Optimizing MAP in Joint Multi-Relational Models for Recommendations in the Academic Network*

Zaihan Yang Dawei Yin Brian D. Davison
Department of Computer Science & Engineering
Lehigh University
{zay206, day207, davison}@cse.lehigh.edu

Abstract

In this paper, we target at four specific recommendation tasks in the academic environment: the recommendation for author coauthorships, paper citation recommendation for authors, paper citation recommendation for papers, and publishing venue recommendation for author-paper pairs. Different from previous work which tackles each of these tasks separately while neglecting their mutual effect and connection, we propose a joint multi-relational model that can exploit the latent correlation between relations and solve several tasks in a unified way. Moreover, for better ranking purpose, we extend the work maximizing MAP over one single tensor, and make it applicable to maximize MAP over multiple matrices and tensors. Experiments conducted over two real world data sets demonstrate the effectiveness of our model: 1) improved performance can be achieved with joint modeling over multiple relations; 2) our model can outperform three state-of-the art algorithms for several tasks.

Keywords: Recommender systems, matrix/tensor factorization, joint modeling, MAP, latent factor model

1 Introduction

People can conduct many activities in academic environment: publishing papers, collaborating with other authors, or citing other papers/authors. Theses activities are sometimes not easy to fulfill. For example, reading and therefore citing new

*Technical Report LU-CSE-14-002, Dept. of Computer Science and Engineering, Lehigh University, Bethlehem, PA, 18015. A summary of this report is published elsewhere [22].

published papers is one of the most important tasks that a researcher should conduct for research, however, to find relevant and referential scientific literature from hundreds of thousands of publications is a time-consuming and labor-intensive task especially with the rapid development of Internet which makes published papers easy to be accessed. To better facilitate such activities, information needs have arisen for developing systems that can automatically help to predict or recommend proper venues to submit, papers to cite, and authors to collaborate. In this paper, we focus on the prediction task in academic environment, and particularly pay attention to the following four tasks: the prediction on publishing venues, collaborators, cited papers for authors, and cited papers for papers.

Even though individual systems or algorithms have been proposed to tackle each of the four tasks separately, which we will review in later sections, limitations still remain. Most of the previous methods only focus on one single type of relationship while neglect to explore the mutual interaction among different relationships. In a real complicated academic environment, which often consists of heterogeneous nodes and links, each scientific factor can play different roles, and participate in different activities. For example, individual researcher can work as an author to write paper, as a collaborator to work with another researcher, or to be cited by another researcher. The entire academic network is composed of multiple relations that mutually affect each other.

To better model this multi-relational academic activities and to provide good recommendations, several challenges remain:

- **Coupled high order data**: as mentioned above, there are multi-typed scientific entities in the academic environment, playing different roles and participating in different activities. These activities are often coupled. It is quite natural for a paper that has a large number of citations from other papers to be cited by more authors.

- **Cold start problem**: the cold start problem is a typical problem in recommender systems. Take the task of citation recommendation for papers as one example, some most recently published papers will hardly be cited since they have never been cited before by other papers or authors, even though they are highly relevant to a topic or may have great contribution in a certain field.

- **Personalization support for authors:** Researchers play an important role in many activities, and they may have different preferences in selecting which paper to cite, or which venue to submit, even though those papers or venues focus on similar topics.

- **Interest evolution for authors:** The interest of researchers evolves over time. Even though they keep on working in one research field, their research focus and methods may change.

To tackle these challenges, we propose a joint multi-relational model referred as the JMRM model which directly models several groups of coupled activities in the academic environment and provide a more general framework that can solve several prediction tasks simultaneously in a unified way.

Our model is fundamentally the latent factor collaborative-filtering(CF)-based model, in which each relation can be represented as a matrix or higher-dimensional matrix. However, the following three characteristics distinguish our model from previous ones. **Firstly**, our model is composed of multiple matrices or tensors, each of which indicate one relation in the academic environment, and are highly coupled with each other. **Secondly**, we integrate the temporal information into the generation of several matrices to better reflect the involution over authors' preferences; **Thirdly**, we choose the objective function for solving the model as maximizing the mean average precision (MAP) as compared to most of the previous work minimizing the predicting error (RMSE). MAP is a ranking-based standard IR measure for which errors at top of the ranking list will lead to a higher penalty than errors at lower places of the ranking list. This top-heavy biased property makes MAP particularly suitable to work as the objective function for recommender systems, since most people will only pay attention to the top ranked results in the recommendation list. For this reason, we choose to maximize MAP as our objective function.

To sum up, the main contribution of our work are as follows:

- we propose a joint multi-relational model which integrates several coupled relations in an academic environment. This model is particularly designed for four recommendations: the prediction task on paper submission for venues, co-authorship prediction, paper citation prediction for authors, and paper citation prediction for papers.

- we choose to maximize MAP as the objective function for solving the model, and extend the tensor factorization approach optimizing MAP into a more general framework that can maximize MAP for coupled multiple matrices and tensors.

- experimental evaluation over two real world data sets demonstrate the capability of our model in four recommendation tasks, as they show improved performance as compared to three state-of-the-art algorithms.

2 Related Work

Recently, researchers have explored to enhance the traditional latent factor models by incorporating additional features or content of participating entities. Typical works include the 'Regression Based Factor Models' [1], the CTR model [20], and the 'feature-based matrix factorization' [3] model. However, all of these three models can only cope with the two-order data interactions, and cannot be model higher-order data structures. 'Factorization Machine' model proposed by Rendle [12] combines latent factorization model with SVM. Compared with these work, our model incorporate both features for papers and authors, making it capable to model more than two-order data interactions.

The second direction of development for latent factor model emphasizes on joint modeling multi-relational relations. One typical work: the 'collective matrix factorization' model [17] however is only limited to be two or three relations. Most recently, Yin et al. [24] proposed a 'Bayesian probabilistic relational-data Analysis' (BPRA) model which extends the BPMF and BPTF model by making it applicable to arbitrary order of coupled multi-relational data structures. However, the model isbased upon point-wise RMSE optimization, different from our targeted ranking-based optimization.

Several ranking-based optimization models have been proposed to replace optimizing point-wise measures, such as RMSE or MSE. One typical work: the 'Bayesian Personalized Ranking' (BPR) model [13] minimizes the AUC metric by using a smooth version of the hinge loss.The method that is most similar to our work is the TFMAP model [15], which proposes a method to approximate and optimize the MAP measure. However, their model is for user-item-context recommendation, and is only able to deal with one single tensor relation, which are both different from our work in this paper.

We then summarize some relevant work with each specific recommendation task considered in this paper. Future paper citation recommendation is the most widely explored problem. We categorized existing works into three groups. In the first group, neighborhood based CF models along with graph-based link prediction approaches are widely used to tackle the citation recommendations for a given author or paper with a partial list of initial citations provided, typical works in this category include [11], [26], [18] and more. In the second group of approach, probabilistic topic modeling is used for citation list generation.In the third group, citation context (the text around citation mentions) is utilized. Typical work includes the context-award citation recommendation work and its extensions proposed by He et al. [7, 6] Despite of these existing work, few work has be developed using CF latent factor models for recommendation, excluding the CTR model.

Coauthor-ship recommendation is mostly tackled by using graph-based link prediction approach. The most representative work is proposed by Liben-Nowell [10], which measures the performance on using several graph-based metrics. The work on predicting future conference(venue) submission is seldom explored. Yang et al. [21] proposed an extended version of the neighborhood CF model to solve this problem recently. In their model, they incorporate stylometric features of papers and further distinguish the importance of four different types of neighboring papers via parameter tuning and optimization. Similar to the paper citation work, few existing work applies the latent factor model based CF approach, which is different from the work in this paper.

3 Preliminary Experiments

In this section, we conducted some simple experiments on two real world data sets: the **ACM data set** and **ArnetMiner data set** to analyze the characteristics of activities and relationships among scientific factors in the academic environments.

3.1 Data sets

The **ACM data set** is a subset of the **ACM Digital Library**, from which we crawled one descriptive web page for each 172,890 distinct papers having both title and abstract information. For each published paper, we extracted the information about its authors and references. Due to possible author names' ambiguity, we represent each candidate author name by a concatenation of the first name and last name, while removing all the middle names. We then use exact match to merge candidate author names. Finally, we obtain 170,897 distinct authors, and 2097 venues.

The **ArnetMiner data set** is the data set 'DBLP-Citation-network V5' provided by Tsinghua University for their ArnetMiner academic search engine [19]. It is the crawling result from the ArnetMiner search engine on Feb 21st, 2011 and further combined with the citation information from ACM. The original data set is reported to have 1,572,277 papers and to include 2,084,019 citation-relationships. After carrying out the same data processing method as we did for the ACM data set, we find 1,558,415 papers, 795,385 authors and 6010 venues.

3.2 Coupled Relations

We are first interested in finding out whether multiple relations in an academic environment are coupled. As a simple test example, we compute for each author in both data sets his/her total number of publications, citations and coauthors, and

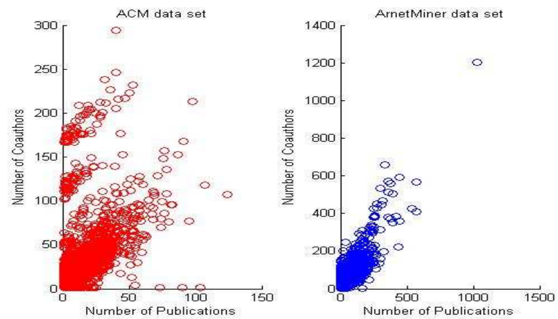

Figure 1: Correlation between Number of Publications and Coauthors

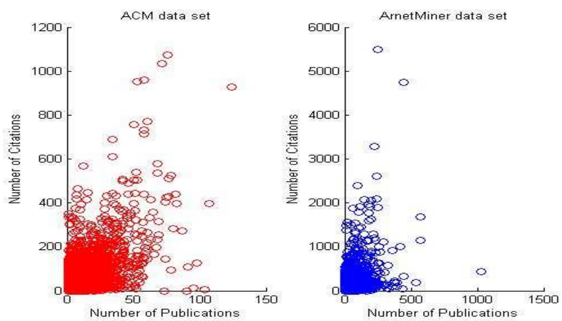

Figure 2: Correlation between Number of Publications and Citations

evaluate the correlation between these three factors. Figure 1 and 2 show our results.

As we can see, there exists an obvious linear positive correlation between number of publications and coauthors, indicating that under most circumstances, the more coauthors you have, the more publications you can achieve. This observation is compatible with our common sense. However, the correlation between publication number and citation number is not so obvious. As shown in Figure 2, we have many data points scattered in the lower-left corner of the figure, indicating that some authors who do not publish many papers can also achieve high citation.

3.3 Cold Start Problem

We evaluate the changes in papers' ability in attracting citation to demonstrate the existence of cold start problem in the academic environment. We average the number of citations each paper retrieves in both data sets on a yearly basis. This simple statistical result, as shown in Table 1, indicates that averagely a newly published paper begins to retrieve citations 2 more years later than its publication. However,

Table 1: Statistics on Papers' Citations

Data set	No. of Papers	First Citation after publication	Avg. Citation Frequency
ACM	55392	2.0350	0.9693
Arnet	315831	2.7599	0.8528

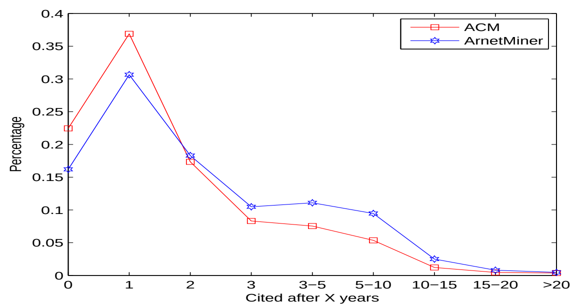

Figure 3: Average number of citations change over time

after that, it just costs around 0.97 years and 0.85 years for papers in ACM and ArnetMiner data set to retrieve one new citation. Another simple statistics, as shown in Figure 3, indicates that papers on average can achieve most of their citations in the following year of its publication, and that number gradually drops as time evolves.

3.4 Interests evolution

We evaluate the evolution of authors' research interests by checking the changes in their publishing venues. For ACM data set, we collect for each author the publishing venues of his papers published before 2003 and after 2003 (including 2003) as two sets, and adopt the Jaccard Similarity method to detect the similarity/difference between these two sets. For ArnetMiner data set, we set the year point as 2006. We choose the year point by guaranteeing that the average number of distinct venues of authors in each separate data set is equivalent before and after that year point. Table 2 shows the results.

As indicated, the average Jaccard values for both data sets are pretty small, indicating that authors have a diversified publishing venue list. Authors chose different venues to submit, indicating that their research focus may evolve over time.

Table 2: Statistics on Changes of Publishing Venues

	ACM ($Y = 2003$)	**ArnetMiner** ($Y = 2006$)
No. of Authors	23358	188143
Avg No.Venues before Y	2.73	5.14
Avg No.Venues after Y	2.75	5.09
Avg Jaccard Similarity	0.0946	0.1188

Table 3: Notations

K	Number of entity types $(K = 6)$
a, p, p_c, v, w, a_f	represents author, citing paper cited paper, venue, word and feature entity type respectively.
a_i	entity of type a with index i
k	entity type. $k \in a, p, p_c, v, w, a_f$
N_k	Number of entities of type k in data corpus
D	Dimension for latent vector
V	Number of relations $(V = 6)$
θ_k	Latent matrix for entities of type k
θ_{k_t}	Latent feature vector for t^{th} entity of type k

4 Joint Multi-Relational Model (JMRM): Model Design and Generation

Inspired by the information needs for developing recommender systems in the academic environment and in order to fulfill the challenges, we propose a joint multi-relation model. Our model is designed for four particular recommendation tasks in the academic environment, each of which represents one academic activity, and induces one relation. Therefore, we have four main relations in the model: the author-paper-venue relation (represented as the APV-tensor), author-author-collaboration relation (AA-matrix), author-paper-citation relation (AP-matrix), and paper-paper-citation relation (PP-matrix). Figure 4 shows the framework of the model. In order to deal with the cold start problem and better support authors' personalization, we further incorporate additional features for papers and authors. In the current work, we only consider the pure paper content as paper features, and we use the PW-matrix to represent it. We model authors and their features as the AF-matrix, and will introduce more detailed features for authors in the following section.

We formally describe as follows the four recommendation/prediction tasks and introduce how the corresponding relation is constructed.

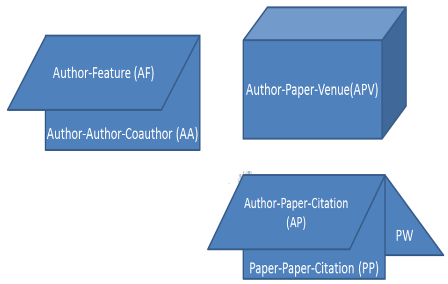

Figure 4: Coupled Matrices and Tensor

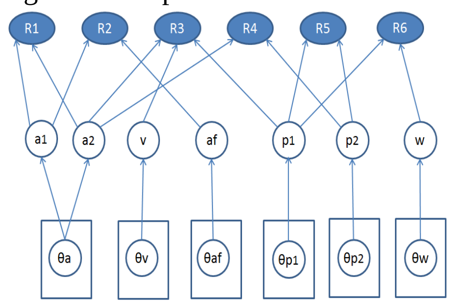

Figure 5: Graphical Model Representation

Task 1 Publishing venue prediction: this task involves predicting the publishing venue v_k for a given author-paper pair (a_i, p_j) such that paper p_j written by author a_i is published in venue v_k.

Corresponding relation **APV-tensor: the author-paper-venue relation** This three-order relation consisting of triples $<$author-paper-venue$>$ indicates the publishing venue selection for papers with known authors.we have: $APV(a_i, p_j, v_k) = 1$ if paper p_j written by author a_i is published in venue v_k; Otherwise, $APV(a_i, p_j, v_k) = 0$.

Task 2 Recommend paper citations for authors: this task involves recommend a set of papers $\boldsymbol{p}$ for a given author a_i such that each paper p_j ($p_j \in \boldsymbol{p}$) is cited at least once by author a_i.

Corresponding relation **AP-matrix: the author-paper-citation relation** The AP matrix models the citation relationship between authors and papers. An author may cite a paper multiple times at different times, and the preference of the author over papers may also change over time. In order to model this temporal evolution property, we first generate a three-order tensor incorporating the temporal factor as the third dimension, and then collapse the tensor into a two-dimensional matrix by aggregating the number of citations at different years with a time decay function. Given an author a_i, and a paper p_j cited by a_i, the number of times p_j is cited by a_i on year t_k (the value for entry $< a_i, p_j, t_k >$) can be retrieved as: $E(a_i, p_j, t_k) = \sum_{p_{ai} \in \boldsymbol{p_{ai}}} \delta(y(p_{ai}) = t_k \wedge p_j \in \boldsymbol{c_{p_{ai}}})$ where p_{ai} is any paper published by a_i, $\boldsymbol{p_{ai}}$ is the publication set of a_i. $\boldsymbol{c_{p_{ai}}}$ is the set of all cited papers of p_{ai}. Function $y(p_{ai})$ retrieves the publication year of p_{ai}, and $\delta(c)$ is a function which returns 1 if condition c is satisfied and 0 otherwise. We aggregate the citations at different time points based upon the hypothesis that authors' interests decay over time, and therefore more recent citation contribute more heavily than older citation. We penalize the old citations by introducing an exponential decay kernel function. The entry $< a_i, p_j >$ for the collapsed author-paper matrix can thus be defined as: $E_{AP}(a_i, p_j) = \sum_{t_k=T_1}^{T_2} e^{-\beta(T_2-t_k)} \cdot E(a_i, p_j, t_k)$ where T_1 and T_2 are the earliest and last year for paper publication in the data set, and β is the decay rate.

Task 3 Recommend coauthors for authors: this task involves recommend a set of authors $\boldsymbol{a}$ for a given author a_i such that for each author $a_j \in \boldsymbol{a}$, there exists a coauthorship between a_i and a_j.

Corresponding relation **AA-matrix: the author-author-collaboration relation** The AA-matrix indicates the collaboration, an important social interactions between pairs of authors. Similar to the changing interests' of authors over papers, researchers may also change to work with others researchers in different time periods. We follow the same procedure as introduced for the AP-matrix generation by first constructing the author-author-time tensor, and then collapse it into author-author matrix. The entry for $< a_i, a_j >$ can thus be determined by: $E_{AA}(a_i, a_j) = \sum_{t_k=T_1}^{T_2} e^{-\beta(T_2-t_k)} \cdot E(a_i, a_j, t_k)$ where $E(a_i, a_j, t_k)$ is the number of times author a_i collaborates with a_j on year t_k.

Task 4 Recommend paper citations for papers: this task involves recommend a set of papers $\boldsymbol{p}$ for a given paper p_i such that each paper p_j ($p_j \in \boldsymbol{p}$) is cited at least once by paper p_i.

Corresponding relation **PP-matrix: the paper-paper-citation relation** The generation of the PP-matrix is different from that of the AP-matrix or AA-matrix, since each paper can only cite another paper once. However, there also exists temporal influence, as a paper may cite a paper published long time ago, or a more recent one. Suppose we have three papers $p1$, $p2$ and $p3$, published in $y1$, $y2$ and $y3$ respectively ($y1 \leq y2 \leq y3$), and we have paper $p3$ cites $p2$ and $p1$. In our work, we assume that $p2$ will have a greater contribution in presenting the topic interests or preferences for $p3$ than $p1$, since in publishing papers, we often need to review and compare with those most recently published and state-of-the-art papers. With this assumption, we have for each entry $< p_i, p_j >$ indicating that paper p_i cites p_j in the PP-matrix as: $E_{PP}(p_i, p_j) = e^{-\beta(y(p_i)-y(p_j))}$, where $y(p_i)$ and $y(p_j)$ returns the publishing year for p_i and p_j respectively.

PW-matrix: the paper-word relation PW-matrix indicates the features of papers. In current work, we only consider the pure content of papers, and therefore we collect the top returned words in the data set with higher frequency. Each entry of $< p_i, w_j >$ indicates the term frequency of word w_j in paper p_i.

AF-matrix: the author-feature relation We identify 20 distinctive features for authors listed in Table 4 to represent the personalized property of an author from three aspects.

Table 4: Author Features

Feature Category	**Feature**
Simple bibliographic	total publicationNo ; total citationNo ; H-index [8]; G-index [4]; Rational H-index distance [14]; Rational H-index X [14]; E-index [25]; Individual H-index [2]; Normalized individual H-index [5]
Network-based	PageRank score on coauthor network; PageRank score on citation network
Temporal-based	CareerTime [23]; LastRestTime [23]; PubInterval [23]; Citation Influence Ratio [23]; Contemporary H-index [16]; AR-index [9] AWCR-index [5]; Avg Publication number; avg Citation number

5 Joint Multi-Relational Model: Algorithm

5.1 Preliminary Notations

As shown in Figure 4, our joint multi-relational model consists of six relations generated by authors, papers, venues, words and features entities. It is noticeable to mention that we distinguish the 'paper' entity into two different entities types: the citing papers and cited papers, and therefore we altogether have six entity types.

The joint multi-relational model is a further extension and generalization of the classical matrix or tensor factorization, in which each entity in the interactions can be represented by a latent feature vector in $\mathbb{R}^D$, where D is typically a small number. By doing this, each tensor or matrix can be factorized into lower rank approximations. Figure 5 shows the graphical model for the data factorization associated with our model. The lower-dimensional latent vectors are denoted by $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_K)$ ($K = 6$), where for each $k \in K$ $\theta_k = (\boldsymbol{\theta}_{k1}, \ldots, \boldsymbol{\theta}_{kN_k}) \in R^{N_k \times D}$.

5.2 Model Factorization maximizing MAP

5.2.1 Computing MAP

We choose to maximize MAP as our objective function due to its top-heavy bias property. Two questions remain for incorporating MAP into matrix/tensor factorization: how to represent the 'rank' of the entities and therefore compute the MAP scores based upon the latent feature vectors. We follow the same idea proposed in paper [15] to smoothly approximate MAP, and make it appropriate to be used for both tensors and matrices. Since our model contains one tensor and five matrices, for better illustration, we choose to take the APV-tensor and AP-matrix as two examples to show how to compute the MAP scores. The same method can be applied to the other four matrices.

In a tensor like APV-tensor, the predicted value for each entry $< a_i, p_j, v_m >$ can be computed as: $\hat{f}_{a_i p_j v_m} = \langle \theta_{a_i}, \theta_{p_j}, \theta_{v_m} \rangle = \sum_{d=1}^{D} \theta_{a_i d} \theta_{p_j d} \theta_{v_m d}$ where D is the dimension for latent vector.

Similarly, In a matrix like AP-matrix, the predicted value for each entry $< a_i, p_j >$ can be computed as:

$\hat{f}_{a_i p_j} = \langle \theta_{a_i}, \theta_{p_j} \rangle = \sum_{d=1}^{D} \theta_{a_i d} \theta_{p_j d}$

Under these schemes, suppose v_m in triple $< a_i, p_j, v_m >$ is the entity that needs to be ranked, and p_j in tuple $< a_i, p_j >$ is the entity that needs to be ranked, then we can directly approximate $1/r_{a_i p_j v_m}$ for v_m and $1/r_{a_i p_j}$ for p_j by:

$\frac{1}{r_{a_i p_j v_m}} \approx g(\hat{f}_{a_i p_j v_m}) = g(\langle \theta_{a_i}, \theta_{p_j}, \theta_{v_m} \rangle)$

$\frac{1}{r_{a_i p_j}} \approx g(\hat{f}_{a_i p_j}) = g(\langle \theta_{a_i}, \theta_{p_j} \rangle)$

where function $g(\cdot)$ is the sigmoid function satisfying $g(x) = \frac{1}{1+e^{-x}}$.

Correspondingly, the loss function in terms of the MAP values for APV-tensor and AP-matrix can be computed as in equation 1:

$$
\begin{aligned}
L_{apv} &= MAP_{apv} = \frac{1}{N_a N_p} \sum_{i=1}^{N_a} \sum_{j=1}^{N_p} \frac{1}{\sum_{t=1}^{N_v} f_{APV_{a_i p_j v_m}}} \\
&\times \sum_{t1=1}^{N_v} f_{APV_{a_i p_j v_{t1}}} g(\langle \theta_{a_i}, \theta_{p_j}, \theta_{v_{t1}} \rangle) \\
&\times \sum_{t2=1}^{N_v} f_{APV_{a_i p_j v_{k2}}} g(\langle \theta_{a_i}, (\theta_{v_{t2}} - \theta_{v_{t1}}), \theta_{p_j} \rangle) \qquad (1) \\
L_{ap} &= MAP_{ap} = \frac{1}{N_a} \sum_{i=1}^{N_a} \frac{1}{\sum_{j=1}^{N_p} f_{AP_{a_i p_j}}} \\
&\times \sum_{t1=1}^{N_p} f_{AP_{a_i p_{t1}}} g(\langle \theta_{a_i}, \theta_{p_{t1}} \rangle) \\
&\times \sum_{t2=1}^{N_p} f_{AP_{a_i p_{t2}}} g(\langle \theta_{a_i}, (\theta_{P_{t2}} - \theta_{P_{t1}}) \rangle) \qquad (2)
\end{aligned}
$$

To compute the loss function for matrix AA, PP, PW and AF, we can follow the same way as we do for the AP matrix.

5.2.2 Optimization

We introduced the loss function for each individual matrix/tensor in the last section. The overall loss function for this multi-relational model is therefore a summation over all individual loss functions plus the regularization terms to prevent over-fitting, as shown in Equation 3. We use Ω to denote the regularization terms, where $\| \cdot \|$ indicates the Frobenius norms.

We choose to use gradient ascent to solve this optimization problem. For each relation (matrix or tensor) in the model, we alternatively perform gradient ascent on the latent feature vector for one entity at each step, while keep the other latent vectors unchanged. The gradients for the same entity across different relations will be merged. The same process will be repeated for a certain number of times, or until it finally converges with no further updates on all latent feature vectors. To better illustrate, we list below the gradients for the author, paper and venue entity in the APV-tensor, and author and paper entity in the AP-matrix. Similar process can be applied into other entities in other relations. We leave the generalized updating forms for a model with K $N \times M$ matrices for future's work.

$$
\begin{aligned}
L = L_{APV} + L_{AA} + L_{AP} + L_{PP} + L_{PW} + L_{AF} + \Omega \\
\Omega = \sum_{k \in a,p,p_c,v,w,a_f} \frac{\lambda_{\boldsymbol{\theta_k}}}{2} \parallel \boldsymbol{\theta_k} \parallel^2
\end{aligned}
\tag{3}
$$

For one particular author a_i, paper p_j and venue v_m in the APV-tensor, the gradients for updating their corresponding latent vector θ_{a_i}, θ_{p_j} and θ_{v_m} can be computed as follows. For notation convenience, we adopt the following substitutions:

$$
\begin{aligned}
\hat{f}_{APV_{a_i p_j v_m}} &= \langle \theta_{a_i}, \theta_{p_j}, \theta_{v_m} \rangle \\
\hat{f}_{APV_{a_i p_j (v_{t_2} - v_{t_1})}} &= \langle \theta_{a_i}, \theta_{p_j}, (\theta_{v_{t_1}} - \theta_{v_{t_1}}) \rangle
\end{aligned}
$$

$$
\begin{aligned}
\frac{\partial L_{APV}}{\partial \theta_{a_i}} &= \sum_{s=1}^{N_p} \frac{1}{\sum_{t=1}^{N_v} f_{APV_{a_i p_s v_t}}} \sum_{t_1=1}^{N_v} f_{APV_{a_i p_s v_{t_1}}} \\
&\times [\delta_1(\theta_{p_s} \odot \theta_{v_{t_1}}) + g(\hat{f}_{APV_{a_i p_s v_{t_1}}}) \\
&\times \sum_{t_2=1}^{N_v} f_{APV_{a_i p_s v_{t_2}}} g'(\hat{f}_{APV_{a_i p_s (v_{t_2} - v_{t_1})}}) \\
&\times (\theta_{p_s} \odot \theta_{v_{t_2}})] - \lambda \theta_{a_i} \\
\frac{\partial L_{APV}}{\partial \theta_{p_j}} &= \sum_{s=1}^{N_a} \frac{1}{\sum_{t=1}^{N_v} f_{APV_{a_s p_j v_t}}} \sum_{t_1=1}^{N_v} f_{APV_{a_s p_j v_{t_1}}} \\
&\times [\delta_1(\theta_{a_s} \odot \theta_{v_{t_1}}) + g(\hat{f}_{APV_{a_s p_j v_{t_1}}}) \\
&\times \sum_{t_2=1}^{N_v} f_{APV_{a_s p_j v_{t_2}}} g'(\hat{f}_{APV_{a_s p_j (v_{t_2} - v_{t_1})}}) \\
&\times (\theta_{a_s} \odot \theta_{v_{t_2}})] - \lambda \theta_{p_j} \\
\frac{\partial L_{APV}}{\partial \theta_{v_m}} &= \sum_{s=1}^{N_a} \sum_{d=1}^{N_p} \frac{f_{APV_{a_s p_d v_m}} (\theta_{a_s} \odot \theta_{p_d})}{\sum_{t_1=1}^{N_v} f_{APV_{a_s p_d v_{t_1}}}} \\
&\times \sum_{t_2=1}^{N_v} f_{APV_{a_s p_d v_{t_2}}} [g'(\hat{f}_{APV_{a_s p_d v_m}}) \\
&\times g(\hat{f}_{APV_{a_s p_d (v_{t_2} - v_m)}}) + (g(\hat{f}_{APV_{a_s p_d v_{t_2}}}) \\
&- g(\hat{f}_{APV_{a_s p_d v_m}})) g'(\hat{f}_{APV_{a_s p_d (v_{t_2} - v_m)}})] \\
&- \lambda \theta_{v_m}
\end{aligned}
\tag{4}
$$

where

$$\delta_1 = g'(\hat{f}_{APV_{a_i p_j v_m}}) \sum_{t_1=1}^{N_v} f_{APV_{a_i p_j v_{t_1}}} g(\hat{f}_{APV_{a_i p_j (v_{t_1} - v_m)}}) \tag{5}$$
$$- g(\hat{f}_{APV_{a_i p_j v_m}}) \sum_{t_1=1}^{N_v} f_{APV_{a_i p_j v_{t_1}}} g'(\hat{f}_{APV_{a_i p_j (v_{t_1} - v_m)}})$$

For one author a_i and paper p_j in the AP-matrix:

$$\begin{aligned}
\frac{\partial L_{AP}}{\partial \theta_{a_i}} &= \frac{1}{\sum_{t=1}^{N_p} f_{AP_{a_i p_t}}} \sum_{t_1=1}^{N_p} f_{AP_{a_i p_{t_1}}} [\delta_2(\theta_{p_{t_1}}) \\
&+ g(\hat{f}_{AP_{a_i p_{t_1}}}) \sum_{t_2=1}^{N_p} f_{AP_{a_i p_{t_2}}} g'(\hat{f}_{AP_{a_i (p_{t_2} - p_{t_1})}})(\theta_{p_{t_2}})] \\
&- \lambda \theta_{a_i} \\
\frac{\partial L_{AP}}{\partial \theta_{p_j}} &= \sum_{s=1}^{N_a} \frac{f_{AP_{a_s p_j}}(\theta_{a_s})}{\sum_{t_1=1}^{N_p} f_{AP_{a_s p_{t_1}}}} \times \sum_{t_2=1}^{N_p} f_{AP_{a_s p_{t_2}}} [g'(\hat{f}_{AP_{a_s p_j}}) \\
&\times g(\hat{f}_{AP_{a_s (p_{t_2} - p_j)}}) + (g(\hat{f}_{AP_{a_s p_{t_2}}}) \\
&- g(\hat{f}_{AP_{a_s p_j}}))g'(\hat{f}_{AP_{a_s (p_{t_2} - p_j)}})] \\
&- \lambda \theta_{p_j}
\end{aligned} \tag{6}$$

where

$$\delta_2 = g'(\hat{f}_{AP_{a_i p_j}}) \sum_{t_1=1}^{N_p} f_{AP_{a_i p_{t_1}}} g(\hat{f}_{AP_{a_i (p_{t_1} - p_j)}})$$
$$- g(\hat{f}_{AP_{a_i p_j}}) \sum_{t_1=1}^{N_p} f_{AP_{a_i p_{t_1}}} g'(\hat{f}_{AP_{a_i (p_{t_1} - p_j)}}) \tag{7}$$

where $g'(x)$ is the derivative of $g(x)$ and $\odot$ denotes element-wise product, and λ is the regularization parameter.

5.3 Recommendation by Factor Matrices

After retrieving the latent matrix for each entity type, it is straightforward to generate the ranking list based upon the recommendation task and the design of matrix/tensor. Take the prediction task for the author-paper citation as one example, given one author a_i, we can achieve the relevance score of each paper p_j in the candidate set by computing $\frac{1}{r_{a_i p_j}} \approx g(\hat{f}_{a_i p_j}) = g(\langle \theta_{a_i}, \theta_{p_j} \rangle)$, and rank all papers in descending order. The same process can be applied to all other recommendation tasks considered in our model.

Table 5: data set statistics

data set	authors	papers	venues	APV records	AA records	AP records	PP records
ACM	24,764	18,121	846	47,810	112,456	366,201	71,396
ArnetMiner	49,298	47,794	1,682	132,186	361,794	1,675,564	237,531

6 Experimental Evaluation

We report in this section the experimental evaluation results for our model, and compare it with several existing state-of-the-art algorithms.

6.1 Data Preprocessing

We conduct our experiments on a subset of the ACM and ArnetMiner data set introduced in section 3. For papers in each data set separately, we collect the papers with complete information (authors, abstract, publishing venue and publishing year) and have been cited at least 5 times in the ACM data set and 10 times in the ArnetMiner data set. Based on these papers, we further collect all their authors and publishing venues.

We construct the tensor and matrices as introduced in section 3 for each data set. The β parameter in AA, AP and PP matrix is set to be 0.5. The PW-relation and AF-relation are constructed for all valid authors and papers. Table 5 shows a brief data statistics for both data sets, and the total number of records for each relation. Five-fold cross validation is conducted over the APV-relation, AA-relation, AP-relation and PP-relation to get the averaged predicting results. In the APV-relation, since each paper can have multiple authors but just one publishing venue, in order to avoid to have overlapped records in the training and testing set, we split the APV-relation into five folds by guaranteeing that one particular paper with all its authors (and the associated records) would appear in either the training or the testing set.

We adopted MAP as our evaluation metric, as the model is specially designed for maximizing MAP. Since the data in each relation is quite sparse (as shown in Table 6), we cannot treat all entries with no observed data as negative samples (consider the situation that paper a should also cite paper b, but unfortunately it did not.), in which case the recommendation results would be deteriorated. To avoid this, we randomly select 200 negative samples (much higher than the average node degree in each relation) for each entity in the testing set. The performance is therefore measured based on the recommendation list that contains the known positive samples and 200 randomly selected negative samples.

In all experiments, we set the latent dimensionality $D = 10$, the regularization parameter $\lambda = 0.001$ and the learning-rate as 0.001.

Table 6: data set statistics

	Avg. node degree			
data set	APV	AA	AP	PP
ACM	1	10.28	17.51	4.71
ArnetMiner	1	18.40	42.03	7.81

Table 7: Performance comparison over different combinations of relations

Combinations	**ACM**			
	APV	AA	AP	PP
C0	**0.0329**	0.0487*	0.0456*	0.0389
C1	0.0263*	**0.0560**	0.0455*	0.0325*
C2	0.0282*	0.0462*	0.0458*	0.0338*
C3	0.0307*	0.0460*	0.0455*	0.0329*
C4	0.0279*	NA	NA	NA
C5	NA	**0.0560**	NA	NA
C6	NA	NA	**0.0465**	NA
C7	NA	NA	NA	**0.0395**
C8	NA	0.0468*	0.0453*	0.0325*

6.2 Co-effects analysis of multiple relations

In this part of experiments, we work on totally eight different kinds of multi-relational combinations, and evaluate the performance over four tasks respectively. Table 7 and 8 shows the results.

In Table 7 and 8, c_0 indicates the single relation respectively. $c_1 = \{apv, aa, ap, pp, pw, af\}$, $c_2 = \{apv, aa, ap, pp, pw\}$, $c_3 = \{apv, aa, ap, pp\}$, $c_4 = \{apv, pw, af\}$, $c_5 = \{a, af\}$, $c_6 = \{ap, pw, af\}$, $c_7 = \{pp, pw\}$, and $c_8 = \{aa, ap, pp\}$.

Several observations can be drawn from the results. 1) Under almost all situations, jointly modeling multiple relations can indeed improve the prediction performance. For the four tasks over two data sets (just except the publishing venue prediction (APV) on ACM data set), the best performance is always achieved when some relations are jointly modeled. 2) There is no clear trend that the more relations we jointly modeled, the better performance we can achieve. For some prediction task, i.e., the paper-paper citation prediction on ACM data set, best performance is obtained when only paper-paper-citation and paper-word relation are incorporated. However, for the ArnetMiner data set, three out of four tasks have the best performance with all relations incorporated.

For each relation in both of the two data sets, we conducted the students' t test between the best performance result with others. Statistically significant improve-

Table 8: Performance comparison over different combinations of relations

Combinations	**ArnetMiner**			
	APV	AA	AP	PP
C0	0.0277*	0.0534*	0.0782*	0.0342*
C1	0.0289*	**0.0566**	**0.0788**	**0.0357**
C2	**0.0317**	0.0541*	0.0786	0.0353
C3	0.0285*	0.0538*	0.0784	0.0353
C4	0.0316	NA	NA	NA
C5	NA	0.0565	NA	NA
C6	NA	NA	0.0786	NA
C7	NA	NA	NA	0.0348*
C8	NA	0.0543*	0.0787	0.0349*

Table 9: Performance Comparison

Approaches	**ACM**			
	APV	AA	AP	PP
JMRM	0.0329*	**0.0560**	0.0465*	**0.0395**
FM	**0.2127**	0.0434*	0.0388*	0.0053*
CTR		0.0374*	**0.0513**	0.0341*
BPRA	0.0161*	0.0558	0.0360*	0.0216*

ments (paired-based $p \leq 0.05$) are labeled with a $*$ in Table 7 and 8.

6.3 Comparison with existing methods

We report the performance comparison with three state-of-the-art approaches: the Factorization Machines (short as FM) [12], the Collaborative Topic Regression (short as CTR) [20] and the Bayesian probabilistic relational-data Analysis [24] approach.

Factorization machines are a generic approach which can effectively combine the generality of feature engineering with the high-prediction accuracy superiority of factorization models. It therefore can mimic most factorization models by simple feature engineering.

CTR model combines traditional collaborative filtering with topic modeling. BPRA jointly models coupled matrices and tensors but optimizes the model by minimizing RMSE.

For FM, CTR and BPRA models, we feed the same training and testing set we used for JMRM, and evaluate the prediction performance on each individual relations separately. For JMRM, the reported results are the best results selected

Table 10: Performance Comparison

Approaches	ArnetMiner			
	APV	AA	AP	PP
JMRM	0.0317*	**0.0566**	0.0788	0.0357*
FM	**0.1595**	0.0402*	0.0613*	0.0047*
CTR		0.0395*	0.0756*	**0.0375**
BPRA	0.0176*	0.0359*	**0.0794**	0.0286*

from different combinations of multiple relations (as shown in Table 7). For using FM method, we regard the tasks as 'regression' tasks; The dimensionality of the factorization machine is set to be '1,1,8', indicating that the global bias, one-way interactions and pairwise interactions are all used, and that the number of factors used for pairwise interactions is set to be 8. Stochastic gradient descent (SGD) is chosen to used as the learning method. For CTR method, we construct paper profiles by their abstracts, and author profiles by concatenating all their publications. The basic LDA is used to retrieve the topic proportion and distribution vectors. The dimension for latent factor is set to be 10, and the number of latent topics is set to 20. Since CTR is only proposed for factorizing two types of entities, we did not adopt it to the task of publishing venue prediction (the APV-relation). Note that both FM and CTR are implemented using publicly available software. We also set the dimension for latent factor in BPRA as 10.

Table 9 and 10 show the results. As indicated, we found that our JMRM mode can outperform FM and CTR in several cases which demonstrates the effectiveness of our model. FM can achieve significantly better results than JMRM in predicting publishing venue, but has a very poor performance in predicting paper-paper citation. Our model shows the best overall performance, since out of 8 cases (four recommendation tasks over two data sets), our model ranks first for three cases, and the second for the other five cases, demonstrating its superiority in providing recommendations for four tasks simultaneously.

7 Conclusions

We proposed an extended latent factor model that can jointly model several relations in an academic environment. The model is specially designed for our recommendation tasks, and is proposed based upon the assumption that several academic activities are highly coupled, and that by joint modeling, we can not only solve the cold start problem but also help in achieving more coherent and accurate latent feature vectors. Moreover, to facilitate ranking, we extend an existing work which directly maximizes MAP over one single tensor into a more generalize form and is therefore able to maximize MAP over several matrices and tensors. Experiments carried out over two real world data sets demonstrate the effectiveness of our model.

References

[1] D. Agarwal and B.-C. Chen. Regression-based latent factor models. In *KDD*, 2009.

[2] P. Batista, M. Campiteli, and O. Kinouchi. Is it possible to compare researchers with different scientific interests? *Scientometrics*, 68:179–189, 2006.

[3] T. Chen, Z. Zheng, Q. Lu, W. Zhang, and Y. Yu. Feature-based matrix factorization. *CoRR*, abs/1109.2271, 2011.

[4] Egghe. Theory and practice of the g-index. *Scientometrics*, 69:131–152, 2006.

[5] A. Harzing. *The publish or perish book.* 2010.

[6] Q. He, D. Kifer, J. Pei, P. Mitra, and C. L. Giles. Citation recommendation without author supervision. In *WSDM*, pages 755–764, 2011.

[7] Q. He, J. Pei, D. Kifer, P. Mitra, and L. Giles. Context-aware citation recommendation. In *WWW*, pages 421–430, 2010.

[8] J. Hirsch. An index to quantify an individual's scientific research output. *Proc.Nat.Acad.Sci.*, 46:16569, 2005.

[9] B. Jin. The ar-index: Complementing the h-index. *Intl. Socienty for Scientometrics and Informetrics Newsletter*, 2007.

[10] D. Liben-Nowell and J. Kleinberg. The link-prediction problem for social networks. *J. Am. Soc. Inf. Sci. Technol.*, 58(7), 2007.

[11] S. M. McNee, I. Albert, D. Cosley, P. Gopalkrishnan, S. K. Lam, A. M. Rashid, J. A. Konstan, and J. Riedl. On the recommending of citations for research papers. In *CSCW*, 2002.

[12] S. Rendle. Factorization machines with libfm. *ACM Trans. Intell. Syst. Technol.*, 3(3):57:1–57:22.

[13] S. Rendle, C. Freudenthaler, Z. Gantner, and L. Schmidt-Thieme. Bpr: Bayesian personalized ranking from implicit feedback. In *UAI*, UAI '09, pages 452–461, 2009.

[14] F. Ruane and R. Tol. Rational (successive) h-indices: An application to economics in the republic of iremand. *Scienctometrics*, 2008.

[15] Y. Shi, A. Karatzoglou, L. Baltrunas, M. Larson, A. Hanjalic, and N. Oliver. Tfmap: optimizing map for top-n context-aware recommendation. In *SIGIR*, pages 155–164, 2012.

[16] A. Sidiropoulos, D. Katsaros, and Y. Manolopoulos. Generalized hirsch h-index for disclosing latent facts in citation networks. *Scientometrics*, 72:253–280, 2007.

[17] A. P. Singh and G. J. Gordon. Relational learning via collective matrix factorization. In *KDD*, 2008.

[18] T. Strohman, W. B. Croft, and D. Jensen. Recommending citations for academic papers. In *SIGIR*, 2007.

[19] J. Tang, J. Zhang, L. Yao, J. Li, L. Zhang, and Z. Su. ArnetMiner: extraction and mining of academic social network. In *KDD*, 2008.

[20] C. Wang and D. M. Blei. Collaborative topic modeling for recommending scientific articles. In *KDD*, 2011.

[21] Z. Yang and B. D. Davison. Venue recommendation: Submitting your paper with style. In *ICMLA (1)*, pages 681–686, 2012.

[22] Z. Yang, D. Yin, and B. Davison. Recommendation in Academia: a Joint Multi-Relational Model. In *ASONAM*, 2014.

[23] Z. Yang, D. Yin, and B. D. Davison. Award prediction with temporal citation network analysis. In *SIGIR*, SIGIR '11, pages 1203–1204, 2011.

[24] D. Yin, S. Guo, B. Chidlovskii, B. D. Davison, C. Archambeau, and G. Bouchard. Connecting comments and tags: improved modeling of social tagging systems. In *WSDM*, pages 547–556, 2013.

[25] C. Zhang. The e-index, complementing the h-index for excess citations. *PLos One*, 4(5):1–4, 2009.

[26] D. Zhou, S. Zhu, K. Yu, X. Song, B. L. Tseng, H. Zha, and C. L. Giles. Learning multiple graphs for document recommendations. In *WWW*, pages 141–150, 2008.