Uncertain Graphs meet Collaborative Filtering

Claudio Taranto, Nicola Di Mauro, and Floriana Esposito

Department of Computer Science, University of Bari "Aldo Moro"
via E. Orabona, 4 - 70125, Bari, Italy
{claudio.taranto,ndm,esposito}@di.uniba.it

Abstract. Collaborative filtering (CF) aims at predicting the user interest for a given item. In CF systems a set of users ratings is used to predict the rating of a given user on a given item using the ratings of a set of users who have already rated the item and whose preferences are similar to those of the user. In this paper we propose to use a framework based on *uncertain graphs* in order to deal with collaborative filtering problems. In this framework relationships among users and items and their corresponding likelihood will be encoded in a uncertain graph that can then be used to infer the probability of existence of a link between an user and an item involved in the graph. In order to solve CF tasks the framework uses an approximate inference method adopting a constrained simple path query language. The aim of the paper is to verify whether uncertain graphs are a valuable tool for CF, by solving classical, complex and structured problems. The performance of the proposed approach is reported when applied to a real-world domain.

1 Introduction

The inherent uncertainty and complexity present in some real world domains has led to the emerging of many probabilistic frameworks, such as probabilistic graphical models [14] and statistical relational learning [6], able to deal with uncertain and structured domains. Learning and reasoning on *uncertain graphs*[1] has become an increasingly important research topic [19, 29, 9, 11]. In this model, each edge is associated with a probability representing the likelihood of its existence in the graph, and the edges existence is assumed to be mutually independent.

Collaborative filtering (CF) aims at predicting the user interest for a given item based on a collection of user profiles. Collaborative filtering is an approach adopted in recommender systems that attracted much of attention in recent years. In CF systems a set of users ratings is used to predict the rating of a given user u on a given item i using the ratings of a set of users who have already rated i and whose preferences are similar to the ones of u.

CF systems need to compare items against users and this task may be solved with a *memory based* approach that may be divided into *user-based* or *item-based* approaches. A typical example of memory based approaches are *neighborhood*

[1] Uncertain graphs are also referred to *probabilistic graphs* as in [29, 9].

based CF methods centered on computing the relationships between items or between users. Given an unknown rating to be estimated, memory-based CF firstly computes similarities between the given user and other users (*user-based* approach), or between the given item and other items (*item-based* approach). Then, the unknown rating is predicted by averaging the known ratings by similar users or by similar items [4, 15].

In this paper we propose to use uncertain graphs to deal with collaborative filtering problems. In particular, relationships among users and items and their corresponding likelihood will be encoded in a uncertain graph that can then be used to infer the probability of existence of a link between an user and an item involved in the graph.

The main questions that we want to answer in this paper are the following:

- **Q1**: are uncertain graphs a valuable tool for collaborative filtering?
- **Q2**: can uncertain graphs solve classical CF user-based and item-bases tasks?
- **Q3**: can uncertain graphs unify user-based and item-based CF approaches?

2 Uncertain graphs

Let $G = (V, E)$, be a graph where V is a collection of nodes and $E \in V \times V$ is the set of edges, or relationships, between the nodes.

Definition 1 (Uncertain graph). *An* uncertain graph *is a system* $G = (V, E, \Sigma, l_V, l_E, P)$*, where* (V, E) *is an undirected graph,* V *is the set of nodes,* E *is the set of edges,* Σ *is a set of labels,* $l_V : V \rightarrow \Sigma$ *is a function assigning labels to nodes,* $l_E : E \rightarrow \Sigma$ *is a function assigning labels to the edges, and* $P : E \rightarrow [0, 1]$ *is a function assigning* existence probability *values to the edges.*

The existence probability $P(e)$ of an edge $e = (u, v) \in E$ is the probability that edge between u and v can exist in the graph. A particular case of uncertain graph is the *certain graph* when the existence probability value on all edges is 1. In this paper we use the possible world semantics. In particular, we can imagine an uncertain graph G as a sampler of worlds, where each world is an instance of G. A certain graph G' is sampled from G according to P, denoted as $G' \sqsubseteq G$, when each edge $e \in E$ is selected to be an edge of G' with probability $P(e)$. Edges labeled with probabilities are treated as mutually independent random variables indicating whether or not the corresponding edge belongs to a certain graph. Assuming independence among edges, the probability distribution over certain graphs $G' = (V, E') \sqsubseteq G = (V, E)$ is given by

$$P(G'|G) = \prod_{e \in E'} P(e) \prod_{e \in E \setminus E'} (1 - P(e)). \tag{1}$$

Definition 2 (Simple path). *Given an uncertain graph* G*, a* simple path *of a length* k *from* u *to* v *in* G *is a sequence of edges* $p_{u,v} = \langle e_1, e_2, \ldots e_k \rangle$*, such that* $e_1 = (u, v_1)$*,* $e_k = (v_{k_1}, v)$*, and* $e_i = (v_{i-1}, v_i)$ *for* $1 < i < k$*, and all nodes in the path are distinct.*

Given G an uncertain graph, and $p_{s,t}$ a path in G from node s to node t, $l(p_{s,t}) = l(e_1)l(e_2)\cdots l(e_k)$ denotes the concatenation of the labels of all edges in $p_{s,t}$. Given a *context free grammar* (CFG) $\mathcal{C}$ a string of terminals s is derivable from $\mathcal{C}$ iff $s \in L(\mathcal{C})$, where $L(\mathcal{C})$ is the language generated from $\mathcal{C}$.

Definition 3 (Language constrained simple path). *Given an uncertain graph G and a context free grammar $\mathcal{C}$, a* language constrained simple path *is a simple path p such that $l(p) \in L(\mathcal{C})$.*

Given an uncertain graph G a main task corresponds to compute the probability that there exists a path between two nodes u and v, that is, querying for the probability that a randomly sampled certain graph contains a path between u and v. More formally, the *existence probability* $P_e(q|G)$ of a path q in an uncertain graph G corresponds to the marginal $P(G'|G)$ with respect to q:

$$P_e(q|G) = \sum_{G' \sqsubseteq G} P(q|G') \cdot P(G'|G) \tag{2}$$

where $P(q|G') = 1$ if there exits the path q in G', and $P(q|G') = 0$ otherwise. In other words, the existence probability of path q is the probability that the path q exists in a randomly sampled certain graph.

Definition 4 (Language constrained simple path probability). *Given an uncertain graph G and a context free grammar $\mathcal{C}$, the* language constrained simple path probability *of $L(\mathcal{C})$ is*

$$P(L(\mathcal{C})|G) = \sum_{G' \sqsubseteq G} P(q|G', L(\mathcal{C})) \cdot P(G'|G) \tag{3}$$

where$P(q|G', L(\mathcal{C}) = 1$ if there exists a path q in G' such that $l(q) \in L(\mathcal{C})$, and $P(q|G', L(\mathcal{C})) = 0$ otherwise.

In particular, the previous definition give us the possibility to compute the probability of a set of simple path queries fulfilling the structure imposed by a context free grammar. In this way we are interested in certain graphs that contain at least one path belonging to the language corresponding to the given grammar.

2.1 Inference

Computing the existence probability directly using (2) or (3) is intensive and intractable for large graphs since the number of certain graphs to be checked is exponential in the number of probabilistic edges. It involves computing the existence of the path in every certain graph and accumulating their probability. A natural way to overcome the intractability of computing the existence probability of a path is to approximate it using a Monte Carlo sampling approach [12]: 1) we sample n possible certain graphs, $G_1, G_2, \ldots G_n$ from G by sampling edges uniformly at random according to their edge probabilities; and 2) we check if the path exists in each sampled graph G_i. This process provides the basic sampling estimator

$$\widehat{P_e}(q|G) \approx P_e(q|G) = \frac{\sum_{i=1}^{n} P(q|G_i)}{n} \tag{4}$$

Note that is not necessary to sample all edges to check whether the graph contains the path. For instance, assuming to use an iterative depth first search procedure to check the path existence. When a node is just visited, we will sample all its adjacent edges and pushing them into the stack used by the iterative procedure. We will stop the procedure either when the target node is reached or when the stack is empty (non existence).

3 Uncertain graphs for collaborative filtering

The most common approach to CF is based on neighborhood models. User-oriented methods estimate unknown ratings based on recorded ratings of similar users, while in item-oriented approaches ratings are estimated using known ratings made by the same user on similar items.

Let U be a set of n users and I a set of m items. A rating r_{ui} indicates the preference by user u of item i, where high values mean stronger preference. Let S_u be the set of items rated from user u. For user-based approaches, the prediction of an unobserved rating $\widehat{r_{ui}}$ is computed as follows

$$\widehat{r_{ui}} = \overline{r_u} + \frac{\sum_{v \in U | i \in S_u} s_{uv} \cdot (r_{vi} - \overline{r_v})}{\sum_{v \in U | i \in S_u} |s_{uv}|} \tag{5}$$

where $\overline{r_u}$ represents the mean rating of user u, and s_{uv} stands for the similarity between users u and v, computed, for instance, using the Pearson correlation:

$$s_{uv} = \frac{\sum_{a \in S_u \cap S_v} (r_{ua} - \overline{r_u}) \cdot (r_{va} - \overline{r_v})}{\sqrt{\sum_{a \in S_u \cap S_v} (r_{ua} - \overline{r_u})^2 \sum_{a \in S_u \cap S_v} (r_{va} - \overline{r_v})^2}} \tag{6}$$

On the other side, item-based approaches predict the rating of a given item using the ratings of the user on the items considered as similar to the target item. Given a similarity measure, such as the Pearson correlation, the rating $\widehat{r_{ui}}$ is estimated as:

$$\widehat{r_{ui}} = \frac{\sum_{j \in S_u | j \neq i} s_{ij} \cdot r_{uj}}{\sum_{j \in S_u | j \neq i} |s_{ij}|} \tag{7}$$

These neighbourhood approaches see each user connected to other users or consider each item related to other items as in a network structure. In particular they rely on the direct connections among the entities involved in the domain. However, as recently proved, techniques able to consider complex relationships among the entities, leveraging the information already present in the network, involves an improvement in the processes of querying and mining [24, 21]. In [24] the authors improved the accuracy of a similarity measures between two annotated nodes in a graph by using link information. They showed that the similarity between nodes annotations may be improved using also the network context. Another approach [20] to enriched a graph representation is the addition of semantic information improving link prediction results in network datasets. In particular, a supervised learning method for building link predictors from structural attributes of the underlying network using some semantic attributes of the nodes has been adopted.

The approach used in this paper is to represent a dataset consisting of user ratings, $\mathcal{K} = \{(u, i, r_{ui}) | r_{ui} \text{ is known}\}$, with an uncertain graph and then performing inference on this graph to solve classical collaborative filtering tasks. Hence the question to be solved is how to build the uncertain graph from the flat rating representation $\mathcal{K}$. The formal characterization we have provided about uncertain graphs gives us the possibility to represent heterogeneous objects and connections.

3.1 Uncertain graph construction

Given the set of ratings $\mathcal{K} = \{(u, i, r_{ui}) | r_{ui} \text{ is known}\}$, we add a node with label `user` for each user in $\mathcal{K}$, and a node with label `item` for each item in $\mathcal{K}$. The next step is to add the edges among the nodes. Each edge is characterized by a label and a probability value, which should indicate the degree of similarity between the two nodes. Two kind of connections between nodes are added. For each user u, we added an edge, labeled as `simU`, between u and the k most similar users to u. The similarity between two users u and v is computed adopting a weighted Pearson correlation between the items rated by both u and v.

In particular, the probability of the edge `simU` connecting two users u and v is computed as:

$$P(\mathtt{simU}(u, v)) = s_{uv} \cdot w_u(u, v),$$

where s_{uv} is the Pearson correlation between the vectors of ratings corresponding to the set of items rated by both user u and user v, and

$$w_u(u, v) = \frac{|S_u \cap S_v|}{|S_u \cup S_v|},$$

where S_u is the set of items rated from user u.

For each item i, we added an edge, with label `simI`, between i and the most k similar items to i. In particular, the probability of the edge `simI` connecting the item i to the item j has been computed as:

$$P(\mathtt{simI}(i, j)) = s_{ij} \cdot w_i(i, j),$$

where s_{ij} is the Pearson correlation between the vectors corresponding to the histogram of the set of ratings for the item i and the item j, and

$$w_i(i, j) = \frac{|\overline{S}_i \cap \overline{S}_j|}{|\overline{S}_i \cup \overline{S}_j|},$$

where $\overline{S}_i$ is the set of users rating the item i.

Edges with probability equal to 1, and with label $\mathtt{r}_k$ between the user u and the item i, denoting the user u has rated the item i with a score equal to k, are added for each element r_{ui} belonging to $\mathcal{K}$.

After having defined the uncertain graph, now we can solve classical collaborative filtering task by computing the probability of some language constrained simple paths. Since the goal is to predict an unknown rating between an user u and an item i, let us assume that the values of r_{ui} are discrete and belonging to a set R. Given the uncertain graph G, the approach we used to predict the rating $\widehat{r_{ui}}$ is to solve the following maximization problem:

$$\widehat{r_{ui}} = \arg\max_j P(\mathtt{r}_j(u,i)|G), \tag{8}$$

where $\mathtt{r}_j(u,i)$ is the unknown link with label $\mathtt{r}_j$ between the user u and the item i. In particular, the maximization problem corresponds to compute the link prediction for each rating value and then choosing the rating with maximum likelihood.

The previous link prediction task is based on querying the probability of some language constrained simple path. For instance, user-based CF may be simulated by querying the probability of the paths, starting from a user node and ending to an item node, belonging to the context free language $L_i = \{\mathtt{simU}^1\mathtt{r}_i^1\}$. In particular, predicting the probability of the rating j as $P(\mathtt{r}_j(u,i)$ in (8) corresponds to compute the probability $P(q|G)$ for a query path in L_i, i.e., computing $P(L_i|G)$ as in (3):

$$\widehat{r_{ui}} = \arg\max_j P(\mathtt{r}_j(u,i)|G) \approx \arg\max_j P(L_i|G). \tag{9}$$

In the same way, item-base CF could be simulated by computing the probability of the paths belonging to the CFL $L_i = \{\mathtt{r}_i^1\mathtt{simI}^1\}$.

The power of the proposed framework gives us the possibility to construct more complex queries such as that belonging to the CFL $L_i = \{\mathtt{r}_i\mathtt{simI}^n : 1 \leq n \leq 2\}$, that gives us the possibility to explore the graph by considering not only direct connections. Finally, we can implement hybrid CF systems solving queries belonging to the CFL $L_i = \{\mathtt{r}_i\mathtt{simI}^n : 1 \leq n \leq 2\} \cup \{\mathtt{simU}^m\mathtt{r}_i^1 : 1 \leq m \leq 2\}$.

4 Experiments

In order to validate the proposed approach two versions of the MovieLens[2] dataset has been used. The MovieLens data sets were collected by the GroupLens Research Project at the University of Minnesota. The first version called MovieLens 100K consists of 100,000 ratings (1-5) from 943 users on 1682 movies, where each user has rated at least 20 movies and there are simple demographic info for the users (age, gender, occupation, zip). The data was collected through the MovieLens web site during the seven-month period from September 19th, 1997 through April 22nd, 1998. The second version called MovieLens 1M consists

[2] http://www.grouplens.org/

of 1,000,209 anonymous ratings of approximately 3,900 movies made by 6,040 MovieLens users. In this paper we used the ratings only without considering the demographic information.

MovieLens 100K dataset is divided in 5 fold, where each fold present a training data (80000 ratings) and a test data (20000 ratings), while MovieLens 1M is divided in 10 fold. For each training/testing fold the validation procedure follows the following steps:

1. creating the uncertain graph from the training ratings data set as reported Section 3;
2. defining a context free language corresponding to a specific CF task;
3. testing the ratings reported in the testing data set $\mathcal{T}$ by computing, for each pair $(u, i) \in \mathcal{T}$ the predicted rating as in (9) and comparing the result with the true prediction reported in $\mathcal{T}$.

In this particular dataset we have a uncertain graph with nodes labeled as `user` or as `film`. There are edges between two `film` nodes labeled as `simF`, and there are edges with label `simU` between two `user` nodes. These edges are added using the procedure presented in the previous section, where we set the parameter $n = 30$, indicating that an user or a film is connected, respectively, to 30 most similar users, resp. films . Finally, for each rating $(u, i, r_{ui} = k)$ belonging to the training set there is an edge between the user u and the film i whose label is $\mathtt{r}_k$. The goal is to predict the correct rating for each instance belonging to the testing set $\mathcal{T}$. The predicted rating has been computed using a Monte Carlo approach by sampling 100 certain graphs and adopting the function reported in (9).

The accuracy of the proposed framework has been evaluated according to the *mean absolute error* (MAE) a most commonly applied evaluation metric for CF rating predictions. Assuming N computed rating predictions:

$$MAE = \frac{1}{N} \sum_{i=1}^{N} |\widehat{r_{ui}} - r_{ui}|. \tag{10}$$

4.1 Results

In order to evaluate the framework, we proposed to query the paths belonging to the context free languages reported in Table 1. The first language constrained simple paths L_1 reported in Table 1 corresponds to solve a user-based CF problem, while the second language L_2 gives us the possibility to simulate a item-based CF approach. As we can see from Table 2 results improve when we go from a user-based approach to a item-based one.

Then we try to build a basic hybrid system by combining both the languages L_1 and L_2 into the language L_3. Now, as we can see in Table 2 results are better than that obtained when we used a single language only. Then, we propose to extend the basic languages L_1 and L_2 in order to consider a neighbourhood with many nested levels. In particular, instead of considering the direct neighbours

$L_1 = \{\mathtt{simU}^1\mathtt{r}_k^1\}$
$L_2 = \{\mathtt{r}_k^1\mathtt{simF}^1\}$
$L_3 = \{\mathtt{simU}^1\mathtt{r}_k^1\} \cup \{\mathtt{r}_k^1\mathtt{simF}^1\}$
$L_4 = \{\mathtt{simU}^n\mathtt{r}_k^1 : 1 \leq n \leq 2\}$
$L_5 = \{\mathtt{r}_k^1\mathtt{simF}^n : 1 \leq n \leq 2\}$
$L_6 = \{\mathtt{r}_k^1\mathtt{simF}^n : 1 \leq n \leq 3\}$
$L_7 = \{\mathtt{simU}^n\mathtt{r}_k^1 : 1 \leq n \leq 2\} \cup \{\mathtt{r}_k^1\mathtt{simF}^n : 1 \leq n \leq 2\}$
$L_8 = \{\mathtt{r}_k^1\mathtt{simF}^n : 1 \leq n \leq 4\}$

Table 1. Language constrained simple paths used for the MovieLens dataset.

only, we inspect the uncertain graph following a path with a maximum length of two edges, labeled respectively as `simU` for the language L_4 and `simF` for the language L_5. Their corresponding results are better than that obtained with the basic language L_1 and L_2 thus proving the validity of the approach. Language L_6 extends language L_5 in order to inspect the uncertain graph following a path with a maximum length of three edges by obtaining better results than others languages.

Finally, the language L_7 combines both the user-based and item-based approach, and the large neighbourhood explored with paths whose length is greater than one. As we can see, this language is the best among all the others in providing a good MAE value.

	Path						
Fold	L_1	L_2	L_3	L_4	L_5	L_6	L_7
1	0.9419	0.8458	0.8228	0.8661	0.7928	0.7837	0.7663
2	0.9337	0.8366	0.8119	0.8513	0.7777	0.7800	0.7670
3	0.9189	0.8141	0.8063	0.8505	0.7739	0.7700	0.7584
4	0.9275	0.8273	0.8096	0.8608	0.7784	0.7724	0.7678
5	0.9528	0.8421	0.8312	0.8637	0.7824	0.7754	0.7785
Mean	0.9349	0.8332	0.8164	0.8585	0.7810	0.7763	0.7676

Table 2. MAE error on MovieLens 100K adopting different path type

Table 3 shows the results on the MovieLens 1M dataset, using a 10-fold cross-validation, comparing the proposed framework with respect to a neighborhood-based recommendation method [4] adopting as similarity weight the Mean Squared Difference (MSD), the Spearman Rank Correlation (SRC) or the Pearson Correlation (PC). In this case we adopted another language, L_8, that extends the neighborhood of the explored graph. As we can see, the obtained results adopting our system are better than those obtained with the neighborhood-based approach. Furthermore, more the portion of the explored graph is considered, adopting the languages L_2, L_5, L_6 and L_8, and more is the predictive accuracy reached by the system.

Method	MAE
MSD	0.7602
SRC	0.7529
PC	0.7518
L_2	0.7916
L_5	0.7381
L_6	0.7293
L_8	0.7198

Table 3. MSE error on MovieLens 1M

5 Related works

Given a snapshot of a graph (network), the goal we are dealing with is to accurately predict edges that could be added to the network in future, sometime called *link prediction problem* [5]. There are a lot of application where link prediction can be used such as identifying the structure of a criminal network, overcoming the data-sparsity problem in recommender systems using collaborative filtering [25], analyzing users navigation history to generate users tools that increase navigational efficiency [26]. A problem close to link prediction is link completion [8]. The data, collected from the real life sources, is usually noisy and might contain gaps, i.e. links may be incomplete, containing one or more unknown members. The problem of link completion addresses the task of determining the missing member given a partial link. This question is similar to those found in the collaborative filtering domain [2]. The link prediction problem is also related to the problem of inferring missing links from an observed network: in a number of domains, one constructs a network of interactions based on observable data and then tries to infer additional links that, while not directly visible, are likely to exist [7, 18, 22].

All these methods assign a connection weight $score(x, y)$ or a similarity $s(x, y)$ to pairs of nodes x, y, based on the input graph, and then produce a ranked list in decreasing order of $s(x, y)$. This approach may be viewed as computing a measure of proximity or a similarity between nodes. The most basic approach to compute this ranked list could be that to rank pairs x, y by the length of their shortest path in the network G . Such a measure follows the notion that collaboration networks are *small worlds*, in which individuals are related through short chains [17]. Shortest path between two nodes defines the minimum number of edges connecting them. If there is no such connecting path then, the value of this attribute is taken as infinite.

Other methods try to compute the similarity between two nodes by looking their corresponding neighborhoods. Given a node x, let $N(x)$ be the set of neighbours of x in a graph G. Given two nodes x and y, there are several approaches that follow the natural intuition that if the set of neighbours $N(x)$ and $N(y)$ have a large overlapping then the node x and the node y should be very similar.

Common neighbours measure the number of neighbors that node x and node y have in common, in particular $s(x, y) = |N(x) \cap N(y)|$. Newman in [16] shows a correlation between the number of common neighbours of x and y at the time t, and the probability they will be similar in the future.

Jaccard's coefficient, used in information retrieval, measures the probability that both x and y have a feature f in common, for a randomly selected feature f. Using neighbours we can compute this as follow $s(x, y) = |N(x) \cap N(y)|/|N(x) \cup N(y)|$. [1] considers the similarity problem between two entities as $s(x, y) = \sum_{z \in N(x) \cap N(y)} \frac{1}{log|N(z)|}$ where z is a set of features shared both by x and y. Finally, *preferential attachment* is based on empirical evidence that the probability of x and y being connected is correlated with the product of the number of connections of x and y ($N(x)$ and $N(y)$). The measure is computed as $s(x, y) = |N(x)| \cdot |N(y)|$.

Other methods are based on ensemble of paths. Katz [13] defines a similarity measure that directly sums over a collection of paths, exponentially damped by length in order to count short paths more heavily. This leads to the measure $s(x, y) = \sum_{l=1}^{\infty} \beta^l \cdot |paths_{x,y}^{\langle l \rangle}|$ where $paths_{x,y}^{\langle l \rangle}$ is the set of all lengh-l paths from x to y. There exists two variants of the Katz measure: unweighted, in witch $paths_{x,y}^{\langle 1 \rangle} = 1$ if x and y have collaborated and 0 otherwise, and weighted, in witch $paths_{x,y}^{\langle 1 \rangle}$ is the number of times that x and y have collaborated.

Another method uses random walks on the graph G [23], where starting from a node x, the selection of next node to visit is done by choosing among the neighbors of x at random. Using this approach it is possible to compute the hitting time $H_{x,y}$ as the expected number of steps required for a random walk starting at x to reach y. SimRank [10] supposes that two nodes are similar to the extent that they are joined to similar neighbors. In particular $s(x, y) = \gamma \cdot \frac{\sum_{a \in N(x)} \sum_{b \in N(y)} s(a,b)}{|N(x)| \cdot |N(y)|}$ for some $\gamma \in [0, 1]$.

All the methods described above consider the space of representation as a graph with nodes of the network indicating the objects of the world and edges with a numeric value that indicates their weight. Over the last few years uncertain graphs have become an important research topic [19, 27, 28]. In these graphs each edge is associated with an edge existence probability that quantifies the likelihood that the edge exists in the graphs. Using this representation it is possible to adopt the *possible world* semantics to model it. One of main issue in uncertain graphs is how to compute the connectivity of the network. The *network reliability problem* [3] is a generalization of *pairwise reachability*, in which the goal is to determine the probability that all pairs of nodes are reachable from one another. Unlike a deterministic graph in which the reachability function is a binary function indicating whether or not there is a path that connects the two provided vertices, in the case of the reachability on uncertain graphs the function assumes probabilistic values. In [19], the authors provide a list of alternative shortest-path distance measures for uncertain graphs in order to discover the k closest vertices to a given vertex. Another work [12] try to deal with the concept of $x - y$ distance-constraint reachability problem. In particular, given two vertices x and y, they try to solve the problem of computing the probability that the distance from x to y is less than or equal to a user-defined threshold. In order to solve this problem, they proposed an exact algorithm and two reachability estimators based on probability sampling.

6 Conclusions

In this paper a framework based on uncertain graphs able to deal with collaborative filtering problems has been presented. The evaluation of the proposed approach has been reported by applying it to a real world dataset and proving its validity in solving simple and complex collaborative filtering tasks. As future development we will conduct further experiments in order to accurately validate the framework. We will study how the size of the neighbourhood of each node, during the graph construction phase, could influence the quality of the predictions.

References

1. Adamic, L.A., Adar, E.: Friends and neighbors on the web. Social Networks 25(3), 211–230 (2003)
2. Breese, J.S., Heckerman, D., Kadie, C.: Empirical analysis of predictive algorithms for collaborative filtering. In: Proceedings of the14th Annual Conference on Uncertainty in Artificial Intelligence (UAI98). pp. 43–52 (1998)
3. Colbourn, C.J.: The Combinatorics of Network Reliability. Oxford University Press (1987)
4. Desrosiers, C., Karypis, G.: A comprehensive survey of neighborhood-based recommendation methods. In: Ricci, F., Rokach, L., Shapira, B., Kantor, P.B. (eds.) Recommender Systems Handbook, pp. 107–144. Springer (2011)
5. Getoor, L., Diehl, C.P.: Link mining: a survey. SIGKDD Explorations 7(2), 3–12 (2005)
6. Getoor, L., Taskar, B.: Introduction to Statistical Relational Learning (Adaptive Computation and Machine Learning). The MIT Press (2007)
7. Goldberg, D.S., Roth, F.P.: Assessing experimentally derived interactions in a small world. Proceedings of the National Academy of Sciences 100(8), 4372–4376 (2003)
8. Goldenberg, A., Kubica, J., Komarek, P., Moore, A., Schneider, J.: A comparison of statistical and machine learning algorithms on the task of link completion. In: KDD Workshop on Link Analysis for Detecting Complex Behavior (2003)
9. Hintsanen, P., Toivonen, H.: Finding reliable subgraphs from large probabilistic graphs. Data Mining and Knowledge Discovery 17(1), 3–23 (2008)
10. Jeh, G., Widom, J.: Simrank: a measure of structural-context similarity. In: Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining. pp. 538–543. KDD '02, ACM (2002)
11. Jin, R., Liu, L., Aggarwal, C.C.: Discovering highly reliable subgraphs in uncertain graphs. In: Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining. pp. 992–1000. KDD '11, ACM, New York, NY, USA (2011)
12. Jin, R., Liu, L., Ding, B., Wang, H.: Distance-constraint reachability computation in uncertain graphs. Proc. VLDB Endow. 4, 551–562 (2011)

13. Katz, L.: A new status index derived from sociometric analysis. Psychometrika 18(1), 39–43 (1953)
14. Koller, D., Friedman, N.: Probabilistic Graphical Models: Principles and Techniques. MIT Press (2009)
15. Koren, Y., Bell, R.M.: Advances in collaborative filtering. In: Ricci, F., Rokach, L., Shapira, B., Kantor, P.B. (eds.) Recommender Systems Handbook, pp. 145–186. Springer (2011)
16. Newman, M.E.J.: Clustering and preferential attachment in growing networks. Phys. Rev. E 64 (2001)
17. Newman, M.E.J.: The structure of scientific collaboration networks. Proceedings of the National Academy of Sciences of the United States of America 98(2), 404–409 (2001)
18. Popescul, A., Ungar, L.H.: Statistical relational learning for link prediction. In: IJCAI03 Workshop on Learning Statistical Models from Relational Data (2003)
19. Potamias, M., Bonchi, F., Gionis, A., Kollios, G.: k-nearest neighbors in uncertain graphs. Proc. VLDB Endow. 3, 997–1008 (2010)
20. Sachan, M., Ichise, R.: Using semantic information to improve link prediction results in network datasets. IACSIT International Journal of Engineering and Technology 2(4), 71–76 (2010)
21. Taranto, C., Di Mauro, N., Esposito, F.: Probabilistic inference over image networks. Italian Research Conference on Digital Libraries 2011 CCIS 249, 1–13 (2011)
22. Taskar, B., Wong, M.F., Abbeel, P., Koller, D.: Link prediction in relational data. In: in Neural Information Processing Systems (2003)
23. von Luxburg, U., Radl, A., Hein, M.: Hitting and commute times in large graphs are often misleading. CORR (2011)
24. Witsenburg, T., Blockeel, H.: Improving the accuracy of similarity measures by using link information. In: Kryszkiewicz, M., Rybinski, H., Skowron, A., Ras, Z.W. (eds.) ISMIS. Lecture Notes in Computer Science, vol. 6804, pp. 501–512. Springer (2011)
25. Zan, H., Xin, L., Hsinchun, C.: Link prediction approach to collaborative filtering. In: Proceedings of the 5th ACM/IEEE-CS joint conference on Digital libraries. pp. 141–142. ACM Press (2005)
26. Zhu, J.: Mining web site link structures for adaptive web site navigation and search. In: PhD thesis. University of Ulster (2003)
27. Zou, Z., Gao, H., Li, J.: Discovering frequent subgraphs over uncertain graph databases under probabilistic semantics. In: Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining. pp. 633–642. ACM (2010)
28. Zou, Z., Li, J., Gao, H., Zhang, S.: Finding top-k maximal cliques in an uncertain graph. International Conference on Data Engineering pp. 649–652 (2010)
29. Zou, Z., Li, J., Gao, H., Zhang, S.: Mining frequent subgraph patterns from uncertain graph data. IEEE Transactions on Knowledge and Data Engineering 22, 1203–1218 (2010)