Journal of Universal Computer Science, vol. 16, no. 16 (2010), 2179-2195
submitted: 31/3/10, accepted: 31/7/10, appeared: 28/8/10

The 3A Personalized, Contextual and Relation-based Recommender System

Sandy El Helou
(École Polytechnique Fédérale de Lausanne, Lausanne, Switzerland
sandy.elhelou@epfl.ch)

Christophe Salzmann
(École Polytechnique Fédérale de Lausanne, Lausanne, Switzerland
christophe.salzmann@epfl.ch)

Denis Gillet
(École Polytechnique Fédérale de Lausanne, Lausanne, Switzerland
denis.gillet@epfl.ch)

Abstract: This paper discusses the 3A recommender system that targets CSCL (computer-supported collaborative learning) and CSCW (computer-supported collaborative work) environments. The proposed system models user interactions in a heterogeneous graph. Then, it applies a personalized, contextual, and multi-relational ranking algorithm to simultaneously rank actors, activity spaces, and assets. The results of an empirical evaluation carried out on an Epinions dataset indicate that the proposed recommendation approach exploiting the trust and authorship networks performs better than user-based collaborative filtering in terms of recall.

Keywords: Recommender systems, CSCW, CSCL, trust, algorithms, design, pagerank
Categories: L.3.2, L.3.6, H.2.8, M.5

1 Introduction

We live in an age of information abundance best described as the "Information Overload Age" [Ram, 2001]. It is an age distinguished by a rapidly changing knowledge society [Burch, 2005], and fraught with information thanks to the rapidly evolving technological advances, the Internet revolution, and the popular social media that particularly facilitated the production, distribution and consumption of digital content. Today, more than anytime before, people are challenged to constantly and actively acquire knowledge in order to stay up-to-date. Moreover, the society is confronted with the adverse effects of information overload such as stress, anxiety, and reduced work efficiency at a personal as well as an organizational level [Heylighen, 1999]. Personalized recommender systems are instrumental in overcoming the problem of information overload as they help online users find relatively interesting information, services and products [Im and Hars, 2007]. In online formal and informal learning environments in particular, recommender systems play a key role in recommending appropriate knowledge artifacts and learning activities depending on learner interests [Drachsler et al., 2008; Koper and Tattersall, 2004; Tang and McCalla, 2009].

This paper discusses the 3A personalized, contextual and relation-based recommender system. The proposed system serves two purposes. First, it can recommend new actors, activities and knowledge assets depending on the target user's interest, hence triggering new collaboration and learning opportunities. Second, it can recommend an ordering of existing entities in a workspace according to their predicted importance to the target user and his or her context, thus increasing the working efficiency. The rest of the paper is organized as follows. Section 2 presents the 3A recommender system. Section 3 discusses the results of an empirical validation of the algorithm using an Epinions dataset. Section 4 reviews related work. Section 5 concludes the paper with hints on future work.

2 The 3A recommender system

The 3A recommender system ranks 3A entities (actors, assets, and activities) according to their relative importance to a target actor and his or her context. The proposed system unobtrusively leverages the entities' relative importance by relying on the 3A interaction model to identify and exploit significant user interactions, established relations, and evaluation metadata. Evaluation metadata involve ratings, bookmarks, tags, and reviews provided by users. Exploiting evaluation metadata is particularly useful for recommender systems[Vuorikari et al. 2007]. The 3A recommender system applies the 3A ranking algorithm to simultaneously rank actors, activities, and assets based on their global popularity and more importantly their local one. Local popularity refers to the popularity in the neighborhood of the target actor and his or her context. In this section, the 3A interaction model is briefly described. Then, the recommendation approach and the adopted ranking algorithm are presented.

2.1 3A Interaction Model

The 3A interaction model takes advantage of existing CSCW theories, namely Activity Theory and Distributed Cognition [Halverson 2002], while positioning itself at an adequate formalization level to ease implementation. The model is domain-independent and involves three main constructs also referred to as entities: **Actors** represent entities capable of initiating an event in a collaborative environment, such as regular users or agents. **Assets** represent any kind of resource that is produced, transformed, annotated, assessed and shared by actors. Assets can consist for example of simple text files, RSS feeds, content of wikis, as well as video and audio files. **Activity** spaces represent mediums created by actors to conduct spontaneous or planned, individual or collaborative activities. A role consists of a label and an associated set of rights granted to an actor within an activity space. Activity spaces can have a concrete planning of expected assets (or deliverables) with concrete submission and evaluation deadlines as well as predefined evaluators and submitters. This is particularly useful in project management communities and online educational environments.

The proposed model adopts social media design paradigms and explicitly accounts for Web 2.0 evaluation metadata. More specifically, SALT (Share, Assess, Link, Tag) is an acronym introduced to denote Web 2.0 features encouraging opinion expression and active participation [El Helou et al., 2010]. Actors, activities, and assets can be shared, assessed, linked and tagged. Quantitative assessment is done through rating, while qualitative assessment is achieved by adding bookmarks and comments. With respect to linking or relating entities together, actors can use default relation types (i.e. “friends” for actors, and “sub-space” for activity spaces), but can also define new bidirectional or unidirectional ones and share them with others. A graphical representation of the 3A model and the relations between its entities is presented in Figure 1. CRUD is an acronym used in relational databases or at the user interface level to refer to the four main actions of Creating, Reading, Updating, and Deleting that could be performed on actors, activities and assets.

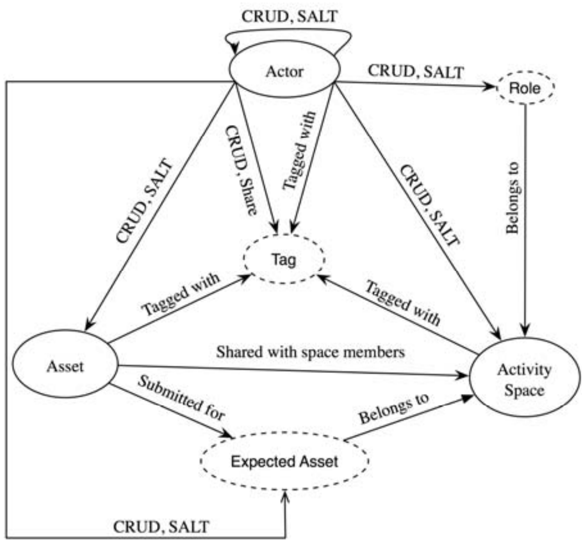

Figure 1: 3A Model Graphical Representation

2.2 The Recommendation Approach

Studies have shown that when it comes to assessing and filtering the information at hand, people are highly influenced by their trusted networks of friends and colleagues [Borgatti and Cross, 2003; Geyer et al., 2008]. Previous research also reports that people are not always ready to explicitly express their preferences and priorities. People perceive such actions as extrinsic to their work and requiring extra effort [Grudin, 1998]. Therefore, the 3A recommender system relies on past interactions involving the target actor and other 3A entities in order to unobtrusively infer the relative importance of a 3A entity for the target actor in a particular context.

The adopted recommendation approach consists of four main steps listed hereafter: graph construction, context definition, importance computation, and ranked lists extraction. These steps can be summarized as follows:

Graph Construction. The proposed recommender system models significant 3A inter-relations in a heterogeneous and multi-relational directed graph. The graph is formed taking as nodes the actors, activity spaces, and assets that the target actor is allowed to access, and as edges the inter-relations between them. Intermediary entities such as roles and tags are also incorporated in the graph as nodes, depending on the relations considered and the level of granularity worth keeping track of. For instance, it might be significant to give different importance weights for space owners and regular members. For that, it is important to include "role" as an intermediary node between spaces and actors, instead of connecting an activity space directly to its members, loosing in the graph the information related to their role. Each bidirectional relation (e.g. "friendship" between two actors) is translated into two directed edges. Additionally, some initially unidirectional relations are complemented by another edge going in the opposite direction, in such a way that the two nodes involved in the relationship reinforce one another. For instance, transforming the initial one-way authorship relation between an asset and its author into two directed edges in opposite directions has two benefits. On the one hand, the edge going from the author to his or her asset allows actors in the author's network to reach this asset through its author. On the other hand, the edge going in the opposite direction (i.e. from the asset to the authors) allows actors that fall on the asset in question, to reach its author and from there discover other potentially interesting assets of the same author. In the same way, if an actor frequently participates in a community's activity space, not only does this imply the actor's interest in the discussion space, but it also indicates the importance of this actor for the community. In other words, if one would like to know what is happening within the collaborative space, it is worth recommending this active participant to him or her.

Context Definition. Whether the 3A ranking algorithm is used for ordering entities already known to the target actor or recommending new ones, it is crucial to take his or her context into account. Context is "any information that can be used to characterize the situation of any entity", an entity being a person, place or object relevant to the user's interaction with the application [Dey, 2001]. It can be measured by relying on implicit interest parameters consisting of users' activities and interactions combined with explicit parameters such as tags [Vuorikari and Berendt, 2009]. Based on the above definitions and on the 3A model's taxonomy, the target actor's context is represented at any point in time by a set of 3A main entities, in addition to intermediary ones (e.g. tags) directly involved in an action performed by the target actor. When an actor performs a search, all tags and entities having attributes (e.g. title, description) that match the search keyword(s) are considered as contextual nodes. Alternatively, when the target user chooses a specific actor, activity space or asset to interact with, all directly related entities constitute the context. For instance, when the selected entity is an activity space, then its members, assets, roles, and directly related activities constitute its context. As it is explained in section 2.5, the algorithm is then expected to bias results towards the context in such a way that entities that have strong connections to contextual nodes get an important ranking. This will lead to suggesting new relevant entities to the target actor depending on the strength of their connection to contextual nodes.

Importance Calculation. Once the graph is formed and the context defined, the 3A ranking algorithm is applied [El Helou, 2009]. The 3A ranking algorithm is based on the key idea of the original pagerank algorithm: a node is recursively important if and only if many other important nodes point to it. With respect to our framework, the idea can be extended and reformulated as follows: *A node is recursively important to a particular root set of nodes (representing the target user and the context) if and* *only if many important nodes connected to the root set via important edge types point to it*.

Ranked List Extraction: Separate lists of actors, activities, and assets are extracted whilst respecting their relative order in the original heterogeneous list. When the aim is to recommend new entities rather than prioritizing existing entities in a workspace, entities that already have a direct connection to the user are skipped. These already related items are more likely to appear first in the recommendation list because they are very close to the target user. Even though it is definitely not beneficial to include them in the recommendation list, they help during the ranking process to reach related nodes that the user is not aware of.

2.3 Original pagerank algorithm

The 3A ranking algorithm takes its roots from the original pagerank algorithm developed by Page and Brin for ranking hypertext documents for Google [Page et al., 1998]. Pagerank is based on the idea that if the owner of a page j links to a page i, he or she is implicitly indicating that page i is important. It follows that the more incoming links a page i has, the more it is considered as "authoritative" or globally important because many pages refer to it. It is not only the number of incoming links that counts but also their quality; the more "authoritative" a page is, the more its outgoing links are valued, and the more importance it can confer to the pages it links to.

The iterative probability equation that translates the algorithm's key idea is described hereafter. A node's conferred importance is divided equally among all nodes it points to. Let *N* denote the total number of Web pages, OutDegree(*j*) the total number of outgoing links from a page (or node) *j*. A transition matrix $T(N \times N)$ is defined such that, each entry T_{ij} is equal to $1/\text{Outdegree}(j)$ if j points to i, and 0 otherwise. Dangling pages are pages with no outgoing links. These pages do not confer any importance to other nodes. To solve this issue, dangling nodes are considered to link to all nodes in the graph with an equal probability. For that, a matrix $D(N \times N)$ is defined such that all entries are 0 except for the dangling columns whose entries are equal to $1/N$. The damping factor *d* represents the probability to follow page links. Since the damping factor *d* is less than 1, the further the nodes are from one another, the less influence they will have on each other's rank. λ defines the probability of falling on a random page. Using λ avoids situations where nodes of a graph component form an importance "sink". λ ensures that no page will have a zero rank and every page is reachable from every other one. Given this, starting with an equal rank of $1/N$ to all nodes, the probability equation of landing on a node i (or rank of a Web page *i*) at each iteration given the ranks of the previous iteration *k*, is given by:

$$p_i^{k+1} = \frac{\lambda}{N} + d\sum_{j=0}^{N}(T_{ij} + D_{ij})p_j^k \ ; \lambda, d > 0; \lambda + d = 1 \qquad \textit{Eq.1}$$

Eq.1 can be understood as a Markov chain where states are pages and the transition between states depends on the link structure of the Web. The equation can be interpreted as the probability for a random surfer to land on a page or node i starting at any node with an equal prior probability, following random links with a probability of *d*, and randomly jumping on a page with a probability of λ.

2.4 Multi-Relational Ranking

Unlike the graph of hypertext documents of the original pagerank algorithm, the social graph of the 3A model involves heterogeneous nodes (i.e. actors, activity spaces, and assets) related by different types of edges that are not necessarily equally important. In such a multi-relational graph, when the surfer falls on a node, he or she can choose to follow different pathways. For instance, if an actor is looking for an expert on a particular topic, he or she can search among his or her friends and "friends of friends" for actors whose profiles match his or her interest. He or she can also choose to traverse different activity spaces, choose one that is relevant to the topic, and from there reach actors who have actively contributed to the space or posted interesting resources. In the same way, given two papers that are equally relevant to a topic, an actor might prefer to first check the one that has been posted or top-rated by a related actor, than the one that have been simply accessed by the same trusted actor. Clearly, the probability to fall on interesting nodes depends upon the probability that the adopted way (combined ways) will lead to them.

To account for the existence of different link types with potentially different importance weights, the original algorithm is modified as follows. The complete multi-relational network is viewed as a combination of separate sub-networks each connecting nodes with one specific edge or relation type. Let *E* denote the set of all edge types. An inner transition matrix $T^e(N \times N)$ and a corresponding weight w_e are defined for each edge type $e \in E$, where w_e is interpreted as the probability for a target actor to follow links within the sub-network *e*. w_e represents the probability to fall on nodes connected by relations of type *e*. Nodes that do not have outgoing links within a sub-network (locally dangling nodes) are considered as linking to all nodes in the sub-network with an equal probability. For that, a matrix $D^e(N \times N)$ is defined for each type of relation *e* such that all entries are 0 except for the dangling node columns where entries equal $1/N$. Then, the iterative stationary probability equation of landing on a node i, is given by:

$$p_i^{k+1} = \frac{\lambda}{N} + d\sum_{e\in E}\left(w_e \sum_{j=0}^{N} (T_{ij}^e + D_{ij}^e) p_j^k \right) ;\sum_{e\in E} w_e = 1;\ \lambda, d > 0;\ \lambda + d = 1 \qquad Eq.2$$

The transition matrix T^e is defined depending on the type of relation it corresponds to. When it comes to relations representing one-time events such as joining a space, T^e is the same as in the original pagerank algorithm. Let $\text{Outdegree}_e(j)$ represent the total number of edges of type *e* coming from *j*, then the entry T_{ij}^e between i and j is expressed as follows:

$$T_{ij}^e = \begin{Bmatrix} \dfrac{1}{\text{Outdegree}_e(j)}, & \text{if } j \text{ points to } i \\ 0, & \textit{otherwise} \end{Bmatrix}$$

Relations resulting from events that can be repeated over time (e.g. updating an asset or accessing an activity space) are treated in a slightly different way. Let R_{ij}^e represent the total number of events of type *e* involving i and j. Then, the probability to jump from j to i is equal to R_{ij}^{e} and normalized by the total number of outgoing relations of type *e* having j as their source node.

$$T_{ij}^{e} = \left\{ \begin{array}{ll} \dfrac{\sum R_{ij}^{\mathrm{e}}}{\sum\limits_{k \in N} R_{kj}^{\mathrm{e}}}, & \text{if } j \text{ points to } i \\ 0, & \textit{otherwise} \end{array} \right\}$$

The "rating" relation type is also handled differently. In this case, the probability to fall on an item i rated by actor j is equal to the corresponding rating value divided by the sum of all ratings issued by actor j and having a value higher than his or her average rating value. In this way, poorly rated assets are not reachable from an actor. Let $\overline{v}_{j}$ denote the average rating given by j and v_{ij} the rating value given by j to i, then T_{ij}^{e} can be written as:

$$T_{ij}^{e} = \left\{ \begin{array}{ll} \dfrac{v_{ij}}{\sum\limits_{k \in N} v_{kj}}, & \text{if } v_{ij} \geq \overline{v}_{j} \\ 0, & \textit{otherwise} \end{array} \right\}$$

Finally, in order to take into account the evolution of the graph over time, one can define significant time frames, and then group relations not only according to their type but also the time frame during which they occurred, giving a higher relative weight to more recent ones.

2.5 Personalized and Contextual Ranking

In [White and Smyth, 2003], pagerank is extended to rank nodes according to their relative importance to a root set of nodes. For that, the initial probability equation is changed in such a way that the random surfer starts at the root set with adequate prior probabilities, follows links with a probability of *d*, and goes back to the root set with a probability of β (where it restarts again). This change leads to a bias towards the root set and the nodes strongly connected with it (because of the iterative process). During their experiment, authors used a value of 0.3 for β while acknowledging that the choice is inherently subjective and dependent upon the objective, nature and structure of the graphs considered.

Similarly, the 3A ranking algorithm is personalized and contextualized by biasing rankings towards the target actor and the context. To do so, we introduce two parameters β_c and β_u. β_c represents the probability to jump back to the contextual nodes and β_u the probability to jump back to the target actor. Also, in order to speed up the algorithm's convergence to the stationary rank vector, the initial probability is set to 0 except for contextual nodes. Let N' be the number of contextual nodes, then each of them receives an equal initial probability of $1/N'$. Also, let R_c represent the set of contextual nodes, and p_c a variable equal to $1/N'$ for contextual nodes, and 0 otherwise. In addition, let u denote the target actor's node and p_u a variable defined such that it is 0 for all nodes except u. To ensure that no node highly connected to the target user but irrelevant (or not enough relevant) to the context gets a high rank, β_u should be considerably smaller than β_c / N'. This choice makes nodes that are relevant to the context (i.e. contextual nodes and those strongly connected to them) achieve top ranks, and those among them that are closer to the target actor achieve better ranking than others. The complete iterative stationary probability equation of landing on node i is given by:

$$p_i^{k+1} = \frac{\lambda}{N} + \beta_u p_u + \beta_c p_c + d\sum_{e\in E}\left(w_e \sum_{j=0}^{N} (T_{ij}^e + D_{ij}^e) p_j^k \right); \qquad Eq.3$$

$$\sum_{e\in E} w_e = 1;\ \lambda, d, \beta_c, \beta_u > 0;\ \lambda + d + \beta_c + \beta_u = 1;$$

$$p_c = \begin{Bmatrix} \frac{1}{N'}, & \text{if } i \in R_c \\ 0, & otherwise \end{Bmatrix};\ p_u = \begin{Bmatrix} 1, & if\ i = u \\ 0, & otherwise \end{Bmatrix}$$

Eq. 3 can be interpreted as the probability to fall on a node in the graph, starting within a set of contextual nodes, following different types of links with a probability of *d* (each with a probability of w_e), jumping to random nodes with a probability of λ, jumping back to the target actor with a probability of β_u, then going back to one of the contextual nodes with a probability of β_c (and restarting again).

2.6 Rank Vector Existence, Uniqueness, and Computation

This section explains how the rank vector whose components are the importance rankings of all graph nodes is obtained. Let $C(N \times N)$ be a matrix such that all row elements are zero except those corresponding to contextual nodes where entries are equal to $1/N'$, and $U(N \times N)$ a matrix having all rows equal to 0, except the one corresponding to the target actor which is equal to 1. Also, let $1(N \times N)$ denote a matrix of 1s. Then, the complete matrix M representing the random walk can be written as follows:

$$M = \frac{\lambda}{N} 1 + d\sum_{e\in E} w_e (T^e + D^e) + \beta_u U + \beta_c C \qquad Eq.4$$

The rank vector I containing the importance rank of each node i can then be written as follows:

$$I^{k+1} = MI^k \qquad Eq.5$$

According to Eq.5, I is an eigenvector of M with eigenvalue 1. To prove the existence and uniqueness of the rank vector I, important properties of the matrix M are discussed hereafter. To start with, each column in M sums to 1 and all entries are positive. This means that every node is reachable from every other one, thanks to the random jump parameter. Thus M is stochastic, irreducible and primitive. As a result, and according to the Perron-Frobenius theorem, M has one positive eigenvalue that is greater (in absolute value) than all other eigenvalues, and one positive eigenvector corresponding to it. Consequently, It is guaranteed that the matrix-free power method will converge to I, the unique leading eigenvector corresponding to the dominant eigenvalue and containing the importance rankings of all graph entities as it is the case for the original pagerank algorithm [Langville and Meyer, 2003]. Authors of the original pagerank algorithm report that with a value of *d* close to 0.85, only 50 to 100 iterations are enough to reach a good approximation of *I* for the Web graph involving billions of hyperlinks [Page et al., 1998].

This paper does not address issues related to space and time complexity or graph and rank update frequency. Still, a fully scalable future implementation can take advantage of reported experiments and proposed solutions related to scaling personalized pagerank [Fogaras et al., 2006; Haveliwala et al., 2003; Jeh and Widom, 2003].

3 Experimental Evaluation on Epinions dataset

This experiment aims at verifying whether the proposed recommendation approach that represents different 3A inter-relations in a multi-layer graph, and computes rankings based on a personalized and contextualized version of pagerank yields relevant top-*N* recommendations.

3.1 Dataset description

The extended Epinions dataset is chosen for the evaluation (http://www.trustlet.org/wiki/Extended_Epinions_dataset). To our best knowledge, it is the only large and publicly available dataset with social networks information in addition to authorship data. The original dataset is reduced to ensure a reasonable validation taking into account the adopted evaluation methods and the ranking approaches compared in this experiment. The evaluation method consists of randomly withdrawing user ratings and trying to predict them. Since unrated reviews cannot be evaluated in this way, they are excluded from the dataset along with their authorship information. While these reviews could have enriched the 3A recommender system, the collaborative filtering method used as a comparison basis cannot make any prediction on them. Actors that share trust information but have not made any rating are also not considered. Distrust information between actors is ignored since none of the ranking approaches exploits them. Since the 3A recommender system considers only ratings values greater or equal to the average, ratings with values less than 3 out of 5 were ignored in the dataset used during the evaluation. Knowing that a bias towards high rating values in Epinions rating distribution has already been reported in (Massa and Avesani, 2007), and that in this dataset in particular, ratings that are below the average only constitute 2.34% of the total number of ratings, removing them is not expected to affect the performance of user-based collaborative filtering in any substantial way. The resulting dataset involves 113,364 actors, 602,309 trust statements, 744,075 rated reviews (47.6% of initial total reviews), 13,348,412 ratings (97.7% of initial total ratings) and 102,652 different topics. Figure 2 shows a mapping of the Epinions data into the 3A model, in which reviews are considered as assets, and subjects or topics are treated as tags. *The unidirectional or bi-directional relations that are translated into bi-directional edges in the graph are displayed in italic*. Relationships between actors, assets and tags are shown along with their relative weight. Weights were chosen empirically taking into account the importance of the different relationships as well as the total number of edges in their corresponding sub-networks.

As noted earlier, the rank of the target user's nodes and contextual nodes are boosted at every step by β_u and β_c respectively. Consequently, and unlike the original pagerank algorithm that only considers the nodes' global popularity, nodes that are directly and indirectly connected to the target user and the context are also boosted. For instance, actors trusted by the target user will rank better than others because they are directly connected to him/her, and they will in turn confer importance to the actors they trust and the assets they authored, rated, and tagged. In the same way, assets and actors linked with tags that are relevant to the context or used by the target actor will also achieve higher ranks than others.

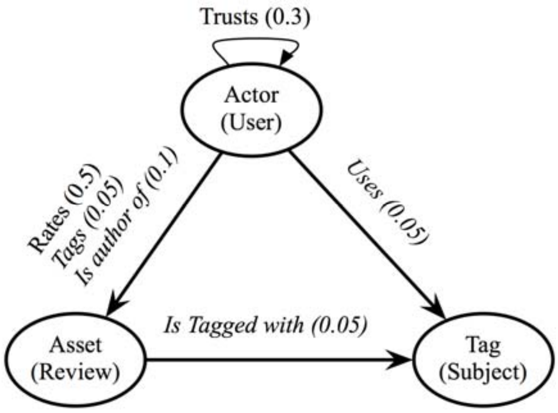

Figure 2: The Epinions network structure

3.2 Evaluation Method

Two different versions of the 3A recommender system are applied: the first exploits trust and ratings relations while the second uses all available relations (trust, ratings, authorship and topic information). The experiment does not involve contextualization. In order to personalize rankings towards every target user, parameters were adjusted as follows $d = 0.75$, $\lambda = 0.05$, $\beta_c = 0$, and $\beta_u = 0.245$. Ranks were obtained using the power method with 5 iterations.

A user-based collaborative filtering method (referred hereafter as CF) is used as a comparison basis. CF computes the similarity in rating behaviour between users. Then CF predicts the rating of an asset unrated by a target user, based on how similar users have rated it. In this experiment, the similarity in rating behaviour $sim(x, y)$ between two actors x and y is calculated using the cosine-based similarity measure [Adomavicius and Tuzhilin, 2005]. Let S_{xy} denote the set of all items co-rated by x and y, $sim(x, y)$ is given by:

$$sim(x,y) = \frac{\sum_{i \in S_{xy}} r_{x,i} r_{y,i}}{\sqrt{\sum_{i \in S_{xy}} r_{x,i}^2} \sqrt{\sum_{i \in S_{xy}} r_{y,i}^2}}$$

The predicted rating $r_{x,i}$ of an item i unrated by a target actor x is computed using the weighted average of all ratings given by users in R_x for item i. Let $r_{y,i}$ denote the rating given by user y to item i respectively. The predicted rating $r_{x,i}$ for a target user x is given by:

$$r_{x,i} = \frac{\sum_{y \in R_x} sim(x,y)\, r_{y,i}}{\sum_{y \in R_x} sim(x,y)}$$

In this experiment, items rated by the top 50 most similar actors to a target user and unrated by the target actor are aggregated, and sorted according to their predicted rating $r_{x,i}$ and then their frequency of occurrence (McLaughlin and Herlocker, 2004).

To compare the performance of the 3 *top-N* recommendation algorithms described above in terms of results accuracy, we use an evaluation approach similar to the one proposed in [Jamali and Ester, 2009]. The *leave-one-out* method is adopted. It consists of withdrawing a rating and trying to predict its rank using the remaining data. In our experiment, random ratings for 1500 randomly selected users are withdrawn. Only top-rated assets by an actor are considered. Actors that have less than 3 ratings are disregarded during the selection process, as user-based collaborative filtering cannot make predictions on them, after one rating is withdrawn.

Recall (or *hit-ratio*) is used to measure the accuracy of a *top-N* recommended list for a target user [Kim et al., 2007]. A hit occurs every time a withdrawn rating for a user appears in *top-N* recommended list for that user. For each of the three recommendation algorithms used in this experiment, recall is computed by dividing the number of hits achieved by the total number of withheld ratings.

3.3 Results

Figure 3 shows the average recall value achieved by the three different ranking algorithms for different values of *N*. Evaluating the algorithm's performance with also big values of *N* is explained by the fact that the dataset involves more than 13 millions ratings, and there are significantly more assets than actors in the dataset. For *N*=10, CF performs slightly better than the 3A ranking algorithm when only rating and trust relationships are taken into account, despite the fact that the latter uses more information. A slight improvement of user-based collaborative filtering over trust-based recommendation algorithms is also reported in other experiments and explained by the fact that users do not necessarily rate the same reviews as the people with whom they have issued trust statements [Jamali and Ester, 2009; Walter et al, 2009]. When the 3A ranking algorithm combines trust, authorship, rating and topic information, it achieves a considerably better recall than CF for small as well as large values of *N*.

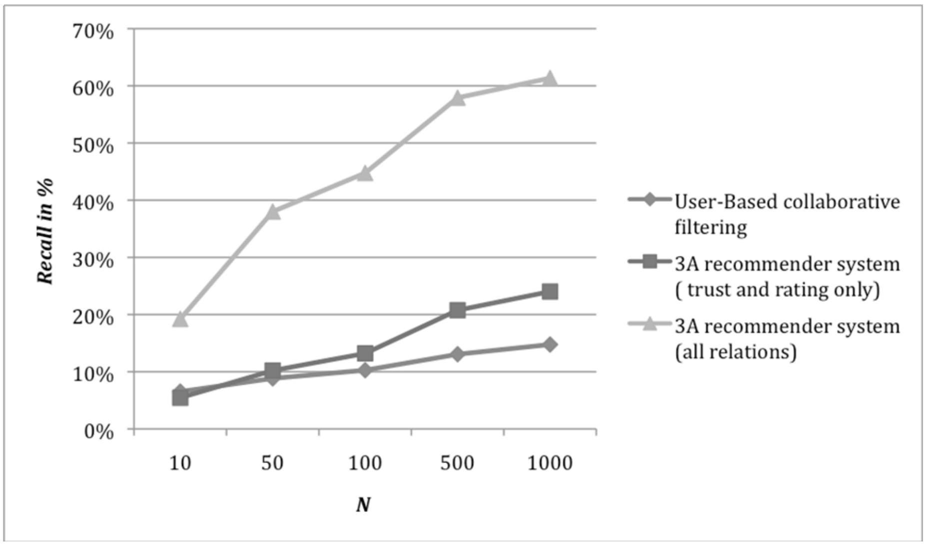

Figure 3: Average recall achieved by the different recommendation approaches

4 Related Work

Many studies on recommender systems for Web applications can be found in the literature. In particular, several recommender systems specifically dedicated to learning environments are proposed [Anderson et al., 2003; Rafaeli et al., 2004; Manouselis et al., 2007]. With respect to the adopted recommendation approaches, most of the existing systems use traditional collaborative filtering where items are recommended based on how “like minded” people rated them. Some combine collaborative filtering with content-based filtering where items are recommended if they are similar in content to items the target user has previously liked. Others use ontology-based filtering that define sequencing rules, model the fine-grained learner’s preferences and competences, and compare them against the characteristics of the learning resource [Shen and Shen, 2004]. This approach is usually computationally expensive and restricted to one domain. In addition, compared to the 3A model that simultaneously ranks actors, activities and assets, most of these cited systems are concerned with recommending learning resources only. Only a few also recommend people such as Altered Vista (Recker et al., 2003) and learning activities such as Cyclades [Avancini et al., 2007]. The difference between Altered Vista and the 3A recommender system is that the former requires explicit and active user input, while the latter rely on user networks and previous actions as implicit preference indicators. On the other hand, Cyclades recommends folders and users using content-based filtering in addition to rating-based measures for finding similar folders. The 3A recommender system proposes a more general framework that does not only rely on ratings and folder ownerships but also exploits other 3A inter-relations and evaluation metadata to leverage user preferences. In particular, relying on Web 2.0 evaluation metadata such as tags, reviews, and ratings is not yet widely used in recommender systems targeting learning environments.

As far as recommender systems targeting general-purpose Web 2.0 applications as concerned, several rely on tagging and social bookmarking behaviour [Gulli et al., 2009; Symeonidis et al., 2008]. For instance, TC-SocialRank [Hotho et al., 2006] presents a link-based algorithm for folksonomy systems that ranks users, bookmarks and shared resources taking into account temporal and user-clicks information. In addition, several recommendation algorithms that rely on both user ratings and social networks (e.g. friendship and/or trust network) are proposed in the literature [Walter et al., 2009; Ben-Shimon et al., 2007]. The difference with the 3A ranking algorithm is that its underlying graph is heterogeneous and multi-relational; it is not limited to actors related by a monolithic relation but also incorporates different node types and combines diverse relations. This is due to the fact that the 3A model targets collaborative environments where users can undertake more actions than merely tagging, bookmarking or rating and where recommendation is not limited to resources such as movies or documents but also extends to people and activity spaces. Therefore, more generalized interaction models and recommendation algorithms are required to be able to infer user interests and preferences from significant inter-relations between actors, activities and assets.

Recommender systems that adopt graph-based approaches and link analysis algorithm already exist in the literature. [Huang et al., 2002] presents a graph-based recommender system for digital libraries where a two-layer graph is used to represent similarity in content between books, similarity in demographic information between people as well as “purchase” relation connecting people to books. Then, the recommendation task consists of traversing the graph to find weighted paths from the target person to different books. Just as in the 3A recommendation model, first-degree associations (in that case, books that users already purchased) are only used to lead to other ones and are skipped in the final recommendation list. The difference with the 3A model is that the latter is applied in a different context and exploits social networks and user interactions. In addition, the 3A model ranks entities by applying a personalized and contextualized version of the original pagerank algorithm based on global and local popularity measures rather than a graph-search technique. On the other hand, [Wang et al., 2008] propose a graph-based approach that combines different object types linked by diverse relations. It relies on a random walk algorithm based on pagerank to compute the importance of objects in an educational portal. In addition, a more general framework called fusion also based on a random-walk algorithm and combining inter and intra-links among multiple-type objects is introduced in [Xi et al., 2004]. None of these two papers addresses the issue of having different weights for different relation types, neither do they personalize or contextualize rankings. Finally, with respect to personalizing recommendations in working and learning environments, a personalized activity prioritization approach that identifies different types of users’ actions and exploits them using a Support Vector Machine model is presented in [Li et al., 2007]. The 3A recommendation algorithm also exploits user actions. However, it is not limited to activities. Finally, it does not only recommend an ordering of existing entities but also recommends new ones taking into account the target user’s context.

5 Conclusion and Future Work

This paper presented a personalized, contextual and multi-relational ranking system that can simultaneously rank actors, activities, and assets in CSCW and CSCL environments. The evaluation carried using a large Epinions dataset shows that the presented approach outperforms a user-based collaborative filtering algorithm used as a comparative basis. More evaluations will be carried out in the future using data from social media sites offering public API's such as LinkedIn (http://www.linkedin.com/). Studies regarding the algorithm's sensitivity to its different parameters will be conducted. As far as the relation types linking 3A entities are concerned, further experiments can help identify the significant relations that should be taken into account to improve recommendations, and study the impact of different relation weights' distribution on the algorithm. Finally, future investigations will also address different challenges related to how best to display recommendations, and how to update them taking into account the user's online feedback.

Acknowledgements

This research was partially funded through Projects (Palette, ROLE, STELLAR) that are part of the Sixth and Seven Framework Program for Research and Technological Development of the European Union in Information and Communication Technologies, as well as through the Innovation Funds for educational projects (FIFO) of the Swiss Federal Institute of Technology in Lausanne (EPFL).

References

[Anderson et al., 2003] Anderson, M., Ball, M., Boley, H., Greene, S., Howse, N., Lemire, D., McGrath, S.: "RACOFI: a Rule-Applying Collaborative Filtering System"; Proc. IEEE/WIC COLA'03, NRC 46507,Halifax, Canada (2003).

[Avancini et al., 2007] Avancini, H., Candela, L., Straccia, U.: "Recommenders in a personalized, collaborative digital library environment"; Journal of Intelligent Information Systems, 28, 3 (2007), 253–283.

[Ben-Shimon et al., 2007] Ben-Shimon, D., Tsikinovsky, A., Rokach, L., Meisles, A., Shani, G., Naamani, L.: "Recommender System from Personal Social Networks"; Advances in Intelligent Web Mastering, 43 (2007), 47-55.

[Borgatti and Cross, 2003] Borgatti, S.P., Cross, R.: "A Relational View of Information Seeking and Learning in Social Networks"; Management Science Archive, 49, 4 (2003), 432-445.

[Burch, 2005] Burch S.: "The Information Society / the Knowledge Society" In Word Matters: multicultural perspectives on information societies; C & F Éditions, Brussels (2005).

[Dey A., 2001] Dey, A. K.: "Understanding and Using Context"; Personal and Ubiquitous Computing, 5, 1 (2001), 4-7.

[Drachsler, 2008] Drachsler, H., Hummel, H., Koper, R.: "Navigation support for learners in informal learning environments"; Proc. Recommender systems, ACM, Lausanne (2009), 303-306.

[El Helou et al., 2009] El Helou, S.; Gillet, D.; Salzmann, C.; and Sire, S.: "The 3A Contextual Ranking System: Simultaneously Recommending Actors, Assets, and Group Activities"; Proc. The third ACM Conference on Recommender Systems (RecSys'09), ACM, New York (2009), 373-376.

[El Helou et al., 2010] El Helou S., Li S., Gillet, D.: "The 3A Interaction Model: Towards Bridging the Gap between Formal and Informal Learning"; Proc. The third International Conference on Advances in Computer-Human Interactions (ACHI'10), IEEE, St. Martin (2010), 179-184.

[Fogaras, 2004] Fogaras, D., Rácz, B., Csalogány, K., Sarlós, T.: "Towards Scaling Fully Personalized PageRank: Algorithms, Lower Bounds, and Experiments"; Internet Mathematics. 2, 3 (2007), 333-358.

[Geyer et al., 2008] Geyer, W., Dugan, C., Millen, D.R, Muller, M., Freyne, J.: "Recommending topics for self-descriptions in online user profiles"; Proc. The ACM Conference on Recommender Systems (RecSys '08), ACM, New York (2008), 59-66.

[Grudin, 1998] Grudin, J.: "Why CSCW applications fail: problems in the design and evaluation of organizational interfaces"; Proc. The 1988 ACM Conference on Computer-Supported Cooperative Work (CSCW '88), ACM, New York (1998), 85-93.

[Gulli et al., 2009] Gulli, A., Cataudella, S., Foschini, L. "TC-SocialRank: Ranking the Social Web."; Proc. The 6th international Workshop on Algorithms and Models For the Web-Graph, LCNS 5427, Springer, Barcelona (2009), 143-154.

[Halverson, 2002] Halverson C.: "Activity theory and distributed cognition: or what does CSCW need to do with theories?"; Computer Supported Cooperative Work, 11, 1(2002), 243-267.

[Haveliwala, 2003] Haveliwala, T., Kamvar, S., Jeh, G.: "An Analytical Comparison of Approaches to Personalizing PageRank"; Technical Report. Stanford, US (2003). Retrieved April 8, 2009 from http://infolab.stanford.edu/~taherh/papers/comparison.ps

[Heylighen, F 1999] Heylighen, F. "Change and information overload: negative effects"; Principia Cybernetica Web. Retrieved May 26, 2009 from: http://pespmc1.vub.ac.be/CHINNEG.html

[Hotho et al., 2006] Hotho, A., Jäschke, R., Schmitz, C., Stumme G. 2006.: "FolkRank: A Ranking Algorithm for Folksonomies"; Proc. Workshop on Information Retrieval (FGIR), Germany (2006).

[Im and Hars, 2007] Im, I., Hars, A. "Does a one-size recommendation system fit all? The effectiveness of collaborative filtering based recommendation systems across different domains and search modes"; ACM Trans. Information Systems, 26,1 (2007), 4.

[Jamali and Ester, 2009] Jamali, M., Ester, M.: "Using a trust network to improve top-N recommendation"; Proc. The third ACM Conference on Recommender Systems (RecSys '09), ACM, New York (2009), 181-188.

[Jeh and Widom, 2003] Jeh, G., Widom, J.: "Scaling personalized web search"; Proc. ACM World Wide Web Conference, ACM, New York (2003), 271-279.

[Kim, 2007] Kim, H.-N., Ji, A.-T., Kim, H.-J., Jo, G.-S.: "Error-Based Collaborative Filtering Algorithm for Top-N Recommendation"; Advances in Data and Web Management , 594-605.

[Koper, 2004] Koper, R., Tattersall, C.: "New directions for lifelong learning using network technologies"; British Journal of Educational Technology, 35, 6 (2004), 689-700.

[Langville and Meyer, 2003] Langville, A.N., Meyer C.D.: “Deeper Inside PageRank”; Internet Mathematics, 1,3 (2003), 335-380.

[Li et al., 2007] Li, L., Muller, M. J., Geyer, W., Dugan, C., Brownholtz, B., Millen, D. R.: “Predicting individual priorities of shared activities using support vector machines”; Proc. The sixteenth ACM Conference on Conference on Information and Knowledge Management (CIKM '07), ACM, New York (2007), 515-524.

[Manouselis et al., 2007] Manouselis N., Vuorikari, R., Van Assche, F.: “Simulated analysis of MAUT collaborative filtering for learning resource recommendation”; Proc. Workshop on Social Information Retrieval for Technology-Enhanced Learning (SIRTEL), 2nd European Conference on Technology Enhanced Learning (EC-TEL'07), Crete, Greece (2007).

[Massa and Avesani, 2007] Massa, P., Avesani, P.:“Trust-aware Recommender Systems”; Proc. Recommender systems, ACM, Minneapolis (2007), 17 - 24.

[Page et al, 1998] Page, L., Brin, S., Motwani, R., Winograd, T.: “The PageRank citation ranking: Bringing order to the Web”; Technical report, Stanford Digital Library Technologies Project (1998).

[Rafaeli et al., 2004] Rafaeli S., Barak M., Dan-Gur Y., Toch E.: “A Web-based environment for learning, assessing and knowledge sharing in communities”: Computers and Education, 43 3 (2004), 273-289.

[Ram, 2001]. Ram S.: “Intelligent Agents and the World Wide Web: Fact or Fiction?”; Journal of database management, 12(2001), 1, 46-47.

[Recker et al., 2003] Recker, M., Walker, A., Lawless, K.: “What do you recommend? Implementation and analyses of collaborative information filtering of web resources for education”; Instructional Science, 31, 4 (2003), 299-316.

[Shen and Shen, 2004] Shen L., Shen R.: “Learning Content Recommendation Service based on Simple Sequencing Specification”; Proc. International Conference on Web-Based Learning, LNCS 3143, Beijing (2004), 363-370.

[Symeonidis et al., 2008] Symeonidis, P., Nanopoulos, A., Manolopoulos, Y.: “Tag recommendations based on tensor dimensionality reduction”; Proc. The 2008 ACM Conference on Recommender Systems (RecSys'08), ACM, New York (2008), NY, 43-50.

[Tang and McCalla, 2009] Tang, T., McCalla, G.: “A Multidimensional Paper Recommender: Experiments and Evaluations”; Internet Computing, IEEE, 13, 4 (2009), 34-41.

[Vuorikari and Berendt, 2009] Vuorikari, R., Berendt, B.: “Study on Contexts in Tracking Usage and Attention Metadata in Multilingual Technology Enhanced Learning”; Proc. Exploitation of Usage and Attention Metadata Workshop, Lübeck (2009).

[Vuorikari et al., 2007] Vuorikari, R., Manouselis, N., Duval, E.:“Metadata for social recommendations: storing, sharing and reusing evaluations of learning resources”. Social Information Retrieval Systems: Emerging Technologies and Applications for Searching the Web Effectively, Goh, D. and Foo S. (Eds.), Idea Group Inc., 87-107.

[Walter et al., 2009] Walter, F. E., Battiston, S., Schweitzer, F.: “Personalised and dynamic trust in social networks”. Proc. The third ACM Conference on Recommender Systems (RecSys '09), ACM, New York, NY(2009), 197-204.

[Wang et al., 2008] Wang, X., Yuan, F., Qi, L.: “Recommendation in Education Portal by Relation Based Importance Ranking”. Proc. 7th international Conference on Advances in Web Based Learning, LCNS 5145, Springer, Jinhua (2008), 39-48.

[White and Smyth, 2003] White, S. Smyth, P.: "Algorithms for estimating relative importance in networks"; Proc. Ninth ACM SIGKDD international Conference on Knowledge Discovery and Data Mining (KDD '03), ACM, New York (2003), 266-275.

[Xi et al, 2004] Xi, W., Zhang, B., Chen, Z., Lu, Y., Yan, S., MA, W.Y.: "Link fusion: a unified link analysis framework for multi-type interrelated data objects"; Proc. 13th international Conference on World Wide Web, ACM, New York(2004), 319-327.