Semantic Social Breadth-first search and Depth-first search Recommendation Algorithms

SSBFS and SSDFS recommendation

Dalia Sulieman[*,] — Maria Malek[*] — Hubert Kadima[*] — Dominique Laurent[**]**

ETIS-ENSEA Université de Cergy-Pontoise CNRS France[**]
EISTI Ecole Internationale des Sciences du Traitement de l'Information France[*]
dalia.sulieman,maria.malek,hubert.kadima@eisti.eu
dominique.laurent@dept-info.u-cergy.fr

ABSTRACT. *Social networks gained more attention over the last years, due to their importance in the users' modern life. Moreover, social networks have become a great source of information, and several applications have been proposed to extract information from them such as: recommender systems. In this paper we present two recommendation algorithms called: Semantic Social Breadth First Search SSBFS and Semantic Social Depth First Search SSDFS. These algorithms are designed to recommend items to users who are connected via a social network. Our algorithms are based on three main features: a social network analysis measure, graph searching algorithms, and semantic relevance measure. We apply these algorithms to a real dataset (Amazon dataset) and we compare them with item-based collaborative filtering and hybrid recommendation algorithms. Our results show good precision as well as good performance in terms of runtime. Furthermore, our results show that SSBFS and SSDFS search a small part of the dataset, compared to the other algorithms.*

RÉSUMÉ. *Nous présentons dans ce papier, deux algorithmes de recommandation fondés sur l'éxploration en profondeur et en largeur du graphe sociale. Ces algorithmes combinent, l'analyse des réseaux sociaux et les profils sémantiques des utilisateurs dans le processus de l'élaboration de la recommandation. Nous intégronsdes les heuristiques pour le parcours de graphe (parcours en profondeur DFS et parcours on largeur BFS) pour explorer le graphe fondé sur la représentation sémantique des profils utilisateurs extrait á partir de l'ontologie du domaine. Nous avons appliqué ces algorithmes sur un ensemble de données réelles extraits des données de Amazon. Nos résultats montrent des valeurs de précision, de rappel et de F-measure satisfaisantes. Nous montrons également comment réduire le temps de calcul par rapport aux approches de recommandation classiques.*

KEYWORDS: *Social Network Analysis, Recommender Systems, Graph Algorithms*

MOTS-CLÉS : *Analyse des Réseaux Sociaux, Systèms de Recommandation, Algorithmes de Graphe, Mesure de Similarité Sémantique*

1. Introduction

It is well known that the information available on the Web increases rapidly in time. As a result, humans are not capable of understanding, exploiting or even handling such a huge amount of information. *Recommender systems* are thus widely used to overcome this information overload, by filtering information in order to help users in making choices according to their interests. Three main categories of recommender systems have been distinguished in [ADO 05]: content-based, collaborative-filtering or social, and hybrid. Moreover, it is now common practice that users be connected through a *social network*, in which vertices and edges represent respectively people and their social interactions (such as friendship and co authorship) [NEW 10]. Furthermore, due to the emergence of the utilization of social networks, and due to its significant applications in our modern life, social networks have become a great source of information, such as: opinion mining and retrieval, finding experts and social recommender systems [JAM 09].

In this paper we introduce a *semantic social recommender system*, in which we suppose a set of users and a set of items such that users are connected through a social network, and users and items are described via a taxonomy. In this setting, given an item, we use a heuristic based depth first search algorithm to search the social network in order to compute a relevant set of users to whom the item can be recommended, while scanning the network as little as possible. Thus, our main contribution is to provide two algorithms that combine all available information (namely, the domain taxonomy relating all the available items, user preferences seen as part of that taxonomy and the social network connecting users) in order to efficiently compute the set of users to whom a given item should be recommended. The experiments reported in the paper show that our algorithms outperform existing approaches that do not fully exploit all of this information.

2. Semantic-Social recommender system

2.1. *Semantic information*

The semantic part of our proposed model relies on three fundamental aspects:

1) *User preferences*: grouped in a user profile, which contains all the possible information about users, such as activities and interests. User profile has several types of representation, as vectorial representation and conceptual representation [GAU 07].

2) *Domain taxonomy*: A taxonomy is defined as a collection of entities, organized in a hierarchical structure (*'is-a'* hierarchy), in order to describe objects of a certain domain. Several recommender systems use taxonomy to *estimate users preferences*, in the case of lack of information about users [ZUB 07].

3) *Semantic similarity measures*: used to compute the relevancy between the ontology concepts [JIA 97].

In our model we use a *domain taxonomy*, to represent the knowledge about users and items. We also attach a *semantic-taxonomy profile* to each user and item, then we use a *semantic similarity measure* to compute the semantic relevance between users and items. For that, we now introduce the following definitions:

Definition 1 *Given a set of items, the Semantic Taxonomy Tree* (STT) *associated with these items is a tree in which* nodes *are the domain terms, and* edges *represent the hierarchy between these terms. A* STT *is represented as a set containing sets of pairs of the form* $(term, level)$*, where* $level$ *is an integer. When a* STT *has* n *levels, the term of level* 0 *is the most general term, while the terms of level* $n-1$ *are the most specific terms in the domain. Moreover, we assume that every item is associated with a* unique *leaf of the* STT*.*

Figure 1 shows a STT describing books, for which the set of pairs is $T(Books) = \{(Books, 0), (Culture\&Tradition, 1), (Science, 1), (Computers, 2), \ldots\}$.

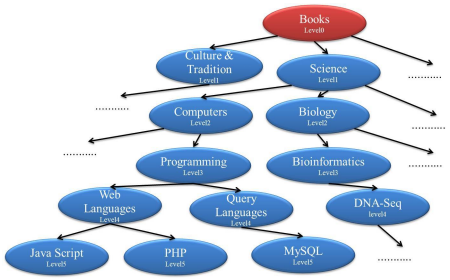

Figure 1. *A* STT *for books*

Given a STT, we associate every item with a subset of this taxonomy as follows:

Definition 2 *Let* x *be an item. The* Item Preference Tree of x*, denoted by* $IPT(x)$*, is the subtree (path) of the* STT *defined by the set of pairs of the form* $(term, level)$ *connecting the root of the* SST *with the leaf to which item* x *is associated.*

In our model, we suppose that every user u is associated with a set of items, denoted by $I(u)$, containing all items u bought and liked in the past. Based on this information, we define the *user preference subtree* of a given user as follows:

Definition 3 *Let* u *be a user and* $I(u)$ *its associated set of items. The* User Preference Tree of u*, denoted by* $UPT(u)$*, is the union of all item preference trees of the items in* $I(u)$*. In other words, we have:* $UPT(u) = \bigcup_{x \in I(u)} IPT(x)$*. See Figure 2.*

Figure 2, shows the UPT associated to a user who bought and liked the items $Item1$, $Item2$ and $Item3$.

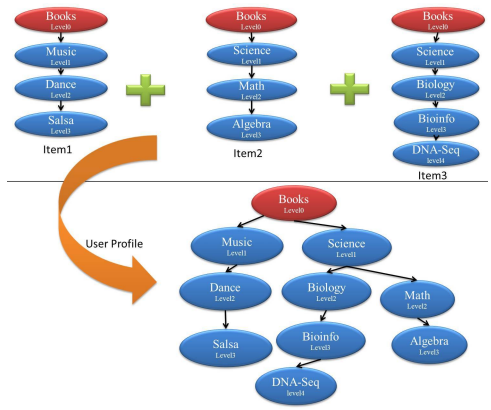

Figure 2. *A User Preference Tree*

2.1.1. *Semantic similarity measure*

Based on the previous definitions, we present a user-item semantic relevance measure, to compute the semantic relevance between users and items. To do so, we introduce the following similarity measure between two prefernce trees. Let P_1 and P_2 be two sets of sets of pairs of the form (t, l) where t is a term and l is a level in the associated STT. The similarity between P_1 and P_2, denoted by $\sigma(P_1, P_2)$, is defined by

$$\sigma(P_1, P_2) = \frac{1}{\mu}\left(\sum_{(t,l)\in P_1\cap P_2} l\right)$$

where $\mu = \min\left(\sum_{(t,l)\in P_1} l, \sum_{(t,l)\in P_2} l\right)$.
The function σ allows to define the similarity between a user and an item.

Definition 4 *Let u be a user and x an item. The similarity measure between u and x, denoted by $sim(u, x)$, is the similarity between their associated preference trees is: $sim(u, x) = \sigma(UPT(u), IPT(x))$*

Example 1 *Figure 2 shows an example of user u, who likes three items: $Item1$, $Item2$ and $Item3$. In this case, the user set of associated items is stated as: $I(u) = \{Item1, Item2, Item3\}$, and the user preferences tree is stated as: $UPT(u) = IPT(Item1) \cup IPT(Item2) \cup IPT(Item3)$. In order to compute the semantic relevancy between u and item x (x is defined as: $IPT(x) = \{(Books, 0), (Science, 1), (Math, 2), (Geometry, 3)\}$), we apply our proposed measure $sim(u, x)$ as follows: since $UPT(u) \cap IPT(x) = \{(Books, 0), (Science, 1), (Math, 2)\}$, $\sum_{(t,l)\in UPT(u)} l = 21$ and $\sum_{(t,l)\in IPT(x)} l = 6$, we obtain $sim(u, x) = 3/6 = 0.5$.*

In the algorithms presented in the next section, the similarity measure sim is used to discard users whose relevancy with the item to be recommended is too low.

2.2. *Social information*

The second component of our model is the "Social Information", which is mainly based on: collaboration social networks [RAM 07], and degree centrality [NEW 10].

2.2.1. *Collaboration Social Networks*

Collaboration networks are extracted from bipartite graphs, using the *one-mode projection* [RAM 07]. Bipartite graphs and one-mode projections have been used in several recommender systems [CAN 06, GRU 08]. Figure 3 (a) shows a user-item bipartite graph G. Figure 3 (b) shows the extracted user one-mode projection and Figure 3 (c) shows the extracted item one-mode projection.

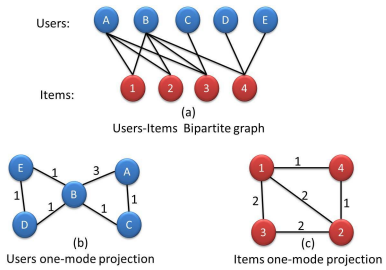

Figure 3. *A bipartite graph and its associated one-mode projections*

In our approach, we consider a collaborative social network based on a user one-mode projection of a user-item bipartite graph, where the edge weight equals to the number of products the connected users have bought in common.

2.2.2. *Degree Centrality*

Degree centrality of vertices is one of the most popular social network analysis measures [NEW 10]. If v is a vertex of a non directed graph, the centrality degree of v is defined as the number of edges involving this vertex [CAN 06].

Our proposed algorithms use the degree centrality as a measure to guide the search in the collaboration social network.

3. Two Semantic-Social recommendation algorithms based on Depth First Search and Breadth First Search

In our approach, the graph to be searched is the user one-mode projection associated with a user-item bipartite graph, we use two algorithms based on DFS (Depth First Search) and BFS (Breadth First Search) algorithms to explore this graph. However, since this graph is huge, we apply heuristics in order to avoid exploring *all* vertices and *all* edges, such heuristic depends on three concepts as follows. Semantic similarity, degree centrality and the graph edges weights, as defined in the previous section. It should be first noted that, in our algorithms, the vertices of the user-item bipartite graph G to be explored are labeled in order to visit each of them at most once. More precisely, every vertex v of G is associated with a label whose possible values are *unvisited* or *visited*.

As we want to avoid visiting all vertices and edges of G, but in order to visit as many relevant vertices as possible, we choose to start the exploration of G through the vertices that have a high *centrality degree*, that is, the vertices that are connected to a high number of other vertices.

To this end, the centrality degree of every vertex is computed (line 5 of Algorithm 1 and line 6 of Algorithm 3) and the N vertices having the highest centrality degree are stored in a vector called Top-N vector (line 6 of Algorithm 1 and line 7 of Algorithm 3). When exploring G, the only paths going through a vertex from this vector are considered (line 7 of Algorithm 1 and line 8 of Algorithm 3). Our experiments show that considering only these vertices, we have good results in the sense that we avoid visiting all vertices while obtaining relevant recommendation lists.

3.1. *Semantic-Social Depth first search SSDF recommendation algorithm*

Starting form the vertices in Top-N-vector, the graph is explored using a DFS algorithm according to a heuristic called Heuristic-Search (line 7 of Algorithm 1). According to this heuristic, considering a vertex from v from Top-N-vector, v is processed if: it is still unvisited, and its similarity with the item x is greater than a given threshold θ. Then, all successors v' of v, v' is recursively processed in the same way, until reaching a vertex that fails to satisfy the similarity requirement or until reaching an edge whose weight is less than a given threshold δ. The corresponding procedures are shown in Algorithm 1 and Algorithm 2.

3.2. *Semantic-Social Breadth First Search SSBFS recommendation algorithm*

Our proposed Semantic-Social Breadth First Search SSBFS algorithm works as follows. In line 8 of Algorithm 3, the algorithm copies all the top-N vector's vertices into a queue. Starting from the first vertex v in the queue, and if the similarity between v and the input item x is greater than a given threshold θ, a breadth first search BFS

Algorithm 1 Semantic-Social Depth first search

Require: (i) A user-item bipartite graph G with vertices V and edges E
(ii) An item x and its Item Preference Tree $IPT(x)$
(iii) A positive integer N
(iv) A user-item similarity threshold θ
(v) An edge weight threshold δ
Ensure: List of recommended users $user\_list$

```
for all vertices v in V do
    v.label = unvisited
end for
user_list = empty list
Compute the degree centrality of every vertex in V
Top-N-vector = all vertices of V with the top-N
                highest centrality degrees
for all v in Top-N-vector do
    Call Heuristic-Search with input v
end for
return user_list
```

Algorithm 2 Heuristic-Search

```
if v.label = unvisited then
    v.label = visited
    if sim(v, x) > θ then
        Add v to the current value of user_list
        for all e = (v, v') ∈ E do
            if e.weight > δ then
                Heuristic-Search(G, v')
            end if
        end for
    end if
end if
```

is applied. Then for all the successors v' of v, a successor v' is processed by BFS if: v' is not visited, the similarity between v' and the input item x is greater than θ, and the weight of the edge connecting v and v' is greater than a given threshold δ. SSBFS continues searching the graph in the same way, while the queue is not empty. See Algorithm 3.

Algorithm 3 Semantic-Social Breadth first search

Require: (i) A user-item bipartite graph G with vertices V and edges E
(ii) An item x and its Item Preference Tree $IPT(x)$
(iii) A positive integer N
(iv) A user-item similarity threshold θ
(v) An edge weight threshold δ
Ensure: List of recommended users $user\_list$

```
for all vertices v in V do
    v.label = unvisited
end for
user_list = empty list
Q = empty queue
Compute the degree centrality of every vertex in V
Top-N-vector = all vertices of V with the top-N
                highest centrality degrees
for all v in Top-N-vector do
    v.label = visited
    if sim(v, x) > θ then
        user_list.add(v)
        enqueue(Q, v)
    end if
end for
while Q ≠ ∅ do
    v = dequeque(Q)
    for all e = (v, v') ∈ E do
        if v'.label = unvisited and e.weight > δ then
            v'.label = visited
            if sim(v', x) > θ then
                user_list.add(v')
                enqueue(Q, v')
            end if
        end if
    end for
end while
return user_list
```

4. Experiments and Results

4.1. *Data Set Description*

In our experiments, we use *Amazon* data [1] as a real dataset to test and compare our algorithms. According to this dataset, all information about users is related to their previous purchases (items the user preferred and bought in the past), and all information about items is related to a taxonomy representation of item preferences.

1. Amazon dataset description is available on the web page of Stanford University http://snap.stanford.edu/data/amazon-meta.html

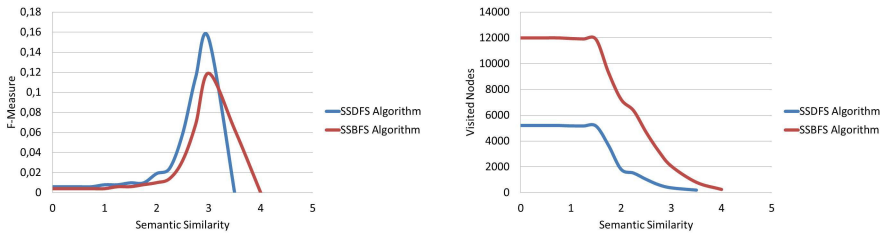

Figure 4. *F-Measure and graph visited nodes according to θ*

In our experiments, we firstly build the collaboration social network, then we apply the recommendation algorithms by submitting recommendation queries. In order to build the Collaboration Social Network, we firstly extract item-user bipartite graph from Amazon dataset. Then we build the users one-mode projection from item-user bipartite graph [ZHO 07]. In the users one-mode projection graph, edges connecting users are weighted as being the number of bought items these users have in common. According to this dataset, users one-mode projection collaboration social network has $38,982$ vertices and more than 5 million edges.

4.2. *Experiment Settings*

In order to evaluate our proposed algorithms, we suggest to compare them with two of the most common recommendation algorithms as follows.

– Item-based collaborative filtering algorithm, using *cosine* similarity measure to compute the similarity between two items [SAR 01].

– Hybrid recommendation algorithm which combines: *Collaborative filtering* recommendation, using cosine similarity measure, and *Content-based* recommendation, using our proposed semantic-similarity measure as defined in 2.1.1.

Algorithms are assessed according to the following criteria:

– Accuracy measures: we use precision, recall and F-measure, as metrics to compare the accuracy of our proposed algorithms [HER 04].

– The percentage of graph vertices (users) which have been visited by the recommendation algorithms, during the recommendation process.

– The time each algorithm takes, to answer a recommendation query.

Our proposed algorithms mainly depend on three important parameters: the *Semantic Similarity* parameter θ, the number N of the *Top-N nodes vector* and the *Edges weight* parameter δ. Furthermore, we found that accuracy and performance are strongly related to these three parameters. See Figure 4, Figure 5, and Figure 6.

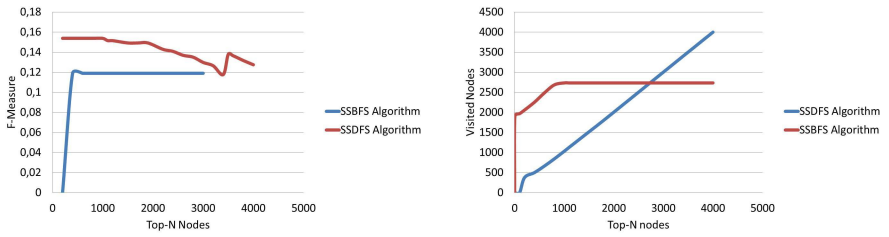

Figure 5. *F-Measure and graph visited nodes according to N*

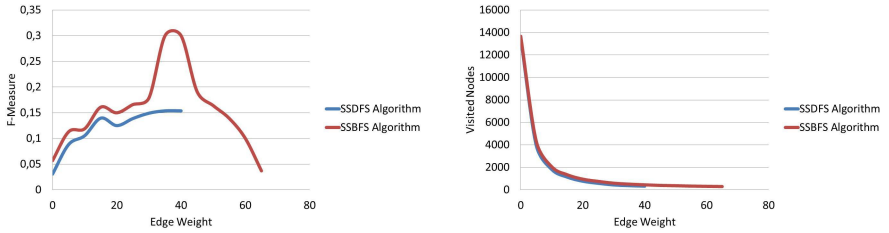

Figure 6. *F-Measure and graph visited nodes according to δ*

4.3. *Experiment Results*

We developed our algorithms using java 6, we also used JUNG (Java Universal Network/Graph) [MAD] as a framework for social network analysis. Moreover, we performed our experiments on an Intel(R) Xeon(R) CPU E5520 $2.27GHz$ with 12 Giga of RAM, using Debian Linux as operating system.

We implemented 58 recommendation queries on the four recommendation algorithms. Then we computed average precision, average recall, average F-Measure, average execution time and average data coverage. The obtained results are described as follows: Figure 7 shows that: SSDFS gives the best precision value and Hybrid al-

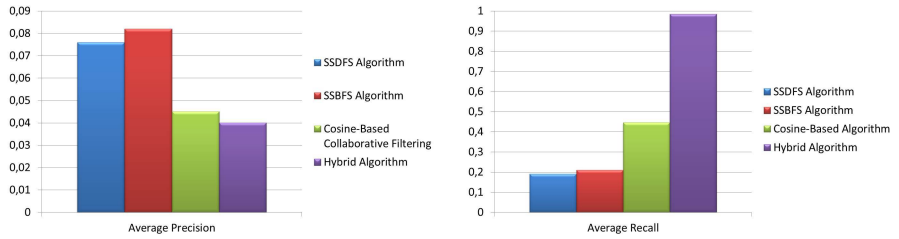

Figure 7. *comparison of average precision and average recall between SSDFS, SSBFS, Cosine-Based CF and hybrid recommendation algorithms*

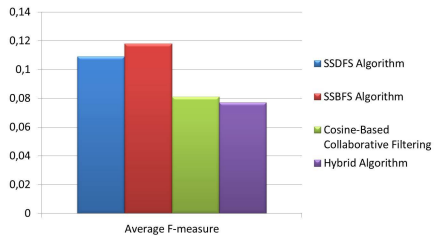

Figure 8. *Comparison of average F-Measure between SSDFS, SSBFS, Cosine-Based CF and hybrid recommendation algorithms*

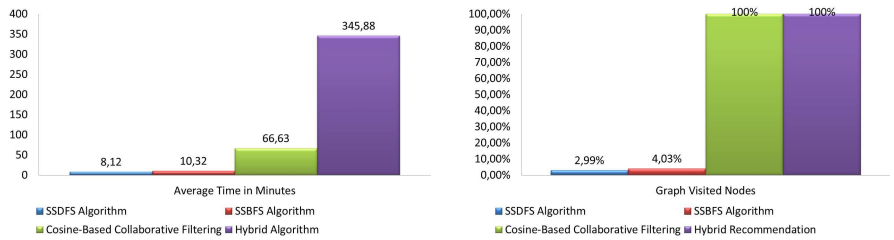

Figure 9. *Comparison of recommendation time and graph visited nodes between SSDFS, SSBFS, Cosine-Based CF and hybrid recommendation algorithms*

gorithm gives the best recall value, while Figure 8 shows that, SSDFS gives the best *average F-Measure* value. Moreover, from Figure 9 we find that, SSDFS answers the recommendation query in a very short time and it explores a very small amount of graph nodes.

5. Related Work

In 1999, IRA (Intelligent Recommendation Algorithm, [AGG 99]) was proposed as a graph-based collaborative filtering recommender system, in which a breadth-first search algorithm is used to compute the shortest paths between graph vertices (users). Moreover, user-item bipartite graph and one-mode projection are used in a movie recommender system proposed in [MIR 03]. In this system a recommendation graph has been defined as the sum of the bipartite graph and the one-mode projection graph, then the shortest path algorithm has been applied on this recommendation graph. In [JAM 09], a random walk algorithm is proposed to recommend items in a trust network. This algorithm recommends items based on ratings expressed by trusted friends, using random walk and probabilistic item selection.

Other recommender systems include semantic aspects, in addition to collaborative filtering aspects. In [Jul 08] a recommendation algorithm is introduced for collabora-

tive URL tagging. In this system, user interests are modeled according to their social ties and the vocabularies they use to tag URLs. In [SHE 08] similar tags are grouped in clusters, these tag clusters are used as intermediate sets between users and items. In [ZIE 04] the authors proposed to represent the users by a vector of scores assigned to topics taken from domain taxonomy; then a semantic similarity measure (between the users vectors and domain taxonomy) is used in a semantic-based (taxonomy) recommender system.

6. Conclusion

In this paper we introduced two semantic social recommendation algorithms called *Semantic Social Depth First Search* and *Semantic Social Breadth First Search*, these algorithms recommend an input item to a group of users. In fact, we assume that, users are connected via collaboration social network, and users and items are described via semantic taxonomy. Our proposed algorithms are mainly based on Depth-First Search and Breadth-First Search algorithms with some modifications, which are related to the *semantic similarity* between users and the input item, and to the *social network analysis measures*. We applied these two algorithms on a real dataset (Amazon Dataset), and we compared them with collaborative filtering and hybrid recommendation algorithms. Our results showed that Semantic Social Depth First Search algorithm give good F-Measure values, with good performance.

Our perspectives are to study and test other graph algorithms, and to study and test other social network analysis measures for the same model.

7. References

[ADO 05] ADOMAVICIUS G., TUZHILIN A., "Toward the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions", *Knowledge and Data Engineering, IEEE Transactions on*, vol. 17, num. 6, 2005, p. 734–749.

[AGG 99] AGGARWAL C. C., WOLF J. L., WU K., YU P. S., "Horting Hatches an Egg: A New Graph-Theoretic Approach to Collaborative Filtering", *KDD '99: Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining*, San Diego, California, United States, 1999, ACM, p. 201–212.

[CAN 06] CANO P., CELMA O., KOPPENBERGER M., BULDÚ M. J., "Topology of music recommendation networks", *Chaos An Interdisciplinary Journal of Nonlinear Science*, vol. 16, 2006.

[GAU 07] GAUCH S., SPERETTA M., CHANDRAMOULI A., MICARELLI A., "User Profiles for Personalized Information Access The Adaptive Web", BRUSILOVSKY P., KOBSA A., NEJDL W., Eds., *The Adaptive Web*, vol. 4321 of *Lecture Notes in Computer Science*, chapter 2, p. 54–89, Springer Berlin / Heidelberg, Berlin, Heidelberg, 2007.

[GRU 08] GRUJIĆ J., "Movies Recommendation Networks as Bipartite Graphs", *Proceedings of the 8th international conference on Computational Science, Part II*, ICCS '08, Berlin, Heidelberg, 2008, Springer-Verlag, p. 576–583.

[HER 04] HERLOCKER J. L., KONSTAN J. A., TERVEEN L. G., RIEDL J. T., "Evaluating collaborative filtering recommender systems", *ACM Trans. Inf. Syst.*, vol. 22, num. 1, 2004, p. 5–53.

[JAM 09] JAMALI M., ESTER M., "TrustWalker: a random walk model for combining trust-based and item-based recommendation", *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, NY, USA, 2009, ACM, p. 397–406.

[JIA 97] JIANG J. J., CONRATH D. W., "Semantic Similarity Based on Corpus Statistics and Lexical Taxonomy", *International Conference Research on Computational Linguistics (ROCLING X)*, 1997.

[Jul 08] JULIA STOYANOVICH, SIHEM AMER YAHIA, CAMERON MARLOW AND CONG YU, "A Study of the Benefit of Leveraging Tagging Behavior to Model UsersÃnterests in del.icio.us", *In AAAI Spring Symposium on Social Information Processing*, 2008.

[MAD] MADAHAIN J. O., FISHER D., SMYTH P., WHITE S., BOEY Y.-B., "Analysis and Visualization of Network Data using JUNG", *Journal of Statistical Software*, vol. VV, num. II.

[MAL 10] MALEK M., SULIEMAN D., "Exhaustive and Guided Algorithm for Recommendation in a Professional Social Network", *7th conference on application of Social Network Analysis*, ZÃ$\frac{1}{4}$rich, September 2010, p. 15-17.

[MIR 03] MIRZA B. J., KELLER B. J., RAMAKRISHNAN N., "Studying Recommendation Algorithms by Graph Analysis", *Journal of Intelligent Information Systems*, vol. 20, num. 2, 2003, p. 131–160, Springer Netherlands.

[NEW 10] NEWMAN M., *Networks An Introduction*, Oxford University Press, 2010.

[PAL 10] PALTOGLOU G., THELWALL M., "A study of information retrieval weighting schemes for sentiment analysis", *ACL '10: Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, Morristown, NJ, USA, 2010, Association for Computational Linguistics, p. 1386–1395.

[RAM 07] RAMASCO J. J., "Social inertia and diversity in collaboration networks", *The European Physical Journal Special Topics*, vol. 143, 2007, p. 47-50.

[RES 94] RESNICK P., IACOVOU N., SUCHAK M., BERGSTROM P., RIEDL J., "GroupLens: an open architecture for collaborative filtering of netnews", *CSCW '94: Proceedings of the 1994 ACM conference on Computer supported cooperative work*, New York, NY, USA, 1994, ACM, p. 175–186.

[SAR 01] SARWAR B., KARYPIS G., KONSTAN J., REIDL J., "Item-based collaborative filtering recommendation algorithms", *Proceedings of the 10th international conference on World Wide Web*, New York, NY, USA, 2001, ACM, p. 285–295.

[SHA 95] SHARDANAND U., MAES P., "Social Information Filtering: Algorithms for Automating "Word of Mouth"", *Proceedings of ACM CHI'95 Conference on Human Factors in Computing Systems*, vol. 1, 1995, p. 210–217.

[SHE 08] SHEPITSEN A., GEMMELL J., MOBASHER B., BURKE R., "Personalized recommendation in social tagging systems using hierarchical clustering", *Proceedings of the 2008 ACM conference on Recommender systems*, RecSys '08, New York, NY, USA, 2008, ACM, p. 259–266.

[ZHA 05] ZHANG J., ACKERMAN M. S., "Searching for expertise in social networks: a simulation of potential strategies", *GROUP '05: Proceedings of the 2005 international ACM*

SIGGROUP conference on Supporting group work, NY, USA, 2005, ACM, p. 71–80.

[ZHA 07] ZHANG J., ACKERMAN M. S., ADAMIC L., "Expertise networks in online communities: structure and algorithms", *WWW '07: Proceedings of the 16th international conference on World Wide Web*, NY, USA, 2007, ACM, p. 221–230.

[ZHO 07] ZHOU T., REN J., MEDO M., ZHANG Y. C., "Bipartite network projection and personal recommendation", *Physical Review E (Statistical, Nonlinear, and Soft Matter Physics)*, vol. 76, num. 4, 2007, APS.

[ZIE 04] ZIEGLER C. N., LAUSEN G., LARS S. T., "Taxonomy-driven computation of product recommendations", *Proceedings of the thirteenth ACM international conference on Information and knowledge management*, CIKM '04, New York, NY, USA, 2004, ACM, p. 406–415.

[ZUB 07] ZUBER V. S., FALTINGS B., "OSS: A Semantic Similarity Function based on Hierarchical Ontologies", *Proceedings of IJCAI 07*, 2007, p. 551–556.