Future Generation Computer Systems 78 (2018) 413–418

Contents lists available at ScienceDirect

Future Generation Computer Systems

journal homepage: www.elsevier.com/locate/fgcs

Editorial

Recommender Systems for Large-Scale Social Networks: A review of challenges and solutions

Magdalini Eirinaki [a], Jerry Gao [a], Iraklis Varlamis [b,*], Konstantinos Tserpes [b]

[a] *Computer Engineering Department, San Jose State University, San Jose, CA, USA*
[b] *Department of Informatics and Telematics, Harokopio University of Athens, Athens, Greece*

ARTICLE INFO

Keywords:
Recommender systems
Social networks
Scalability
Big data

ABSTRACT

Social networks have become very important for networking, communications, and content sharing. Social networking applications generate a huge amount of data on a daily basis and social networks constitute a growing field of research, because of the heterogeneity of data and structures formed in them, and their size and dynamics. When this wealth of data is leveraged by recommender systems, the resulting coupling can help address interesting problems related to social engagement, member recruitment, and friend recommendations.

In this work we review the various facets of large-scale social recommender systems, summarizing the challenges and interesting problems and discussing some of the solutions.

1. Introduction

The fields of computational intelligence and knowledge management have made significant advances over the past decades. The potential ability to create intelligence from the analysis of raw data has been successfully applied to diverse areas such as business, industry, sciences, social media etc. The coupling of online social networks with recommender systems created new opportunities for businesses that consider the social influence important for their product marketing, as well as the social networks that want to improve the user experience by personalizing the content that is provided to each user and enabling new connections. At the same time, these changes have created new challenges for researchers in the area of recommender systems and social network analysis. The large volume of social network interactions that expand the size of the social graph with increased velocity, the variety of information provided in the form of written reviews, ratings or permanent and volatile relations, and the veracity of data expressed in the form of trust or distrust between users who become product reviewers or opinion influencers, are only some of the factors that make social networks and the associated recommender systems an ideal case of big-data research.

In this work we focus on large-scale recommender systems that take advantage of the characteristics of the underlying social network and focus on the variety and volatility of social bonds, tackle the problems of size and speed of change of social graphs, test the scalability of traditional recommender systems and/or present solutions that can take recommender systems to the next level.

2. Recommender systems and social networks

The most popular techniques used in recommender systems are Content-based (CB) filtering and Collaborative Filtering (CF). Each user in such systems is typically represented by a user profile, that includes the "items" this user has rated (or purchased). In content-based filtering, new items are recommended based on their similarity to items already present in the user's profile. To achieve this, more details on each item are needed. On the other hand, collaborative filtering approaches are "agnostic" to the items. Instead, they employ the ratings assigned by the user, to find other similar users (or items) based on their rating patterns. The generic nature of collaborative filtering systems was the reason of their broad success, as they are used to recommend a wide range of products such as movies, music, news, books, research articles, search queries, social tags, etc.

While a bipartite user-item graph containing the ratings used to be the sole input of such systems, such a structure is no longer sufficient to represent all available information, such as content, context, social information, and metadata. As a result, many attempts have been made to incorporate more information about the user, his/her context (spatial, temporal, social etc) and its evolution over time.

* Corresponding editor.
E-mail addresses: magdalini.eirinaki@sjsu.edu (M. Eirinaki), jerry.gao@sjsu.edu (J. Gao), varlamis@hua.gr (I. Varlamis), tserpes@hua.gr (K. Tserpes).
URLs: http://www.sjsu.edu/people/magdalini.eirinaki/ (M. Eirinaki), http://www.sjsu.edu/people/jerry.gao/ (J. Gao), https://www.dit.hua.gr/~varlamis/ (I. Varlamis), https://www.dit.hua.gr/~tserpes/ (K. Tserpes).

http://dx.doi.org/10.1016/j.future.2017.09.015
0167-739X/

2.1. Beyond user-item ratings: Context-aware recommender systems

Context-aware recommender systems (CARS) generate more relevant recommendations by adapting them to the specific context of the user. The contextual factors that must be considered by a recommender system relate to the time, location, and purpose of the targeted user. According to [1], the user context can be static or change over time. CARS assume a pre-filtering step, where context information is used to select the set of relevant items, a classic recommendation step that ranks relevant items according to the predicted ratings and a contextual post-filtering step, that re-ranks and filters the output of the traditional recommender [2].

Context-aware systems combine information from multiple sources within the social network in order to refine the context space and solve major recommender system issues such as "scalability" and the "cold-start" problem [3]. For example, in [4] contextual information is encoded in or reflected by the user-specific and item-specific latent factors. The user-item rating matrix is split into partitions, by grouping users and items with similar contexts and matrix factorization is applied to the generated sub-matrices.

A context-aware multimedia recommender system, which considers in tandem user preferences (in previous items' metadata), opinions (textual comments in user reviews), behavior (past items observations and actions) and feedbacks (expressed in the form of ratings) within the same framework is presented in [5].

2.1.1. Time-aware recommender systems

Time-aware recommender systems (TARS) can be considered as specialized CARS focusing on exploiting contextual information in the form of time. They assume that user preferences are drifting over time and user taste is evolving as new items become available and new trends appear. This in turns affects item popularity, which is constantly changing, bringing old items to the long tail of user preferences and moving new items to the head [6].

Handling the temporal dynamics of user preferences in recommender systems raises new challenges since the change on each individual user interests is different from the concept drift problem [7]. In a social network with multiple users and items, many different features are changing simultaneously, and influence each other, whereas in the general concept drift problem, only a single concept is tracked. Using sliding windows and preference decay functions increases the sparsity of an already sparse problem (since past information is discarded or lost) and is usually avoided. In [8] Koren extends the static matrix factorization model and the associated baseline predictors with functions that capture the gradual drift of user and item bias and introduce the timeSVD++ algorithm, which outperforms its predecessors.

Several techniques have been used in the literature, to adopt CF algorithms to temporal changes, by boosting recent ratings and penalizing older ratings, such as discrete time windows [9] or continuous decay function [10]. Recently, an algorithm based on association rule and community identification approach has been proposed to handle the drift problem in recommender systems [11]. Bayesian Probabilistic Tensor Factorization [12] has also been proven an appropriate temporal CF model.

2.1.2. Location-aware recommender systems

Location-aware recommender systems (LARS) have become popular mainly through place recommendation systems in the travel and tourism industry, as discussed by Chen and Tsai [13]. The advent of location-based social networks (LBSNs), such as Facebook Places and Foursquare, increased the available data and challenges for researchers [14]. LARS exploit location ratings when partitioning user-location bi-partite rating graphs with spatial criteria, so that locations that are spatially close to the users are employed and those in a distance are ignored, in a manner that maximizes system scalability while not sacrificing recommendation quality [15]. LARS consists of two components, an offline modeling component, which learns the interest of each individual user and the local preference of each individual location, by capturing item co-occurrence patterns and exploiting item contents and an online recommender component, which automatically combines the learned user's local preferences and produces the top-k recommendations [16,17].

Time is a crucial factor in Location-aware recommender systems, so recent approaches in location or activity recommendations employ unified spatio-temporal frameworks [18–20].

2.1.3. Community-aware or Social recommender systems

Community-aware or "Social recommender systems", have gained the attention of researchers since they leverage social relationships in order to improve the recommendation process. Authors in [21] give a narrow definition of social recommendation as "*any recommendation with online social relations as an additional input, i.e., augmenting an existing recommendation engine with additional social signals*". A broader definition, by [22] refers to recommender systems targeting social media domains.

The main premise in this line of research is that users' preferences are influenced more by the preferences of their friends, than these of unknown users. Such attempts enhance the typical recommendation process with social data, assuming that item ratings are available and that some form of influence/trust propagation exists within the user network. For example, a common approach is to enhance the memory-based collaborative filtering process by forming the user's neighborhood using similarities derived from the users' ratings and/or their social relationships, focusing on trust.

Community-aware systems employ user preferences, user connectedness, or any other social information, in order to detect user clusters and consequently partition the recommendation problem into smaller problems. Based on the concept of *homophily* in social networks, that a user's preferences are likely to be similar to, or influenced by these of her friends, such systems manage to fill the gap in cold-start problem and find similarities between users [23]. This can be done through co-factorization, where the assumption is that the users share the same preference vector in both the rating and the social spaces (e.g. [24]), ensemble methods, where the resulting recommendation is derived by the linear combination of two systems (e.g. [25]), or regularization, where priority is given to the social-based ratings (e.g. [26]). For example, in [27] authors propose a preference-aware community detection method to group users based on their social relations, whereas in [28], the users' social information (user-to-user friendship network) is used to partition the large user-to-item bipartite graph into smaller partitions and perform collaborative filtering in a narrower social context. When users belong to more than one community (i.e. we have overlapping communities), multi-label propagation based methods are used on the user-user graphs [29].

2.2. Beyond simple item recommendations

Recommender systems (RSs) have become popular since they can personalize the user experience by providing automated recommendations. They first appeared in e-commerce sites [30], used to recommend individual items, products of potential interest for customers to purchase. However, their usage has span multiple other domains in the past few years, from digital collections (e.g. news and research articles [31], to database queries [32,33] or even web services [34,35]).

2.2.1. Group and package recommendations

Package Recommender Systems extend the classical RSs by proposing to their users sets of items (packages) instead of single items. Package recommendation is extremely useful in a number of application domains (e.g. recommendation of packages of academic courses to students, packages of meals for a weekly diet, travel packages and sets of movies or books). In [36], the authors propose a system for recommending a team of experts, who have a set of predefined skills and a minimum communication cost. In [37] a course recommendation system, based on the maximum flow algorithm, is used to personalize Stanford university curricula. In [38], authors create trip packages that maximize user preference score and satisfy given user constraints. Finally, in [39], the authors propose a generic framework for package recommendations that meets users' preferences whilst satisfying several budget and time constraints.

In order to target the cold-start problem that mainly affects the performance of recommender systems that target new individual users, group recommender systems [40,41] have been proposed for recommending items that can be immediately experienced by a group of users on the same place (physical or virtual) [42]. Group recommendations have also been applied to construct some "stereotypes", which can be applied to a single-user recommendations. Group recommender systems raise a number of challenging issues [43], such as the need of members to examine each others preferences, and to negotiate and reach a final decision on the recommended items [44].

2.2.2. User recommendations/Link prediction

The problem of recommending to the users of a social network other users (i.e. new connections) has been extensively studied [45,46]. When explicit user-to-user trust information is available, trust propagation models can be employed to infer which new interactions among the social network members are likely to occur in the near future, and recommend new connections to the users [47–50]. In the absence of explicit user-to-user ratings, latent factor models that leverage implicit users feedback have been employed [51].

From an algorithmic point of view, link prediction algorithms perform graph traversals in different directions (path-based or random walk-based algorithms) in order to compute the final ranking of vertices or edges and consequently recommend links. In order to handle large-scale networks, proposed methods [52,53] perform a partitioning of the initial adjacency matrix, and then compute a low-rank approximation of each partition (i.e., diagonal block), which is then applied to link prediction. In their implementations, they process large-scale graphs using frameworks such as MapReduce [54], Fork-Join [55], Spark [56], Pregel [57] or GraphLab [58].

2.3. Beyond simple evaluation metrics

Developing an effective recommender system is not only limited to predicting user preferences and recommending the most prominent items to each user [31,59]. Especially in e-commerce systems, where new items are added in a rapid pace, it is important for the recommender systems to move from popular to newly introduced items with similar features, using only the few ratings available. This challenge is defined as the "long tail" problem in the literature [60] and can be leveraged using content-based methods to define the similarity between popular and newer items and replace popular items with new ones, or cluster the items in the long tail and leverage the sparse ratings using aggregated ratings per cluster [61]. In general, the same approaches that solve the cold-start problem are used to tackle the long tail problem.

With the above restrictions in mind, the "diversity" and "novelty" of recommendations have bacome key features for efficient RSs, from a business perspective [62]. In order to increase novelty, the authors in [63] invert the recommendation task and instead of selecting items to recommend to a user, their system decides the users each item should be recommended to. For this purpose, they invert the rating matrix introducing interesting reformulations of nearest-neighbor algorithms, which in turn introduce a new neighbor selection policy.

3. Major challenges for Large-scale Recommender Systems

The main challenges for large-scale recommender systems that employ the social network structure to enhance the recommendation process, are: (a) to take advantage of all the available information in order to analyze the social, rating and content similarity graphs that are formed, (b) to adapt to dynamically evolving graphs, and (c) to scale to large graphs [28].

3.1. Data variety

Two of the major problems in recommender systems are data sparsity and the poor performance of cold-start recommendations to new users. Social networks and the multitude of personal and preference information they keep for their users, helped in alleviating the cold-start problem and also filled many gaps in the user-item preference data, through implicit user preferences extraction [64].

However, the use of all the information available in social networks for users, made state-of-the-art collaborative filtering algorithms (e.g. matrix factorization [65]) insufficient to handle the volume and complexity of the new information. Consequently, modifications and extensions to the popular models have been suggested in order to incorporate latent preference or profile information to the existing models (e.g. in matrix factorization). Social regularization [66] and social-based matrix factorization [67,68] are the key terms used to describe the approaches in this direction.

3.2. Data volatility

From the early works on recommender systems that capture the user interest drifts [69,70], to more recent works that model the dynamics of user interest in activity streams [71], the volatility of user preferences is a parameter to consider when designing a recommender system. The flourish of social recommender systems that produce data in a streaming, transactional format and the evidence that capturing the temporal dynamics of user preferences improves the recommendation performance, makes data volatility a major challenge for modern recommender systems [72].

Authors in [73] propose a semi-supervised framework for stream-based recommendations, which extends a matrix factorization algorithm by the ability to add new dimensions to the matrix at runtime and perform semi-supervised learning. Similarly, authors in [74] propose a novel Collaborative Evolution model, based probabilistic factorization of the user-item rating matrix. The factor matrices for users and items evolve over time, in order to capture the evolution of users profiles through the sparse historical data and be able to output the prospective user profile of the future.

3.3. Data volume

In order to address the continuously growing amounts of social network data, researchers have focused on parallel and distributed systems, and work on transforming existing recommendation algorithms to the parallel environment. And since collaborative filtering (CF) is probably the most popular model-based technique for recommender systems, several implementations of CF algorithms have been developed on parallel and distributed frameworks.

Among the many frameworks that have been developed to facilitate the processing of high-volume data, the ones that have been frequently used for CF are OpenMP, Pthreads, and Java Threads for shared memory parallel programming and CUDA and OpenCL for GPU computing. Among the distributed solutions, Hadoop [75] that is based on the "key-value pairs" model of MapReduce [76], has been the basis for more recent distributed frameworks, such as Mahout [77] and Spark [78]. Storm [79] and GraphLab [58] are competitive frameworks for graph-based implementations and algorithms based on the master-worker and the bulk synchronous parallel (BSP) model respectively.

The work presented in [80] provides an interesting overview of the challenges faced by social networking data analysis tools that handle huge volumes of data and the work presented in [81] gives a comprehensive survey of parallel and distributed implementations of collaborative filtering algorithms.

Authors in [5] develop a context-aware recommender system for large-scale social networks, that considers all the possible user-to-content relationships that can occur in a social network. The system has been implemented on Apache Spark using the Hadoop technological stack, AllegroGraph triplestore and SPARK SQL facilities.

4. Recommender systems for large-scale social networks a brief review of accepted articles of this special issue

From the early work of Sarwar et al. [82] on scalable collaborative filtering algorithms until today, the growth of social networks and the abundance of user preference data has created the need for recommender systems that can handle in tandem the variety, volatility and volume of data generated by large-scale social networks.

This special issue contains research works that go beyond the traditional user-item rating information and take advantage of information extracted from social networks. The proposed solutions are examined from various big-data perspectives related to the variety and volume of social data and the volatility of user preferences over time.

In [83], Rezaeimehr et al. propose a time-aware recommender system, called TCARS. The TCARS recommendation algorithm, is based on identifying overlapping communities of users and modeling the dynamics of users preferences in order to minimize sparsity effects and keep communities updated over time. The results obtained using TCARS are comparable to that of state-of-the-art methods.

In [53], Corbellini et al. propose a novel graph processing model, called Distributed Partitioned Merge (DPM). DPM is a hybrid model for processing large social networks, which combines the simplicity of the Fork-Join programming style and the performance and scalability of the Pregel framework. DPM has been evaluated for its ability to quickly compute path-based and random walk-based ranking of large networks' vertices and thus its ability to serve as a link prediction component in a user recommender system in a social network.

In [84], Guo et al. address the volatility of user interests in an e-commerce environment, where user purchase history is the input for a sequential pattern mining algorithm that models the drift of users interest for different categories. The proposed model diversifies recommendations and improves their accuracy by incorporating both multi-category purchase interval and price preference information. The latter is modeled using fuzzy set theory and the proposed recommender system is evaluated with real purchase records showing improved performance against competitive methods.

In [33], Margaris et al. present a social network-based query personalization algorithm, which considers the influence from the users social network, when searching for personalizing information. The algorithm extends the state of the art query personalization algorithms by (a) exploiting social graphs for improving personalization quality and (b) proposing an efficient query rewriting technique, which handles the mapping of the personalization procedure results. Authors evaluate their approach in a typical movie information database, where past user preferences and influence from other network users are employed to filter the long list of results that match a user query. For example, when a user searches for movies of a specific period in which an actor stared in, the ranking of the movies depends query matching the user preferences but also user's friends preferences. The proposed algorithm has been experimentally validated regarding its performance, and the accuracy of result ordering with encouraging results.

In [85], Gan et al. analyze correlations between triangular motifs and social relations, and conclude that if a person has a large number of social relations, the connections among his friends will be large as well. They find that potential social relations are related with common connections and demonstrate the necessity of fusing global and local associations. The authors develop a novel method, called FLOWER (Fusing GLobal and LOcal Associations toWards PErsonalized Social Recommendation) to integrate global and local associations and show that methods based on FLOWER significantly outperform others in social recommendation in various types of social network settings.

In [68], Gurini et al. propose a novel people-to-people recommender system for social networks, which is based on the identification of users attitudes in terms of: sentiment, volume and objectivity. In order to do this at large-scale on traditional social networks, the system employs a three-dimensional matrix factorization model (one dimension for each attitude) and includes the temporal alterations of users attitudes into consideration in the factorization model. The evaluation shows that the recommenders accuracy and diversity increases with attitudes and temporal features.

5. Conclusions—future directions

In order to address the sparsity of rating information and the cold-start problem plaguing the traditional Collaborative Filtering algorithms, several deep learning approaches have been introduced recently in the literature, which combine item ratings with topic information about items, that comes from other sources. For example the hierarchical Bayesian model in [86], which jointly performs deep representation learning for the content information and collaborative filtering for the ratings matrix, or the deep network of stacked denoising autoencoder in [87], where each layer is trained to minimize the error in reconstructing its input (which is the output of the previous layer).

Deep learning also opened the road for new types of recommendations, such as session-based recommendations using recurrent neural networks to model click-session data [88], the cross-domain recommender systems where items are mapped to a joint latent space [89], or the social trust ensemble learning model of [90].

A large number of research works for collaborative filtering that go beyond matrix completion [91] is based on Nonnegative matrix factorization (NMF) [92] and tensor decomposition [93,94] of the ratings matrix. Weighted and graph regularization NMF have also been proposed to incorporate information from the social graph to the ratings model, and in order to reduce the memory requirements for large graphs and avoid over-fitting, the Laplacian matrix of the graph regularization has been replaced by other regularizations (e.g. Tikhonov [95]).

Finally, as the list of possible recommendations grows, the importance of algorithms that make valid but also novel recommendations rises [96]. Apart from accuracy, the diversity, serendipity, novelty and freshness of recommended items [97] and user familiarity [98] with them, and avoiding user boredom [71], are a few of the new criteria for evaluating the recommendations' quality.

References

[1] G. Adomavicius, B. Mobasher, F. Ricci, A. Tuzhilin, Context-aware recommender systems, AI Mag. 32 (3).

[2] G. Adomavicius, A. Tuzhilin, Context-aware recommender systems, in: F. Ricci, L. Rokach, B. Shapira (Eds.), Recommender Systems HandBook, Springer, 2015, pp. 191–226.

[3] L. Song, C. Tekin, M. van der Schaar, Online learning in large-scale contextual recommender systems, IEEE Trans. Serv. Comput. 9 (3) (2016) 433–445.

[4] E. Zhong, W. Fan, Q. Yang, Contextual collaborative filtering via hierarchical matrix factorization, in: Proceedings of the 2012 SIAM International Conference on Data Mining, SIAM, 2012, pp. 744–755.

[5] F. Amato, V. Moscato, A. Picariello, F. Piccialli, Sos: A multimedia recommender system for online social networks, Future Gener. Comput. Syst.

[6] P.G. Campos, F. Díez, I. Cantador, Time-aware recommender systems: a comprehensive survey and analysis of existing evaluation protocols, User Model. User-Adapt. Interact. 24 (1–2) (2014) 67–119.

[7] A. Tsymbal, The Problem of Concept Drift: Definitions and Related Work, Vol. 106, no. 2, Computer Science Department, Trinity College Dublin.

[8] Y. Koren, Collaborative filtering with temporal dynamics, Commun. ACM 53 (4) (2010) 89–97.

[9] O. Nasraoui, J. Cerwinske, C. Rojas, F. Gonzalez, Performance of recommendation systems in dynamic streaming environments, in: Proceedings of the 2007 SIAM International Conference on Data Mining, SIAM, 2007, pp. 569–574.

[10] Y. Ding, X. Li, Time weight collaborative filtering, in: Proceedings of the 14th ACM International Conference on Information and Knowledge Management, ACM, 2005, pp. 485–492.

[11] H. Feng, J. Tian, H.J. Wang, M. Li, Personalized recommendations based on time-weighted overlapping community detection, Inf. Manage. 52 (7) (2015) 789–800.

[12] L. Xiong, X. Chen, T.-K. Huang, J. Schneider, J.G. Carbonell, Temporal collaborative filtering with bayesian probabilistic tensor factorization, in: Proceedings of the 2010 SIAM International Conference on Data Mining, SIAM, 2010, pp. 211–222.

[13] C.-C. Chen, J.-L. Tsai, Determinants of behavioral intention to use the personalized location-based mobile tourism application: An empirical study by integrating tam with issm, Future Gener. Comput. Syst.

[14] N.S. Savage, M. Baranski, N.E. Chavez, T. Höllerer, Im feeling loco: A location based context aware recommendation system, in: Advances in Location-Based Services, Springer, 2012, pp. 37–54.

[15] J.J. Levandoski, M. Sarwat, A. Eldawy, M.F. Mokbel, Lars: A location-aware recommender system, in: Data Engineering (ICDE), 2012 IEEE 28th International Conference on, IEEE, 2012, pp. 450–461.

[16] J. Bao, Y. Zheng, M.F. Mokbel, Location-based and preference-aware recommendation using sparse geo-social networking data, in: Proceedings of the 20th International Conference on Advances in Geographic Information Systems, ACM, 2012, pp. 199–208.

[17] H. Yin, Y. Sun, B. Cui, Z. Hu, L. Chen, Lcars: a location-content-aware recommender system, in: Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, 2013, pp. 221–229.

[18] Q. Yuan, G. Cong, Z. Ma, A. Sun, N.M. Thalmann, Time-aware point-of-interest recommendation, in: Proceedings of the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval, ACM, 2013, pp. 363–372.

[19] H. Gao, J. Tang, X. Hu, H. Liu, Content-aware point of interest recommendation on location-based social networks, in: AAAI, 2015, pp. 1721–1727.

[20] P. Kefalas, P. Symeonidis, Y. Manolopoulos, Recommendations based on a heterogeneous spatio-temporal social network, World Wide Web (2017) 1–27.

[21] J. Tang, X. Hu, H. Liu, Social recommendation: a review, Soc. Netw. Anal. Min. 3 (4) (2013) 1113–1133.

[22] I. Guy, D. Carmel, Social recommender systems, in: Proceedings of the 20th International Conference Companion on World Wide Web, WWW '11, ACM, New York, NY, USA, 2011, pp. 283–284. http://dx.doi.org/10.1145/1963192.1963312.

[23] S. Sahebi, W.W. Cohen, Community-based recommendations: a solution to the cold start problem, in: Workshop on Recommender Systems and the Social Web, RSWEB, 2011.

[24] J. Tang, H. Gao, X. Hu, H. Liu, Exploiting homophily effect for trust prediction, in: Proceedings of the Sixth ACM International Conference on Web Search and Data Mining, ACM, 2013, pp. 53–62.

[25] H. Ma, I. King, M.R. Lyu, Learning to recommend with social trust ensemble, in: Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval, ACM, 2009, pp. 203–210.

[26] M. Jamali, M. Ester, A matrix factorization technique with trust propagation for recommendation in social networks, in: Proceedings of the Fourth ACM Conference on Recommender Systems, ACM, 2010, pp. 135–142.

[27] J.-C. Ying, B.-N. Shi, V.S. Tseng, H.-W. Tsai, K.H. Cheng, S.-C. Lin, Preference-aware community detection for item recommendation, in: Technologies and Applications of Artificial Intelligence, TAAI, 2013 Conference on, IEEE, 2013, pp. 49–54.

[28] C. Sardianos, I. Varlamis, M. Eirinaki, Scaling collaborative filtering to largescale bipartite rating graphs using lenskit and spark, in: Big Data Computing Service and Applications, BigDataService, 2017 IEEE Second International Conference on, IEEE, 2017.

[29] S. Gregory, Finding overlapping communities in networks by label propagation, New J. Phys. 12 (10) (2010) 103018.

[30] J.B. Schafer, J. Konstan, J. Riedl, Recommender systems in e-commerce, in: Proceedings of the 1st ACM Conference on Electronic Commerce, ACM, 1999, pp. 158–166.

[31] J. Beel, B. Gipp, S. Langer, C. Breitinger, Research-paper recommender systems: a literature survey, Int. J. Digit. Libr. 17 (4) (2016) 305–338.

[32] M. Eirinaki, S. Abraham, N. Polyzotis, N. Shaikh, Querie: Collaborative database exploration, IEEE Trans. Knowl. Data Eng. 26 (7) (2014) 1778–1790.

[33] D. Margaris, C. Vassilakis, P. Georgiadis, Query personalization using social network information and collaborative filtering techniques, Future Gener. Comput. Syst. 78 (2018) 440–450.

[34] K. Tserpes, F. Aisopos, D. Kyriazis, T. Varvarigou, A recommender mechanism for service selection in service-oriented environments, Future Gener. Comput. Syst. 28 (8) (2012) 1285–1294.

[35] Z. Zheng, H. Ma, M.R. Lyu, I. King, Wsrec: A collaborative filtering based web service recommender system, in: Web Services, 2009, ICWS 2009 IEEE International Conference on, IEEE, 2009, pp. 437–444.

[36] T. Lappas, K. Liu, E. Terzi, Finding a team of experts in social networks, in: Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, 2009, pp. 467–476.

[37] A. Parameswaran, P. Venetis, H. Garcia-Molina, Recommendation systems with complex constraints: A course recommendation perspective, ACM Trans. Inf. Syst. 29 (4) (2011) 20.

[38] M. Xie, L.V. Lakshmanan, P.T. Wood, Breaking out of the box of recommendations: from items to packages, in: Proceedings of the Fourth ACM Conference on Recommender Systems, ACM, 2010, pp. 151–158.

[39] P. Kouris, I. Varlamis, G. Alexandridis, A package recommendation framework based on collaborative filtering and preference score maximization, in: Engineering Applications of Neural Networks: 18th International Conference, EANN 2017, Athens, Greece, August, 2017, Proceedings, Springer, 2017.

[40] M. Kompan, M. Bielikova, Group recommendations: Survey and perspectives, Comput. Inform. 33 (2) (2014) 446–476.

[41] L. Boratto, Group recommender systems, in: Proceedings of the 10th ACM Conference on Recommender Systems, ACM, 2016, pp. 427–428.

[42] J. Masthoff, Group recommender systems: Combining individual models, in: Recommender Systems HandBook, Springer, 2011, pp. 677–702.

[43] A. Jameson, More than the sum of its members: challenges for group recommender systems, in: Proceedings of the Working Conference on Advanced Visual Interfaces, ACM, 2004, pp. 48–54.

[44] L. Quijano-Sanchez, J.A. Recio-Garcia, B. Diaz-Agudo, G. Jimenez-Diaz, Social factors in group recommender systems, ACM Trans. Intell. Syst. Technol. 4 (1) (2013) 8.

[45] M.Al. Hasan, M.J. Zaki, A survey of link prediction in social networks, in: Social Network Data Analytics, Springer, 2011, pp. 243–275.

[46] V. Martínez, F. Berzal, J.-C. Cubero, A survey of link prediction in complex networks, ACM Comput. Surv. 49 (4) (2016) 69.

[47] M. Eirinaki, M.D. Louta, I. Varlamis, A trust-aware system for personalized user recommendations in social networks, IEEE Trans. Syst. Man Cybern. 44 (4) (2014) 409–421.

[48] Y. Dong, J. Tang, S. Wu, J. Tian, N.V. Chawla, J. Rao, H. Cao, Link prediction and recommendation across heterogeneous social networks, in: Data Mining (ICDM), 2012 IEEE 12th International Conference on, IEEE, 2012, pp. 181–190.

[49] K.-Y. Chiang, N. Natarajan, A. Tewari, I.S. Dhillon, Exploiting longer cycles for link prediction in signed networks, in: Proceedings of the 20th ACM International Conference on Information and Knowledge Management, ACM, 2011, pp. 1157–1162.

[50] P. Massa, B. Bhattacharjee, Using trust in recommender systems: an experimental analysis, in: International Conference on Trust Management, Springer, 2004, pp. 221–235.

[51] H. Chen, X. Cui, H. Jin, Top-k followee recommendation over microblogging systems by exploiting diverse information sources, Future Gener. Comput. Syst. 55 (2016) 534–543.

[52] X. Sui, T.-H. Lee, J.J. Whang, B. Savas, S. Jain, K. Pingali, I. Dhillon, Parallel clustered low-rank approximation of graphs and its application to link prediction, in: International Workshop on Languages and Compilers for Parallel Computing, Springer, 2012, pp. 76–95.

[53] A. Corbellini, D. Godoy, C. Mateos, S. Schiaffino, DPM: A novel distributed large-scale social graph processing framework for link prediction algorithms, Future Gener. Comput. Syst. 78 (2018) 474–480.

[54] J. Dean, S. Ghemawat, Mapreduce: simplified data processing on large clusters, Commun. ACM 51 (1) (2008) 107–113.

[55] C. Mateos, A. Zunino, M. Campo, An approach for non-intrusively adding malleable fork/join parallelism into ordinary javabean compliant applications, Comput. Lang. Syst. Struct. 36 (3) (2010) 288–315.

[56] R.S. Xin, J.E. Gonzalez, M.J. Franklin, I. Stoica, Graphx: A resilient distributed graph system on spark, in: First International Workshop on Graph Data Management Experiences and Systems, ACM, 2013, p. 2.

[57] G. Malewicz, M.H. Austern, A.J. Bik, J.C. Dehnert, I. Horn, N. Leiser, G. Czajkowski, Pregel: a system for large-scale graph processing, in: Proceedings of the 2010 ACM SIGMOD International Conference on Management of Data, ACM, 2010, pp. 135–146.

[58] Y. Low, D. Bickson, J. Gonzalez, C. Guestrin, A. Kyrola, J.M. Hellerstein, Distributed graphlab: a framework for machine learning and data mining in the cloud, Proc. VLDB Endowment 5 (8) (2012) 716–727.

[59] J.L. Herlocker, J.A. Konstan, L.G. Terveen, J.T. Riedl, Evaluating collaborative filtering recommender systems, ACM Trans. Inf. Syst. 22 (1) (2004) 5–53.

[60] C. Anderson, The long tail: Why the future of business is selling less of more, in: Hachette Books, 2006.

[61] Y.-J. Park, A. Tuzhilin, The long tail of recommender systems and how to leverage it, in: Proceedings of the 2008 ACM Conference on Recommender Systems, ACM, 2008, pp. 11–18.

[62] S. Vargas, P. Castells, Rank and relevance in novelty and diversity metrics for recommender systems, in: Proceedings of the Fifth ACM Conference on Recommender Systems, ACM, 2011, pp. 109–116.

[63] S. Vargas, P. Castells, Improving sales diversity by recommending users to items, in: Proceedings of the 8th ACM Conference on Recommender Systems, ACM, 2014, pp. 145–152.

[64] G. Guo, J. Zhang, D. Thalmann, Merging trust in collaborative filtering to alleviate data sparsity and cold start, Knowl.-Based Syst. 57 (2014) 57–68.

[65] Y. Koren, R. Bell, C. Volinsky, Matrix factorization techniques for recommender systems, Computer 42 (8).

[66] H. Ma, D. Zhou, C. Liu, M.R. Lyu, I. King, Recommender systems with social regularization, in: Proceedings of the Fourth ACM International Conference on Web Search and Data Mining, ACM, 2011, pp. 287–296.

[67] H. Ma, I. King, M.R. Lyu, Learning to recommend with explicit and implicit social relations, ACM Trans. Intell. Syst. Technol. 2 (3) (2011) 29.

[68] D.F. Gurini, F. Gasparetti, A. Micarelli, G. Sansonetti, Temporal people-to-people recommendation on social networks with sentiment-based matrix factorization, Future Gener. Comput. Syst. 78 (2018) 430–439.

[69] H. Cao, E. Chen, J. Yang, H. Xiong, Enhancing recommender systems under volatile user interest drifts, in: Proceedings of the 18th ACM Conference on Information and Knowledge Management, ACM, 2009, pp. 1257–1266.

[70] C. Castro-Herrera, J. Cleland-Huang, B. Mobasher, A recommender system for dynamically evolving online forums, in: Proceedings of the Third ACM Conference on Recommender Systems, ACM, 2009, pp. 213–216.

[71] K. Kapoor, K. Subbian, J. Srivastava, P. Schrater, Just in time recommendations: Modeling the dynamics of boredom in activity streams, in: Proceedings of the Eighth ACM International Conference on Web Search and Data Mining, ACM, 2015, pp. 233–242.

[72] S. Chang, Y. Zhang, J. Tang, D. Yin, Y. Chang, M.A. Hasegawa-Johnson, T.S. Huang, Streaming recommender systems, in: Proceedings of the 26th International Conference on World Wide Web, International World Wide Web Conferences Steering Committee, 2017, pp. 381–389.

[73] P. Matuszyk, M. Spiliopoulou, Stream-based semi-supervised learning for recommender systems, Mach. Learn. (2017) 1–28.

[74] Z. Lu, S.J. Pan, Y. Li, J. Jiang, Q. Yang, Collaborative Evolution for User Profiling in Recommender Systems, IJCAI, New York City, USA, 2016, p. 1.

[75] Apache hadoop, http://hadoop.apache.org. (Online; Accessed 15 June 2017).

[76] J. Zhao, J. Pjesivac-Grbovic, Mapreduce: The programming model and practice, tutorial (2009). https://research.google.com/archive/papers/mapreduce-sigmetrics09-tutorial.pdf.

[77] S. Owen, R. Anil, T. Dunning, E. Friedman, Mahout in Action, Manning Publications Co. Greenwich, CT, USA, 2011.

[78] J.G. Shanahan, L. Dai, Large scale distributed data science using apache spark, in: Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, 2015, pp. 2323–2324.

[79] M. Yang, R.T. Ma, Smooth task migration in apache storm, in: Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data, ACM, 2015, pp. 2067–2068.

[80] A. Sapountzi, K.E. Psannis, Social networking data analysis tools & challenges, Future Gener. Comput. Syst.

[81] E. Karydi, K. Margaritis, Parallel and distributed collaborative filtering: A survey, ACM Comput. Surv. 49 (2) (2016) 37.

[82] B.M. Sarwar, G. Karypis, J. Konstan, J. Riedl, Recommender systems for large-scale e-commerce: Scalable neighborhood formation using clustering, in: Proceedings of the fifth international conference on computer and information technology, Vol. 1, 2002.

[83] F. Rezaeimehr, P. Moradi, S. Ahmadian, N.N. Qader, M. Jalili, Tcars: Time-and community-aware recommendation system, Future Gener. Comput. Syst. 78 (2018) 419–429.

[84] J. Guo, Z. Gao, N. Liu, Y. Wu, Recommend products with consideration of multi-category inter-purchase time and price, Future Gener. Comput. Syst. 78 (2018) 451–461.

[85] M. Gan, R. Jiang, Flower: Fusing global and local associations towards personalized social recommendation, Future Gener. Comput. Syst. 78 (2018) 462–473.

[86] H. Wang, N. Wang, D.-Y. Yeung, Collaborative deep learning for recommender systems, in: Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, 2015, pp. 1235–1244.

[87] J. Wei, J. He, K. Chen, Y. Zhou, Z. Tang, Collaborative filtering and deep learning based recommendation system for cold start items, Expert Syst. Appl. 69 (2017) 29–39.

[88] B. Hidasi, A. Karatzoglou, L. Baltrunas, D. Tikk, Session-based recommendations with recurrent neural networks, in: International Conference on Learning Representations.

[89] J. Lian, F. Zhang, X. Xie, G. Sun, Cccfnet: A content-boosted collaborative filtering neural network for cross domain recommender systems, in: Proceedings of the 26th International Conference on World Wide Web Companion, International World Wide Web Conferences Steering Committee, 2017, pp. 817–818.

[90] S. Deng, L. Huang, G. Xu, X. Wu, Z. Wu, On deep learning for trust-aware recommendations in social networks, IEEE Trans. Neural Netw. Learn. Syst. 28 (5) (2017) 1164–1177.

[91] D. Jannach, P. Resnick, A. Tuzhilin, M. Zanker, Recommender systems—: beyond matrix completion, Commun. ACM 59 (11) (2016) 94–102.

[92] A. Hernando, J. Bobadilla, F. Ortega, A non negative matrix factorization for collaborative filtering recommender systems based on a bayesian probabilistic model, Knowl.-Based Syst. 97 (2016) 188–202.

[93] P. Symeonidis, Matrix and tensor decomposition in recommender systems, in: Proceedings of the 10th ACM Conference on Recommender Systems, ACM, 2016, pp. 429–430.

[94] J. He, X. Li, L. Liao, D. Song, W.K. Cheung, Inferring a personalized next point-of-interest recommendation model with latent behavior patterns, in: Thirtieth AAAI Conference on Artificial Intelligence, 2016.

[95] Y. Li, D. Wang, H. He, L. Jiao, Y. Xue, Mining intrinsic information by matrix factorization-based approaches for collaborative filtering in recommender systems, Neurocomputing 249 (2017) 48–63.

[96] Q. Zhao, G. Adomavicius, F.M. Harper, M.C. Willemsen, J.A. Konstan, Toward better interactions in recommender systems: Cycling and serpentining approaches for top-n item lists, in: CSCW, 2017, pp. 1444–1453.

[97] M. Kaminskas, D. Bridge, Diversity, serendipity, novelty, and coverage: A survey and empirical analysis of beyond-accuracy objectives in recommender systems, ACM Trans. Interact. Intell. Syst. 7 (1) (2016) 2.

[98] S. Berkovsky, R. Taib, D. Conway, How to recommend?: User trust factors in movie recommender systems, in: Proceedings of the 22nd International Conference on Intelligent User Interfaces, ACM, 2017, pp. 287–300.