International Journal of Computer Information Systems and Industrial Management Applications.
ISSN 2150-7988 Volume 5 (2013) pp. 499-508

Toward ontology-based personalization of a Recommender System in social network

Hubert Kadima, Maria Malek
LARIS/EISTI Laboratory
Ave du Parc 95011, Cergy Pontoise - FRANCE

Abstract: **Personalized search, navigation and content delivery techniques have attracted interest in the recommender systems as a means to decrease search ambiguity and return results most relevant to a particular user preferences. In this paper, we study the effect of incorporating user semantic profile derived from past user's behavior and preferences on the accuracy of a recommender system. We present a preliminary work which aims at tackling most technical issues due to the integration of an ontology-based semantic user profile within a hybrid recommender system based on our early released guided recommender algorithm. A semantic user profile context is represented as an instance of a reference domain ontology in which concepts are annotated by interest scores.**

Keywords: Ontological User Profiles, User Context, Personalized Information Retrieval, Ontology, Social Semantic Web.

I. Introduction

Personalization encompasses a set of techniques that make application sensitive to user's profiles and contexts. In recent years, personalized search has attracted interest in the research community as a means to decrease search ambiguity and return results most relevant to a particular user preferences and thus providing more effective and efficient information access [31, 3, 6]. One of the key factors for accurate personalized information access is user profile context. A user profile, used in recommender systems, is a structured construct containing information both directly and indirectly pertaining to a user's preferences, behavior and context. Most of social networks already propose to their users recommendations based on their profiles. In a similar fashion, they help users to find people for sharing common social activities and preferences. In this paper, we study the effect of incorporating semantic user profile derived from past user behavior and preferences on the accuracy of a hybrid recommender system. We present a preliminary work which aims at tackling the most technical issues due to the integration of an ontology-based semantic user profile within a hybrid recommender system based on our early released guided recommender algorithm [22] in social network. Content-based collaborative filtering mechanisms applied in this algorithm require services to build and maintain accurate models of a user's preferences and to organize the information in such a way that matches the particular user context. Typically, the recommender systems have to maintain a model of each user profile with respect to his knowledge of domain concepts and user preferences evolution. A variety of classical techniques ranging from simple statistics to machine learning algorithms (k-nearest neighbour, naive Bayes, SVM) may be applied to deliver personalization solutions. Recently, the emergence of the social semantic web provides us opportunity to revisit personalization approaches for the social networks environment. In general, the different ontologies in social networks should help to distinguish models based on the representation of business data and those established for the representation of document structures and social interactions on the web. Our unified user profile context is represented as an instance of a reference domain ontology in which concepts are annotated by interest scores. We use semantic user profile and the RDF-based user model exchange language UserML [19] to maintain consistency between individual user profile contexts.

The organization of the rest of this paper is as follows. Section 2 presents a general overview about expert identification in social networks Section 3 presents our recommender algorithms [22] pointing out data organization of user profile, matching mechanisms applied and experimental-based accuracy of these algorithms. Section 3 presents main technical issues due to the integration of this algorithm with a semantic user profile. Section 4 presents some semantic interoperability issues. Section 5 presents conclusion and main orientations of our future work.

II. Expert identification in social networks

A social network is a set of people or groups of people with some pattern of contacts or interactions between them. Social networks analysis is defined as the study of social entities such as people in organizations called actors, and their interactions and relationships. A social network is modeled by a graph or network, where each vertex is a node (actor) and each edge is a relationship. We can study the structural properties as well as the role and the social prestige of each actor [25, 12, 10]. We can also find different types of sub graphs such as communities formed by groups of actors with common interests, by isolating the group individuals with a high density [5]. The social network can be also a source for the development of recommendations: find an expert in a given field, suggest products to sell, offer a friend, etc. This development may be based on paths exploration algorithm, degree analysis.

Given a particular task and a set of experts, the problem of the

expert location is to identify the most suitable set of experts. We consider the case when experts are organized in networks that correspond to social networks or organized structures of companies. In a company, the network may capture hierarchical organizations. In a research community, the network captures previous collaboration among scientists. We are dealing with the expert idenfication with score propagation which is defined as [2] : *Given a query Q that consists of a list of skills, and a social graph G, identify a subset of candidates who have the skill specified by the query. Use the graph to propagate scores to re-rank experts.*
A candidate's expertise can be inferred by the skills of other people he is connected with. Many popular webpage ranking algorithms such as PageRank [7] and Hits [20] can be used for the expert location problem. In [8], Authors proposed to use the email communication network to refine their expertise identification. In this network, people who have received many mail inquiries are defined as the *authorities* or the experts and people who are able to forward questions to many experts are defined as the *hubs*. In [35] and [21], authors employed both pageRank and Hits on community-based question answering networks; a user A who replied to another user B's question often indicates that A has more knowledge on the subject than B. If B answered questions from C then A's expertise score should be boosted, etc.

Many implemented systems start to use the connections among individuals to identify experts. We cite the Arnet-Miner system developed by [34] for academic search. Given a query the system returns a list of exerts on this topic. The system suggests also the top conferences and papers on this topic. We mention also other examples of social systems as the "Spree" [24] and expertise recommender [23].

On the other hand, many graph algorithms have been used for experts' recommendation in social networks. These strategies are essentially [36]:

- Breadth First Search which broadcasts the query to every person in a social network.
- Random Walk Search (RWS) that randomly chooses one of the current's neighbor to whom to spread the query.
- Best Connected Search (BCS) proposed by [1] which makes use of the skewed degree distribution of many networks.
- Weak and Strong Ties algorithms are based on the idea that the connections between two individuals can have different strengths. The strength of association varies and is not always symmetric.
- Hamming Distance Search (HDS) picks the neighbor which has the most uncommon friends with the current user.
- The Information Scent Search (IIS) picks the next person who has the highest match score between the query and the profile.

Searching expertise in social network has been approached in Zhang and Ackerman work since 2005 ([36]). Graph search strategies were applied and evaluated on the Enron email data ([36]). The evaluation criteria are: the number of people used per query, the depth of the query chain. The IIS is not obviously better than out degree based strategies (BCS and HDS). Weak Ties have been found to be important in helping people get new information. There will be found that weak ties are critical for automated expertise finding. The out degree strategy such BCS and HDC in networks like the Enron's have clear advantages over other strategies ([36]). In [8], the problem of expertise identification using Email communications is treated. A content-based algorithm is compared with a graph based algorithm using the HITS algorithm and taking into consideration both text and communication. Results show that the graph based algorithm performs better. The same idea is developed in [13] showing that social networks analysis techniques as the expertise propagation algorithm leads to significant performance improvement. In [35], the recommendation is formalized as a ranking problem over a heterogeneous social network. Random Walk Search is used to elaborate a recommendation when a person is doing a search or when browsing the information.
In [16], the structure of social network of mathematical papers and the relations between authors in mathematical field are studied, the nodes of this network are the mathematicians and the edges are the common papers between them, the evolution of this network over the time (number of authors, number of papers) is also presented.

We describe in the next section an original algorithm that uses of the most representative spanning tree of the network. The expert identification algorithm is based on a guided search algorithm which uses an heuristic in order to search efficiently the spanning tree. We apply this algorithm on bibliographic data.

III. The search-recommender algorithms

In our previous released paper[22], we compare two recommendation algorithms based on classical spanning tree algorithm. They are based on three types of knowledge: The first type deals with information concerning the person. This information is stored in the actor vertex level and can be annotated by an ontology describing user profiles. The second type of information is computed and derived from the network structure itself. Actually, this consists of exploring the links starting from the initial actor exploring the maximum spanning tree which the root is the initial actor. We can thus reduce the search space of target actors. While the third type of information is based on the betweenness centrality measure associated to each actor. This measure enables to estimate the control of an actor over other pairs of actors. We use this measure to extract the best paths from the previous spanning tree.

A. Data description : extraction of the professional network

Social network we deal with, is composed of authors extracted from bibliographic data. In this graph, nodes are authors, while evaluated edges are the similarity degree between these authors. Each author Z has a given profile Pro_Z. This profile is described by a weighted vector of keywords

T_i, these keywords present the topics the authors' interests. $Pro_Z = \{(T_1, P_1), (T_2, P_2) \ldots, (T_1, P_1)\}$. The goal of the system is to recommend, in response to a given author query, a group of ranked authors according to the similarity between their profiles (terms of interests T_m) and the query terms. For that we had to extract a social network that presents the authors and the relations between them.
The social network has been extracted from Microsoft Academic search website libra.msra.cn.
Firstly, we have extracted a connected network of authors from this site. The obtained network is described as a valued directed graph (see figure 1 and 2), nodes of this graph are authors while edges of this graph are the citations between these authors, each edge has a value representing the number of citations between two connected authors. This social network is represented by a matrix L. In this matrix: L_{ij} equals n if author i cites author j n times. In fact, this network represents the citations number between authors (not the similarity between authors), then, we have extracted another social network which is the similarity network depending on this network as described in the next section.

The similarity social network is represented by a undirected graph, its nodes present authors and its edges present the similarity between authors. For every node, a weighted vector of keywords is extracted and stored to describe the user's profile as mentioned above.
We suppose that two authors are *structurally similar* if they: cite a given number of authors in common or if they are cited by a given number of authors in common.
The similarity relation in this network is based on two matrices, the co-citation matrix and the bibliographic coupling matrix.

1) Co-citation matrix

The co-citation matrix measures the similarity between authors. It is computed by:

$$C_{ij} = \sum_{k=1}^{n} L_{ki} L_{kj} \tag{1}$$

where L is the matrix representing the social network of citations as mentioned above (see figure 1).
According to this matrix, if two authors cite a given number of other authors in common, then we can say these two authors have similar interests.

2) Bibliographic coupling

The bibliographic coupling matrix is another similarity measure between authors which is given by :

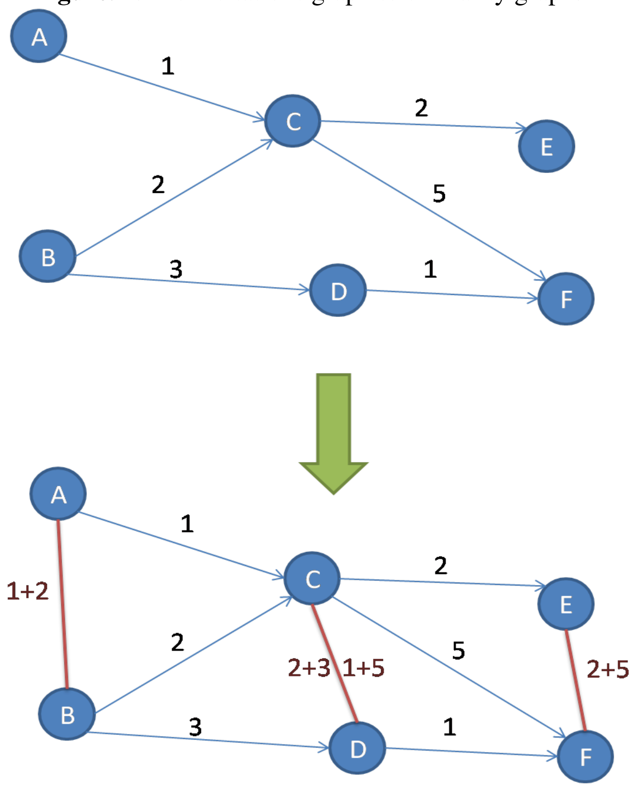
$$B_{ij} = \sum_{k=1}^{n} L_{ik} L_{jk} \tag{2}$$

According to this matrix, if two authors are cited by a given number of other authors (they are in the bibliography of other authors), then these two authors are similar.

3) Structural similarity graph

The similarity graph is defined as the sum of the two previous matrices the co-citation matrix C and the bibliographic coupling matrix B. A similarity relation between two authors is created if they cite the same authors or if they are cited by a common author and if the two nodes i and j satisfy the condition $[B + C][i][j] >= threshold$. In this case we obtain a similarity based social network from the citations based social network (see figure 1).

Figure. 1: From citations graph to similarity graph.



B. The algorithm idea

The idea is to propose a search algorithm which combines semantic, structural and social proprieties:

The semantic part is the information stocked about the actor (the person) within each node. In other terms, it is consists of the user profile.

The structural part is the information described by the network structure. Our contribution consists of using the maximum spanning tree in order to enhance the search performance.

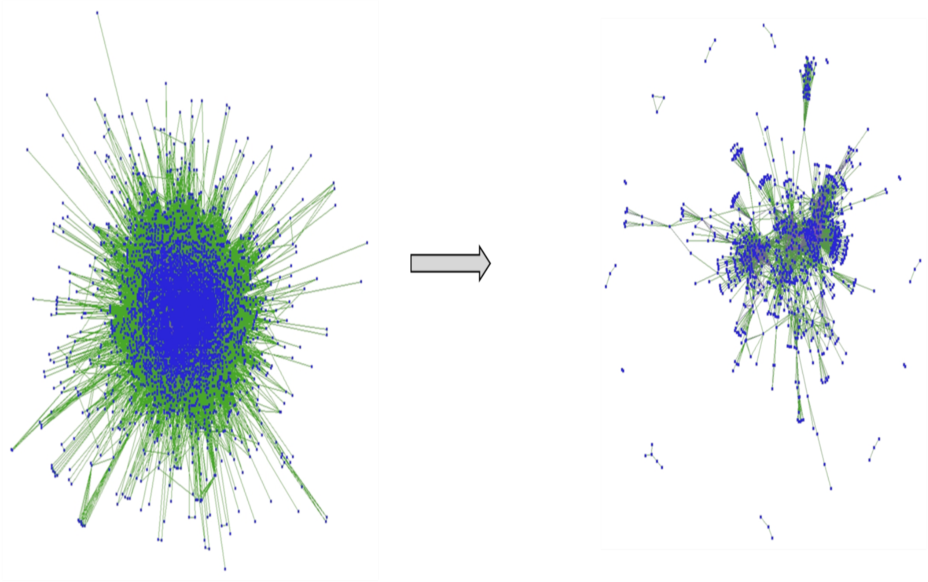
The social part consists of using the betweenness of actors in order to retain certain paths which are more prestigious than others.

1) The semantic part

We compute the similarity between the query R_x of an author X, and the profile of an author Z :

- R_X is the query of X and is composed of a set of terms $T_i : R_X = \{T_1, T_2 .., T_n\}$
- Pro_Z is the profile associated to the actor Z presented by a set of weighted terms : $Pro_Z = \{(T_1, P_1), (T_2, P_2) .., (T_m, P_m)\}$.

Figure. 2: Two undirected similarity graphs extracted from the global directed graph, the first one is denser.



The similarity is given by:

$$sim(R_x, Pro_Z) = \frac{\sum_{j \in inter(R_X, Pro_Z)} Pro_Z.P_j}{\sum_{i=1}^{m} Pro_Z * P_j + |R_X \backslash Pro_Z|} \quad (3)$$

With: $inter(R_X, Pro_Z) =$

$$\{k \in \{1, \dots m\}, \text{such as, } Pro_Z \cdot T_K \in R_X\}$$

2) The structural part

We extract the maximum spanning tree from the evaluated similarity graph using the Kruskal algorithm ([11, 4]) and by taking the maximum edge values instead of the minimum values. We aim to enhance the research by finding an optimized navigation in the spanning tree, instead of exploring the whole or even a part of the graph.

3) Nodes' beetweenness

The betweenness centrality is given by the equation:

$$C_B(i) = \sum \frac{P_{jk}(i)}{Pjk} \quad (4)$$

Where:
$P_{jk}(i)$ is the number of the shortest paths between j and k, which pass from the node i.
P_{jk} is the number of the shortest paths between j and k.
The use of the betweenness allows to prefer certain more privileged search paths for the requested recommendation. The intuition behind using the beetweenness is that choosing theses actors who are often in the shortest paths may lead to enhance the performance of the algorithm. Experiment results on the guided algorithm, which uses an heuristic based on the beetweeness, will illustrate this idea.

C. The algorithm

To elaborate some recommendation, we propose to navigate a covering spanning tree instead of considering the whole graph. This will help to take significant navigation paths and to enhance the system performance.
The recommendation algorithm finds a response to the user query by searching the extracted spanning tree. The *algorithm input* is composed of a query R_x submitted by an author X, this query is formed as a chain of keywords T_i. $R_x = \{T_1, T_2 \dots, T_n\}$. The *algorithm output* corresponds to a response to the author X query which is presented by a weighted sequence of recommended authors $\{(Z_1, P_1), (Z_2, P_2).., (Z_n, P_n)\}$; as well as the semantic chain connecting the two actors X, Z_i [1] (see figure 3).
The algorithm is given as follow:

1. Compute the maximum spanning tree.
2. Compute and store the betweenness of all the nodes.
3. Extract from the spanning tree a ranked list of actors to recommend by using the exhaustive algorithm or the guided one.

D. The exhaustive version

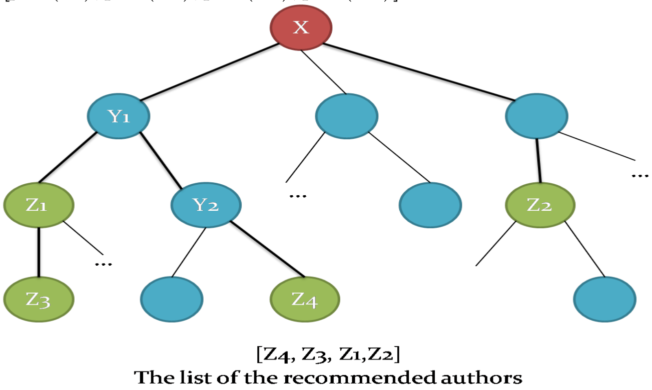
1. Search the spanning tree starting by the user X (figure 3) and using the breadth first strategy. We search for the nodes Z_i where: $sim(R_X, Pro_{Z_i}) >= threshold$ to recommend to X.
2. Compute the rating P_i associated to each author Z_i, this rating depends on two values: the similarity and the betweenness centrality of authors on the path of the solution.

$$P_i = \left(\begin{array}{ll} sim(R_X, Pro_{Z_i}) * \frac{\sum_{j-1}^{l} C_B(Y_j)}{l} & \text{if } l \geq 1 \\ sim(R_X, Pro_{Z_i}) & \text{if not} \end{array} \right) \quad (5)$$

$Y_1, Y_2, \dots, Y_{l-1}$ is the set of authors present on the path relating X to Z_i.

[1]The semantic chain connecting the two actors X, Z_i is constituted of the list of terms extracted form the profile of nodes (authors) relating X to Z_i

Figure. 3: Searching the spanning tree using the breadth first search algorithm - An example of an authors list to recommend can be $[Z_4, Z_3, Z_1, Z_2]$ ranked according to their rating measurements, the semantic chain between X and Z_4 is $[pro(X), pro(Y_1), pro(Y_2), pro(Z_4)]$.



E. The guided version

We propose a second version which is more efficient that allows to search solution, by finding more quickly the search path in the spanning tree instead of applying the breadth first strategy. We use an heuristic allowing to choose the next node to visit among a set of candidates ones; we apply the A* algorithm that allows to choose the node Y that maximise the following heuristic:

$$h(Y) = sim(Pro_X, Pro_Y) * C_B(Y)$$

until we reach the node Z that verifies:

$$sim(X, Z) >= threshold.$$

We can prove that our heuristic is monotone and that it decreases slowly on the solution's path, we can prove also that it recognizes the solution.
On the other hand, we show with experiments (see next section) that this version converges more quickly to the solution and succeeds to explore from 11% to 49% from the spanning tree explored by the exhaustive version.

F. Experimentations

Table 1 presents some statistics about the social network (that describes the similarity between authors): nodes number, edges number and graph density (in social network the graph density is small).

Table 1: Some statistics about the social network describing the structural similarity between authors.

Nodes number	7065
Edges number	1 009 940
Graph density	4, 05. 10-2

We have evaluated the guided version compared to the exhaustive one. We have done ten experiments: each experiment begin with a query elaborated by an author X (which becomes the root of the spanning tree). For each query, we apply both versions of the algorithm and we pick up the following measurements (see table 2):

The rank of the found (recommended) author by the guided algorith remember that the exhaustive algorithm propose for the same query a set of recommended authors and their ranks.

The number of visited nodes by the guided algorithm

The computation time

We notice that for 8 experiments (see table 2), the rank number 1 is found by the guided version, while the rank number 2 is found for the 2 other experiments. Only a part of the spanning tree is searched by the guided version. The search space is thus reduced 11% to 49%. The computation time is also reduced.

IV. Integrating a semantic user profile within guided recommended algorihtm

In guided recommender algorithm, user information is stored in the actor vertex level and can be represented by an ontology-based user profile. This integration raises a number of technical issues which we are investigating in this paper. We address these issues with semantic web technologies to solve various types of concerns, mainly: - data integration from disparate sources constituted by various social networks. - semantic interoperability due to the need of sharing user profile contexts and domain ontologies. The description of relevant main technical issues and possible solutions come in four points:

A. User profile modeling and ontologies

A semantic user profile is a description of a user's interests and disinterests. User profiles will be much more than just a list of interest keywords-they hold information regarding user behavior, context and other preferences. Ontology support inference mechanisms that can be used to enhance recommendation. Previous user profiles (or privileges) data have to be enriched.
Our assumption is that semantic knowledge is an essential part of the user profile context. That needs to use domain ontologies as the fundamental source of semantic knowledge. For the semantic services and profiles to work together, it is indispensable that the things offered in the service definition use the same ontology as the interests from the profile definition.
Ontologies have to be populated with semantic concepts associated to terms extracted from the different social networks. Common terms are easily identifiable because their corresponding concepts can be found in English dictionaries like WordNet. From these extracted terms, an automatic mechanism will create ontology instances using the domain concepts of these terms. The basic idea of our proposal is to match the categories of extracted terms with classes of the ontologies and then link them with the matched ontology class that is most similar to the domain concepts. Each concept in our user profile is annotated with an interest score. Initial user profile ontology reflects a typical user profile in terms of semantic subsumption of domain ontology. Relationships between concepts are easily defined in standard RDF. All information about a user is expressed and stored in

Table 2: Comparaison of the breadth first search algorithm and the A* algorithm: each experience corresponds to a query sent by an author root, the recommended authors correspond to those who have the most important rating. We notice that the first author found by the exhaustive algorithm is the same found by the guided one for 8 experiences. For te two other experiences, the second authors according to the rating measures are found. The search space is considerably reduced when using the guided version

N	The exhaustive algorithm			The A* algorithm		
	Recommended author	**Rating**	**Computation time**	**Recommended author**	**Computation time**	**explored graph**
1	Andrew Emili	0.00064	159,41s	Andrew Emili (1)	109,27s	39.25%
2	G V Belle	0.00141	159,35s	G V Belle (1)	17,45s	21.13%
3	Hans A Kestler	0.00060	150,41s	*Yuichi Asahiro (2)*	11,66s	13.86%
4	Jimin Pei	0.00002	160,61s	Jimin Pei (1)	32,52s	20.02%
5	John F Canny	0.00003	159,99s	John F Canny (1)	21,77s	11.77%
6	C Wang	0.00010	157,37s	C Wang (1)	233,99s	49.13%
7	J Michael Brady	0.00001	162,68s	J Michael Brady (1)	118,74s	41.14%
8	Peter G Neumann	0.00022	160,72s	*Elizabeth J O neil (2)*	40,49s	24.88%
9	Peter Eades	0.00004	153,95s	Peter Eades (1)	54,47s	30.95%
10	Liang Chen	0.00019	161.71s	Liang Chen (1)	14,14s	16.67%

the form of RDF triples. User profile CRUD operations are all based on the SPARQL-based query of these RDF triples.
User preferences could be obtained either manually or automatically. A user profile editor will allow the users to manually create and update their semantic preferences according the existing domain ontologies. Accurate information about the user's interests must be collected and represented with minimal user intervention. This can also be done by passively observing the user's browsing behavior over time and collecting, for example, web pages in which the user has shown interest. Several factors, including the frequency of visits to a page, the amount of time spent on the page, and other user actions such as bookmarking a page can be used as bases for heuristics to automatically collect these information. At this point of view the semantic user profile context is represented as a set of weighted concepts from domain ontology. This set is obtained by collecting the concepts that have been involved in user's actions during a given unit of time. As the user interacts with the system by selecting or viewing new documents, the ontological user profile is updated and the annotations for existing concepts are modified by spreading activation mechanisms.
Use of ontologies may also serve to derive new concepts that are likely to be of interest to the user through semantic spreading activation networks that has been intensively studied as well [14, 30]. A number of previous studies have shown that the spreading process improves accuracy and overcomes the challenges caused by inherent relationships and polysemy in word sense disambiguation process [17, 33] and ontology mapping [30]. Spreading process improves the semantic similarity computation.

B. User profile information extraction process

This process requires extracting user interests and clustering them.

1) Extraction of user profiles

During this step, we extract user interests from various social networks, e.g., Facebook or LinkedIn and determine interlink social data between them. These user interests not only gather the list of activities and preferences that the user filled in during profile creation, but also include the groups to which the user subscribes, the pages marked and the tags he has added. All of this information is represented by a label (e.g., the name of a group, or the description of a bookmark). The extractor uses common processes (tokenization, lemmatization and stemming) to normalize each interest term for the matcher component.

2) Clustering interests around domain-level concepts

The problem here is to discover concepts that gather several interests from the information extracted in the previous step. The idea is to cluster similar interests around a high-level concept. In fact, we want to create clusters, each composed of a high-level concept defined over domain ontologies and a list of interests that are related to this concept. The discovering of these concepts and the matching of an interest towards a concept are performed using state-of-the-art matching approaches.
In the clustering operation, we have to interlink social data from distinct sources distributed across the social networks to build user profile in conformance with our user profile model. In terms of Twitter for example, we define a single social data fragment as being a microblog post. In terms of Flickr and Picassa for example, a single social data fragment is an image.
In the case of Twitter, querying Twitter for all the social data fragments that a user has produced we are provided with an ordered XML response of the microblogs. Based on these informations, we build an RDF representation. To begin with, we create a URI for the data fragment using the derefenceable URL describing the microblog post. We define this as an instance of sioc:Post from the SIOC (Semantically Interlinked Online Community) Ontology [10]. We then associate the data fragment with the person who created it using the URI of the Twitter user. This allows queries to be performed which gather all the microblogs published by that user. The content of the microblog is then associated with sioc:Post instance using the sioc:content property. This forms the full description of the topic of the social data fragment. To enable easier discovery of social data for a given topic we extract all the tags from a given social data fragment.

C. A semi-automatic distributed semantic annotation and indexing of resources

To improve efficient accessing to resources, the system has to analyze and annotate the textual information with concepts that exist in the domain ontologies. An automatic mechanism could achieve this operation in applying matching techniques. The concepts are morphologically compared with the names of the classes and instances of the domain ontologies before to be indexed. Social tagging and the results it generates (folkosonomies), can be considered as new opportunities to free users from the constraints of traditional indexing. Several tags can be associated by a single user to the same resource, and the same tag can be associated to the same resource by different users, tagging actions are usually not isolated. The problem arises here of conflict detection and maintenance of semantic consistency of tag clouds based on ontologies. These operations essentially require the implementation in social networks of a collaborative process that lets move from simple keyword indexing to content indexing guided by ontologies, while retaining the flexibility and the social aspect. The interest of this practice is to allow a further search of content guided by ontologies, rather than by simple keywords. In fact, the tag-based systems suffer from many shortcomings in terms of information retrieval, both caused by the problems of ambiguity and synonymy of keywords by their nature totally flat and the absence of links between tags. In this case, the model-MOAT Meaning Of A Tag would make the relationship between ontologies and tags to solve the problems of conventional systems based on tags (ambiguity, heterogeneity, lack of organization) through the use of formal knowledge bases supporting such systems. It would take into account and model the meaning of the tags via the semantic web concepts, thus offering the possibility of establishing a flexible link between folksonomies and ontologies.

D. Using of semantic matching techniques

Matching services are the runtime on line services to provide effective personalization to recommender systems. The goal of semantic matching is to determine whether a given profile is semantically compatible to a particular service and, if so, how well both do match. Ontologically speaking, from the concepts describing a user's interests and dislikes, we check the subsumption relationships between concepts Match, UsersInterest and ServiceOffer to determine a match degree. Entities may be user profiles, contexts or content descriptors Horrocks et al. have proposed a matching approach that achieves this task.

A number of other approaches have already been proposed in literature to determine the similarity between two ontology concepts. These determine similarity by: measuring the path distance between them [15], evaluating shared information between them [27], recursively matching sub-graphs [9], combining information from various sources [37], analyzing structure of the ontology, and combining content analysis and web search. All these approaches are only able to determine closeness between two concepts,

We solve the matching issues in user profile matching through effective use of ontologies in considering the notion of semantic similarity between two user profiles as inherent relationships between concepts appearing in their respective representation. We use the process of spreading to include additional related terms to a user profile by referring to an ontology.

V. Semantic Interoperability issues

The personalization in a large scale social network leads to a process of user profiling which is inherently distributed. Shared ontologies are critical to guarantee consistent social network portability. In social network, to effectively share and exchange user information, we need to know the semantics of this user information, and therefore resolve the issues related to semantic interoperability. We also need shared vertical ontologies for social network applications covering the same domain. So, users should be able to manage their social profile independently from any social networking platform and application. We have adopted UserML [19] as the user model exchange language. UserML statements represent the user model information. UserML forms the syntactic description in the knowledge exchange process. The merging of partial, decentralized user models is realized by combining the different user profile models, while the inferential integration is done by filters and conflict resolution strategies. To facilitate the use of annotations, high-data visualization RDF and semantic mash-ups should be developed and dedicated search engine should be implemented to use ontologies and semantic annotations to ease searching of annotated resources.

For interoperability purposes, our user model is based on semantic technologies (OWL as description schema, SPARQL query language for information retrieval). SPARQL seems to be the optimal choice for this, since the user model has an OWL representation. This choice is based on current research initiative to leverage semantic technologies for a richer user profile representation. Several user profile models, like GUMO (General User Model Ontology) [18] or UPOS (User-Profile Ontology with Situation-Dependent Preferences Support) [32] are highly relevant to this initiative. GUMO is commonly accepted as top level ontology for user models and is becoming of great importance for the user modeling research community. The combination of the GUMO ontology with the exchange language UserML together with the decentralized u2m.org user model service seems to be promising.

VI. Conclusion and perspective

For personalization purposes, multiple mechanisms may be conjointly used for extending user profiles (set and graph based spreading) and semantic matching (set intersection and bipartite graphs) in recommender systems.

In this paper, we have pointed out the main technical issues and possible solutions raised by integration of an ontology-based semantic user profile within a hybrid recommender system based on our guided recommender algorithm.

We are now working on the elaboration of semantic user preferences by using a domain ontology. The Amazon data sets (http://snap.stanford.edu/) contains 548551 products described by:

- Two identifiers: Id , ASIN: Amazon Standard Identification Number.
- Title,group : (Book, DVD, Video or Music), salesrank.
- Similar : ASINs of co-purchased products.
- Categories : location in product category hierarchy.
- Reviews : Product review information: time, user id, rating, total number of votes on the review, total number of helpfulness votes (how many people found the review to be helpful).

The data preparation process consists of:

1. Elaboration of the products taxonomy.
2. Extraction of the collaboration network (nodes are users).
3. Elaboration of basic and semantic preferences for users.

We use a domain ontology which is a taxonomy as in the Amazon application. The single relation between some pairs of concepts is the relation "Is-a". Such an ontology is defined by a set of concepts terms : $\{C_1, C_2, .., C_n\}$ and a set of pairs of concepts related by the relation Is-a. This ontology can be represented by a tree and forms a taxonomy. We aim to integrate the ontology concepts terms as well as its structure in order to enhance the recommendation quality [28, 29]. As mentioned above, user preferences are actually represented by a weighted vector of terms. We propose to extend user preferences by annotating this weighted vector of terms, using the ontology concept terms ; we thus can derive the semantic profile of user which will be represented by a weighted vector of concepts. For example, suppose that the user u_1 has as basic preferences the weighted vector : $\{(I_1, S_1), (I_2, S_2), (I_3, S_3), (I_4, S_4)\}$ where I_i is a given item and S_i is the score associated to this item. We suppose that these items belong to concept terms of the ontology domain : $\{(I_1, C_2), (I_2, C_3), (I_2, C_4), (I_3, C_4), (I_4, C_2)\}$. Derived semantic preferences will be : $\{(C_2, \frac{S_1+S_4}{2}), (C_3, S_2), (4_4, \frac{S_2+S_3}{2})\}$.

We propose different categories of relevance measures. The basic relevance measure is the classical one which matches the preference terms to the query terms. The semantic relevance measure is the one that matches the semantic preferences to the query concept terms both extracted from the ontology definition. The semantic-structural relevance measure integrates the structural and hierarchical organization of concepts in the ontology. We aim now to test different types of relevance measures in order to propose a metric that enables to elaborate the best recommendation.

References

[1] L. A. Adamic, O.Buyukkokten, E.Adar. A social network caught in the web. *First Monday* 8 (6),2003.

[2] C. Aggarwal,(Ed.). *Social Network Data Analytics. Springer Science and Business Media, Ch. A survey of algorithms and systems for expert location in social networks*, pp. 215–241, 2011.

[3] M. Aktas, M. Nacar, and F. Menczer. Using hyperlink features to personalize web search. In Advances in Web Mining and Web Usage Analysis, *Proceedings of the 6th International Workshop on Knowledge Discovery from the Web, WebKDD 2004*, Seattle, WA, August 2004.

[4] C. Berge. *Graphes*. Gauther-Villars, 1983.

[5] V. D. Blondel, J.-L. Guillaume, R. Lambiotte, E. Lefebvre. Fast unfolding of community hierarchies in large networks. *Jouranl of Statisical Mechanics : Theory and Experiment*,2008.

[6] O. Boydell and B. Smyth. Capturing community search expertise for personalized web search using snippet-indexes. in *Proceedings of the 15th ACM International Conference on Information and Knowledge Management, CIKM 2006*, Arlington, VA, November 2006, pp. 277-286.

[7] S. Brin, L. Page. The anatomy of a large-scale hypertextual web serach engine. In: *7th international conference on World Wide Web*, AUSTRIA,1998.

[8] Christopher S. Campbell, Paul P. Maglio, Alex Cozzi, and Byron Dom. Expertise identification using email communications. In *CIKM*, pages 528–531, 2003.

[9] P. A. Chirita, W. Nejdl, R. Paiu, and C. Kohlschutter. Using odp metadata to personalize search. In Proceedings of the *28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2005*, Salvador, Brazil, August 2005, pp. 178-185.

[10] M. G. Everett , S. P. Borgatti. The centrality of groups and classes. *Journal of Mathematical Sociology* 23 (3), 181–201,1999.

[11] J-C. Fournier. *Théorie de Graphes et applications*. Lavoisier, 2006.

[12] L. C. Freeman. *Centrality in social networks: Conceptual clarification. Social Networks* 1 (3), 215–239, 1979.

[13] Y. Fu, R. Xiang, Y. Liu, M. Zhang,S. Ma. Finding experts using social network analysis. In:*Web Intelligence*. pp. 77–80, 2007.

[14] F. Gasparetti and A. Micarellli. User profile generation based on a memory retrieval theory. In *Proceedings of the 1st International Workshop on Web Personalization, Recommender Systems and Intelligent User Interfaces, WPRSIUI 2005*, Reading, UK, October 2005.

[15] S. Gauch, J. Chaffee, and A. Pretschner. Ontology-based personalized search and browsing. *Web Intelligence and Agent Systems*, vol. 1, no. 3-4, 2003.

[16] J. Grossman. The evolution of the mathematical research collaboration graph. *Congressus Numeratium*, 2002.

[17] H. Haav and T. Lubi. A survey of concept-based information retrieval tools on the web. In *5th East-European Conference, ADBIS 2001*, Vilnius, Lithuania, September 2001, pp. 29-41.

[18] D. Heckmann, T. Schwartz, B. Brandherm, M. Schmitz, M. von Wilamowitz-Moellendorff. GUMO - the General User Model Ontology, in *Proc. of the International Conference on User Modeling, Edinburgh*, UK, 2005.

[19] D. Heckmann, A. Krueger. A User Modeling Markup Language (UserML) for Ubiquitous Computing. In *Proceedings of the 9th International Conference on User Modeling*, Pittsburgh, PA, pp. 393-397, 2003.

[20] Kleinberg, J. M. Authoritative sources in a hyperlinked environment. *Journal of the AM* 46, 604–632, 1999.

[21] Y. Lu, X. Quan, X. Ni, W. Liu, Y. Xu. Latent link analysis for expert finding in user-interactive question answering services. In: *fifth International Conference on Semantics, Knowledge and Grid*, 2009

[22] M. Malek, Dalia Sulieman. Exhaustive and Guided Algorithms for Recommendation in a Professional Social Network, in *Applications of Social Network Analysis Conference*, Zurich 15-17 September, 2010.

[23] D. W. McDonald, M. S. Ackerman. Expertise recommender: a flexible recommendation system and architecture. In: *CSCW '00: Proceedings of the ACM conference on Computer supported cooperative work*. ACM, New York, NY, USA, pp. 231–240, 2000.

[24] Metze, F., Bauckhage, C., Alpcan, T. The "spree" exeprt finding system. In: *ICSC07 : Proceedings on the international conference on Semantic Computing.*, IEEE computer Society, Washington, USA, 2007.

[25] Newman, M. E. J. The structure and function of complex networks. *SIAM Review 45*, 167–256, Mar 2003.

[26] M. E. J. Newman. Coauthorship networks and patterns of scientific collaboration. *Proceedings of the National Academy of Science of the United States (PNAS)*, 101:5200–5205, 2004.

[27] D. Ravindran and S. Gauch. Exploting hierarchical relationships in conceptual search. In *Proceedings of the 13th International Conference on Information and Knowledge Management*, ACM CIKM 2004, Washington DC, November 2004.

[28] Vincent Schickel-Zuber and Boi Faltings. Oss: A semantic similarity function based on hierarchical ontologies. In *IJCAI*, pages 551–556, 2007.

[29] Vincent Schickel-Zuber and Boi Faltings. Using hierarchical clustering for learning theontologies used in recommendation systems. In *KDD*, pages 599–608, 2007.

[30] A. Sieg, B. Mobasher, and R. Burke. Web search personalization with ontological user profiles. In *ACM Sixteenth Conference on Information and Knowledge Management, CIKM 2007*, Lisbon, Portugal, November 2007.

[31] A. Singh and K. Nakata, "Hierarchical classification of web search results using personalized ontologies," in Proceedings of *the 3rd International Conference on Universal Access in Human-Computer Interaction, HCI International 2005*, Las Vegas, NV, July 2005.

[32] Michael Sutterer, Olaf Droegehorn, Klaus David, "UPOS: User Profile Ontology with Situation-Dependent Preferences Support," International Conference on Advances in Computer-Human Interaction, pp. 230-235, em First International Conference on Advances in Computer-Human Interaction, 2008.

[33] J. Trajkova and S. Gauch. Improving ontology-based user profiles. In *Proceedings of the Recherche d'Information Assiste par Ordinateur, RIAO 2004*, University of Avignon (Vaucluse), France, April 2004, pp. 380-389.

[34] J. Tang, J. Zhang, L. Yao , J. Li, L. Zhang, Z. Su, Arnetminer: extraction and mining of academic social networks. In: em KDD, 2008.

[35] Jun Zhang, Mark S. Ackerman, and Lada Adamic. Expertise networks in online communities: structure and algorithms. In *WWW '07: Proceedings of the 16th international conference on World Wide Web*, pages 221–230, New York, NY, USA, 2007. ACM.

[36] Jun Zhang and Mark S. Ackerman. Searching for expertise in social networks: a simulation of potential strategies. In *GROUP*, pages 71–80, 2005.

[37] C. Ziegler, K. Simon, and G. Lausen. Automatic computation of semantic proximity using taxonomic knowledge. In *Proceedings of the 15th ACM International Conference on Information and KnowledgeManagement, CIKM 2006*, Arlington, VA, November 2006, pp. 465- 474.

Author Biographies

Hubert KADIMA received his Master Degree in Computer Science and his PhD in Automatics both from University of Paris VII in France. He has worked as Senior Researcher for more than 30 years in a number of industrial research laboratories (ALCATEL, Schneider, Bull) and has successfully coordinated several research projects funded by various funding European and French agencies. He has also a significant teaching university experience in computing science. Currently, he is Research Director and Head of computing science EISTI/LARIS Laboratory at Cergy Pontoise in France. He has presented many publications in international and national conferences and has authored 5 books. His current research areas include distributed systems, Cloud computing, semantic web and collaborative Engineering Systems.

Maria MALEK is born at Damascus in 1968. She has received her engineering degree in computer science from HIAST (Higher Institute of Applied Sciences & Technology) at Damascus in 1990. She has obtained her Ph.D in computer science of artificial intelligence from the Joseph fourier University of Grenoble, France in 1996. She is actually an assistant professor at EISTI (Ecole Internationale des Sciences du Traitement de l'Information) at Cergy in France, where she acts as head of the decision support engineering track. Her research and teaching activities cover different fields in artificial intelligence, including machine learning, data mining and case-based reasoning. She has published many papers in international and national journals and conferences.