Title: Recommender Systems using Social Network Analysis:
Challenges and Future Trends

Name: Johann Stan, Fabrice Muhlenbach, Christine Largeron

Affil./Addr.: Laboratoire Hubert Curien, Université Jean Monnet
Saint-Étienne, France
johann.stan@univ-st-etienne.fr, fabrice.muhlenbach@univ-st-etienne.fr
christine.largeron@univ-st-etienne.fr

Recommender Systems using Social Network Analysis: Challenges and Future Trends

Synonyms

Recommendation systems, Information filtering, Collaborative filtering, Content-based filtering

Glossary

Recommender System (RS): Special type of information filtering system that provides a prediction that assists the user in evaluating items from a large collection that the user is likely to find interesting or useful.

Status update (micropost): Short message, shared in an online social platform, expressing an activity, state of mind or opinion.

Folksonomy: Whole set of tags that constitutes an unstructured collaborative knowledge classification scheme in a social tagging system.

Definition

Recommender systems (RSs) are software tools and techniques dedicated to generate meaningful suggestions about new items (products and services) for particular customers (the users of the RS). These recommendations will help the users to make decisions in multiple contexts, such as what items to buy, what music to listen to, what online news to read [19], or, in the social network domain, which user to connect to or which users to consider as a trustful adviser.

Overview

Main Components of a Web 2.0 Social Network

A social network can be defined as a set of entities interconnected and it is usually represented as a graph where the entities are described by nodes and their relationships by links. It should be noticed that this concept is not limited to the case of online social networks such as *Facebook*, *LinkedIn*, *MySpace* or *Twitter*, the main focus of our work. A common characteristic of these networks, and more specifically modern online social networks, is that they are composed of (i) users (with a user profile, activities and connections) and (ii) social objects representing the intermediations, e.g. topics of user interactions, shared videos, photos.

The user profile generally includes static personal information, such as the name, email and address, as well as more dynamic information about the interests and information needs of the user. The role of the user profile is essential in online communities. Generally user profiles are different from one application to another, as users present themselves differently, based on the targeted population of the given application (which are sometimes very specific). Another dimension of users is represented by the activities they perform in the social platform. This includes content sharing, media uploading and content description (such as photo tagging). Finally, the third dimension of users is represented by the social connections they establish with others in the network. Users in these online networks are generally connected to different communities, belonging to different social spheres (e.g., friends, family, coworkers).

Another important user characteristic is related to trust. Indeed, the different applications on social content sites allow users to be closer to their communities and to be aware of peer activities and opinions. This brings new dimensions to trust and allows users to have higher confidence in the recommendations, suggestions and sentiment of friends.

Shared social objects influence interactions between users. An object in this context has a concrete and perceptible, physical and/or numeric, manifestation. Some objects are the source of conversational interactions and keepers of collective attention. They constitute a conversation support. In our actual digital context objects are mainly multimedia ones as articles (*Wordpress*, *Wikipedia*), videos (*Youtube*, *Dailymotion*), pictures (*Flickr*, *Picasa*) or specific status updates shared by users.

In such systems, users can employ different types of annotations to describe social objects: structured annotations (in this case, the terms employed in the annotation are regulated by a common domain vocabulary that must be used by the members of the system), semi-structured annotations (these annotation are generally freely selected keywords without a vocabulary in the background, and a collection of these annotations is called a *folksonomy*). The last category of such annotations is unstructured, which is the most frequently used in social platforms, and therefore we describe it in more detail.

This can be found in the majority of social networks and microblogging systems and primarily consists of free texts in the form of short messages describing a resource, a finding, an impression, a feeling, a recent activity, mood or future plan. A common practice is either to express an opinion about the resource (e.g., web page) or to provide its short summary for the community.

The limitations of this kind of content sharing from the viewpoint of information retrieval and knowledge management are similar to that of social tagging, as users have complete freedom in the formulation of these messages. More concretely, it is difficult to extract interesting topics or named entities from such messages, given the fact that there is an ambiguous, frequently changing underlying vocabulary.

Recommender Systems and Social Platforms: the Mutual Benefits

Nowadays, the wide use of Internet around the world allows a lot of people to connect. This explosion of the Web 2.0 (blogs, wikis, content sharing sites, social networks, etc.) gives rise to a growing need for RSs based on social and information network mining methods. For such systems, the underlying social structure, also called social network or virtual community, can be leveraged.

The substantial growth of the social web poses both challenges and new opportunities for research in RSs. The main reason for this is the fact that the social web transforms information consumers into active contributors, allowing them to share their status, comment or rate web content. Finding relevant and interesting content at the right time and in the right context is challenging for existing recommender approaches.

At the same time, the major added value of social platforms is to encourage interaction between users. Each interaction can be extracted and used as an input for the RS, as it helps to better understand the user interests and information needs. Also, the structure of the underlying social network in a social platform can contribute to generate recommendations that are more trusted by users (e.g., by considering the social distance in the recommendation process, as generally we trust more recommendations from closer connections). Therefore, we can conclude that the social web provides a huge opportunity for improving RSs (Fig. 1).

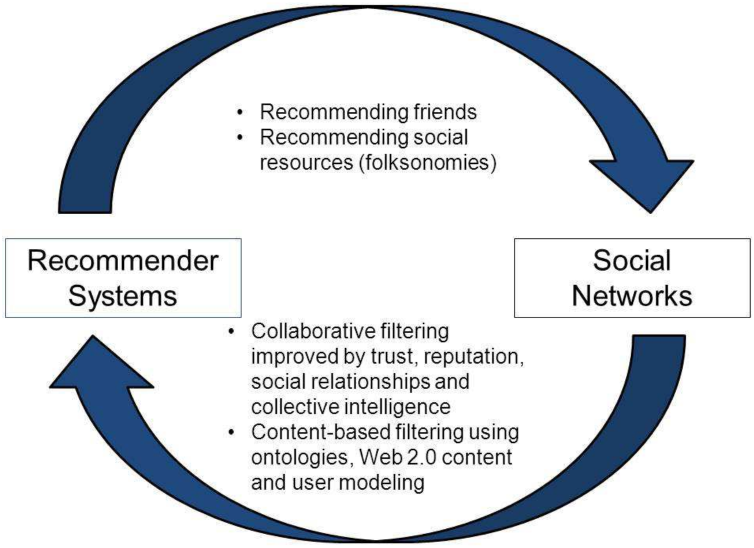

Fig. 1. Reciprocal contributions made by recommender systems to social networks

On the other hand, RSs can clearly help to improve user participation in social systems, as they can recommend new friends or interesting content. Thus, the user will be more motivated to keep on-going participation in the social platform, because the more content he/she shares, the more relevant connections the system can recommend, having a precise profile about him/her.

Using this connection between social platforms and RSs, new scenarios can be defined for advanced applications, such as people recommendation or various content recommendations (e.g., tags for photo annotation).

Introduction and State of the Art

Social Network Analysis

Social network analysis and social mining can be very useful in this context where RSs can take benefit from social networks and conversely, where the formation and evolution of the network can be affected by the recommendations. In order to illustrate this point, we can mention three well known tasks in social network analysis and social network mining:

- The first one is the identification of key actors which play a particular role or which have a particular position in the network. Different indicators, such as the centrality or the prestige were initially introduced mainly in order to highlight the "most important" actors in the network [22]. With the appearance of online social networking, these measures were recently revisited to detect actors called, depending on the authors, mediators, ambassadors or experts. Among the actors who have received a lot of attention appears notably the influencer who can be defined as an actor who has the ability to influence the behaviour or opinions of the other members in the social network [2]. The identification of the influencers can be seen as an optimization problem better known as "influence maximization" (or "spread maximization") that is NP-complete but approximated solutions can be determined thanks to greedy algorithms like "Cost-Effective Lazy Forward" (*CELF*) algorithm or its extensions *Newgreedy*, *Mixedgreedy* or *Celf++* [12; 4].
- Another well known problem in the context of social networks is that of community detection. This problem has mainly been studied in the literature in the case where the community structure is described by a partition of the network actors where each actor belongs to one community [20; 13] and among the core methods we can mention those that optimize a quality function to evaluate the goodness of a given partition, like the modularity, the ratio cut, the min-max cut, or the normalized cut, the hierarchical techniques like divisive algorithms based on the minimum cut, spectral methods or Markov Clustering algorithm and its extensions. However, in real networks, an actor can often belong to several

groups and these overlapping communities can be detected using for example the clique percolation algorithm implemented in *CFinder* or *OSLOM* (Order Statistics Local Optimization Method). Other recent works have attempted to detect communities, taking into account the profile of the users and their relationships [3]. These methods can be applied to determine groups of users with similar characteristics or the same interests and consequently, they can be integrated in neighbourhood-based collaborative systems.

- The evolution of the network is another challenge. Indeed, in many networks, the structure of the network, in other word the actors as well as their relationships, changes quickly over time. The identification of evolving communities or their detection over time is also a subject of recent research which can be integrated in systems to improve recommendations but the dynamic analysis of the network is also related to the link prediction problem which aims to determine the appearance of new links or the deletion of links in the network [18; 15; 5; 8]. It is obvious that link prediction can be useful for people recommendation and, conversely, recommendation approaches can allow to predict the evolution of the network. This temporal dimension is notably important in the context of mobile applications in which moving actors are interacting with each other.

Recommender Systems

The field of social network analysis is a complex and rapidly changing area. To understand the mutual contributions of social networks in recommender systems (and vice versa), it is necessary to clarify the basic principles of these systems.

RSs are dedicated to the help of the users when they must make a decision, taking as basis the fact that in ordinary life people often make decisions based on the recommendation of others. At work, employers count on recommendation letters when they want to recruit new employees; with friends, we talk about books that we loved to read, music or movies that we liked, purchases that have given us satisfaction, or products that disappointed us; and more generally, we trust reviews of specialists before seeing a TV show, an art exhibit, or purchasing an item. This behaviour is based both on the belief that our friends have similar tastes to ours, and on the trust that we can provide to the expert opinion. The recommendations provided by automated systems are trying to mimic those two principles, depending on the available information, and they are supplied to the users in the form of a prediction or a list of items.

The information used for the recommendation process can be extracted from the content available from the users and the items, or it can be inferred from the explicit ratings when the users are asked to rate the items. Depending on the way of how the information is used, the RS is considered to be a content-based, a collaborative filtering or a hybrid (where both information, collaborative and content-based, are used) RS [1].

Whatever approach is used, the key elements of an RS are (i) the users, (ii) the items, and (iii) the transactions. The users of an RS, which may have very diverse goals and characteristics, are both those who benefit from the system and those who supply it with information. Items are the objects (products or services) that are recommended, and they may be characterized by their complexity and their value or utility for a given user. Transactions are the recorded interactions between a user and the RS, especially the relation between a user and a given item, which can be an explicit feedback, e.g., the rating of a user for a selected item.

In the content-based approach, which has its roots in information retrieval and information filtering research, an item is recommended to a user based upon a description of the item and a profile of the user interests [19]. This family of RSs has some advantages (user independence, transparency, easy recommendation of new items) but also some drawbacks: content analysis is limited and the system suffers from over-specialization that leads to homophily (a person is only recommended by people who think like he or she).

In the collaborative filtering approach, an item is recommended to a given user by following another way: the collaborative filtering methods produce user specific recommendations of items based on patterns of ratings without need for exogenous information about either items or users [19]. The preferences of the users are explicit: the users are asked to rate the items (e.g., in terms of 1–5 star scale or "I like" / "I don't like"). This approach needs only a set of ratings of users on sets of items: a list of n users, a list of m items and a rating $r_{x,t}$ indicates the rating of user x on the item t. In a typical collaborative filtering scenario, it is very rare (if not impractical) for a user x to rate all the m items, so the R matrix of all ratings $users \times items$ is sparse. To result in recommendation, the collaborative filtering can be either neighbourhood-based (memory-based) or model-based [17; 19]. The model-based approaches try to propose a model able to predict the unknown rating of a user x for an item t by discovering the underlying preference class of users and the category class of the items. In neighbourhood-based collaborative filtering, the rating matrix R is directly used to predict ratings for new items, either when the neighbourhood derives from a similarity between the users (for user-based systems), or when the neighbourhood derives from a similarity between the items (for item-based systems), e.g., two items are considered as neighbours if several users have rated these items in a similar way. In most cases, the similarity estimated between users or items in these approaches are Pearson correlation or vector cosine-based similarity.

The efficiency of an RS is measured in terms of relevance of the recommendations and forecast accuracy, in particular seeking to narrow the difference between the predicted ratings made by the system and the real ratings made by the users.

Moreover, the system has to be a good filtering system and not present to users uninteresting items, while not missing interesting items (e.g., in the case of commercial RS, for increasing the number of items sold). It is important to propose to the users items that might be hard to find without a precise recommendation. Many systems suffers from novelty discovery, i.e. they fail to find serendipitious items. All these properties will increase the user satisfaction and the fidelity to the use of the system.

The latest trends in RS domain seek to take into account how human beings function with their peers, especially in their interpersonal behaviors, which brings it closer to the field of social network analysis. Some users try to find credible recommenders so they can follow them, it is thus interesting to investigate the most influential members. It is also important to develop a method to better understand each user of the system and improve the understanging of their profiles, to identify what they like and dislike, or are expecting from the system. The RS must seek to enable individual mechanisms that users can work together, because some users like to contribute to the system with their ratings and express their opinions and beliefs or can be happy to help the others by contributing with information. However, it should be cautious as there are malicious users who seek to influence others in the system just to promote or penalize certain items. A detailed overview of these properties is presented in the different chapters of the collective book edited by Ricci et al [19].

Social Search Systems

Frameworks that specifically target recommendation services based on user profiles are mostly in the category of people recommendation and question answering systems. Such systems explore either the topology of the network or the content of the exchanges between communities and peers. The main difference to content-based social search is the fact that the result of a recommendation is not a document, but another user or group of users. In this way, the person can interact directly with the recommended user, which provides a more secure and trusted environment for the communication process. Also, such people-to-people interactions are more interesting for the service provider, as they can contribute to the growth of the social platform, which is generally measured by the number of users and connections between them.

Guy et al [6] present a people recommendation strategy specially adapted for the enterprise ecosystem. The recommendation engine uses information from an organization Intranet for computing similarity scores between employees. Such information include: (i) paper or patent co-authorship, (ii) commenting of each others' blogs or profiles, (iii) mutual connection in other social networks, internal to the organization. Based on an aggregated score computed for each relationship, people are recommended to be added in an employee internal messenger system. For each recommendation, an explanation is generated, considered an important component of such systems [9]. A limitation of this approach can be considered the fact that the recommendation only uses statistical information to infer the social proximity between users. More concretely, the content of interactions and exchanges is not taken into account to measure the similarity of interests or information needs. We also mention here the fact that most people recommendation strategies in popular social networks, such as *Facebook* or *Orkut*, are also based on this statistical similarity schema.

Lin et al [16] also target the issue of expertise location in the enterprise environment. The proposed system, *SmallBlue* [16], similarly to Guy et al [6], employs data mining and statistical data analysis techniques to extract profile information for employees. More specifically, the system uses company email as a source of information. Keywords are extracted from each email and a bag-of-words based profile is constructed for employees. An innovative feature of the system is the social explanation of people recommendations, by displaying the social path that connects the user to the recommended person on a specific topic.

Hannon et al [7] go beyond the previous approach and build a recommendation strategy using the content of interactions (e.g., status updates) as input. Designed for recommending people to follow in *Twitter*, the *Twittomender* system allows users to expand their network by connecting to people that they do not know directly, but with whom they share similar interests. Each user in the system is represented by a vector, comprised of terms extracted from their shared messages. A kind of *social expansion* of this basic profile is performed, by taking into account messages shared by people connected to the user. This is based on the observation that connected people share close interest. The computation of profile similarities is achieved by the traditional *tf-idf* weighting schema in information retrieval and cosine similarity. The *Twittomender* system is original and different from existing collaborative filtering approaches, as it takes into account the structure of the underlying social network to better approximate the interests of the user. It is however a considerable limitation in the system that no disambiguation or semantic expansion of profile terms are considered. More concretely, the user profile is composed of keywords that might have multiple meanings and this could be a considerable drawback for the relevance of recommendations.

A new generation of social search engines is represented by so-called *Question Answering Systems*. The main difference to the previous approaches is the fact that in this case the system builds a user profile from some kind of user activity (content production or consumption) and uses it to match them with a question formulated by another user.

Aardvark [10] is certainly the most promising social search engine. *Aardvark* introduced several innovations in the field of social search. First of all, it is the first system that models the users based on their generated content. For this reason, users provide topics of interest to the system when they subscribe. Then, a crawler extracts further topics from the user's profiles and status updates in social platforms to expand the initially entered profile items. The extraction of topics from social updates is achieved by linear classifiers, such as Support Vector Machines and probabilistic classifiers. *Aardvark* is not built on top of existing social platforms and lacks a global approach for conceptualizing user profiles.

In another recent social search engine, *CQA* [14], the objective is similar to that of *Aardvark*: route a question to the right person in a community of answerers. In their paper, Li and King [14] introduce two important dimensions for such systems: (i) the consideration of the answerer availability and (ii) the question of the quality of answers. The quality of answers is estimated by taking into account statistical information about the length of the answer, the time the user took to send it and the feedback of other users. In the case of availability, the system monitors the user logins and performs a prediction of whether the user will be available at a specific time and date in the future.

We can finally conclude that in current social search systems that offer a people recommendation service, the issue of recommendation explanation is still not well tackled (which is also strongly related to privacy management). Also, few frameworks benefit from semantic web technologies on a data storage or data enrichment level.

Another possibility to build an RS is to leverage the content shared by users in the social network. More specifically, we consider the content productions of users in order to better understand their interests and information needs and more concretely, build expertise profiles. In such way, the recommender engine is able to recommend people that have similar or complementary interests. From a conceptual viewpoint, such a recommender engine is composed of two parts: (i) the identification of semantic data (e.g., entities extracted from status updates) that will compose the profile and (ii) the scoring of said semantic data (measuring the user expertise).

We consider X the domain of all n users involved in the social platform. T_x represents the set of items correlated with user x, i.e., $T_x = t|Weight(t,x) > 0$. Therefore, user x and item t are correlated when $Weight(t,x) > 0$, $Weight$ being the weight of the item in the profile.

An item in the user profile can be represented by a keyword or a concept. The main difference is that concepts have URIs, that provide them the exact semantic meaning. Generally, such URIs can be retrieved from so called semantic knowledge graphs, such as *DBPedia*. Each profile item is an entity (keyword, named entity) extracted from at least one content production of the user and connected to at least one semantic concept present in at least one semantic knowledge base. The main arguments for this choice is that this kind of representation is richer and less ambiguous than a keyword-based or item-based model. It provides an adequate grounding for the representation of coarse to fine-grained user interests. A semantic knowledge base provides further formal, computer-processable meaning on the concepts (who is coaching a team, an actor filmography, financial data on a stock), and makes it available for the system to take advantage of knowledge base-originated semantic concepts that are more precise, and reduce the effect of the ambiguity caused by simple keyword terms.

Normally in a conversation, we depend essentially on the context of the conversation to disambiguate a word. Similarly, in order to associate keywords or entities in a social update to the right concept in Linked Data, contextual cues are necessary to allow restricting the semantic field of the social update. In traditional documents, generally there are sufficient contextual cues to overcome such ambiguous situations, where the meaning of a term is not straightforward.

In the case of social platforms, the short nature of posts requires to find these cues elsewhere, so we may consider two main additional sources of contextual cues:

- The first contextual cue is user-related, which consists in building incrementally a *vocabulary* from all social updates of the user. The assumption behind this first additional context is that there is a probability that the user previously shared some content in a related semantic field (e.g., a user who posted about "Apple" might have shared before about other Apple products, such as the "iPhone").
- The second additional contextual cue is community-related. On social platforms users are members of different communities, which influence each other in terms of interests. Users participate in a group or a community because they are interested in what community members say and as a consequence of this participation, users have intention of using commonly known keywords to make his/her contents easily understandable by the community. This second contextual cue is used only if the user-related one is not yet available or not sufficiently rich (e.g., user has shared few messages, but has lots of friend connections). More specifically, it is a solution for the so-called cold-start situation and consists of aggregating the most recent messages of friends connected to the user and constructing a vocabulary from the content of these messages.

After the construction of the vector containing also such items that represent the context of the keyword, several similarity measures can be used to compare it with the description of candidate concepts in the knowledge base, and the best matching concept selected. A further, optional step is to leverage the semantic neighborhood of the concept to better describe the user expertise (e.g., include more general concept into the profile). This could be interesting in case of profile extracted from status updates, as such messages are short and therefore we have little available information about the user information needs or interests.

Future Directions/Open Questions

As seen in the previous section, over the last two decades, some major advances have been achieved in the area of RSs using techniques of social network analysis and mining.

In this section we present some current challenges and open questions, that we think, will be a major preoccupation for scientific communities, but also the industry in the upcoming years. We will consider two practical future directions and list the corresponding open challenges that need to be considered.

Recommender Systems in the Enterprise

Nowadays, more and more companies show increasing interest towards the integration of RSs in the Intranet in order to further improve communications and internal knowledge management. Several reasons push companies to invest in such infrastructures:

- it can improve social interactions between employees (e.g., a people recommender in the enterprise may help in finding the best expert for a specific problem [11], which may reduce costs and increase efficiency);
- it can provide new means for the dynamic composition of teams for a specific project, as the expertise of employees can be easily retrieved. Also, internal documents, videos can be recommended for a project or learning;
- such a system may provide specific tools for employees in order to keep motivation and a good atmosphere in the company, e.g. associating specific tags to colleagues, such as expertise tags and specific badges, when being an active contributor in providing help to colleagues or other scenarios;
- with such a system, an implicit internal social network can be built, that links employees with similar interests and activities. This can help the company in improving its organization and also optimize human resources management (changing dynamically teams, etc.).

The deployment of an RS in a company faces several challenges and its design depends on several criteria, such as the type of activity the company performs or the degree of sensibility of the information they share. A first challenge, but also, the most important, is what kind of internal content to use as input for the RS. Company e-mails are a rich source for learning more about each employee expertise and interests, but there may privacy and security concerns. Another, more acceptable source for such an RS may be represented by content employees share on internal or web-based social networks, such as *Twitter* [21] or Yammer. Such content is shorter and generally does not contain confidential information. Furthermore, the content of web-pages employees read may also represent and additional source for such systems for the construction of the expertise profiles [11].

Amongst the challenges for building such a system, the most important are technical and related to human-computer interaction (HCI). More concretely, technical challenges include the implementation of content extraction tools from internal mail servers, the microblogging platforms and web browsers. All the extracted content must be aggregated and stored in a secure database. Challenges related to HCI include the design of user interfaces that allow users to control what content to share with the system (e.g., there may be e-mails for private usage).

An important issue when designing RSs is to generate an explanation for each recommendation. Such explanations could be useful as they increase trust. They can be of several types: (i) the explanation of the social path between the two users, i.e. by showing part of the social graph and the paths in the employee social network that connect them or (ii) a semantic explanation, that includes areas of expertise of the recommended employee. According to the social distance, such areas of expertise may be shown with different levels of granularity, by using hierarchical paths of concepts in semantic knowledge bases, such as *DBPedia* (e.g., expert in *Twitter* is more specific than expert in microblogging platforms).

In a nutshell, the following questions should be considered for building a successful RS specifically targeted to an enterprise:

- How to extract the named entities from short, unstructured messages, status updates? In other words, how to transform each social interaction that occurs in the company or that employees share into useful knowledge for the RS;
- How to combine structural and semantic analysis for recommendation ranking;
- What are the next generation privacy protection mechanisms that would allow an easy adoption of such a system in a company;
- How to generate useful and meaningful explanations for a recommendation;
- How to make good recommendations without violating privacy concerns;

The use of such RSs in an enterprise may be useful also for generating a profile for the entire company, e.g., by aggregating all individual user profiles. Such a profile may be useful for the next generation enterprise social networks, where each node in the network is a company. Such a network could facilitate collaboration between companies, e.g., by finding the best company for a collaborative European project.

Recommendation in Mobile Social Networks: a Multi-Agent Approach

A second scenario for RSs concerns mobility and ubiquity, as more and more users have smartphones, capable of sensing context. The most widely used context in such a scenario is the user location, which may significantly improve recommendation (other context data may include available networks (Wifi, Bluetooth) or other physical data). By integrating location in an RS, a preliminary filtering of items can be performed, by selecting only a subset that is in a well-defined perimeter. Such items may include other users with similar interests (e.g., looking for people who like similar artists in a given location), as well as restaurants, cinema or other services the city provides. The deployment of such an RS faces several challenges, depending on its design. A first important design principle which needs to be fixedearly is whether the system is centralized or decentralized. Clearly, a centralized system would face important performance and scalability issues. A decentralized system is more interesting, as a local server can be associated to each location in the city, which could support this recommendation service.

A further step towards decentralization can be considered, by integrating multi-agent principles to the RS, i.e., to design and implement a customizable approach where different autonomous decision-making entities (agents) have to communicate, exchange knowledge and cooperate in order to achieve individual and/or collective objectives. It allows the creation of different communities, with different possible functions and modes of exchanges. Such an approach aims to meet several challenges, such as decentralization of the community management, personalized automatic management and discovery of communities, and flexibility so that any agent can create its own community. In addition, it should cover all levels of abstractions (agent, environment and organization) that are required for the development of sophisticated multi-agent system. In this design, each smartphone is equipped with an agent, capable of exchanging knowledge with other agents, using the local server associated to a given location in the city.

Using a multi-agent approach for an RS in mobility, agents can act as a personal assistant on the behalf of each user, present in a given location. The agent perceives knowledge from the communities of individual interests and acts upon the communities to meet their goals. Thus, agents can bring the appropriate people having common goals or interests together share their knowledge with each other at ease.

Other scientific challenges for such advanced RSs include the traditional cold-start problem, i.e., how to provide recommendations to users with little information about their profile, or how to recommend items with few ratings. Also, an important general challenge is how to make recommendation users trust, i.e., how to provide users an easy way for giving feedback on recommendations. With regards to trust, recent works try to integrate the notion of distrust, i.e., how to deal with users or items that cannot be trusted?

Cross-References

Centrality Measures - 00227

Combining Link and Content for Community Detection - 00214

Recommender Systems, Overview - 00116

Recommender Systems: History - 00088

Research Designs for Social Network Analysis - 00246

Semantic Annotation - 00119

Social Media Mining and Knowledge Discovery - 00188

Social Recommender Systems, Techniques in - 00190

References

1. Adomavicius G, Tuzhilin A (2005) Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. IEEE Transactions on Knowledge and Data Engineering 17(6):734–749
2. Anagnostopoulos A, Kumar R, Mahdian M (2008) Influence and correlation in social networks. In: Li Y, Liu B, Sarawagi S (eds) Proceedings of the 14th ACM SIGKDD International Conference

on Knowledge Discovery and Data Mining, Las Vegas, Nevada, USA, August 24-27, 2008, ACM, pp 7–15

3. Combe D, Largeron C, Egyed-Zsigmond E, Gery M (2012) Getting clusters from structure data and attribute data. In: Proceedings of the International Conference on Advances in Social Networks Analysis and Mining, ASONAM 2012, Istanbul, Turkey, 26-29 August, 2012, IEEE Computer Society, to appear
4. Domingos P (2005) Mining social networks for viral marketing. IEEE Intelligent Systems 20(1):80–82
5. Getoor L (2010) Link mining and link discovery. In: Encyclopedia of Machine Learning, pp 606–609
6. Guy I, Ronen I, Wilcox E (2009) Do you know?: recommending people to invite into your social network. In: Conati C, Bauer M, Oliver N, Weld DS (eds) Proceedings of the 2009 International Conference on Intelligent User Interfaces (IUI), February 8-11, 2009, Sanibel Island, Florida, USA, ACM, pp 77–86
7. Hannon J, Bennett M, Smyth B (2010) Recommending twitter users to follow using content and collaborative filtering approaches. In: Amatriain X, Torrens M, Resnick P, Zanker M (eds) Proceedings of the 2010 ACM Conference on Recommender Systems, RecSys 2010, Barcelona, Spain, September 26-30, 2010, ACM, pp 199–206
8. Hasan MA, Zaki MJ (2011) A survey of link prediction in social networks. In: Aggarwal CC (ed) Social Network Data Analytics, Springer, pp 243–275
9. Herlocker JL, Konstan JA, Riedl J (2000) Explaining collaborative filtering recommendations. In: CSCW '00: Proceedings of the 2000 ACM Conference on Computer Supported Cooperative Work, ACM, New York, NY, USA, pp 241–250
10. Horowitz D, Kamvar SD (2010) The anatomy of a large-scale social search engine. In: Rappa M, Jones P, Freire J, Chakrabarti S (eds) Proceedings of the 19th International Conference on World Wide Web, WWW 2010, Raleigh, North Carolina, USA, April 26-30, 2010, ACM, pp 431–440
11. Joly A, Maret P, Daigremont J (2010) Contextual recommendation of social updates, a tag-based framework. In: An A, Lingras P, Petty S, Huang R (eds) Active Media Technology, 6th International Conference, AMT 2010, Toronto, Canada, August 28-30, 2010. Proceedings, Springer, Lecture Notes in Computer Science, vol 6335, pp 436–447
12. Kempe D, Kleinberg JM, Tardos É (2003) Maximizing the spread of influence through a social network. In: Getoor L, Senator TE, Domingos P, Faloutsos C (eds) Proceedings of the Ninth ACM

SIGKDD International Conference on Knowledge Discovery and Data Mining, Washington, DC, USA, August 24 - 27, 2003, ACM, pp 137–146

13. Lancichinetti A, Fortunato S (2009) Community detection algorithms: a comparative analysis. Physical review E 80(5):056,117
14. Li B, King I (2010) Routing questions to appropriate answerers in community question answering services. In: Huang J, Koudas N, Jones GJF, Wu X, Collins-Thompson K, An A (eds) Proceedings of the 19th ACM Conference on Information and Knowledge Management, CIKM 2010, Toronto, Ontario, Canada, October 26-30, 2010, ACM, pp 1585–1588
15. Liben-Nowell D, Kleinberg J (2007) The link-prediction problem for social networks. Journal of the American Society for Information Science and Technology (JASIST) 58(7):1019–1031
16. Lin CY, Cao N, Liu S, Papadimitriou S, Sun J, Yan X (2009) Smallblue: Social network analysis for expertise search and collective intelligence. In: Ioannidis YE, Lee DL, Ng RT (eds) Proceedings of the 25th International Conference on Data Engineering, ICDE 2009, March 29 2009 - April 2 2009, Shanghai, China, IEEE, pp 1483–1486
17. Melville P, Sindhwani V (2010) Recommender systems. In: Sammut C, Webb GI (eds) Encyclopedia of Machine Learning, Springer, pp 829–838
18. Namata G, Getoor L (2010) Link prediction. In: Encyclopedia of Machine Learning, pp 609–612
19. Ricci F, Rokach L, Shapira B, Kantor PB (eds) (2011) Recommender Systems Handbook. Springer
20. Schaeffer SE (2007) Graph clustering. Computer Science Review 1(1):27–64
21. Stan J, Do VH, Maret P (2011) Semantic user interaction profiles for better people recommendation. In: International Conference on Advances in Social Networks Analysis and Mining, ASONAM 2011, Kaohsiung, Taiwan, 25-27 July 2011, IEEE Computer Society, pp 434–437
22. Wasserman S, Faust K (1994) Social Network Analysis: Methods and Applications, 4th edn. No. 8 in Structural analysis in the social sciences, Cambridge University Press, New York, USA