Transaction-Based Link Strength Prediction in a Social Network

Hassan Khosravi
School of Computer Science
Simon Fraser University
hkhosrav@cs.sfu.ca

Ali Bozorgkhan
School of Computer Science
Simon Fraser University
abozorgk@cs.sfu.ca

Oliver Schulte
School of Computer Science
Simon Fraser University
oschulte@cs.sfu.ca

Abstract—**The revolution of social networks and methods of analyzing them have attracted interest in many research fields. Predicting whether a friendship holds in a social network between two individuals or not, link prediction, has been a heavily researched topic in the last decade . In this paper we investigate a related problem, link strength prediction: how to assign ratings or strengths to friendship links. A basic approach would be matrix factorization applied to only friendship ratings. However, the existence of extensive transactions among users may be used for better predictions. We propose a new type of multiple-matrix factorization model for incorporating a transaction matrix. We derive gradient descent update equations for learning latent factors that predict values in the target rating matrix. Multiple-matrix factorization can be seen as a data fusion technique, that combines evidence from different sources. In the social network application, the target matrix contains friendship ratings and the evidence matrices specify transaction intensities between users. To evaluate the model, we introduce data from Cloob, a popular Iranian social network as well as synthetic data.**

I. Introduction

Online social networks (Facebook, Orkut, LinkedIn, Myspace) and methods of analyzing them have attracted extensive interest in many research fields. Much of the past work has focused on social networks with binary relational ties (e.g., whether two people are friends or not), which is known as link prediction [9]. Because of the low-cost or effort for becoming friends in on-line social networks, the resulting networks have both strong and weak ties with little or no information to differentiate the two, so these binary indicators provide only a coarse indication of the nature of the relationship. Treating both strong and weak ties the same, when analyzing the characteristics of an individual, increases the noise level and leads to misleading results. A recent trend in on-line social network, such as Google+ and Facebook, has been to allow users to differentiate among their friendship links by forming lists or circles. Gilbert [2] presents research on restricting Twitter lists to a subset of people followed by a user. We formulate the link strength problem to extend link prediction to estimate the strength of a tie given a pair of linked individuals where larger values indicate a stronger link [6, 3]. Lists can be viewed as a simple discretization of the continuous tie strength concept that we model.

The topological structure of the network defines an explicit network that contains information about the strength of links. A basic approach would be to apply matrix factorization to this network to learn latent factors for each user. Although this model performs reasonably well, the existence of a large volume of data recording transactions between users is a motivation for building more complex models that better predict strengths of friendships. The strength of a tie directly impacts the frequency of transactional events between users, as users tend to interact mostly with their strong ties. These transactional intensities generate an implicitly weighted network among users that may be used in addition to the explicit network. Previous approaches have derived features from the implicit networks and then used the features to build a classification model for distinguishing weak from strong ties [6, 3]. Instead, multiple-matrix factorization performs data fusion, which is the process of combining data from multiple sources into one model for analysis.

Collective matrix factorization is a general approach for analyzing multiple matrices [15, 10]. In this model, for each individual a single latent feature vector is introduced to explain all the links that the individual participates in. Collective matrix factorization is well-suited to generative models that represent the joint distribution over all relationships. In contrast, we use a discriminative model where the goal is to predict the values of a single target matrix—strength of friendship—and implicit matrices, like transactional intensities, are used to weight the importance of connections between users.

Research on social recommendation systems has shown that friendship information can improve item recommendation [5, 15]. Investigations of social networks have shown that it is friends with strong ties, rather than weak ties, that exhibit similar preferences [3]. Therefore a promising application of link strength prediction is to combine it with social recommendation models (i.e., a user-item matrix) [14].

Evaluation. We use data from Cloob[1], a popular Iranian website, as well as synthetic data to validate our work. To our knowledge, the Cloob dataset is the only dataset that contains explicit friendship ratings, which serve as a ground truth for our evaluation. Experimental results show that our Transaction-MF model outperforms collective matrix factorization, single table matrix factorization, as well as models that analyze the data matrices separately. An important challenge for link strength prediction is the presence of *zero-*

[1] http://www.cloob.com

978-1-4673-5895-8/13/$31.00 ©2013 IEEE

transaction friendships, between user dyads that are friends but have no recorded activities. Our experiments illustrate that even in the absence of direct transactions between two friends, their transactional behavior with their other friends is informative and leads Transaction-MF model to perform better than competitors. Our experiments also examine the impact of a number of model parameters, including the dimensionality of the latent factors and a trade-off parameter that controls the importance of the transaction intensities.

Contributions.

- To our knowledge, ours is the first paper using matrix factorization model to predict the strength of an existing link, as opposed to the existence of a link.
- Since the user-user link strength rating is directed, we distinguish two types of latent feature vectors associated with each user: one that models the user's behavior as a rater, and a second that models the user's behavior as a ratee.
- We present the first evaluation of link-strength prediction on a real-world data set with continuous ratings, as ground truth, as opposed to binary link strength labels.

II. Problem Definition, Background, and Notation

Suppose the social network consists of a set of N users. The link strengths expressed by users for other users are given in a link strength matrix $R_{N\times N}$. This matrix is not symmetric, because users u and v can express different strengths for their friendship. Instead of using indices like i, j for a generic user, we use the index u for users as raters, and v for users as ratees. The goal of link strength prediction is to infer the value of $r_{u,v}$ given that we know u and v are friends. The baseline or bias parameters b_u^{rater} and b_v^{ratee} indicate the observed deviation of user u as rater and user v as rated from average μ over all expressed strengths in the training dataset. Each user is assigned two baseline parameters [7]. For example, suppose user John wants to rate his friendship with user Mary. Assume that the average of the ratings is μ which is 3.4 and Mary is a popular user whom users rate 0.8 above average. On the other hand, John is a critical user who rates 0.3 below average. We would predict John will rate his friendship with Mary as 3.4 + 0.8 - 0.3 = 3.9.

The basic idea of our data fusion model is to learn informative latent factors for each individual. Since the patterns of how a user rates others and how others rate him are usually distinct, we associate two sets of latent factors to each user. As an example, a popular movie star may be linked to many fans that indicate a strong tie to him, but he would probably indicate a weak tie to fans that he does not personally know. Let $U_{N\times K}$ be the latent factors modeling how users rate and let $V_{N\times K}$ be the latent factors modeling how users are rated. In our example, the V factors of the movie star indicate that he tends to be rated high and U factors of the movie star show that he tends to rate low.

We employ matrix factorization techniques to learn the latent characteristics of users. Matrix factorization maps users to a joint latent factor space of dimensionality k, such that the rating between two users would be the inner products of their latent factor like $\hat{r}_{uv} = {U_u}^T V_v$. The baseline predictors may be integrated with the raw matrix factorization model

$$\hat{r}_{uv} = \mu + b_u^{rater} + b_v^{ratee} + {U_u}^T V_v$$

Each user has two sets of direct friends or neighbors: First, N^{rater} for the friends that the user has rated, N^{ratee} for the friends that have given the user a rating. We translate transactional intensities between two neighbors into real numbers. The transaction intensities are stored in the **transaction matrices** $T^i_{N\times N}$. The expression $T^i_{u,v} = x$ denotes that x is the intensity of transaction type i between u and v that were carried out by u. In general, transactions are directed (e.g., sending a message), so T^i is asymmetric. In this paper, we focus on one transaction matrix $T_{N\times N}$; however, our techniques extend to multiple matrices. We use the terms transactions and activities interchangeably.

III. Transactional-Based Link Strength Model

In this section we go through the mathematical details of the proposed TMF model. Basically we use a Gausian prediction model where the mean of the predictive distribution is a linear combination of the baseline predictors and the product of the latent factors.

We discuss how the baseline predictors b_u^{rater} and b_v^{ratee} and latent factors for U and V are learned given the strength of the friendship and transactional intensities between users. The corresponding graphical model is presented in Figure 1 which can be computed using the following Gaussian,

$$p(R|U, V, \sigma_R^2) = \prod_{u=1}^{N}\prod_{v=1}^{N}\left[\mathcal{N}\left(r_{u,v}|g(\mu + b_u^{rater} + b_v^{ratee} + U_u^T V_v), \sigma_R^2\right)\right]^{I^r_{u,v}}$$

Here $\mathcal{N}(x|\mu, \sigma^2)$ is the normal distribution with mean μ and variance σ^2, and $I^r_{u,v}$ is the indicator function that is equal to 1 if u has rated v and equals to 0 otherwise. The expression $g(x) \equiv \frac{1}{1+e^{-x}}$ denotes the logistic function. The logistic function was previously used for probabilistic matrix factorization [5].

In plain English, the model embodies the following assumptions about latent features. 1) The strength of a link between two users depends on how well the latent features of the two users match. 2) The latent features of a rater should be related to the latent features of her friends as ratees, where friends carry more weight if the intensity of transactions between the rater and the friend was higher. 3) The latent features of a ratee should be related to the latent features of her friends as raters, where friends carry more weight if the intensity of transactions between the ratee and the friend was higher.

Learning. We first learn the baseline estimates without considering the latent factors, following [7]. The intuition is that the baseline terms capture the general rating trend of a rater resp. ratee, independently of the transaction intensities. In contrast, the latent factors U and V depend on both the

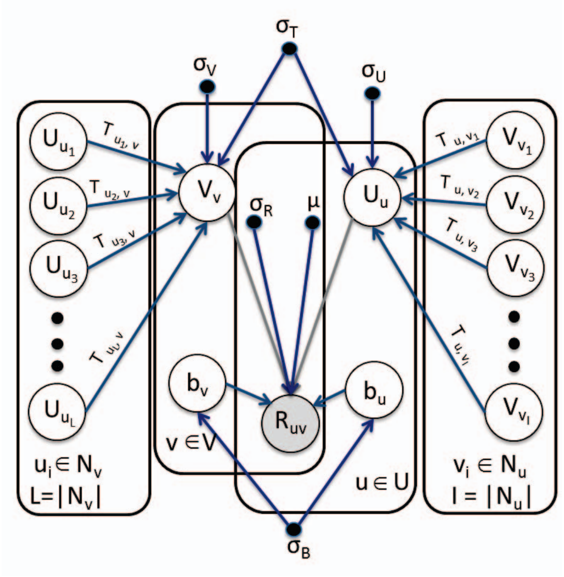

Fig. 1. The Transaction-MF or (TMF) plate model. Observed quantities are the ratings R_{uv} and the transaction intensities $T_{u,v}$.

ratings and the transaction intensities. In order to estimate b_u^{rater} and b_v^{ratee} for each user, we therefore maximize the log-likelihood of the ratings without considering the latent factors. Adding a L2-regularizer and trade-off parameter λ_b then leads to minimizing the following objective:

$$\frac{1}{2}\sum_{u=1}^{N}\sum_{v=1}^{N} I^r_{u,v}\Big(r_{u,v} - g(\mu + b_u^{rater} + b_v^{ratee})\Big)^2$$
$$+\frac{\lambda_b}{2}\sum_{u=1}^{N}(b_u^{rater})^2 + \frac{\lambda_b}{2}\sum_{v=1}^{N}(b_v^{ratee})^2$$

We use gradient descent to learn the baseline predictors. The following formula gives the update rules.

$$b_u^{rater} := b_u^{rater} - \gamma\Big(\sum_{v \in N_u} g'(\mu + b_v^{ratee} + b_u^{rater})$$
$$\times\Big(g(\mu + b_v^{ratee} + b_u^{rater}) - r_{uv}\Big) + \lambda_b b_u^{rater}\Big)$$
$$b_v^{ratee} := b_v^{ratee} - \gamma\Big(\sum_{u \in N_v} g'(\mu + b_v^{ratee} + b_u^{rater})$$
$$\times\Big(g(\mu + b_v^{ratee} + b_u^{rater}) - r_{uv}\Big) + \lambda_b b_v^{ratee}\Big)$$

Similar to the Social-MF model [5], we make some simplifying assumptions to approximate the optimal latent factors. (1) The rater factors U and ratee factors V are mutually dependent. We approximate their joint distribution as the product of the conditional distributions. (2) The conditional distribution of one latent factor conditional on the model parameters and the other latent factors are split into two terms: (i) the shrinkage prior and (ii) the conditional distribution given the latent factors of friends. (3) The latent factor that describes how a user rates her friends depends linearly on the latent factors that describe how her friends are rated, with the transaction intensities between her and her friends as weights.

We can apply Bayes' theorem and the assumptions stated to obtain a tractable approximation to the posterior probability of the latent variables U and V, shown in the following formula.

$$p(U,V|R,T,\sigma_R^2,\sigma_T^2,\sigma_U^2,\sigma_V^2)$$
$$\approx p(R|U,V,\sigma_R^2) \times p(U|V,T,\sigma_U^2,\sigma_T^2) \times p(V|U,T,\sigma_V^2,\sigma_T^2)$$
$$= \prod_{u=1}^{N}\prod_{v=1}^{N}\Big[\mathcal{N}\Big(r_{u,v}|g(\mu + b_u^{rater} + b_v^{ratee} + U_u^T V_v), \sigma_R^2\Big)\Big]^{I^r_{u,v}}$$
$$\times \prod_{u=1}^{N}\mathcal{N}(U_u|\sum_{v \in N_u} T_{u,v}V_v, \sigma_T^2\mathbf{I}) \times \prod_{u=1}^{N}\mathcal{N}(U_u|0,\sigma_u^2\mathbf{I})$$
$$\times \prod_{v=1}^{N}\mathcal{N}(V_v|\sum_{u \in N_v} T_{u,v}U_u, \sigma_T^2\mathbf{I}) \times \prod_{v=1}^{N}\mathcal{N}(V_v|0,\sigma_v^2\mathbf{I})$$

The log of the approximate posterior probability can be computed using the following formula. In the following formula set $\lambda_U = \sigma_R^2/\sigma_U^2$, $\lambda_V = \sigma_R^2/\sigma_V^2$ and $\lambda_T = \sigma_R^2/\sigma_T^2$.

$$L(R,T,U,V)$$
$$= \frac{1}{2}\sum_{u=1}^{N}\sum_{v=1}^{N} I^r_{u,v}\Big(r_{u,v} - g(\mu + b_u^{rater} + b_v^{ratee} + U_u^T V_v)\Big)^2$$
$$+\frac{\lambda_U}{2}\sum_{u=1}^{N} U_u^T U_u + \frac{\lambda_V}{2}\sum_{v=1}^{N} V_v^T V_v$$
$$+\frac{\lambda_T}{2}\sum_{u=1}^{N}\Big((U_u - \sum_{v \in N_u} T_{u,v}V_v)^T(U_u - \sum_{v \in N_u} T_{u,v}V_v)\Big)$$
$$+\frac{\lambda_T}{2}\sum_{v=1}^{N}\Big((V_v - \sum_{u \in N_v} T_{u,v}U_u)^T(V_v - \sum_{u \in N_v} T_{u,v}U_u)\Big)$$

This formula is the main objective function with the predictive baselines and transactional intensities included. Taking the derivative with respect to U_u and V_v yields gradient descent update formulas for the baseline predictors and latent factors.

$$\frac{\partial L}{\partial U_u} = \sum_{v=1}^{N} I^r_{u,v}\Big[V_v g'(\mu + b_u^{rater} + b_v^{ratee} + U_u^T V_v)$$
$$\times\Big(g(\mu + b_u^{rater} + b_v^{ratee} + U_u^T V_v) - r_{u,v}\Big)\Big] + \lambda_U U_u$$
$$+\lambda_T(U_u - \sum_{v \in N_u} T_{u,v}V_v) + \lambda_T\Big(\sum_{v \in N_u} T_{u,v}(U_u T_{u,v} - V_v)\Big)$$
$$\frac{\partial L}{\partial V_v} = \sum_{u=1}^{N} I^r_{u,v}\Big[U_u g'(\mu + b_u^{rater} + b_v^{ratee} + U_u^T V_v)$$
$$\times\Big(g(\mu + b_u^{rater} + b_v^{ratee} + U_u^T V_v) - r_{u,v}\Big) + \lambda_V V_v$$
$$+\lambda_T(V_v - \sum_{u \in N_v} T_{u,v}V_v) + \lambda_T\Big(\sum_{u \in N_v} T_{u,v}(V_v T_{u,v} - U_u)\Big)$$

IV. Related Work

a) Link Prediction: Matrix factorization models have been developed for predicting the existence of a link (e.g., [4, 12]). While such models are broadly similar to ours, the task of link strength prediction has different characteristics from the task of link existence prediction [6]: We model link strength as continuous, whereas link existence is binary. A big difference for learning methods is that links are sparse, so link existence shows a high class imbalance, whereas link strength applies only to existing links and does not exhibit the class imbalance problem.

b) Link Strength Prediction: Most previous link strength models treat the binary classification problem of distinguishing weak from strong ties [3, 6]. These models use transaction intensities as observed predictor variables. We use continuous link strength as dependent variable, and the predictors are latent user factors that are learned using the continuous transaction intensities. A major advantage of latent factor models is that they can utilize parameters for each individual rather than a weight for each attribute.

Xiang *et al.* [14] predict continuous link strength with a latent variable model. However, their model does not use matrix factorization, that is, it does not assign a latent factor to each user that explains the user's ratings (given and received). Instead, the link strength itself is the latent variable, related to user similarity and transaction records as observed features. The evaluation does not compare predicted link strength against true link strength ratings as their datasets do not provide ground truth ratings.

c) Single Matrix Factorization: Link prediction uses latent factors to model a link matrix, where links are typically undirected and have binary values. Our models feature two directed and continuous matrices, link strength and transaction intensity. Recommendation systems also predict a continuous rating. Koren et al in [8, 7] combine single matrix factorization with baseline predictors for the Netflix challenge. Our model adopts this idea.

d) Multiple Matrix Factorization: Social recommender systems can use double-matrix factorization, one item-user matrix and another user-user friendship or trust matrix [5, 11, 10, 10]. The Social-MF model of [5] is a discriminative model where the goal is to predict the values of a single target matrix (user-item). A trust matrix is used to weight the strength of connections between users. The Social-MF model features a *recursive dependency* in which the latent factors of a user depend on the latent factors of his friends.

Our Transaction-MF model (TMF) incorporates multiple (user-user) matrices and the goal is to predict the values of a user-user target rating matrix. The predictions are made on the basis of two sets of latent factors U and V, that model the behavior of each user, one for rating and one for being rated. TMF adapts the idea of a recursive dependency among latent factors, where a transaction intensity matrix is used to weight the strength of connections between users. The main new aspects of our model compared to Social-MF are the following.

1) The prediction target is a directed user-user rating, not a user-item rating. Therefore we introduce *two* sets of latent features, one that models each user's behavior as a rater, and another that models each user's behavior as a ratee.
2) In Social-MF, the latent feature of each user depends on those of her neighbors, so the latent features can be interpreted in terms of a similarity metric between users. In TMF, the latent factor of a rater (ratee) depends on the latent factors of her neighbors as ratees (raters). This condition can be interpreted as a regularization [5], but not as expressing that latent factors of neighbors should be similar.
3) The trust information that is used to weight the latent neighbor factors in Social-MF is replaced by information about transaction intensities, which reflect multiple types, with different scales, of actitivities between users. Whereas trust values are usually not directly expressed by users, specially continuous ones, transactions are logged in the system.
4) We adapt Koren's use of baseline or bias parameters for each user to the link strength prediction setting. The baseline parameters reflect a user's general tendency to rate and be rated, independent of transaction information. Such baseline parameters are not used in Social-MF.

V. Experiments

In this section we examine the predictive performance of the TMF model on different criteria. All of the experiments are reported over 5-fold cross validation. We distribute the friendship links for each user and the associated transaction information, in either direction, evenly into the folds as much as possible, so we train on approximately 80% of the friendship links available for a user and test on the rest of the friendships [13].

A. Datasets

We use synthetic and real world data to validate the model.

1) Cloob.com: It is a challenge to find data of reasonable size on real social networks for evaluating link strength. We introduce Cloob, which is the biggest social network and the 8th most popular website in Iran. We were given full access to cloob's anonymized data for research purposes. The network has records of over 1.5 million users and 25 million friendships between 2005 to 2010.

A special feature of Cloob is that users can enter a value between 0 and 5 to indicate the strength of a friendship, which provides us with a ground truth for evaluating link strength models. Since the website does not require users to rate their friends, many of the friendships have the default rating. We omit the ties with default value from our experiments and concentrate on ties of other strength because: (1) The strength of default friendships are unknown since they contain both strong and weak ties. (2) Due to the large skew on the target class, including the default values in the learning procedure of the latent factors, makes the model perform poorly in predicting other values of strength. After removing all friendship ratings with the default value, the strengths are in the range [1,5]. Table I shows the number of friendships and activities after removing the default values. We evaluate the ratings given by users that have rated at least five of their friendships, and the ratings given to users that have been rated at least five times. The network contains 20 million transactional events between 2005 to 2010. Users can apply five types of activities to one another: comment, paint, send pictures, like, and write testimonial. To convert information about the five types of activities into a single entry in a transaction intensity matrix, we add up the five activity counts for every pair of friends. An interesting extension for future work would be to learn with five separate activity matrices.

# Friendships	502,060
# Users	45,150
# Activities	7,913,596

TABLE I
NUMBER OF FRIENDSHIPS AND ACTIVITIES AFTER REMOVING THE DEFAULT VALUES FOR THE CLOOB DATASET.

# Friendships	50000
# Users	27000
# Activities	50000

TABLE II
NUMBER OF FRIENDSHIPS AND ACTIVITIES FOR THE SYNTHETIC DATASET.

2) Synthetic Data: The idea behind this paper is that the implicit network created from the transactional intensities among users correlates with explicit network generated by the users ratings. We first generate a synthetic dataset in which there is a very strong correlation between strengths of ties and the intensity of transaction between users. In this dataset, the following two assumptions are exaggerated compared to what you expect in a real social network. (1) All of the friendship ties generated in this dataset are directly dependent on the number of transactions. This assumption does not generally hold as two users u and v may have known each other for a long time, leading to a strong tie, but have only recently become friends in an on-line social network, so they don't have many recorded transactions. (2) The average number of activities between users is much higher than what it normally is; there are usually many ties in social networks with zero or very few activities. The exaggerations are introduced to create a dataset where the basic assumption of the TMF model is true, so our expectation is that if the model assumptions are true, our learning algorithm for TMF should outperform other methods. This dataset is created using the following method.

First we generate users and their friendships using four distributions. The first two distributions are used to generate the baseline values b_u^{rater} and b_v^{ratee} for each user. We assume that each user could have one of three different types of behavior when defining strength for a friendship: assign higher than, same as, or lower than average strength for their friendships. The same three behaviors could be realized for how they have been rated. We assign a uniform distribution over the nine different types of people. The following formulas show how the b_u^{rater} and b_v^{ratee} are generated for users using the normal distribution. The mean value of the distributions depend on the user's type, which is -1, 0, or 1:

$$b_u^{rater} \sim \mathcal{N}(\{-1,0,+1\},\sigma_b^2), b_v^{ratee} \sim \mathcal{N}(\{-1,0,+1\},\sigma_b^2)$$

We then introduce a third parameter acc_u for users to generate their tendency to make friends. This parameter is the probability of acceptance of friendships for users. To generate friendships, we randomly sample two users u and v and produce a random value x between 0 and 1. If x is smaller than the acc_u, the link is kept, otherwise we repeat the step and sample two other users. We set the acceptance parameter to achieve a power law distribution on the number of friendships.

Finally, the rating between users is randomly generated according to the following distribution:

$$r_{uv} \sim \mathcal{N}(\mu + b_u^{rater} + b_v^{ratee} + a_{uv}, \sigma_R^2)$$

The a_{uv} parameter produces the number of activities between users u and v, distributed as $a_{uv} \sim max(\mathcal{N}(\mu_a,\sigma_a^2),0)$. Thus the number of activities between user pairs in social networks is modelled with a normal distribution. For simplicity, only one type of activity may be carried out between users. The minimum number of activities between users is set to 0 to avoid a negative number of transactions. In our experiment we arbitrarily set σ_b^2 = 0.5, σ_R^2 = 0.1, μ = 3, and μ_a = 25.

Table II shows the number of generated friendships and activities and table III summarizes the parameters used for the synthetic data set.

B. Experimental Setup

As in previous evaluations [5, 7], the metric used in our experiments is root mean squared error: $RMSE = \sqrt{\frac{\sum_{(u,i)\in R_{test}}(r_{u,v}-\hat{r}_{u,v})^2}{|R_{test}|}}$, where R_{test} is the set of all pairs (u, v) in the test data. RMSE penalizes large errors more than mean absolute error. The error is evaluated at parameter settings obtained after gradient descent has converged. For different models and methods, convergence is attained after different numbers of iterations, which we also report. Note that by running the gradient descent until convergence we focus on comparing the predictive power of the models, rather than the computational difficulty of optimizing parameters in each. We compare the following seven methods.

LRT Linear Regression on Transactions (LRT) learns a weight for each of the five transactions to predict the strength of the friendships. Our implementation uses the Weka linear regression package [1].

MF Uses Matrix factorization to learn the latent factors without using the baseline predictors and activities.

MF+Base Uses Matrix factorization to learn the latent factors using the baseline predictors, but not using activities. [8]

LRT+MF A weighted combination of the LRT and the MF model. The weights are learned using regression, again with Weka.

CMF Collective matrix factorization uses gradient descent to

Parameter	Interpretation	Generation Method
μ	Average friendship rating in population	Set to 3
σ_b^2	Variance for generating baseline predictors for each user	Set to 0.5
b_u^{rater}	observed deviation of user u as rater from μ	$\mathcal{N}(\{-1, 0, +1\}, \sigma_b^2)$
b_v^{ratee}	observed deviation of user v as rated from μ	$\mathcal{N}(\{-1, 0, +1\}, \sigma_b^2)$
μ_a	Average number of activities between two users	Set to 25
σ_a^2	Variance for generating activities between a pair of users	Set to 15
a_{uv}	Number of activities between user u and v	$max(\mathcal{N}(\mu_a, \sigma_a^2), 0)$
acc_u	Acceptance rate of friendships for users	Set to 0.9,0.7,0.5,0.3,0.1
x	To decide whether a friendship link is kept or not	Randomly generated between 0 and 1
σ_R^2	Variance for generating rating between two users	Set to 0.1
r_{uv}	Rating between user u and v	$\mathcal{N}(\mu + b_u^{rater} + b_v^{ratee} + a_{uv}, \sigma_R^2)$

TABLE III
PARAMETERS USED FOR GENERATING THE SYNTHETIC DATA

learn hidden factors to explain both the strength of the links and the transactional intensities. This model has previously been used mostly for recommendation systems in which the explicit matrix is of type (user-item) and the implicit matrix is of type (user-user) [15, 10]. We adapt the model to fit two (user-user) matrices.

CMF+Base The addition of the baseline predictors to Collective matrix factorization.

TMF Transaction-based Matrix factorization is our model discussed in Section III.

Table V lists the parameters and their meaning used in this evaluation. Since the focus of this paper is not on the prior probabilities, we fix the values of $\lambda_b = 0.001$, $\lambda_u = 0.001$ and $\lambda_v = 0.001$ for all experiments.

C. Data Fusion Experiments

Data fusion, as discussed, is the process of combining data from multiple sources for analysis instead of using each of the sources individually. Among our comparison methods, the CMF and TMF approaches can be viewed as performing data fusion, whereas the other approaches use only one information source (LRT, MF), or model the information sources independently (LRT+MF). We perform experiments to evaluate whether a data fusion approach leads to more accurate predictive performance or not. Table VI illustrates the results for this experiment on the Cloob dataset.

For the parameters of MF, CMF and TMF we used the latent factor dimension $k = 5$ and the optimal value of λ_t, as established by cross-validation. LRT assigns very low weights to each of the activities and predicts values very close to the mean of the strengths of ties, which indicates that the activities are almost independent of the strength of ties. This occurs mainly because many pairs of friends lack any transactions—we refer to this situation as a *zero-transaction* friendship. The MF model performs better than LRT and is able to find factors that perform reasonably well in the dataset. LRT+MF outperforms both the LRT and MF model by using the information from both matrices separately. The TMF approach uses data fusion and leads to the best performance compared to learning weighted combination of the two models.

Zero-Transaction Friendships. To investigate further the importance of data fusion, we separately report results regarding zero-transaction friendships. Such friendships are numerous; the Cloob database records 329,940 of them, or 65% of all friendships. In this case a model based on explicit transaction information does not apply, whereas the effect of latent factor learning is to propagate information from different third-party transactions. For example, if Jack and James are friends with no transactions between them, but Jack has recorded transactions with 20 other friends, and James has recorded transactions with 30 other friends, this information will influence the latent factors for Jack and James, and hence the prediction of the link strength between them. Table VII shows the cross-validation RMSE for zero-transaction friendship pairs. Collective Matrix Factorization (CMF-base) and single matrix factorization (MF-base) show the same error rate on zero-transaction user pairs. This suggests that Collective Matrix Factorization does not propagate information as effectively as Transaction-based Matrix Factorization. More effective propagation would also explain the superior performance of TMF on the entire dataset, shown in Table VI.

Method	RMSE
LRT	1.17 ± 0.012
MF	0.61 ± 0.023
MF + LRT	0.47 ± 0.019
CMF	0.50 ± 0.015
TMF	**0.28 ± 0.008**

TABLE VI
DATA FUSION.

Method	Result
MF	0.55 ± 0.032
MF-Base	0.31 ± 0.023
CMF	0.47 ± 0.012
CMF-Base	0.31 ± 0.041
TMF	**0.24 ± 0.014**

TABLE VII
PERFORMANCE ON
ZERO-TRANSACTION USER PAIRS

D. Impact of Transaction Information

It is interesting to study the effect of the transactional intensities on the link strength. Parameter λ_T in the TMF

Method	k = 2		k = 5		k = 10	
	RMSE	itr	RMSE	itr	RMSE	itr
MF	0.67 ± 0.045	143	0.61 ± 0.023	88	0.71 ± 0.078	110
MF-Base	0.40 ± 0.014	246	0.35 ± 0.025	284	0.38 ± 0.030	331
CMF	0.90 ± 0.010	269	0.50 ± 0.015	422	2.52 ± 0.239	300
CMF-Base	0.38 ± 0.018	299	0.33 ± 0.010	308	0.36 ± 0.005	662
TMF	**0.35 ± 0.015**	403	**0.28 ± 0.008**	627	**0.22 ± 0.032**	1165

TABLE IV
AVERAGE RMSE AND NUMBER OF ITERATIONS FOR DIFFERENT DIMENSIONALITY IN THE CLOOB DATASET.

Parameter	Interpretation	Estimation Method
λ_b	L2-regularizer for baseline predictors	Fixed for all methods
λ_u	L2-regularizer for raters	Fixed for all methods
λ_v	L2-regularizer for ratees	Fixed for all methods
λ_t	L2-regularizer for transaction intensities	Evaluated with different values
k	dimensionality of the latent factors	Evaluated with different values
U_u	latent factors modeling the behavior of raters	Gradient descent
V_v	latent factors modeling the behavior of ratees	Gradient descent

TABLE V
THE INTERPRETATION AND THE METHODS OF ESTIMATION FOR THE NEW PARAMETERS USED IN THE CLOOB MODELS. PREVIOUS PARAMETERS (TABLE III) WERE ESTIMATED AS FOLLOWS. μ :EMPIRICAL MEAN; b_u^{rater}, b_v^{ratee}: GRADIENT DESCENT.

model controls the influence of the transactional intensities. Setting $\lambda_T = 0$ totally ignores the transactions to make the model similar to the MF+base model. Large values of λ_T indicate a strong influence of the transaction information used to derive predictive features, moving close to the LRT model. Figures 2 and 3 shows the transactional intensities' influence. Figure 2 illustrates that TMF does much better on the synthetic dataset, which was to be expected since the data satisfy the assumptions of the model. The MF methods do not perform well because they fail to make use of the transaction information. Figure 3 shows that methods that incorporate base-line predictors do well on the Cloob dataset. In recommendation problems like the Netflix challenge, the MF+base approach is a state-of-the-art method, so its strong performance is to be expected. Still, data fusion with the TMF model leads to further improvement. In both datasets, we observe that for the data fusion methods (TMF and CMF), as we increase λ_T, the performance first improves up to a point and then starts to drop. Therefore, the transactional intensities are helpful with link strength; however, if too much weight is assigned to them, they would override the effects of the friendship ratings. For the parameters of MF and TMF we used the latent factor dimension $k = 5$.

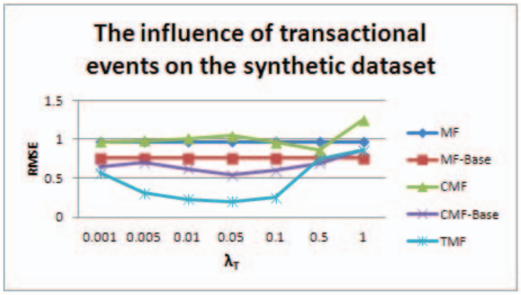

Fig. 2. Transactional intensities influence on Link strength in the synthetic dataset.

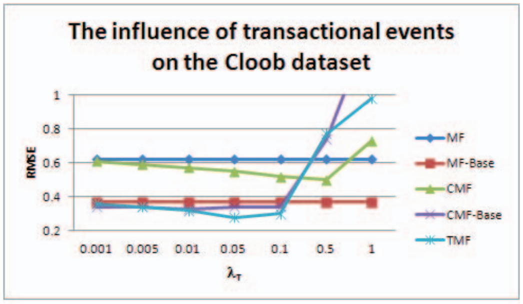

Fig. 3. Transactional intensities influence on Link strength in the Cloob dataset.

E. Number of Latent Factors

We examine the impact of the dimensionality of the latent factors k. In experiments from other sections the λ_T with the best performance is used. Tables VIII and IV report the average RMSE and the number of iterations for different dimensions of the latent factors. The number of iterations is reasonable for a gradient descent method, especially considering the high number of hidden factors to be assigned. On both datasets, the TMF model performs best on each dimension.

On the synthetic dataset, MF, MF-Base, and CMF have their best performance when $k = 1$. This is because the data size is small and bigger values of k lead to over fitting. CMF-

Method	k = 1		k = 5		k = 10	
	RMSE	itr	RMSE	itr	RMSE	itr
MF	0.80 ± 0.026	191	0.97 ± 0.283	344	3.37 ± 0.299	445
MF-Base	0.65 ± 0.011	258	0.76 ± 0.217	485	2.71 ± 0.310	678
CMF	0.70 ± 0.007	115	0.86 ± 0.174	434	2.52 ± 0.239	300
CMF-Base	0.64 ± 0.011	283	0.54 ± 0.091	389	2.58 ± 0.190	586
TMF	**0.61 ± 0.012**	295	**0.20 ± 0.002**	340	**0.21 ± 0.003**	413

TABLE VIII
AVERAGE RMSE AND NUMBER OF ITERATIONS FOR DIFFERENT DIMENSIONALITIES IN THE SYNTHETIC DATASET.

Base and TMF have their best performance on $k = 5$. On the cloob dataset, since the dataset is much larger, higher dimensionality improves all of the models. MF, MF-Base, CMF, and CMF-Base have their best performance when $k = 5$. The performance of TMF improves with dimensionality, which suggests that the model is well-suited to the dependencies in the data.

VI. LIMITATIONS AND FUTURE WORKS

1) The experiments of this paper *aggregated* the values of different transactional intensities to create a single implicit network. This method may be inappropriate, if the effect of more frequent type of activities can overcome the effect of more important, yet less frequent type of activities. We will investigate the use of multiple implicit networks with different weights in the future.

2) Social networks often contain useful temporal information. The temporal behavior of users may play an important role in link strength prediction. Consider a retired user that has not had any activities or new friendships over a long period, but used to be a very active user. Without considering the temporal side of his behavior, the false prediction is made that this user is very likely to obtain new strong friendships. The Cloob dataset contains timestamps, and supports investigation of temporal information for better link strength prediction.

3) Link prediction models often combine latent features with observed features [4, 12]. Observed features may include descriptive attributes of users (e.g., culture, behavior, personality), and features derived from the link topology (e.g.,number of friends in common). A promising direction is to combine latent and observed features for link strength prediction.

VII. CONCLUSION

We introduced a discriminative matrix factorization model where the goal is to predict the values of a single explicit matrix, and implicit matrices are used to weight the importance of connections between users. We derived gradient descent equations for learning two sets of latent factors for each user, as a rater and as a ratee. In experiments on generated synthetic data and on real-world data, our transaction-MF model outperforms collective matrix factorization, single table matrix factorization, as well as regression models that analyze the data matrices separately.

REFERENCES

[1] Mark Hall et al. The weka data mining software: an update. *SIGKDD Explorations*, 11(1):10–18, 2009.

[2] Eric Gilbert. Predicting tie strength in a new medium. In *CSCW '12*, pages 1047–1056. ACM, 2012.

[3] Eric Gilbert and Karrie Karahalios. Predicting tie strength with social media. In *CHI*, pages 211–220, 2009.

[4] P. D. Hoff. Multiplicative latent factor models for description and prediction of social networks. *Computational and Mathematical Organization Theory*, 2007.

[5] Mohsen Jamali and Martin Ester. A matrix factorization technique with trust propagation for recommendation in social networks. In *RecSys*, pages 135–142, 2010.

[6] Indika Kahanda and Jennifer Neville. Using transactional information to predict link strength in online social networks. In *ICWSM*, 2009.

[7] Yehuda Koren. Collaborative filtering with temporal dynamics. *Commun. ACM*, 53(4):89–97, 2010.

[8] Yehuda Koren, Robert M. Bell, and Chris Volinsky. Matrix factorization techniques for recommender systems. *IEEE Computer*, 42(8):30–37, 2009.

[9] David Liben-Nowell and Jon Kleinberg. The link-prediction problem for social networks. *J. Am. Soc. Inf. Sci. Technol.*, 58:1019–1031, May 2007.

[10] Hao Ma, Irwin King, and Michael R. Lyu. Learning to recommend with social trust ensemble. In *SIGIR*, pages 203–210. ACM, 2009.

[11] Paolo Massa and Paolo Avesani. Trust-aware recommender systems. RecSys '07, pages 17–24, 2007.

[12] A. Menon and C. Elkan. Link prediction via matrix factorization. *Machine Learning and Knowledge Discovery in Databases*, pages 437–452, 2011.

[13] Jennifer Neville, Brian Gallagher, and Tina Eliassi-Rad. Evaluating statistical tests for within-network classifiers of relational data. In *ICDM*, pages 397–406, 2009.

[14] R. Xiang, J. Neville, and M. Rogati. Modeling relationship strength in online social networks. In *WWW'*, pages 981–990. ACM, 2010.

[15] Shuang-Hong Yang, Bo Long, Alex Smola, Narayanan Sadagopan, Zhaohui Zheng, and Hongyuan Zha. Like like alike: joint friendship and interest propagation in social networks. WWW '11, pages 537–546.