are stochastically selected, recombined and randomly mutated to produce a new population for the next generation. The algorithm stops iterating when the optimal solution is produced or a maximal number of iterations is reached. To apply GA in our approach, a chromosome is represented by a vector of the coefficients $\alpha$ (the genes). Each chromosome is then evaluated using a fitness function that maximizes the precision@N on the training data. We set the following GA parameters: population size=30, iterations=80, crossover rate=0.9 and mutation rate=0.01.

## 4.2 Particle Swarm Optimization

Particle Swarm Optimization (PSO) [7] is a population based stochastic optimization technique inspired by the social behavior of bird flocking or fish schooling. The algorithm initializes a population of random solutions called swarms (particles), and searches for optima of a fitness function by updating generations. Compared with GA, PSO is easy to implement with few parameters to adjust, and has no evolution operators such as crossover and mutation. In each generation, each particle accelerates in the direction of its own personal best solution found so far, as well as in the direction of the global best position discovered so far by any of the particles in the swarm. Each particle $i$ in the swarm has the following attributes: a current position $x_i$, a current velocity $v_i$, and a personal best position $p_i$ in the search space, and the global best position $p_{gbest}$ among all the $p_i$. In each iteration, the velocity and the position of each particle is updated as following:

$$v_i(t+1) = w.v_i(t) + c_1 r_1 (p_i - x_i(t)) + c_2 r_2 (p_{gbest} - x_i(t))$$
$$x_i(t+1) = x_i(t) + v_i(t+1)$$

where $c_1$ and $c_2$ are the acceleration coefficients, $r_1$ and $r_2$ are random numbers uniformly distributed within [0,1], and $w$ is the inertia weight. The velocity and acceleration are responsible for changing the position of the particle to explore the space of all possible solutions, instead of using existing solutions to reproduce. The personal and the global best positions are the optima of a fitness function respectively in each iteration and for all past iterations. Following the same setting in GAs, a particle is represented by a vector of the weights $\alpha$, and the fitness function aims at maximizing the precision results. We set the following PSO parameters: population size=30, iterations=80, $c_1$=1.494, $c_2$=1.494 and w=0.729 (more details in [7]).

## 5. EVALUATION

We carried out a set of experiments measuring the precision and recall metrics to assess the contribution of each step and the quality of our recommendation compared with existing approaches.

### 5.1 Real World Dataset

As described in Section 2, our RDF dataset is obtained from three large public event directories (Last.fm, Eventful and Upcoming). In particular, we have been interested in Last.fm directory containing a large number of active users, that explicitly announce their participation to future events. Using SPARQL, we collected 2,436 events, 481 active users whose the attendance rates are within [15,50] generating 12,729 distinct consumptions. This set of events are related to 14,748 different artists, 897 different locations and 4265 distinct tags (music domain). For evaluation purpose, we used a test set containing the most recent 30% of the consumptions and a training test with the remaining 70% consumptions. Then, we calculated two metrics used in top-N recommendation task: Precision is the ratio of correctly recommended items and the length of the recommendation N; Recall is the ratio of correctly recommended items and the total number of future consumptions. Precision and recall were computed at different N values.

| Task | location | agent | subject |
|------|----------|-------|---------|
| (1) | 0.9942 | 0.9174 | 0.3175 |
| (2) | 0.6854 | 0.7392 | 0.2843 |

Table 1: Sparsity rates of the similarity matrix for each property before (1) and after the similarity-based interpolation and linked data enrichment (2)
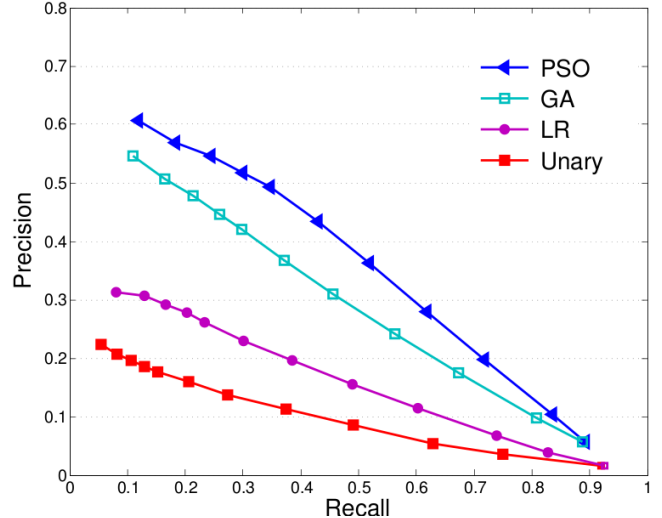


Figure 5: Recall and Precision based on different approaches to estimate the vector $\alpha$

### 5.2 Experiment

To highlight the importance of the similarity-based interpolation and linked data enrichment, we report the sparsity rates in Table 1. We see that our method succeed to discover latent similarities between items attributes especially for discriminant properties. This similarity was not considered in keyword-based recommender system, and its interpolation becomes straightforward thanks to the ontology-based modeling.

As a second step of our evaluation, we assess the performance of the training methods to learn the coefficients $\alpha$ in the hybrid recommendation algorithm. Note that, for this experiment, we do not include the user diversity and set the $\beta_{subject} = 1$ since our aim is rather to compare the behavior of our training methods. Figure 5 shows the Precision and Recall curves computed using different approaches. It is evident that setting all the coefficients equal to 1 achieves the worst performance because no adaptive optimization was considered. Moreover, it is shown that precision optimization methods (GA and PSO) deliver considerably better top-N results compared with error (RMSE) minimization method (LR). This also confirms recent works [4] showing that methodologies based on error metrics do not necessarily improve the accuracy of top-N recommendation task. One given explanation is that the RMSE-oriented methods account only for known ratings to train the system, and do not consider the unrated items. Finally, Figure 5 highlights the better performance of PSO compared with GA algorithm. We observed a faster convergence to the optimal solution in PSO compared with GA which may need more iterations. This owed to the inherent behavior of PSO where the evolu-
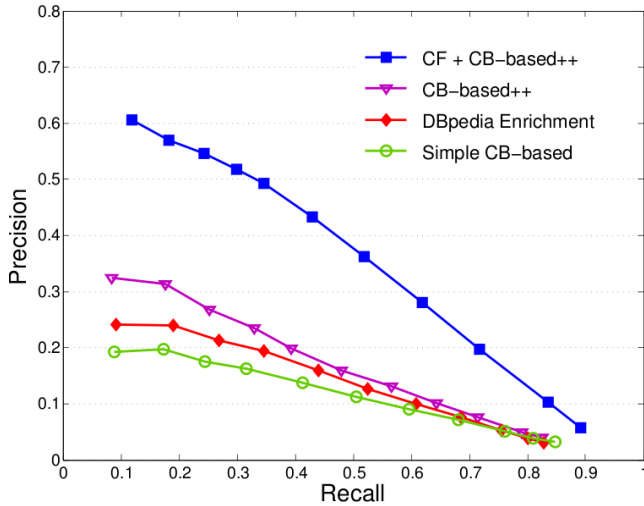
**Figure 6: Evolution of the recommendation accuracy by incorporating the DBpedia enrichment, user diversity (CB-based++) and collaborative filtering (CF)**



**Figure 7: Comparison of hybrid event recommendation with pure CF algorithms**

tion is only guided by the best particle, while it is guided by a group of solutions in GA (in which even weak candidates continue to survive after some iterations). In the following, we rely on PSO algorithm to compute the coefficients $\alpha$ and the weight $\beta_{subject}$.

To gain insights into the influence of the different steps in our recommendation, we decided to examine the evolution of the system performance by adding in each experiment (by order) the Linked Data enrichment, the user diversity and the collaborative filtering. Results are illustrated in Figure 6. We can observe that enriching data with DBpedia categories slightly improves both precision and recall. Indeed, introducing more coherent and qualitative data is one solution to reduce the noise that can be found in the collective knowledge of crowd tagging. Then, the integration of the user diversity model also enhances the system performance. For this experiment, we fixed the coefficients $\alpha$ obtained with PSO, and trained the system to compute the coefficient $\beta_{subject}$ which depends on the peaks of the user interests. We obtained $\beta_{subject}$=0.4 when the past event is not included in an interest peak, and $\beta_{subject}$=1.6 (4 times more) otherwise. This confirms the importance to account for the peaks of interest when recommending items, especially for the users who have strong propensity towards specific topics and when items are topically diverse. Finally, combining these results with the collaborative filtering notably increases the recommendation accuracy. We believe that this improvement is perfectly tangible with the use of a real world dataset. Indeed, according to the user-centric study in [18], social aspects such as people and friends who are attending an event have strong priority and influence on decision making.

In the last part of our evaluation, we assess to which extent a hybrid event recommendation outperforms the existing pure collaborative filtering. We compared our algorithm with the traditional user-based CF and the Probability based Extended Profile Filtering (UBExtended) proposed for event recommendation in [14]. This method em-
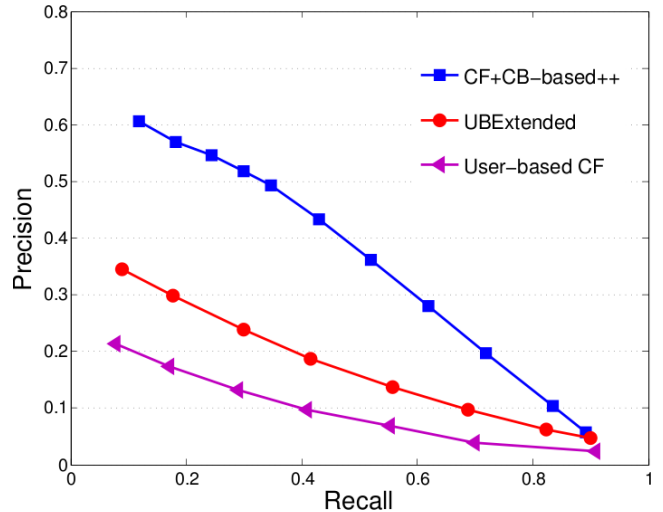
ploys a cascade of user-based CF systems attempting to recommend popular (most consumed) events among the users. The rationale behind is that the probability to consume an event is proportional to the current popularity of the event, an important aspect that will be considered in a future study. The comparison results are depicted in Figure 7. It is shown that the UBExtended improves the outcome of the system compared with the pure user-based CF algorithm. We can also observe that our hybrid algorithm exhibits the best accuracy in terms of precision and recall. This owed to the benefits of hybridization technique which have been extensively highlighted in several recommender systems.

## 6. RELATED WORK

In the research area of recommender systems, many approaches have been proposed to recommend movies, but few are the studies that deal with event recommendation. Considered as one-and-only items, events are particularly hard to recommend due to their short life time and the system often suffers from highly sparse data. Some works have been proposed to overcome these issues and improve the recommendation accuracy. Cornelis et al. [3] build a hybrid approach within a fuzzy relational framework which allows to reflect the uncertain information in the domain. The rationale behind is to recommend future events if they are similar to past ones that similar users have liked. However, this framework was not evaluated and there is no clear insight about its performance especially for large scale dataset. Minkov et al. [13] follow the same rationale and propose a low rank collaborative method to predict the ratings of future events. They highlight the performance of the collaborative filtering over the pure content based system. Nevertheless, their approach has mainly focused on the recommendation of scientific talks in the same building, and there is no consideration of the user location. Some other systems have been also developed such as "Pittcult" [11] and "Eventer" [8] which position the user within a social network and leverage the trustworthiness between users, a valuable information that is not available in many systems. Finally, a user-centric evaluation [6] showed that the straightforward com-

bination of CF and CB recommendations outperforms both individual algorithms on almost qualitative metrics such as accuracy, novelty, diversity, satisfaction and trust.

Another innovative direction also interesting in our research is the recent studies proposed to harness Linked Data in recommender systems. In [5], Di Noia et al. use the Linked Data as the only background knowledge to recommend movies. They highlight the performance of the system compared to the keyword-based methods. However, there is no deep exploitation of latent similarities that may exist between movies attributes. In event recommendation and to the best of our knowledge, our system is one of the first initiative that combines the benefits of the hybrid algorithms and Linked Data.

# 7. CONCLUSION

In this paper, we have presented an event recommendation approach which combines the CB and CF methods, and utilizes Linked Data to enrich the item profile. We have also employed a statistical model of user interests to overcome the topical diversity of rated items and improve the system performance. The evaluation particularly highlights the importance of social information and the user diversity model to improve the event recommendation. In the future, we plan to involve other significant features such as event popularity and temporal indexing of recent consumptions to enhance the scalability. Moreover, we want to utilize some advanced similarity techniques such as tensor factorization in Linked Data, which has shown excellent performance in sparse and high dimensional domains.

## Acknowledgments

# 8. REFERENCES

[1] C. Bizer, T. Heath, K. Idehen, and T. Berners-Lee. Linked data on the web. In *LDOW, 1st Workshop on Linked Data on the Web*, Beijing, China, 2008.

[2] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022, 2003.

[3] C. Cornelis, X. Guo, J. Lu, and G. Zhang. A fuzzy relational approach to event recommendation. In *2nd Indian International Conference on Artificial Intelligence*, Pune, India, 2005.

[4] P. Cremonesi, Y. Koren, and R. Turrin. Performance of recommender algorithms on top-n recommendation tasks. In *ACM Conference on Recommender Systems (RecSys'10)*, Barcelona, Spain, 2010.

[5] T. Di Noia, R. Mirizzi, V. C. Ostuni, D. Romito, and M. Zanker. Linked open data to support content-based recommender systems. In *8th International Conference on Semantic Systems, I-SEMANTICS*, Graz, Austria, 2012.

[6] S. Dooms, T. D. Pessemier, and L. Martens. A user-centric evaluation of recommender algorithms for an event recommendation system. In *Workshop on Human Decision Making in RecSys'11*, Chicago, IL, USA, 2011.

[7] R. C. Eberhart and Y. Shi. Particle swarm optimization: developments, applications and resources. In *IEEE Congress on Evolutionary Computation*, volume 1, pages 81–86, 2001.

[8] M. Kayaalp, T. Özyer, and S. T. Özyer. A collaborative and content based event recommendation system integrated with data collection scrapers and services at a social networking site. In *International Conference on Advances in Social Networks Analysis and Mining*, Athens, Greece, 2009.

[9] H. Khrouf, V. Milicic, and R. Troncy. Eventmedia live: Exploring events connections in real-time to enhance content. In *Semantic Web Challenge at 11th International Semantic Web Conference*, Boston, USA, 2012.

[10] H. Khrouf and R. Troncy. Eventmedia: a LOD dataset of events illustrated with media. *Semantic Web Journal, Special Issue on Linked Dataset descriptions*, 2012.

[11] D. H. Lee. Pittcult: trust-based cultural event recommender. In *ACM Conference on Recommender Systems (RecSys'08)*, Lausanne, Switzerland, 2008.

[12] X. Liu, Q. He, Y. Tian, W.-C. Lee, J. McPherson, and J. Han. Event-based social networks: Linking the online and offline social worlds. In *18th ACM SIGKDD conference on Knowledge Discovery and Data Mining*, KDD'12, Beijing, China, 2012.

[13] E. Minkov, B. Charrow, J. Ledlie, S. J. Teller, and T. Jaakkola. Collaborative future event recommendation. In *19th ACM Conference on Information and Knowledge Management*, Toronto, Ontario, Canada, 2010.

[14] T. D. Pessemier, S. Coppens, K. Geebelen, C. Vleugels, S. Bannier, E. Mannens, K. Vanhecke, and L. Martens. Collaborative recommendations with content-based filters for cultural activities via a scalable event distribution platform. *Multimedia Tools Appl.*, 58(1):167–213, 2012.

[15] D. Quercia, N. Lathia, F. Calabrese, G. Di Lorenzo, and J. Crowcroft. Recommending social events from mobile phone location data. In *10th IEEE International Conference on Data Mining*, Sydney, Australia, 2010.

[16] G. Salton, A. Wong, and C. S. Yang. A vector space model for automatic indexing. *Communications of The ACM*, 18:613–620, November 1975.

[17] D. Song and J. Heflin. Automatically generating data linkages using a domain-independent candidate selection approach. In *10th International Semantic Web Conference (ISWC'11)*, Bonn, Germany, 2011.

[18] R. Troncy, A. T. S. Fialho, L. Hardman, and C. Saathoff. Experiencing events through user-generated media. In *1st International Workshop on Consuming Linked Data*, Shanghai, China, 2010.

[19] H. Wu, V. Sorathia, and V. Prasanna. When diversity meets speciality: Friend recommendation in online social networks. *ASE Human Journal*, 1(1):52–60, 2012.

[20] J. yuan Yeh, J. yi Lin, H. ren Ke, and W. pang Yang. Learning to rank for information retrieval using genetic programming. In *SIGIR Workshop on Learning to rank for Information Retrieval*, 2007.