Multi-Relational Matrix Factorization using Bayesian Personalized Ranking for Social Network Data

Artus Krohn-Grimberghe, Lucas Drumond, Christoph Freudenthaler, and
Lars Schmidt-Thieme
Information Systems and Machine Learning Lab, University of Hildesheim, Germany
{artus, ldrumond, freudenthaler, schmidt-thieme}@ismll.de

ABSTRACT

A key element of the social networks on the internet such as Facebook and Flickr is that they encourage users to create connections between themselves, other users and objects.

One important task that has been approached in the literature that deals with such data is to use social graphs to predict user behavior (e.g. joining a group of interest). More specifically, we study the cold-start problem, where users only participate in some relations, which we will call social relations, but not in the relation on which the predictions are made, which we will refer to as target relations.

We propose a formalization of the problem and a principled approach to it based on multi-relational factorization techniques. Furthermore, we derive a principled feature extraction scheme from the social data to extract predictors for a classifier on the target relation. Experiments conducted on real world datasets show that our approach outperforms current methods.

Categories and Subject Descriptors

H.3.3 [**Information Search and Retrieval**]: Information filtering; Search process

General Terms

Algorithms

Keywords

Recommender Systems, Cold-Start, Multi-Relational Learning, Social Network, Matrix Factorization, Item Prediction, Item Recommendation, Ranking, Joint Factorization

1. INTRODUCTION

Social media applications allow users to connect to each other and to interact with items of interest such as songs, videos, web pages, news, groups and the like. Users generally tend to connect to other users due to some commonalities they share, often reflected in similar interests. One of the most important applications in this domain is to provide each user with a list of items ranked according to the likelihood that they could be interesting to him or her. Moreover, in many real-life applications it may be the case that only social information about certain users is available while interaction data between the items and those users has not yet been observed. In this paper we address how to make use of social information on users for item prediction, especially when no collaborative information on the test users is available.

This task is exactly the one demanded of recommender systems, which are designed to predict whether and how much a user will like an item and, in some cases, whether she will buy, watch, use or evaluate it. Since users usually tend to connect with others due to common interests and preferences, the connections between them in social networks may contain useful information for predicting their preferences. In [3], it has been shown that such information may indeed help to improve the quality of rating prediction (i.e. estimating the rating that a user will give to an item), especially for cold start users. In this work we restrict ourselves to the related, but yet distinct task of item prediction, where we present users a ranked list of items that are likely to suit their interests. More specifically, we study the problem of recommending items to users where only social information is available, but no previous direct interactions with items has been recorded, i.e. the cold-start problem.

This problem has been approached by Tang and Liu [14][15], who extract user features from social networks and use them as predictors of a multi-label classifier that predicts whether each of the items is interesting to a given user.

One of the most prominent approaches for recommender systems is matrix factorization. Since social media data is usually presented as two or more matrices (the social and the target relation), in order to apply factorization approaches to social data, one has to resort to their multi-relational variants [12]. However, special care is necessary if, as in the cold-start scenario analyzed here, no information about the test users is given on the target relation.

Approaching this problem with multi-relational matrix factorization can be seen as a variant of [14], where both the feature extraction and the item prediction are performed by sparse factorization models. However, one crucial difference is that, whereas Tang and Liu perform the two steps sequentially, plain applications of multi-relational factorization methods simply simultaneously factorize all the matrices, which means that both feature extraction and item pre-

WSDM'12, February 8–12, 2012, Seattle, Washington, USA.

diction are performed jointly. It is however unclear whether the sequential or the joint approach is more appropriate.

Since the task at hand is to rank items according to the preferences of a given user, it makes sense to optimize the models for a suitable criterion. The Bayesian personalized ranking (BPR) [9] has been shown to provide strong results in many item recommendation tasks [9][10]. We argue that it is a more suitable optimization criterion for the task analyzed and present an extension of it to the multi-relational case.

The main contributions of this work are:

1. We formalize the problem of recommendation in social networks as a multi-relational learning problem and present it in a multi-relational factorization framework;
2. We extend the Bayesian personalized ranking (BPR) framework to the multi-relational case and show how it can be adapted to optimize for different evaluation measures;
3. We provide empirical evidence that factorizing the relations jointly is at least as good as the sequential approach, yielding most of the times better results;
4. Our experiments show that our multi-relational BPR approach outperforms state-of-the-art approaches for recommendation in social networks and for cold-start recommendations in real world datasets.

2. RELATED WORK

The cold-start problem [11] has a long history in the area of recommender systems and several approaches both attribute-based and multi-relational have been proposed.

Tang and Liu [14, 15] offer two bi-relational approaches. Both methods can handle the cold-start problem when no information is present in the target relation for the entity receiving the predictions. In their setting, the first relation comprises two different entities, users and labels, whereas the second relation is an interaction between the users (friendship). They predict whether users should be connected to labels, and their methods treat this problem as a multi-label relational classification task, with the links between users and labels being the class labels. In both algorithms, the training data is derived from the friendship relation during a unique pre-processing step and fed into a SVM for learning. For modularity maximization (ModMax) [14], the training data are first k eigenvectors of the modularity matrix [8] of the friendship relation. For edge clustering [15], a sparse k-means clustering of the edges between the entities of the friendship relation is used as a training feature. The labels of the target relation are treated independently of each other; the ones with the highest scores are suggested to the user.

Gantner et al. [1] focus on cold-start item recommendations. Their work is different from the work of Tang and Liu insofar as they do not use auxiliary relational data but attributes instead. Furthermore, in their work the attributes belong to the items, but after a transposition of the user-item-relation this difference is removed. The method proposed by Gantner et al.—BPR-map—starts with a factorization of the sparse user-item matrix and subsequently learns a (linear) mapping on the user features from the factorization and the auxiliary user-attribute matrix. The attributes serve as predictors for the user features, making the model able to offer predictions for non-observed users; the user features predicted are in turn used to reconstruct the target relation.

Ma et al. [6, 5] exploit a friendship relation for rating prediction which is shown to be effective especially for users with few ratings. They do not evaluate full-on cold-start situations, though. SoRec [6] proposes a joint multi-relational factorization of both the user-item relation and the user-user relation. As a pre-processing step, they require a normalization of the user-user relation. STE (Social Trust Ensemble) [5] is a weighted combination of a factorization on the user-item relation and neighborhood score, derived from the sum of the dot products of the item and the friends.

Jamali and Ester [3] extend the work of Ma et al. on improving rating prediction by exploiting a social relation. Their method, SocialMF, employs an additional regularizer for a standard matrix factorization. The regularization term forces a user's features learnt during the factorization of the target relation to be similar to her friends' features. This is shown to improve rating prediction performance at least for users with few ratings. On the other hand, this method cannot work for users with no ratings for which only social information is available.

Singh and Gordon [12, 13] propose a framework for multi-relational factorization models. They subsume models on any number of relations as long as their loss function is a twice differentiable decomposable loss (including Bregman Divergences). In their work, they address both rating prediction and item recommendation.

Zhang et al [17] contribute an extension of the probabilistic matrix factorization to the multi-relational case. They propose sharing only the user features among different relations via a covariance matrix learnt from the data. Additionally, the authors review different link functions in order to transform the data within the relations, but their loss function is always the square loss. Interestingly, they propose breaking up the item dimension into several smaller dimensions (given additional information).

Yang et al. [16] model item prediction in a bi-relational social network as a problem that consists of multi-relational data and/or attribute data. They evaluate on a ranking measure but still use a logistic loss or square loss derivatives for learning. They also show that taking the non-negative structure of the data into account is rewarded with better prediction accuracy.

We will review the differences to our approach in section 3.2 after formulating the problem.

3. COLD-START ITEM RECOMMENDATION IN SOCIAL NETWORKS

3.1 Problem Formulation

This problem can be seen as a specific case of a more general relational learning setting. In the general case, there is a set of entity types $\mathcal{E} := \{E_1, E_2, ..., E_{|\mathcal{E}|}\}$. Each entity type is a set of entity instances $E_i := \{e_i^{(1)}, ..., e_i^{(|E_i|)}\}$. Entity types are related through a set of binary relations between them $\mathcal{R} = \{R_1, R_2, ..., R_{|\mathcal{R}|}\}$, where $R_m \subseteq E_i \times E_j$. For almost all applications, there is only one relation for which the model employed should make a prediction. Consequently, we call this relation the *target relation* and denote it by Y. The

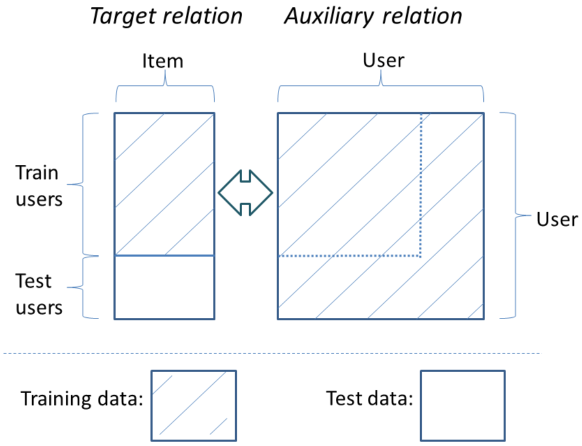

Figure 1: An overview of the multi-relational cold-start setting in social networks.

other relations are called *auxiliary relations* $\mathcal{A} := \mathcal{R} \setminus \{Y\}$ which contain information that is used to improve or enable the prediction of the target values. In the specific case considered in this work, there are two entities, $\mathcal{E} = \{U, I\}$, where U is the set of users and I is the set of items. The target relation represents information about interaction between users and items is represented as a set of pairs $Y \subseteq U \times I$. Finally, auxiliary information is available in the form a social relation $A \subseteq U \times U$, which represents the existence of a relationship between a pair of users, e.g. the users are friends. The items could be videos, for instance, blog posts, or communities that the users can join; the edges of A may represent friendship relationships among users; and the implicit feedback Y means that users have watched videos, read posts or joined communities, thus allowing one to assume that they are interested in them. Figure 1 depicts the general setting.

The task now is to derive a ranked list of items, sorted according to the likelihood that a given user will be interested in each of them. More formally, given a user u, the task is to derive a total order $>_u$ of all items I. In line with [9], this can be achieved by learning a scoring function $\hat{y} : U \times I \to \mathbb{R}$ capable of generating predictions for users U^{test} and sorting the items according to their scores.

In a full cold-start problem, the task gets increasingly intricate due to additional requirements on the training data: while auxiliary information in form of a social graph $G := (U, A)$ is still available for all users, the training data $Y \subseteq U^{train} \times I$ does not contain any information about the users U^{test}. Mitigating cold-start effects and the incorporation of information beyond the target relation are among the hardest problems in recommender systems nowadays. It should also be noted that besides a full cold-start, our approach also handles slow start situations when the first few user-item interactions become available as well as regular recommender scenarios.

3.2 State-of-the-Art Approaches

Table 1 lists three state-of-the-art approaches capable of cold-start item recommendations, namely *ModMax* [14], *edge clustering* [15], and *BPR-map* [1]. All of them follow a two step learning model. *ModMax* and *edge clustering* first extract user features from the social graph (the auxiliary relation) and then feed those features into a classifier that predicts whether each item will be interesting to a given user. Both approaches use a one-vs-rest linear SVM as a classifier, but they use different feature extraction approaches. *ModMax* works with the top-k eigenvectors of the modularity matrix of the social graph. *edge clustering* first clusters the edges in the *edge* × *user* representation of the auxiliary matrix, and then uses the resulting relation of users and their cluster membership as predictors. *BPR-map* differs from this in that it starts with learning from the target relation, first extracting latent user features and item features from the target relation using a factorization method. Subsequently, it learns a mapping between the user features as labels and the rows of the auxiliary matrix as predictors.

The predictions for the target relation are made by a (linear multi-label) SVM for *ModMax* and *edge clustering* and a (linear) BPR regression for *BPR-map* using the features learned from and the auxiliary relation.

The cited methods have in common that they learn their models from the target and auxiliary relations separately in a sequential way. In the next section, we propose to use an approach that is able to learn from both relations simultaneously and show how it can be generalized to the sequential case.

4. MULTI RELATIONAL LEARNING WITH BPR

Following the formalization in section 3, the recommendation task described here can be viewed as an instance of a multi-relational learning problem which consists of two entities, the users and the items, as well as of at least two relations, namely the auxiliary relation A (represented by the social graph) and the target relation Y, represented by the user-item interaction (figure 1).

Multi-relational matrix factorization models [12] provide a framework for jointly factorizing multiple relations. In this section we cast the problem of ranking in social networks as a multi-relational factorization problem and propose a model and learning algorithms for it that address the issues discussed here.

4.1 Ranking in Social Networks–A Multi-Relational Factorization Problem

Matrix factorization models [4] represent a matrix as the product of two lower-rank matrices. [1] The goal is, given a partially observed matrix $\mathbf{R}$, to find two matrices $\mathbf{E_1}$ and $\mathbf{E_2}$ such that

$$\underset{\mathbf{E_1},\mathbf{E_2}}{\operatorname{argmin}}\, L(\mathbf{R}, \mathbf{E_1}\mathbf{E_2}^T) + Reg(\mathbf{E_1}, \mathbf{E_2}) \qquad (1)$$

where L is a loss function that measures the reconstruction error of R given $\mathbf{E_1}$ and $\mathbf{E_2}$ and Reg a regularization term to prevent overfitting.

A matrix can be viewed as a relation between two entity types. When we follow equation (1), we may represent each entity type E_i by a matrix $\boldsymbol{E_i} \in \mathbb{R}^{|E_i| \times k}$ where the rows

[1] Matrices are denoted in bold face to avoid confusion with entity type and relation identifiers.

	ModMax	**EdgeClustering**	**BPR-map**
Target relation treatment	dense binary	dense binary	sparse unary
Model on target relation	N/A	N/A	matrix factorization
Aux. relation treatment	dense modularity matrix	dense graph	dense matrix
Model on aux. relation	PCA	k-means	N/A
Meta Model	eigenvectors as predictors to multi-class SVM with labels as classes	see ModMax	auxiliary matrix as predictor to a regression on latent user features

Table 1: Baseline methods for the cold-start item prediction task.

are latent feature vectors of the entity type instances and k is the number of latent features, chosen by model selection. The two entities participating in a relation R_i are denoted by E_{R_i1} and E_{R_i2}. Each relation R is represented by a matrix $\mathbf{R}^{|E_{R1}|\times|E_{R2}|}$ where each entry is given by

$$\mathbf{R}(e^{(i)}_{E_{R1}}, e^{(j)}_{E_{R2}}) := \begin{cases} 1 & \text{if } (e^{(i)}_{E_{R1}}, e^{(j)}_{E_{R2}}) \in R, \\ \text{unobserved} & \text{else} \end{cases}$$

Thus the factorization of relation R_i is given by $\mathbf{R_i} \approx \mathbf{E_{R_i1}}\mathbf{E_{R_i2}}^T$. To make notation simpler, we define the set of all model parameters $\Theta := \{\mathbf{E_1}, ...\mathbf{E}_{|\mathcal{E}|}\}$.

Singh and Gordon [12] generalized the problem in equation (1) in cases where there is more than one relation. The problem is now to find a set of matrices Θ that minimizes the sum of the losses on all relations:

$$\underset{\Theta}{\operatorname{argmin}} \sum_{R\in\mathcal{R}} \alpha_R L(\mathbf{R}, \mathbf{E_{R1}}\mathbf{E_{R2}}^T) + Reg(\Theta)$$

where α_R is an application specific weight for the loss of relation $\mathbf{R}$ and weights are normalized such that:

$$\sum_{R\in\mathcal{R}} \alpha_R = 1$$

In the specific case considered in this work, there are two entities, i.e $\mathcal{E} = \{U, I\}$ and two relations, $R = \{Y, A\}$. Consequently, the specific loss to the problem approached in this paper is the following:

$$\underset{(\mathbf{U},\mathbf{I})}{\operatorname{argmin}} \alpha_Y L(\mathbf{Y}, \mathbf{U}\mathbf{I}^T) + \alpha_A L(\mathbf{A}, \mathbf{U}\mathbf{U}^T) + Reg(\mathbf{U}, \mathbf{I}) \quad (2)$$

The advantage of formalizing the problem this way, in comparison to the multi-label classification view in [14, 15] is twofold. Firstly, it allows for a more straightforward way to incorporate new entities and relations into the model, as it would just require adding the loss for the new relation and the regularization terms for the new entities on the general loss. Secondly, the models in [14, 15] are able to generate predictions for one relation alone. With the formalization proposed here, recommendations can be made for all entities on all relations, without the need to retrain.

4.2 Multi-Relational Factorization with BPR

Since we are dealing with a ranking problem, it makes sense to use a loss function that is optimized for ranking. The Bayesian personalized ranking optimization criterion (BPR-Opt) [9] has been shown to be a suitable criterion for ranking in general and item prediction in particular. Another reason that makes BPR suitable for the task of ranking in social networks is that it is tailored to data where only positive feedback is available (unary data). In fact, for the task at hand the Y relation is a typical representative of positive feedback only recommender data. Thus, $(u, i) \in Y$ can be interpreted as a positive feedback given by the user u about item i. The opposite, however, $(u, j) \notin Y$, only tells us that the user has not yet interacted with item j. For most applications, it would be unjustified to treat all those j as being equally valued by the respective user, e.g., assign all items that he is unaware of and all those items explicitly ignored the same score (i.e. 0). The same holds true for the social relation: the fact that a connection between two users is not observed in the data does not necessarily imply that there is no connection in the real world.

BPR-Opt makes use of the assumption that, for a given user u, an item i where $(u, i) \in Y$ should be ranked higher than an item j where $(u, j) \notin Y$. For convenience, we define for each relation R, a set D_R as being the set of triples $D_R := \{(u, i, j)|(u, i) \in R \wedge (u, j) \notin R\}$. Throughout this section, we stick to the (u, i, j) notation, although these values do not necessarily denote users and items as in the example above; u nonetheless denotes an instance of the first entity involved in the relation, and i and j instances of the second entity.

If $\hat{r}(u, i)$ is the predicted score for the (u, i) pair on relation R and $\hat{x}^R_{u,i,j} = \hat{r}(u, i) - \hat{r}(u, j)$, BPR-Opt can be defined as in equation 3.

$$\text{BPR-Opt}(\mathbf{R}, \mathbf{E_{R1}}\mathbf{E_{R2}}^T) := \sum_{(u,i,j)\in D_R} \ln \sigma(\hat{x}^R_{u,i,j}) \quad (3)$$

where $\sigma(x) := \frac{1}{1+e^{-x}}$ is the sigmoid logistic function. It is worth noting that BPR-Opt should be maximized. Optimizing for BPR-Opt is a smoothed version of optimizing for the well-known ranking measure Area under the ROC Curve (AUC). For a proof and more details about BPR-Opt, please see [9].

Having chosen a loss function, an appropriate regularization term still needs to be chosen. In this case, we use L_2 regularization, since the L_2-regularization terms are differentiable, allowing us to apply gradient-based methods.

In the case of item recommendations for multi-relational cold-starts, there is more than one relation involved. Consequently, an extension of BPR-Opt for the multi-relational case is necessary. Following the framework of [12, 13], we propose the following extension for multi-relational ranking in social networks with Bayesian personalized ranking:

```
procedure LearnMR-BPR($D, \mathcal{R}, \mathcal{E}$)
    initialize all $\mathbf{E} \in \Theta$
    repeat
        for $R \in \mathcal{R}$ do
            draw $(u, i, j)$ from $D_R$
            $\Theta \leftarrow \Theta + \mu \left( \alpha_R \frac{e^{-\hat{x}^R_{uij}}}{1+e^{-\hat{x}^R_{uij}}} \cdot \frac{\partial}{\partial \Theta} \hat{x}^R_{uij} + \lambda_\Theta \cdot \Theta \right)$
        end for
    until convergence
    return $\Theta$
end procedure
```

Figure 2: Multi-relational extension of the LearnBPR algorithm proposed in [9] with learning rate μ and regularization λ_Θ.

```
procedure LearnMR-BPR-Pivotization($D, \mathcal{R}, \mathcal{E}$)
    initialize all $\mathbf{E} \in \Theta$
    repeat
        for $R \in \mathcal{R}$ do
            draw $(u, i, j)$ from $D_R$
            $\Theta \leftarrow \Theta + \mu \left( \alpha_R \frac{e^{-\hat{x}^R_{uij}}}{1+e^{-\hat{x}^R_{uij}}} \cdot \frac{\partial}{\partial \Theta} \hat{x}^R_{uij} + \lambda_\Theta \cdot \Theta \right)$
            draw $(i, u, v)$ from $D_{R^T}$
            $\Theta \leftarrow \Theta + \mu \left( \alpha_R \frac{e^{-\hat{x}^R_{iuv}}}{1+e^{-\hat{x}^R_{iuv}}} \cdot \frac{\partial}{\partial \Theta} \hat{x}^R_{iuv} + \lambda_\Theta \cdot \Theta \right)$
        end for
    until convergence
    return $\Theta$
end procedure
```

Figure 3: Optimizing models for BPR with bootstrapping based stochastic gradient descent. With learning rate α and regularization λ_Θ.

$$\text{MR-BPR}(\mathcal{R}, \Theta) = \sum_{R \in \mathcal{R}} \alpha_R \, \text{BPR-Opt}(\mathbf{R}, \mathbf{E_{R1}} \mathbf{E_{R2}}^T) + \sum_{E \in \mathcal{E}} \lambda_E ||\mathbf{E}||^2 \tag{4}$$

where λ_E is the regularization constant for entity E.

The optimization problem described in equation (4) can be solved using a stochastic gradient descent algorithm. In [9], just such an algorithm, called LearnBPR, is proposed for the unirelational case, i.e. when there is only one matrix to be factorized. Figure 2 provides an extension to LearnBPR that considers the multi-relational case. At each iteration, one sample is uniformly drawn from D_R for each relation R and the parameters are updated in the opposite direction of the loss function's gradient at the point sampled. The derivative of the loss function presented in equation (4) is:

$$\frac{\partial \, \text{MR-BPR}(\mathbf{R}, u, i, j)}{\partial \Theta} = \alpha_R \frac{-e^{-\hat{x}^R_{uij}}}{1 + e^{-\hat{x}^R_{uij}}} \cdot \frac{\partial}{\partial \Theta} \hat{x}^R_{uij} + \lambda_\Theta \cdot \Theta$$

The $\hat{x}^R_{uij}$ partial derivatives are:

$$\frac{\partial}{\partial \Theta} \hat{x}^R_{uij} = \begin{cases} (\mathbf{i}_f - \mathbf{i}_f) & \text{if } \theta = \mathbf{u}_f, \\ \mathbf{u}_f & \text{if } \theta = \mathbf{i}_f, \\ -\mathbf{u}_f & \text{if } \theta = \mathbf{j}_f, \\ 0 & \text{else} \end{cases}$$

where f denotes the f-th latent feature of the entity instance in question. The parameters are optimized using equation (4).

4.3 Pivotization

The problem of item recommendation can be viewed from two perspectives:

1. Given a user, which items could be interesting to her?
2. Given an item, which users could be interested in it?

The first perspective is the one taken into account for BPR optimization as presented before. This is reflected by the samples D_Y, which are composed of a user u and two items i and j, one a positive example, one a negative. In order to train a BPR model for the second perspective, $\mathbf{Y}$ can be transposed and samples drawn from $D_{Y^T} := \{(i, u, v) | i \in I \wedge u, v \in U \wedge (u, i) \in Y \wedge (v, i) \notin Y\}$. This is the perspective adopted in [14] and [15] by one of the evaluation criteria.

The question is: which one of the perspectives is the most appropriate? Clearly, this depends on the measure employed to evaluate the algorithm. In a classical recommender scenario, where the ranking performance is evaluated per user and then averaged, it makes sense to resort to the first perspective. On the other hand, in an evaluation setup like the one in [14, 15] where the recommender performance is evaluated per item, learning under the second perspective should provide better results. We propose to take both views into account by sampling from D_R and D_{R^T} alternately, as described in Algorithm 3. We call this horizontal and vertical sampling *pivotization*.

4.4 Sequential Learning

The multi-relational extension of the LearnBPR algorithm, LearnMR-BPR (figure 2), goes in line with the guidelines in Singh and Gordon [12], where the parameters are learned simultaneously by sampling examples from all the relations. In order to investigate the benefits of this joint factorization approach, we propose to compare it with an analogous model that learns in a step-by-step fashion, such as *ModMax*, *edge clustering*, and *BPR-map*. In order to do this, we will first learn the user parameters on the auxiliary relation using the original LearnBPR algorithm, and then factorize the target relation with the traditional LearnBPR algorithm, making sure to only update the item parameters (keeping the user parameters constant at the values obtained during the factorization of the auxiliary relation).

Figure 4 details the sequential MR-BPR variant of our multi-relational learning approach.

5. EVALUATION

5.1 Datasets

We have three social network datasets: one from a larger blogging website (Blogcatalog), one from flickr.com (Flickr), and one crawled from youtube.com (YouTube); each dataset consists of two relations, with one relation between users and labels (target), and an additional social relation between users and other users (auxiliary). Table 2 shows detailed statistics.

Dataset	relation	dimensionality	# observations	avg. observations per user	avg. observations per label	sparsity
Blogcatalog	target	10312x39	14,476	1.4	371.2	96.40%
	auxiliary	10312^2	667,966	64.7 (med 21)	-	99.37%
Flickr	target	80513x195	107,741	1.3	552.5	99.31%
	auxiliary	80513^2	11,799,764	146.6 (med 46)	-	99.82%
YouTube	target	31,703x47	50,691	1.6	1,078.5	96.60%
	auxiliary	$1,138,499^2$	5,980,886	5.3 (med 1)	-	99,9995%

Table 2: Statistics of the BlogCatalog, Flickr, and YouTube datasets.

```
procedure LearnBPR-Sequential($D_Y, D_S, U, I$)
    initialize $\mathbf{U}$
    repeat
        draw $(u, i, j)$ from $D_S$
        $\mathbf{U} \leftarrow \mathbf{U} + \mu \left( \alpha_S \frac{e^{-\hat{x}^R_{uij}}}{1+e^{-\hat{x}^R_{uij}}} \cdot \frac{\partial}{\partial \mathbf{U}} \hat{x}^R_{uij} + \lambda_U \cdot \mathbf{U} \right)$
    until convergence
    initialize $\mathbf{I}$
    repeat
        draw $(u, i, j)$ from $D_Y$
        $\mathbf{I} \leftarrow \mathbf{I} + \mu \left( \alpha_S \frac{e^{-\hat{x}^R_{uij}}}{1+e^{-\hat{x}^R_{uij}}} \cdot \frac{\partial}{\partial \mathbf{I}} \hat{x}^R_{uij} + \lambda_I \cdot \mathbf{I} \right)$
    until convergence
    return $\mathbf{U}, \mathbf{I}$
end procedure
```

Figure 4: Sequential MR-BPR learning algorith

The high sparsity of all relations is instantly noticeable, although there are some differences between the three datasets. In comparison with Blogcatalog, on Flickr the target relation is almost an order of magnitude sparser. Blogcatalog and YouTube show virtually the same sparsity level on the target relation, but YouTube is three orders of magnitude sparser on the network relation with a median of one user being connected to another.

5.2 Baseline Methods

For better evaluation of the proposed MR-BPR method, we will field the following well-known and strong baseline methods (already have been detailed in section 3.2):

- Modularity maximization (ModMax). ModMax [14] is essentially a PCA on the auxiliary matrix followed by a SVM-multilabel regression between the labels on the target relation and the top eigenvectors of the PCA as features. The interesting aspect of ModMax is that the auxiliary relation gets converted into a modularity matrix [8] before the eigenvectors are extracted. ModMax already outperforms wvRN, LBC, and other classical relational learning algorithms.
- Edge clustering. Edge clustering [15] does a k-means clustering of the edges of the auxiliary relation interpreted as a graph. The resulting sparse clusters are subsequently normalized in a row-wise manner and used as features for a multi-label SVM like with ModMax.
- BPR-map. BPR-map [1] also is a two-step process. In the first step, a factorization model is learnt on the observed (sparse) part of the target relation. As a regular factorization model, this does not yield any features for elements not observed in the training data. In the second step, a linear mapping between the user features from the factorization model and the corresponding entries in the auxiliary relation is learnt. The auxiliary relation serves as a predictor for this mapping and thus, given information in the auxiliary relation, the model is able to make predictions for users not observed in the initial factorization of the target relation.

5.3 Measures

For a consistent evaluation with the social analysis literature, two classical measures for multi-label classification are employed: Micro-F1 (cf. equation (7)) and Macro-F1 (cf. equation (6)).

To better capture the overall ranking performance of the algorithms, the area under the ROC curve, AUC (cf. equation (9)), is analyzed too.

$$F1(i) := \frac{2 * \mathit{correct}(i)}{\mathit{true}(i) + \mathit{predicted}(i)}, \tag{5}$$

$$MacroF1 := \frac{\sum_{i \in I} F1(i)}{|I|} \tag{6}$$

$$MicroF1 := \frac{2 * \sum_{i \in I} \mathit{correct}(i)}{\sum_{i \in I} \mathit{true}(i) + \sum_{i \in I} \mathit{predicted}(i)} \tag{7}$$

$$AUC(u) := \frac{1}{|I_u^+||I \setminus I_u^+|} \sum_{i \in I_u^+} \sum_{I \setminus I_u^+} \delta(\hat{x}_{uij} > 0), \tag{8}$$

$$AUC := \frac{\sum_{u \in U} AUC(u)}{|U|} \tag{9}$$

The functions $\mathit{correct}(i)$ counts the "hits" of the prediction model for an item i where an element from the ground truth is among the elements predicted. The function $\mathit{true}(i)$ emits the number of elements in the ground truth, and $\mathit{predicted}(i)$ returns the number of elements in the respective recommendation list.

5.4 Experiment Protocol

For the experiment protocol, we follow [15] for better comparability.

We evaluated a cold-start scenario in which some users already have social information in the auxiliary relation but are not yet present in the target relation. The aim is to predict which labels the cold-start users interacted with on the target relation.

The experiments were set up as ten-fold cross-validated experiments at different train-test split ratios. For each train-test split ratio, the respective ratio of the available users in the target relation is used for training, together with all information from the social relation. The remaining percentage of users in the target relation was used as test data for evaluation. For each train-test split ratio, ten different splits were generated randomly and continued to provide equal conditions for all algorithms and to get more reliable performance estimates. Test data was never used during training.

As suggested in [15], we remove the dependency on the length of the top-N list for performance evaluation w.r.t. Micro-F1 and Macro-F1 by assuming that the number of true instances is known at prediction time.

On Blogcatalog we created nine different split percentages from 10% to 90% of the target relation being used for training. For Flickr and YouTube, we used 1% to 10%—a setting in which hardly any interaction information on the target relation remains available for training.

5.5 Results

The prediction performance on all three datasets is shown in figures 5–7. Consistent with [15] we did not run modularity maximization on the YouTube dataset due the to the vast runtime and memory requirements of the given method.

The overall picture is the same across all three datasets and all three evaluation criteria employed. In most cases, the BPR-map yields the lowest performance, although this method can sometimes tie with ModMax an/or edge clustering, in rare cases even outperform those methods. Edge clustering outperforms ModMax on Micro-F1 and Macro-F1, except on Blogcatalog, where they are tied (figures 5–6). For AUC, the performance is the complete opposite, with ModMax easily outpacing edge clustering (figure 7). Sequential MR-BPR is the runner-up method, while MR-BPR performs best.

From a datasets perspective, it is interesting to note that Flickr seems to be the toughest dataset with respect to Micro- and Macro-F1, but not so for AUC (where we even observe a reversal of the order of sequential MR-BPR and jointly learning MR-BPR). The top-performing methods on Flickr (AUC), MR-BPR and sequential MR-BPR, achieve very stable and highly correct ordering in pairs. By comparison, Blogcatalog and YouTube seem to be rather easy datasets, except for AUC. On Blogcatalog, edge clustering manages to tie with the proposed MR-BPR algorithms for the splits from 50% on for AUC.

The curves for the BPR-map and our MR-BPR are very similar across all datasets and evaluation measures. Nonetheless, the performance of both approaches is very different with MR-BPR easily outperforming BPR-map. Sequential MR-BPR manages to tie with the jointly learning MR-BPR on several occasions, such as on Flickr and YouTube AUC and on YouTube Micro-F1. Though, in general MR-BPR performs significantly better.

Finally, we note that for Macro-F1 the pivotization of MR-BPR (section 4.3) leads to significantly improved results for all three datasets compared to the version without pivotization. Results omitted for brevity.

Criterion	BPR-SVM	ModMax*	Edge Clustering*
MicroF1 @ 10%	0,2702	0,2756	0,2815
MicroF1 @ 90%	0,3838	0,3756	0,3592
MacroF1 @ 10%	0,1724	0,1735	0,1640
MacroF1 @ 90%	0,2697	0,2390	0,2477

Table 3: Using an SVM on top of features learnt through LearnBPR scheme vs. features learnt using modularity maximization and edge clustering. The results are on the Blogcatalog dataset. Methods with an (*) used optimized hyper-parameters. Higher values are better.

5.6 Discussion

ModMax and edge clustering are strong state-of-the-art methods for the multi-relational cold-start item recommendation problem. Both are known to outperform classical relational learning algorithms in the present problem setting[15]. While BPR-map is also a strong method, it is not designed for a multi-relational setting, but rather for use with i.i.d. attributes as auxiliary information; this is reflected by the results it can achieve.

As the ModMax [14] algorithm proposes the conversion of the auxiliary relation into a dense modularity matrix [8], we wanted to investigate the strength of this pre-processing step. To do so, we replaced this feature extraction process with a plain factorization of the auxiliary relation. The user features learnt this way are treated as vectors, normalized by their norm and subsequently fed into the same multi-label SVM process as Tang and Liu (using the same hyper-parameters they report). From the results in table 3, it is clear that there is virtually no difference between the multi-label SVM on the expensive eigenvectors of the modularity matrix or the multi-label SVM on the user features from the factorization.

This is an interesting observation, as the sparse matrix decomposition [4] of the LearnBPR [9] methods is easy to implement, of low complexity and additionally very fast, yet principled. We may add that optimized hyper-parameters have not been searched - neither for the BPR factorization of the auxiliary relation nor for the SVM learning with the BPR features as parameters. Instead, we simply re-used the hyper-parameters from Tang and Liu for the SVM and from MR-BPR for the factorization.

An important factor in the performance we saw on the various datasets and evaluation methods, we believe, should be attributed to interpreting the underlying multi-relational data in a more appropriate way: BPR treats input data as sparse and unary. This benefits MR-BPR in two ways. Firstly, it treats the auxiliary relation as sparse and unary while BPR-map, ModMax, and edge clustering do not do so. Secondly, ModMax and edge clustering do not take this data property for the target relation into account, while MR-BPR and BPR-map do. We argue that edge clustering performs better than its cousin, ModMax, because it generates sparse features (from a dense interpretation of the matrix) for its final learning step. Further to this, we believe that the similarity in shape of the performance curves of BPR-map and MR-BPR may result from the conceptual similarity with respect to the interpretation of the target relation as positive-only feedback and the application of the BPR-Opt ranking criterion.

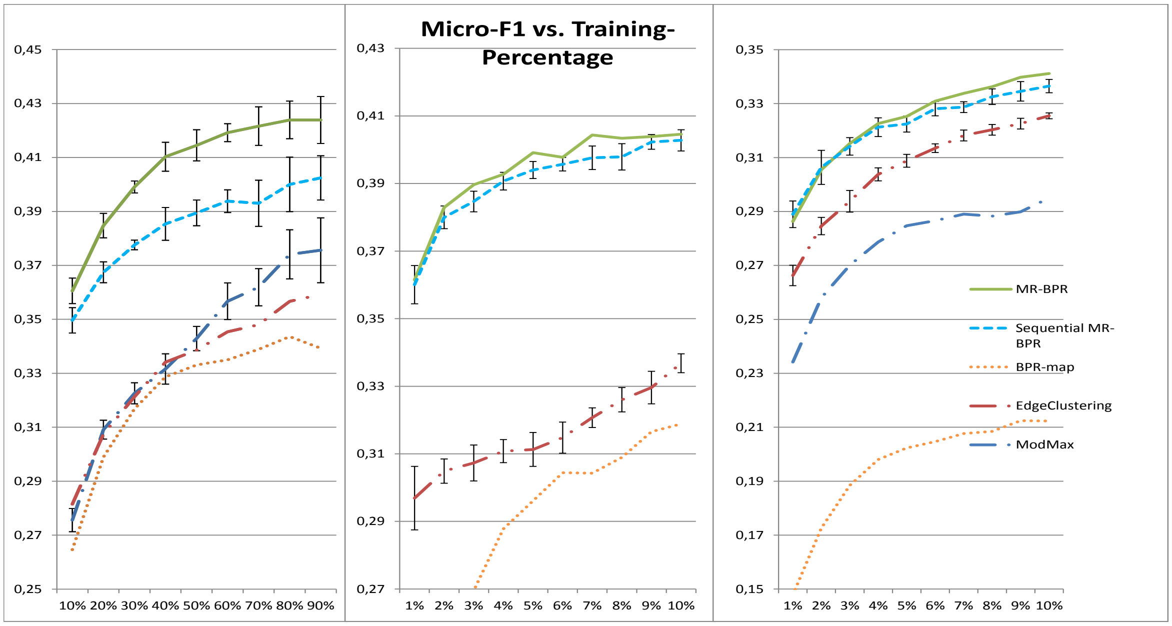

Figure 5: Micro-F1 results on a. Blogcatalog, b. YouTube, and c. Flickr vs. percentage of data used for training. The standard deviation is reported for some methods for better assessment of the statistical significance.

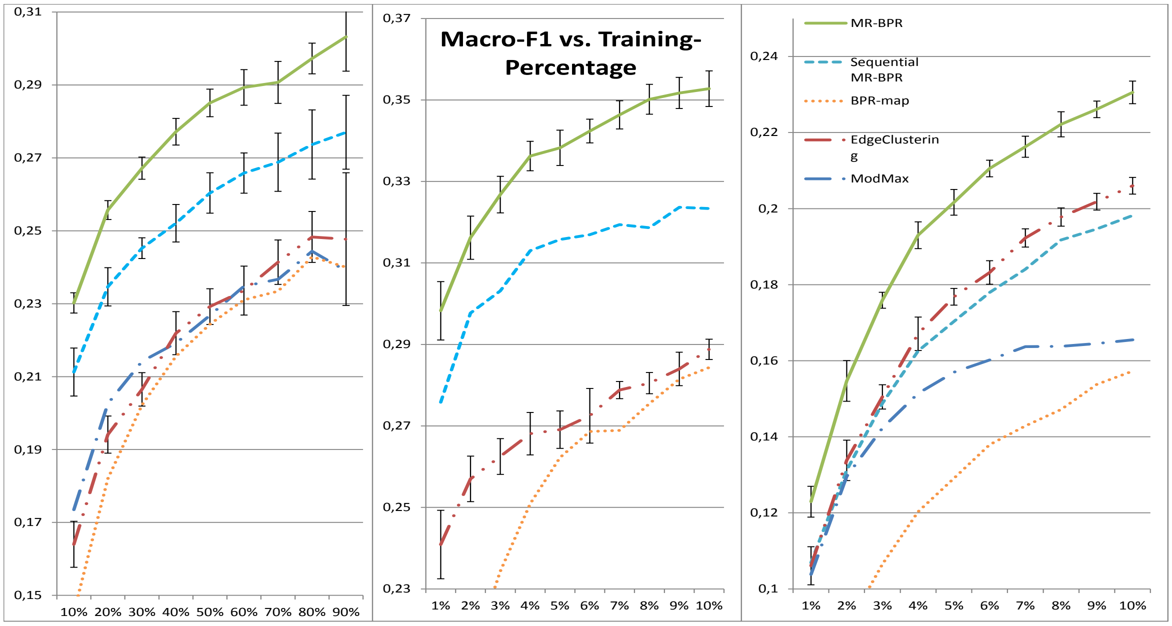

Figure 6: Macro-F1 results on a. Blogcatalog, b. YouTube, and c. Flickr vs. percentage of data used for training. The standard deviation is reported for some methods for better assessment of the statistical significance.

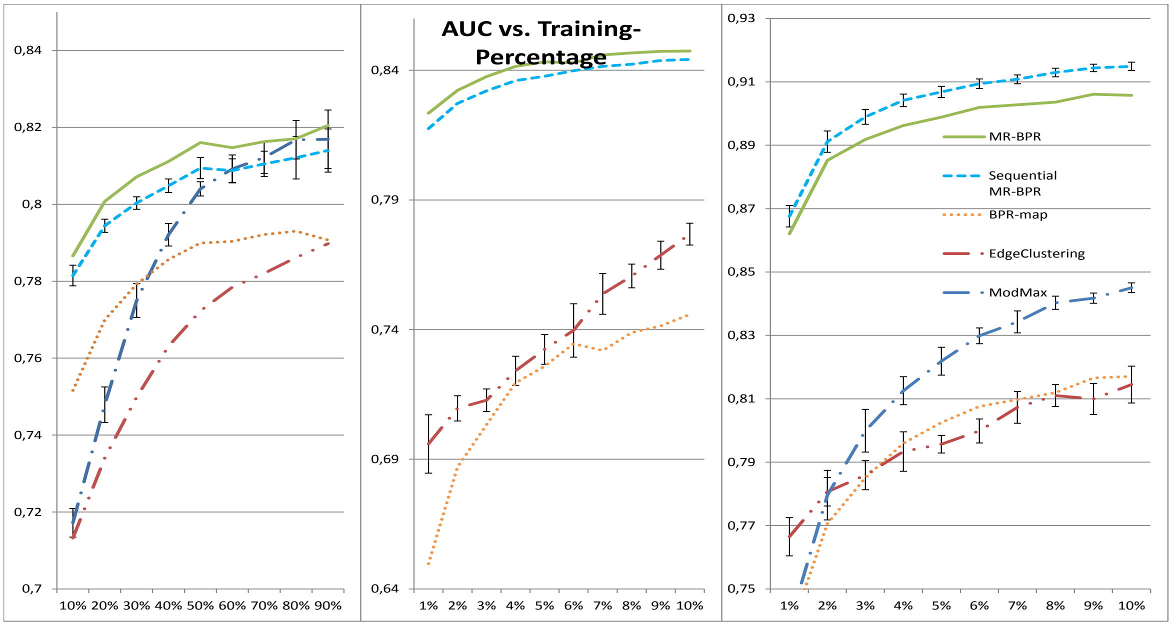

Figure 7: AUC results on a. Blogcatalog, b. YouTube, and c. Flickr vs. percentage of data used for training. The standard deviation is reported for some methods for better assessment of the statistical significance.

We should add that there may also have been more potential for edge clustering and ModMax (and BPR-SVM for that matter) if the parameters had been optimized for the F-measure evaluation scheme. Although it is hard to optimize directly for F-measure, Musicant et al. [7] have shown that for the SVM both, C_+ and C_- need to be searched for an approximate optimization. It is worth noting that the BPR learning framework does not directly optimize, but simply leaves enough room for general optimizations to arbitrary evaluation criteria. For that matter, all hyper-parameters for the BPR family algorithms have been searched such that optimized conditions for an evenly weighted combination of the evaluation criteria were created.

The joint learning of the full multi-relational factorization is what we consider a major improvement over previous methods. Not only is it in line with the findings from [12], but it is also highly scalable (i.e., virtually for free) and performs well. However, we do agree that it would be difficult for the competing SVM approaches to realize such joint learning without an expectation maximization approach. We would have expected the jointly learnt MR-BPR to outperform its sequentially learnt cousin on a more pronounced level. For this reason, more detailed research on this this trait will be a part of our work in future. Our conjecture is that a better factorization on the auxiliary matrix is meaningless when not accompanied by an improved factorization on the target relation.

5.7 Reproducibility of the Experiments

For modularity maximization and edge clustering, the optimal hyper-parameters (the number of latent dimensions for the feature selection and the SVM cost parameter) for all datasets are taken from [15]. For BPR-map and MR-BPR, the optimal hyper-parameters have been estimated via grid search on the largest training split of each respective dataset. BPR-map has two sets of hyper-parameters, where the first set is used for the initial factorization of the target relation ($\mu = 0.01$, $\lambda_u = 0.01$, $\lambda_i = 0.01$, $\lambda_j = 0.00005$ and 300 iterations for all datasets and $k = 500$ for Blogcatalog, $k = 200$ for Flickr and YouTube) and the second set is used for the subsequent mapping of the latent feature to the auxiliary relation ($\lambda_u = 0.0025$, $\lambda_i = 0.0025$, $\lambda_j = 0.00025$, $\mu = 0.01$, 300 iterations, $\lambda_{mapping} = 0.01$, 15000 mapping iterations, $\mu_{mapping} = 0.1$ for all datasets and $k = 500$ for Blogcatalog and $k = 200$ for Flickr and YouTube). For MR-BPR the hyper-parameters are $k = 500$, $\mu = 0.02$, $\lambda_{user} = 0.0125$, $\lambda_{item} = 0.0005$, 300 iterations, $\alpha = 0.5$ for Blogcatalog; $k = 200$, $\mu = 0.05$, $\lambda_{user} = 0.005$, $\lambda_{item} = 0.0005$, $\alpha = 0.5$, 300 iterations for Flickr; and $k = 200$, $\mu = 0,1$, $\lambda = 0,0125$, $\lambda_{item} = 0,0125$, $\alpha = 0,5$ and 300 iterations for YouTube.

The code for MR-BPR is available at: `http://www.ismll.uni-hildesheim.de/software`.

The code for BPR-map is available in the MyMediaLite framework [2]. The code for ModMax and EdgeClustering is available from the homepage of the first author of [14].

The datasets are available from the homepage of the first author of [14].

6. CONCLUSION

In this work, we formalized the special link prediction task of item recommendation in social networks in a multi-relational framework. We then presented a multi-relational factorization approach to the problem. Since the task of item prediction is a ranking task, the Bayesian personalized ranking (BPR) framework was extended to the multi-relational case, so that the approach proposed here offers a model optimized for a ranking loss function.

Our experiments on real world datasets show that our approach outperforms state-of-the-art competitors in different evaluation measures without the need higher runtime.

One of the advantages of multi-relational matrix factorization techniques against state-of-the-art approaches like ModMax and edge clustering is that there is no need for any kind of preprocessing of the auxiliary relation, such as modularity matrix computation or clustering of edges. Instead of factorizing relations separately, our experiments suggest that learning them jointly yields at least comparable results. A thorough comparison between joint an sequential learning of the auxiliary and target relations is left for future work.

7. ACKNOWLEDGMENTS

The authors gratefully acknowledge the co-funding of their work by the Multi-relational Factorization Models project granted by the Deutsche Forschungsgesellschaft[2]. Lucas Drumond is sponsored by a scholarship from CNPq, a Brazilian government institution for scientific development. We also want to thank the reviewers for their comments and Lei Tang for providing the datasets, the implementations of his baseline methods and his friendly and swift replies to our questions and requests.

8. REFERENCES

[1] Z. Gantner, L. Drumond, C. Freudenthaler, S. Rendle, and L. Schmidt-Thieme. Learning attribute-to-feature mappings for cold-start recommendations. In *Proceedings of the 2010 IEEE International Conference on Data Mining*, ICDM '10, pages 176–185, Washington, DC, USA, 2010. IEEE Computer Society.

[2] Z. Gantner, S. Rendle, C. Freudenthaler, and L. Schmidt-Thieme. Mymedialite: A free recommender system library. In *Proceedings of the 5th ACM International Conference on Recommender Systems (RecSys 2011) (to appear)*, 2011.

[3] M. Jamali and M. Ester. A matrix factorization technique with trust propagation for recommendation in social networks. In *Proceedings of the fourth ACM conference on Recommender systems*, RecSys '10, pages 135–142, New York, NY, USA, 2010. ACM.

[4] Y. Koren, R. Bell, and C. Volinsky. Matrix factorization techniques for recommender systems. *Computer*, 42(8):30–37, 2009.

[5] H. Ma, I. King, and 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*, SIGIR '09, pages 203–210, New York, NY, USA, 2009. ACM.

[6] H. Ma, H. Yang, M. R. Lyu, and I. King. Sorec: social recommendation using probabilistic matrix factorization. In *Proceeding of the 17th ACM conference on Information and knowledge management*, CIKM '08, pages 931–940, New York, NY, USA, 2008. ACM.

[7] D. R. Musicant, V. Kumar, and A. Ozgur. Optimizing f-measure with support vector machines. In *Proceedings of the Sixteenth International Florida Artificial Intelligence Research Society Conference*, pages 356–360. Haller AAAI Press, 2003.

[8] M. E. J. Newman. Modularity and community structure in networks. *Proc. Natl. Acad. Sci. USA*, 103:8577–8582, 2006.

[9] S. Rendle, C. Freudenthaler, Z. Gantner, and L. Schmidt-Thieme. BPR: Bayesian personalized ranking from implicit feedback. In *Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence*, UAI '09, pages 452–461, Arlington, Virginia, United States, 2009. AUAI Press.

[10] S. Rendle and L. Schmidt-Thieme. Pairwise interaction tensor factorization for personalized tag recommendation. In *Proceedings of the third ACM international conference on Web search and data mining*, WSDM '10, pages 81–90, New York, NY, USA, 2010. ACM.

[11] A. I. Schein, A. Popescul, L. H. Ungar, and D. M. Pennock. Methods and metrics for cold-start recommendations. In *Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '02, pages 253–260, New York, NY, USA, 2002. ACM.

[12] A. P. Singh and G. J. Gordon. Relational learning via collective matrix factorization. In *Proceeding of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '08, pages 650–658, New York, NY, USA, 2008. ACM.

[13] A. P. Singh and G. J. Gordon. A unified view of matrix factorization models. In *Proceedings of the European conference on Machine Learning and Knowledge Discovery in Databases - Part II*, ECML PKDD '08, pages 358–373, Berlin, Heidelberg, 2008. Springer-Verlag.

[14] L. Tang and H. Liu. Relational learning via latent social dimensions. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '09, pages 817–826, New York, NY, USA, 2009. ACM.

[15] L. Tang and H. Liu. Scalable learning of collective behavior based on sparse social dimensions. In *Proceeding of the 18th ACM conference on Information and knowledge management*, CIKM '09, pages 1107–1116, New York, NY, USA, 2009. ACM.

[16] Yang, Long, Smola, Sadagopan, Zheng, and Zha. Like like alike – joint friendship and interest propagation in social networks. In *Proceedings of the WWW 2011*, 2011.

[17] Y. Zhang, B. Cao, and D.-Y. Yeung. Multi-domain collaborative filtering. In *Proceedings of the 26th Conference on Uncertainty in Artificial Intelligence (UAI)*, pages 725–732, Catalina Island, California, 2010.

[2] http://www.ismll.uni-hildesheim.de/projekte/dfg_multirel_en.html