FEMA: Flexible Evolutionary Multi-faceted Analysis for Dynamic Behavioral Pattern Discovery

Meng Jiang[123], Peng Cui[123], Fei Wang[4], Xinran Xu[123], Wenwu Zhu[123], Shiqiang Yang[123]

[1]Tsinghua National Laboratory for Information Science and Technology
[2]Department of Computer Science and Technology, Tsinghua University, Beijing, China
[3]Beijing Key Laboratory of Networked Multimedia, Tsinghua University, China
[4]IBM Watson Research Center, Yorktown Heights, NY, USA
jm06@mails.tsinghua.edu.cn, cuip@tsinghua.edu.cn, feiwang03@gmail.com
xxr10@mails.tsinghua.edu.cn, wwzhu@tsinghua.edu.cn, yangshq@tsinghua.edu.cn

ABSTRACT

Behavioral pattern discovery is increasingly being studied to understand human behavior and the discovered patterns can be used in many real world applications such as web search, recommender system and advertisement targeting. Traditional methods usually consider the behaviors as simple user and item connections, or represent them with a static model. In real world, however, human behaviors are actually complex and dynamic: they include correlations between user and multiple types of objects and also continuously evolve along time. These characteristics cause severe data sparsity and computational complexity problem, which pose great challenge to human behavioral analysis and prediction. In this paper, we propose a Flexible Evolutionary Multi-faceted Analysis (FEMA) framework for both behavior prediction and pattern mining. FEMA utilizes a flexible and dynamic factorization scheme for analyzing human behavioral data sequences, which can incorporate various knowledge embedded in different object domains to alleviate the sparsity problem. We give approximation algorithms for efficiency, where the bound of approximation loss is theoretically proved. We extensively evaluate the proposed method in two real datasets. For the prediction of human behaviors, the proposed FEMA significantly outperforms other state-of-the-art baseline methods by 17.4%. Moreover, FEMA is able to discover quite a number of interesting multi-faceted temporal patterns on human behaviors with good interpretability. More importantly, it can reduce the run time from hours to minutes, which is significant for industry to serve real-time applications.

Categories and Subject Descriptors

I.5.3 [**Computing Methodologies**]: Pattern Recognition - Clustering; J.4 [**Computer Applications**]: Social and Behavioral Sciences

Keywords

Behavior Modeling; Behavioral Pattern; Evolutionary Analysis; Tensor Factorization; Flexible Regularizers

KDD'14, August 24–27, 2014, New York, NY, USA.

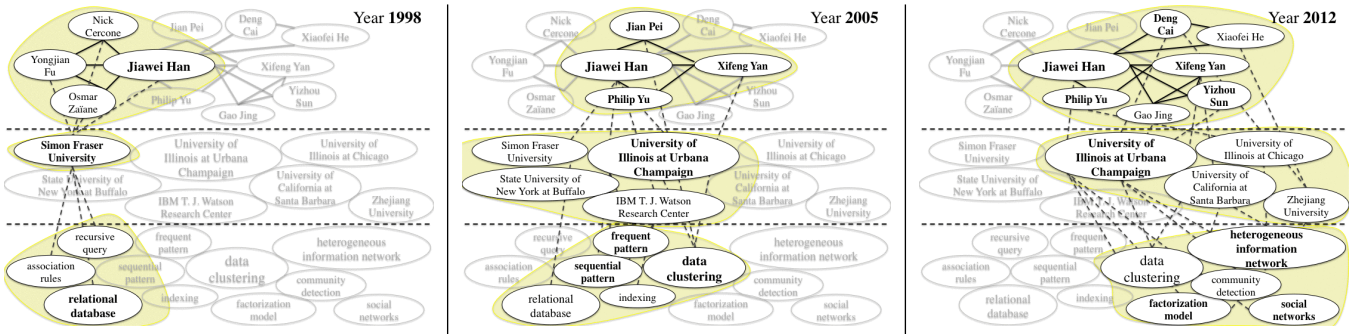
http://dx.doi.org/10.1145/2623330.2623644.

1. INTRODUCTION

Scientists study human behavior from a variety of cultural, political, and psychological perspectives, looking for consistent patterns of individual and social behavior and for scientific explanations on those patterns. It is well accepted that human behavior is the product of a multitude of interrelated factors. The factors such as physical environment, social interaction, and social identity, affect how the behavior takes place with our personalities and interests. As an example, if a researcher changes his affiliation, he will start to collaborate with new friends, join in new projects and eventually study new topics. Given the complexity of multi-faceted factors influencing human behaviors, it is difficult to concisely summarize what they are and how they interact. Moreover, psychological studies [21] demonstrate that human behaviors naturally evolve with the changing of both endogenous factors (e.g., personality) and exogenous factors (e.g, environment), resulting in different dynamic (temporal) behavioral patterns over time. For example, in early 1990s, many researchers focused on database systems and query processing. In late 1990s, with various data collective methods emerging and scales of unlabeled data increasing, they turned to work on clustering and pattern mining problems. In 2000s, people started to focus on social networks and communities since Facebook and Twitter become popular. Consequently, the patterns of human behaviors differ from place to place, era to era and across environments. The complexity and dynamic characteristics pose great challenges to understanding and predicting human behaviors. However, there is a lack of research to support behavioral modeling with both multi-faceted and temporal information.

Traditional methods of data analysis have long been used to discover patterns of human behaviors. Sun *et al.* [27] perform 3-mode analysis on the click-through data with user, query and web page. Chen *et al.* [2] models tagging behavior with the decomposition of ternary relationships of user, tag and item. However, their static views on human behavior are not able to learn from temporal information, or capture the dynamic characteristic. Radinsky *et al.* [22] use several time-series models for representing and predicting web search behavior and content change. Xiang *et al.* [33] use session nodes to capture short-term interests of paper-tagging behavior through session-item connections. However, their representations cannot learn from multi-faceted information, or fully describe the complex characteristic of human behavior. Hence, temporal multi-faceted behavioral patterns are rarely investigated, and how to accurately predict these behaviors still remains as an open problem.

There are two key challenges to learn human behavioral patterns from the multi-faceted and temporal information.



(a) When he was a professor at Simon Fraser University and worked on database systems and data warehousing

(b) When he was a professor at University of Illinois at Urbana-Champaign and worked on data mining from spatio-temporal data

(c) When he was a professor at University of Illinois at Urbana-Champaign and focused on mining social/information network data

Figure 1: A showcase of temporal behavioral pattern discovery: how and why Professor Jiawei Han and his group change their academic research topics? The research behavior includes author, affiliation and keyword (3 layers), and evolves over time.

- **High sparsity.** Multi-faceted data in real applications is often very sparse. In researcher-affiliation-topic case, for example, researchers cannot work in many affiliations or study many problems. The problem is even disastrous when we add the temporal dimension to the multi-faceted behavioral information.
- **High complexity.** Considering the dynamic characteristic, new multi-faceted human behaviors continuously generate along time. The continuously generated data of high volume, high dimension and high sparsity pose great challenge for modeling and analysis due to high computational complexity. Sun *et al.* proposed a tensor based method DTA [26], which saves time of matricizing tensors by storing and updating unfolded matrices. But it still costs too much time on computing eigenvalues and eigenvectors. The issue of fast processing increments is still critical for modeling and predicting human behavior.

To address these challenges, in this paper, we propose a *Flexible Evolutionary Multi-faceted Analysis* (FEMA) method based on a dynamic scheme of tensor factorization for temporal multi-faceted behavior prediction and pattern mining, where flexible regularizers are imposed to alleviate the problems brought by high sparsity. In order to fast decompose high-order tensor sequences, we give approximation algorithms to factorize the new tensor with sparse increments, where the bound of approximation loss is theoretically proved. We evaluate FEMA on two real datasets: publication data from Microsoft Academic Search database and tweet data from Tencent Weibo, a Twitter style website in China. The proposed method achieves 30.8% higher accuracy when it uses multi-faceted factors and 17.4% higher accuracy when it uses flexible regularizers. Moreover, it can reduce the run time from hours to minutes, which is of significant interest to serve real-time applications.

Fig. 1 is a showcase of temporal patterns of academic research behavior discovered by FEMA. It can be seen that the evolutions in Professor Jiawei Han's group from database systems, data clustering to social/information networks can be effectively and efficiently discovered by FEMA. And the pattern clearly shows the co-evolution and interplay between affiliations and his co-authorships, which gives us comprehensive understanding on the temporal and multi-faceted characteristics of behavioral patterns.

The main contributions of this paper are:

(1) Enlightened by the psychological studies on human behaviors, we move one step forward to incorporate temporal dimension into multi-faceted analysis for temporal multi-faceted behavior prediction and pattern mining, which is of paramount importance for various applications, such as web search and recommendation.

(2) We propose the FEMA framework based on tensor factorization to predict temporal multi-faceted behaviors. The model incorporates flexible regularizers to alleviate the sparsity problem. We design approximation algorithms to fast conduct evolutionary analysis of human behaviors.

(3) Our algorithm is efficient and has a theoretical guarantee: it runs in near-quadratic time compared to the near-cubic time of the existing algorithms (see Section 4).

(4) We conduct extensive experiments to predict human behaviors in academic research and social networks. The results show that the proposed FEMA can outperforms other methods on both datasets. More importantly, we demonstrate that the approximation algorithms lead to significant time reduction and the loss is small.

We have the usual organization: Survey, problem definition, proposed method, experiments and conclusions.

2. RELATED WORKS

There is a significant body on research related to our problem, which we categorize into three groups: behavior modeling, behavioral dynamics, and tensor factorization.

Behavior modeling: Matrix factorization has long been used for modeling and predicting human behavior when it includes two types of objects, such as user-item adoption [18] and drug-target interaction [35]. When the number of types is more than two, there has been a great deal of interest in using high-order tensors to model behaviors, for example, web search [27], image and book tagging [29, 23], and recommender systems [11, 12, 2, 10, 20]. These works summarize a static view of the behavioral pattern, but they cannot capture its temporal characteristics.

Behavioral dynamics: There have been attempts to use temporal information to understand past users' behaviors in order to predict future ones in different applications such as recommender systems [6, 3], research themes [32], semantic graphs [28], and online media topics [19, 5, 34]. Xiang *et al.* [33] divide user interests into long-term and short-term and make use of the difference, using a time factor. In contrast to this approach, we consider the group-level dynamics instead of an individual user behavior. Radinsky *et al.* [22] develop a learning algorithm capable of selecting an appropriate model depending on the time. However, how to appropriately use the time information to discover the underlying dynamics of human behavior still remains an important research challenge.

Matrix/tensor factorization: There has been active research on matrix factorization [30, 31], tensor analysis [4, 26, 13], tensor de-

compositions [15, 16, 9] and scalable methods [17, 1]. Here we focus on how to efficiently process the increments in tensor decomposition by matrix and tensor perturbation theory [25].

3. PROBLEM FORMULATION

In this section, we first give the modeling of two different types of human behavior including academic research and mentioning to someone in tweets. Then we give a general definition of our problem or the task of our method.

Modeling the behavior of academic research.

Let the bibliographic dataset be an example of our problem: we focus on finding temporal patterns of academic research behavior. Let the dataset be a list of tuples (a, f, k, t) denoting that an author a in an affiliation f (university, research center, etc.) publishes about a keyword k at time t ($t = 1, \ldots, T$). We model the data as a 3-order tensor sequence $\mathcal{X}_\mathbf{t} \in \mathbb{R}^{n^{(a)} \times n^{(f)} \times n^{(k)}}$, where $n^{(a)}$ is the number of authors, $n^{(f)}$ is the number of affiliations, and $n^{(k)}$ is the number of keywords. $\mathcal{X}_\mathbf{t}(a, f, k)$ has a value of the number of existing tuples (a, f, k, t') ($t' \leq t$). Our goal is to factorize the tensor sequence

$$\mathcal{X}_\mathbf{t} \approx \mathcal{Y}_\mathbf{t} \times_{(a)} \mathbf{A_t} \times_{(f)} \mathbf{F_t} \times_{(k)} \mathbf{K_t} \tag{1}$$

where

- $\mathcal{Y}_\mathbf{t} \in \mathbb{R}^{r^{(a)} \times r^{(f)} \times r^{(k)}}$ is the core tensor sequence, which encodes the temporal behavioral patterns, i.e., the relationship among author, affiliation and keyword groups. $\mathcal{Y}_\mathbf{t}(j^{(a)}, j^{(f)}, j^{(k)})$ indicates the probability of the behavior before time t if the $j^{(a)}$-th author group in the $j^{(f)}$-th affiliation group publishes the $j^{(k)}$-th keyword group.
- $\mathbf{A_t} \in \mathbb{R}^{n^{(a)} \times r^{(a)}}$ is the authors' projection matrix before time t. $\mathbf{A}(i^{(a)}, j^{(a)})$ represents the probability that the $i^{(a)}$-th author belongs to the $j^{(a)}$-th group before time t.
- $\mathbf{F_t} \in \mathbb{R}^{n^{(f)} \times r^{(f)}}$ is the affiliations' projection matrix before time t. $\mathbf{F_t}(i^{(f)}, j^{(f)})$ represents the probability that the $i^{(f)}$-th affiliation belongs to the $j^{(f)}$-th group before time t.
- $\mathbf{K_t} \in \mathbb{R}^{n^{(k)} \times r^{(k)}}$ is the keywords' projection matrix before time t. $\mathbf{K_t}(i^{(k)}, j^{(k)})$ represents the probability that the $i^{(k)}$-th keyword belongs to the $j^{(k)}$-th group before time t.

Note that the key to solving the sparsity problem in tensor decompositions is to learn the flexible regularizers such as the authors' co-authorship, the affiliations' geographical distance and the keywords' semantic information. The regularizers can be encoded as Laplacian matrices $\mathbf{L^{(a)}}$, $\mathbf{L^{(f)}}$, $\mathbf{L^{(k)}}$, where the (i, j)-th element represents the similarity between the i-th and j-th entities (authors, affiliations, keywords): the similarity can be how many papers the authors collaborate or how close the affiliations locate.

The problem is now how to compute the factorizations for the core tensor sequence and projection matrices, given the tensor sequence and constraints. Note that the scale of the tensors are large but the changes are very small. We denote by $\Delta\mathcal{X}_\mathbf{t}$ the increment at time t, which is very sparse: for any $1 \leq t < T$, $\Delta\mathcal{X}_\mathbf{t} = \mathcal{X}_{\mathbf{t+1}} - \mathcal{X}_\mathbf{t}$. The problem can be summarized into two steps:

- **Given** the first tensor $\mathcal{X}_\mathbf{1}$ and the constraints $\mathbf{L^{(a)}}$, $\mathbf{L^{(f)}}$, $\mathbf{L^{(k)}}$, **find** the projection matrix $\mathbf{A_1}$, $\mathbf{F_1}$, $\mathbf{K_1}$, and the first core tensor $\mathcal{Y}_\mathbf{1}$.
- At time t ($1 \leq t < T$), **given** the tensor $\mathcal{X}_\mathbf{t}$, the increment $\Delta\mathcal{X}_\mathbf{t}$, the old projection matrices $\mathbf{A_t}$, $\mathbf{F_t}$, $\mathbf{K_t}$, and the constraints $\mathbf{L^{(a)}}$, $\mathbf{L^{(f)}}$, $\mathbf{L^{(k)}}$, **find** the new projection matrices $\mathbf{A_{t+1}}$, $\mathbf{F_{t+1}}$, $\mathbf{K_{t+1}}$, and the new core tensor $\mathcal{Y}_{\mathbf{t+1}}$.

Modeling the behavior of "mentions" in tweets.

Let the tweet dataset be another example of our problem and thus we come to find temporal patterns of the mention in tweets. Let the dataset be a list of tuples (s, d, w, t) denoting that a Twitter user s ("source") uses the "@username" format to mention a user d ("target", or "destination") in the body of a tweet which includes a word w at time t ($t = 1, \ldots, T$), so that the user d will see the tweet in his/her "Mentions" tab. Similarly to the modeling of academic research behavior, we model the data as a 3-order tensor sequence $\mathcal{X}_\mathbf{t} \in \mathbb{R}^{n^{(s)} \times n^{(d)} \times n^{(w)}}$, where $n^{(s)}$ is the number of sources, $n^{(d)}$ is the number of targets, and $n^{(w)}$ is the number of words. $\mathcal{X}_\mathbf{t}(s, d, w)$ is the number of tuples (s, d, w, t') ($t' \leq t$). Our goal is to factorize the tensor sequence

$$\mathcal{X}_\mathbf{t} \approx \mathcal{Y}_\mathbf{t} \times_{(s)} \mathbf{S_t} \times_{(d)} \mathbf{D_t} \times_{(w)} \mathbf{W_t} \tag{2}$$

where $\mathcal{Y}_\mathbf{t} \in \mathbb{R}^{r^{(s)} \times r^{(d)} \times r^{(w)}}$ is the core tensor sequence; $\mathbf{S_t} \in \mathbb{R}^{n^{(s)} \times r^{(s)}}$ is the source users' projection matrix, $\mathbf{D_t} \in \mathbb{R}^{n^{(d)} \times r^{(d)}}$ is the target users' projection matrix, and $\mathbf{W_t} \in \mathbb{R}^{n^{(w)} \times r^{(w)}}$ is the words' projection matrix.

Here, to solve the sparsity problem, the flexible regularizers such as the users' social relations (e.g., the number of common friends), and the words' semantic information, can be encoded as Laplacian matrices $\mathbf{L^{(s)}}$, $\mathbf{L^{(d)}}$, $\mathbf{L^{(w)}}$. Similarly, the problem can be summarized into two steps:

- **Given** the first tensor $\mathcal{X}_\mathbf{1}$ and constraints $\mathbf{L^{(s)}}$, $\mathbf{L^{(d)}}$, $\mathbf{L^{(w)}}$, **find** projection matrix $\mathbf{S_1}$, $\mathbf{D_1}$, $\mathbf{W_1}$, and core tensor $\mathcal{Y}_\mathbf{1}$.
- At time t ($1 \leq t < T$), **given** the tensor $\mathcal{X}_\mathbf{t}$, the increment $\Delta\mathcal{X}_\mathbf{t}$, the old projection matrices $\mathbf{S_t}$, $\mathbf{D_t}$, $\mathbf{W_t}$, and the constraints $\mathbf{L^{(s)}}$, $\mathbf{L^{(d)}}$, $\mathbf{L^{(w)}}$, **find** the new projection matrices $\mathbf{S_{t+1}}$, $\mathbf{D_{t+1}}$, $\mathbf{W_{t+1}}$, and the new core tensor $\mathcal{Y}_{\mathbf{t+1}}$.

The general problem definition.

Our problem is quite different from previous research. First, we incorporate multi-faceted information and constraints into a unified framework. Second, we conduct evolutionary analysis to efficiently deal with sparse increments, which is in contrast with the majority of existing works that decompose a single tensor. We extend the formulation from 3 to M dimensions and give a general definition.

Definition 1 (Flexible Evolutionary Multi-faceted Analysis (FEMA))
(1) Initialization:
Given *the first M-way tensor $\mathcal{X}_\mathbf{1} \in \mathbb{R}^{n^{(1)} \times \ldots \times n^{(M)}}$ and the constraints $\mathbf{L^{(m)}}|_{m=1}^{M} \in \mathbb{R}^{n^{(m)} \times n^{(m)}}$,* **find** *the first projection matrices $\mathbf{A_1^{(m)}}|_{m=1}^{M} \in \mathbb{R}^{n^{(m)} \times r^{(m)}}$ and the first core tensor $\mathcal{Y}_\mathbf{1} \in \mathbb{R}^{r^{(1)} \times \ldots \times r^{(M)}}$.*
(2) Evolutionary analysis:
At time t ($1 \leq t < T$), **given** *the tensor $\mathcal{X}_\mathbf{t} \in \mathbb{R}^{n^{(1)} \times \ldots \times n^{(M)}}$, the increment $\Delta\mathcal{X}_\mathbf{t}$, the old projection matrices $\mathbf{A_t^{(m)}}|_{m=1}^{M}$, and the constraints $\mathbf{L^{(m)}}|_{m=1}^{M}$,* **find** *the new projection matrices $\mathbf{A_{t+1}^{(m)}}|_{m=1}^{M}$ and the new core tensor $\mathcal{Y}_{\mathbf{t+1}}$.*

4. ALGORITHM

In this section, we provide approximation algorithms for the two steps in Flexible Evolutionary Multi-faceted Analysis (FEMA). We also give a discussion on the computational efficiency and approximation quality.

Initialization.

Here we present how we incorporate the multi-faceted information and constraints into the tensor decomposition. We denote by $\mu^{(m)}$ the weight of the mode-m Laplacian matrix $\mathbf{L^{(m)}}$. The covariance matrix of the m-th mode at time $t = 1$ is

$$\mathbf{C_1^{(m)}} = \mathbf{X_1^{(m)}}\mathbf{X_1^{(m)T}} + \mu^{(m)}\mathbf{L^{(m)}} \tag{3}$$

where $\mathbf{X_1^{(m)}} \in \mathbb{R}^{n^{(m)} \times \prod_{i \neq m} n^{(i)}}$ is the mode-m matricizing of the tensor $\mathcal{X}_1$. The projection matrices $\mathbf{A_1^{(m)}}|_{m=1}^{M}$ can be computed by diagonalization: they are the top $r^{(m)}$ eigenvectors of the covariance matrix $\mathbf{C_1^{(m)}}|_{m=1}^{M}$. The pseudocode is listed in Algorithm 1.

Algorithm 1 Initialization in FEMA

Require: $\mathcal{X}_1, \mathbf{L^{(m)}}|_{m=1}^{M}$
  for $m = 1, \ldots, M$ **do**
    Construct covariance matrix $\mathbf{C_1^{(m)}}$ using Eq.3;
    $\lambda_1^{(m)}$/$\mathbf{A_1^{(m)}}$ are the top $r^{(m)}$ eigenvalue/eigenvector of $\mathbf{C_1^{(m)}}$
  end for
  $\mathcal{Y}_1 = \mathcal{X}_1 \prod_{m=1}^{M} \times_{(m)} \mathbf{A_1^{(m)T}}$;
  return $\mathbf{A_1^{(m)}}|_{m=1}^{M}, \lambda_1^{(m)}|_{m=1}^{M}, \mathcal{Y}_1$

Evolutionary analysis.

Next we introduce an efficient technique based on tensor perturbation to adjust the projection matrices according to changes of the tensor. We denote by $\mathbf{X_t^{(m)}} \in \mathbb{R}^{n^{(m)} \times \prod_{i \neq m} n^{(i)}}$ the mode-m matricizing of the tensor $\mathcal{X}_\mathbf{t}$. We define the covariance matrix $\mathbf{C_t^{(m)}} = \mathbf{X_t^{(m)}}\mathbf{X_t^{(m)}}^\top + \mu^{(m)}\mathbf{L^{(m)}}$ and define $(\lambda_{t,i}^{(m)}, \mathbf{a_{t,i}^{(m)}})$ as one eigenvalue-eigenvector pair of the matrix $\mathbf{C_t^{(m)}}$. The vector $\mathbf{a_{t,i}^{(m)}}$ is exactly the i-th column of the projection matrix $\mathbf{A_t^{(m)}}$. Then we can rewrite $(\lambda_{t+1,i}^{(m)}, \mathbf{a_{t+1,i}^{(m)}})$ as

$$\lambda_{t+1,i}^{(m)} = \lambda_{t,i}^{(m)} + \Delta\lambda_{t,i}^{(m)} \tag{4}$$

$$\mathbf{a_{t+1,i}^{(m)}} = \mathbf{a_{t,i}^{(m)}} + \Delta\mathbf{a_{t,i}^{(m)}} \tag{5}$$

To simplify the denotions, we omit "t" in the terms and equations when it is unnecessary. Thus we can obtain

$$\begin{aligned}&[(\mathbf{X^{(m)}} + \Delta\mathbf{X^{(m)}})(\mathbf{X^{(m)}} + \Delta\mathbf{X^{(m)}})^\top + \mu^{(m)}\mathbf{L^{(m)}}] \\ &\cdot(\mathbf{a_i^{(m)}} + \Delta\mathbf{a_i^{(m)}}) = (\lambda_i^{(m)} + \Delta\lambda_i^{(m)})(\mathbf{a_i^{(m)}} + \Delta\mathbf{a_i^{(m)}})\end{aligned} \tag{6}$$

Now the key questions are how to compute changes to the eigenvalue $\Delta\lambda_i^{(m)}$ and eigenvector $\Delta\mathbf{a_i^{(m)}}$, respectively.

Expanding Eq.6, we obtain

$$\begin{aligned}&\mathbf{X^{(m)}}\mathbf{X^{(m)}}^\top\mathbf{a_i^{(m)}} + \mathbf{X^{(m)}}\mathbf{X^{(m)}}^\top\Delta\mathbf{a_i^{(m)}} \\ &+\mathbf{X^{(m)}}\Delta\mathbf{X^{(m)}}^\top\mathbf{a_i^{(m)}} + \mathbf{X^{(m)}}\Delta\mathbf{X^{(m)}}^\top\Delta\mathbf{a_i^{(m)}} \\ &+\Delta\mathbf{X^{(m)}}\mathbf{X^{(m)}}^\top\mathbf{a_i^{(m)}} + \Delta\mathbf{X^{(m)}}\mathbf{X^{(m)}}^\top\Delta\mathbf{a_i^{(m)}} \\ &+\Delta\mathbf{X^{(m)}}\Delta\mathbf{X^{(m)}}^\top\mathbf{a_i^{(m)}} + \Delta\mathbf{X^{(m)}}\Delta\mathbf{X^{(m)}}^\top\Delta\mathbf{a_i^{(m)}} \\ &+\mu^{(m)}\mathbf{L^{(m)}}\mathbf{a_i^{(m)}} + \mu^{(m)}\mathbf{L^{(m)}}\Delta\mathbf{a_i^{(m)}} \\ &= \lambda_i^{(m)}\mathbf{a_i^{(m)}} + \lambda_i^{(m)}\Delta\mathbf{a_i^{(m)}} + \Delta\lambda_i^{(m)}\mathbf{a_i^{(m)}} + \Delta\lambda_i^{(m)}\Delta\mathbf{a_i^{(m)}}\end{aligned} \tag{7}$$

In this paper, we concentrate on first-order approximation, i.e., we assume all high order perturbation terms (such as $\mathbf{X^{(m)}}\Delta\mathbf{X^{(m)}}^\top\Delta\mathbf{a_i^{(m)}}$, $\Delta\mathbf{X^{(m)}}\mathbf{X^{(m)}}^\top\Delta\mathbf{a_i^{(m)}}$, $\Delta\mathbf{X^{(m)}}\Delta\mathbf{X^{(m)}}^\top\mathbf{a_i^{(m)}}$, $\Delta\mathbf{X^{(m)}}\Delta\mathbf{X^{(m)}}^\top\Delta\mathbf{a_i^{(m)}}$ in the above equation) are neglectable. By further using the fact that $(\mathbf{X^{(m)}}\mathbf{X^{(m)}}^\top + \mu^{(m)}\mathbf{L^{(m)}})\mathbf{a_i^{(m)}} = \lambda_i^{(m)}\mathbf{a_i^{(m)}}$, we can obtain

$$\begin{aligned}&\mathbf{X^{(m)}}\mathbf{X^{(m)}}^\top\Delta\mathbf{a_i^{(m)}} + (\mathbf{X^{(m)}}\Delta\mathbf{X^{(m)}}^\top + \Delta\mathbf{X^{(m)}}\mathbf{X^{(m)}}^\top)\mathbf{a_i^{(m)}} \\ &+\mu^{(m)}\mathbf{L^{(m)}}\Delta\mathbf{a_i^{(m)}} = \lambda_i^{(m)}\Delta\mathbf{a_i^{(m)}} + \Delta\lambda_i^{(m)}\mathbf{a_i^{(m)}}\end{aligned} \tag{8}$$

Now multiplying both sides of Eq.8 with $\mathbf{a_i^{(m)}}^\top$ and because of the symmetry of $\mathbf{X^{(m)}}\mathbf{X^{(m)}}^\top$ and $\mathbf{L^{(m)}}$, we get

$$\Delta\lambda_i^{(m)} = \mathbf{a_i^{(m)}}^\top(\mathbf{X^{(m)}}\Delta\mathbf{X^{(m)}}^\top + \Delta\mathbf{X^{(m)}}\mathbf{X^{(m)}}^\top)\mathbf{a_i^{(m)}} \tag{9}$$

Since the eigenvectors are orthogonal to each other, we assume that the change of the eigenvector $\Delta\mathbf{a_i^{(m)}}$ is in the subspace spanned by those original eigenvectors, i.e.,

$$\Delta\mathbf{a_i^{(m)}} \approx \sum_{j=1}^{r^{(m)}} \alpha_{ij}\mathbf{a_j^{(m)}} \tag{10}$$

where $\{\alpha_{ij}\}$ are small constants to be determined. Bringing Eq.10 into Eq.8, we obtain

$$\begin{aligned}&(\mathbf{X^{(m)}}\mathbf{X^{(m)}}^\top + \mu^{(m)}\mathbf{L^{(m)}})\sum_{j=1}^{r^{(m)}}\alpha_{ij}\mathbf{a_j^{(m)}} \\ &+(\mathbf{X^{(m)}}\Delta\mathbf{X^{(m)}}^\top + \Delta\mathbf{X^{(m)}}\mathbf{X^{(m)}}^\top)\mathbf{a_i^{(m)}} \\ &= \lambda_i^{(m)}\sum_{j=1}^{r^{(m)}}\alpha_{ij}\mathbf{a_j^{(m)}} + \Delta\lambda_i^{(m)}\mathbf{a_i^{(m)}}\end{aligned} \tag{11}$$

which is equivalent to

$$\begin{aligned}&\sum_{j=1}^{r^{(m)}}\lambda_j^{(m)}\alpha_{ij}\mathbf{a_j^{(m)}} + \mathbf{X^{(m)}}\Delta\mathbf{X^{(m)}}^\top\mathbf{a_i^{(m)}} \\ &+\Delta\mathbf{X^{(m)}}\mathbf{X^{(m)}}^\top\mathbf{a_i^{(m)}} = \lambda_i^{(m)}\sum_{j=1}^{r^{(m)}}\alpha_{ij}\mathbf{a_j^{(m)}} + \Delta\lambda_i^{(m)}\mathbf{a_i^{(m)}}\end{aligned} \tag{12}$$

Multiplying $\mathbf{a_k^{(m)}}^\top$ $(k \neq i)$ on both sides of the above equation, we get

$$\begin{aligned}&\lambda_k^{(m)}\alpha_{ik} + \mathbf{a_k^{(m)}}^\top\mathbf{X^{(m)}}\Delta\mathbf{X^{(m)}}^\top\mathbf{a_i^{(m)}} \\ &+\mathbf{a_k^{(m)}}^\top\Delta\mathbf{X^{(m)}}\mathbf{X^{(m)}}^\top\mathbf{a_i^{(m)}} = \lambda_i^{(m)}\alpha_{ik}\end{aligned} \tag{13}$$

Therefore,

$$\alpha_{ik} = \frac{\mathbf{a_k^{(m)}}^\top(\mathbf{X^{(m)}}\Delta\mathbf{X^{(m)}}^\top + \Delta\mathbf{X^{(m)}}\mathbf{X^{(m)}}^\top)\mathbf{a_i^{(m)}}}{\lambda_i^{(m)} - \lambda_k^{(m)}} \tag{14}$$

To get α_{ii}, we use the fact that

$$\begin{aligned}(\mathbf{a_i^{(m)}} + \Delta\mathbf{a_i^{(m)}})^\top(\mathbf{a_i^{(m)}} + \Delta\mathbf{a_i^{(m)}}) &= 1 \\ \iff 1 + 2\mathbf{a_i^{(m)}}^\top\Delta\mathbf{a_i^{(m)}} + O(\|\Delta\mathbf{a_i^{(m)}}\|^2) &= 1\end{aligned}$$

Discarding the high order term, and bringing in Eq.10, we get $\alpha_{ii} = 0$. Therefore,

$$\Delta\mathbf{a_i^{(m)}} = \sum_{j \neq i}\frac{\mathbf{a_j^{(m)}}^\top(\mathbf{X^{(m)}}\Delta\mathbf{X^{(m)}}^\top + \Delta\mathbf{X^{(m)}}\mathbf{X^{(m)}}^\top)\mathbf{a_i^{(m)}}}{\lambda_i^{(m)} - \lambda_j^{(m)}}\mathbf{a_j^{(m)}} \tag{15}$$

Note that the constraints $\mathbf{L^{(m)}}$ do not appear in the eigenvalue and eigenvector updating functions Eq.9 and Eq.15. Note that the constraints have to be learnt only *once*.

Algorithm 2 Evolutionary Analysis in FEMA

Require: $\mathcal{X}_\mathbf{t}$, $\Delta\mathcal{X}_\mathbf{t}$, $\mathbf{A_t^{(m)}}|_{m=1}^M$, $\boldsymbol{\lambda}_\mathbf{t}^{(\mathbf{m})}|_{m=1}^M$
  for $m = 1, \ldots, M$ **do**
    for $i = 1, \ldots, r^{(m)}$ **do**
      Compute $\Delta\lambda_{t,i}^{(m)}$ using Eq.9, and compute
      $\lambda_{t+1,i}^{(m)} = \lambda_{t,i}^{(m)} + \Delta\lambda_{t,i}^{(m)}$;
      Compute $\Delta\mathbf{a}_{\mathbf{t,i}}^{(\mathbf{m})}$ using Eq.15, and compute
      $\mathbf{a}_{\mathbf{t+1,i}}^{(\mathbf{m})} = \mathbf{a}_{\mathbf{t,i}}^{(\mathbf{m})} + \Delta\mathbf{a}_{\mathbf{t,i}}^{(\mathbf{m})}$ and $\mathbf{A}_{\mathbf{t+1}}^{(\mathbf{m})} = \{\mathbf{a}_{\mathbf{t+1,i}}^{(\mathbf{m})}\}$;
    end for
  end for
  $\mathcal{Y}_{\mathbf{t+1}} = (\mathcal{X}_\mathbf{t} + \Delta\mathcal{X}_\mathbf{t}) \prod_{m=1}^M \times_{(m)} \mathbf{A}_{\mathbf{t+1}}^{(\mathbf{m})\mathbf{T}}$;
  return $\mathbf{A}_{\mathbf{t+1}}^{(\mathbf{m})}|_{m=1}^M$, $\boldsymbol{\lambda}_{\mathbf{t+1}}^{(\mathbf{m})}|_{m=1}^M$, $\mathcal{Y}_{\mathbf{t+1}}$

Computational complexity.

Here we analyze the computational complexity of Algorithm 2 before the computation of the core tensor. For the m-th mode, we define $D^{(m)}$ as the number of features of each point on the m-th dimension. Since the tensors are usually extremely sparse, we know $D^{(m)} \leq E \ll \prod_{m' \neq m} n^{(m')}$, where E is the number of non-zero entries in the tensors. In order to compute the increment on the eigenvalue and eigenvector using Eq.9 and Eq.15 for the m-th mode, we need to compute $\mathbf{v}_\mathbf{i}^{(\mathbf{m})}$, which requires $O(n^{(m)}D^{(m)})$ time. As $\Delta\mathbf{X}^{(\mathbf{m})}$ is very sparse, $\Delta\mathbf{X}^{(\mathbf{m})}\mathbf{v}_\mathbf{i}^{(\mathbf{m})}$ only requires constant time $O(D^{(m)})$. Therefore, for computing $\Delta\lambda_i^{(m)}$ and $\Delta\mathbf{a}_\mathbf{i}^{(\mathbf{m})}$, we need $O(r^{(m)}n^{(m)}D^{(m)} + r^{(m)}D^{(m)})$ time, and updating eigenvalues and eigenvectors for T times requires $O(T\sum_{m=1}^M r^{(m)}(n^{(m)}+1)D^{(m)})$ time. In comparison, if we redo the eigenvalue decomposition on $\mathcal{X}_{\mathbf{t+1}}$, it costs $O(T\sum_{m=1}^M (D^{(m)}(n^{(m)})^2 + (n^{(m)})^3))$ time, which is much higher.

Approximation quality.

We now present two theorems that bound the magnitude of $\Delta\lambda_i^{(m)}$ and $\Delta\mathbf{a}_\mathbf{i}^{(\mathbf{m})}$. Both theorems confirm our intuition that the magnitude of $\Delta\lambda_i^{(m)}$ and $\Delta\mathbf{a}_\mathbf{i}^{(\mathbf{m})}$ is directly related to the norm of $\Delta\mathbf{X}^{(\mathbf{m})}$. Also since the higher order terms are ignored in the approximation, FEMA algorithms only works when those terms are relatively small.

Theorem 1 *The magnitude of the variation on the eigenvalue, i.e., $|\Delta\lambda_i^{(m)}|$, $(\forall i = 1, \ldots, r^{(m)})$, satisfies the following inequality*

$$|\Delta\lambda_i^{(m)}| \leq 2(\lambda^{\max}_{\mathbf{X}^{(\mathbf{m})\top}\mathbf{X}^{(\mathbf{m})}})^{\frac{1}{2}} \|\Delta\mathbf{X}^{(\mathbf{m})}\|_2 \tag{16}$$

where $\lambda^{\max}_{\mathbf{X}^{(\mathbf{m})\top}\mathbf{X}^{(\mathbf{m})}}$ is the maximum eigenvalue of the data inner product matrix $\mathbf{X}^{(\mathbf{m})\top}\mathbf{X}^{(\mathbf{m})}$, $\|\Delta\mathbf{X}^{(\mathbf{m})}\|_2$ is the 2-norm of $\Delta\mathbf{X}^{(\mathbf{m})}$.

PROOF. According to Eq.9, we have

$$|\Delta\lambda_i^{(m)}| = |\mathbf{a}_\mathbf{i}^{(\mathbf{m})\top}(\mathbf{X}^{(\mathbf{m})}\Delta\mathbf{X}^{(\mathbf{m})\top} + \Delta\mathbf{X}^{(\mathbf{m})}\mathbf{X}^{(\mathbf{m})\top})\mathbf{a}_\mathbf{i}^{(\mathbf{m})}| \tag{17}$$

By Cauchy-Schwarz inequality,

$$\begin{aligned} &|\mathbf{a}_\mathbf{i}^{(\mathbf{m})\top}(\mathbf{X}^{(\mathbf{m})}\Delta\mathbf{X}^{(\mathbf{m})\top} + \Delta\mathbf{X}^{(\mathbf{m})}\mathbf{X}^{(\mathbf{m})\top})\mathbf{a}_\mathbf{i}^{(\mathbf{m})}| \qquad (18)\\ &\leq 2\|\Delta\mathbf{X}^{(\mathbf{m})}\mathbf{X}^{(\mathbf{m})\top}\mathbf{a}_\mathbf{i}^{(\mathbf{m})}\|_2\|\mathbf{a}_\mathbf{i}^{(\mathbf{m})}\|_2 = 2\|\Delta\mathbf{X}^{(\mathbf{m})}\mathbf{X}^{(\mathbf{m})\top}\mathbf{a}_\mathbf{i}^{(\mathbf{m})}\|_2 \end{aligned}$$

where in the first step we use the symmetry of $\mathbf{X}^{(\mathbf{m})}\Delta\mathbf{X}^{(\mathbf{m})\top} + \Delta\mathbf{X}^{(\mathbf{m})}\mathbf{X}^{(\mathbf{m})\top}$ and in the second step we use the fact that $\|\mathbf{a}_\mathbf{i}^{(\mathbf{m})}\| = 1$. By the definition of matrix 2-norm, we have that

$$\|\Delta\mathbf{X}^{(\mathbf{m})}\mathbf{X}^{(\mathbf{m})\top}\|_2 = \sup_{\|\mathbf{w}\|_2=1} \|\Delta\mathbf{X}^{(\mathbf{m})}\mathbf{X}^{(\mathbf{m})\top}\mathbf{w}\|_2 \tag{19}$$

Therefore

$$\begin{aligned} &|\mathbf{a}_\mathbf{i}^{(\mathbf{m})\top}(\mathbf{X}^{(\mathbf{m})}\Delta\mathbf{X}^{(\mathbf{m})\top} + \Delta\mathbf{X}^{(\mathbf{m})}\mathbf{X}^{(\mathbf{m})\top})\mathbf{a}_\mathbf{i}^{(\mathbf{m})}| \qquad (20)\\ &\leq 2\|\Delta\mathbf{X}^{(\mathbf{m})}\mathbf{X}^{(\mathbf{m})\top}\|_2 \leq 2\|\mathbf{X}^{(\mathbf{m})}\|_2\|\Delta\mathbf{X}^{(\mathbf{m})}\|_2\\ &= 2(\lambda^{\max}_{\mathbf{X}^{(\mathbf{m})\top}\mathbf{X}^{(\mathbf{m})}})^{\frac{1}{2}}\|\Delta\mathbf{X}^{(\mathbf{m})}\|_2 \quad \square \end{aligned}$$

Theorem 2 *The magnitude of the variation on the eigenvector, i.e., $|\Delta\mathbf{a}_\mathbf{i}^{(\mathbf{m})}|$, $(\forall i = 1, \ldots, r^{(m)})$, satisfies the following inequality*

$$|\Delta\mathbf{a}_\mathbf{i}^{(\mathbf{m})}| \leq 2\|\Delta\mathbf{X}^{(\mathbf{m})}\|_2 \sum_{j\neq i} \frac{(\lambda^{\max}_{\mathbf{X}^{(\mathbf{m})\top}\mathbf{X}^{(\mathbf{m})}})^{\frac{1}{2}}}{|\lambda_i^{(m)} - \lambda_j^{(m)}|} \tag{21}$$

where $\lambda^{\max}_{\mathbf{X}^{(\mathbf{m})\top}\mathbf{X}^{(\mathbf{m})}}$ is the maximum eigenvalue of the data inner product matrix $\mathbf{X}^{(\mathbf{m})\top}\mathbf{X}^{(\mathbf{m})}$, $\|\Delta\mathbf{X}^{(\mathbf{m})}\|_2$ is the 2-norm of $\Delta\mathbf{X}^{(\mathbf{m})}$.

PROOF. From Eq.15, we have that

$$\begin{aligned} |\Delta\mathbf{a}_\mathbf{i}^{(\mathbf{m})}| &= 2\left|\sum_{j\neq i} \frac{\mathbf{a}_\mathbf{j}^{(\mathbf{m})\top}\Delta\mathbf{X}^{(\mathbf{m})}\mathbf{X}^{(\mathbf{m})\top}\mathbf{a}_\mathbf{i}^{(\mathbf{m})}}{\lambda_i^{(m)} - \lambda_j^{(m)}}\mathbf{a}_\mathbf{j}^{(\mathbf{m})}\right| \qquad (22)\\ &\leq 2\sum_{j\neq i}\left\|\frac{\mathbf{a}_\mathbf{j}^{(\mathbf{m})\top}\Delta\mathbf{X}^{(\mathbf{m})}\mathbf{X}^{(\mathbf{m})\top}\mathbf{a}_\mathbf{i}^{(\mathbf{m})}}{\lambda_i^{(m)} - \lambda_j^{(m)}}\mathbf{a}_\mathbf{j}^{(\mathbf{m})}\right\|\\ &\leq 2\sum_{j\neq i}\frac{\|\mathbf{a}_\mathbf{j}^{(\mathbf{m})}\|}{|\lambda_i^{(m)} - \lambda_j^{(m)}|}\|\mathbf{a}_\mathbf{j}^{(\mathbf{m})\top}\Delta\mathbf{X}^{(\mathbf{m})}\mathbf{X}^{(\mathbf{m})\top}\mathbf{a}_\mathbf{i}^{(\mathbf{m})}\|\\ &\leq 2\|\Delta\mathbf{X}^{(\mathbf{m})}\|_2\sum_{j\neq i}\frac{(\lambda^{\max}_{\mathbf{X}^{(\mathbf{m})\top}\mathbf{X}^{(\mathbf{m})}})^{\frac{1}{2}}}{|\lambda_i^{(m)} - \lambda_j^{(m)}|} \quad \square \end{aligned}$$

5. EXPERIMENTS

In this section, we evaluate the effectiveness, efficiency and robustness of our proposed FEMA for the tasks of behavior prediction. We also provide interesting discovery of the temporal behavioral patterns for strong additional evidence of the effectiveness.

5.1 Datasets and Experimental Settings

We use the following two real-world datasets in our experiments:

- MAS data [24]: This is a publicly available dataset from Microsoft Academic Search database, which comprises of three files. The first file contains profile information about 250K authors such as author name and affiliation. The second file contains data about 2.5M papers, such as paper title, year and keywords. The third file contains data of corresponding paper-author connections. We first join these three files over author, affiliation, keyword and year of the paper, and then pre-process the data to generate a subset such that each author, affiliation and keyword occur at least 10 times in the dataset. The resulting dataset has 171,519 tuples (author,affiliation,keyword,year) with 7,777 authors, 651 affiliations and 4,566 keywords in 32 years (from 1980 to 2012). The average density of the tensor in each year is less than $3 \times 10^{-5}\%$, while the density of the co-authorship matrix is as large as 0.2%.
- WEIBO data: Tencent Weibo is one of the largest microblogging platforms in China, on which users can post a mention

Statistics	MAS	Statistics	WEIBO
Author	7,777	Source user	6,200
Affiliation	651	Target user	1,813
Keyword	4,566	Word in tweet	6,435
Time	32 years	Time	43 days
Co-authorship	98,671	Social relation	465,438
Number of tuples	171,519	Number of tuples	519,624

Table 1: Characteristics of datasets.

by typing their tweets with "@username". This dataset comprises of two files. The first file contains posting time, user and content of the tweet. From the content, we can recognize the mentioned (target) users. The second file contains social network information of the source and target users. After the preprocessing, we have 519,624 tuples (source,target,word,time) with 6,200 source users, 1,813 target users and 6,435 words in 43 days (from Nov. 9, 2011 to Dec. 21, 2011). The average density of the tensor in each week is less than $2\times10^{-5}\%$, while the density of the social relation matrix is 0.7%.

Tab. 1 summarizes the characteristics of these academic and tweet datasets. We use the two datasets to perform on two different *behavior prediction* tasks. Both the tasks are set up to continuously predict the future behaviors using new-arriving data.

- **2W** (Who-What and Who-Whom) **prediction:** It is to predict the behaviors of the given *author* u to study the given *keyword* v, or the behaviors of the given *source user* u to mention the given *target user* v in their tweets, no matter where the author is or what the tweet content is.
- **3W** (Who-Where-What and Who-Whom-What) **prediction:** The goal is to predict the behaviors of the given *author* u to study the given *keyword* v in the given *affiliation* w, or the behaviors of the given *source user* u to mention the given *target user* v in tweets of the given *word* w.

Fig. 2 shows how we use the data to set up the experiments. The two datasets were split into three parts: training for initialization, training for evolutionary analysis and testing. We use the earliest 30% for initialization, and then let the behavioral data come 5% by 5% for evolutionary analysis, and use the next 20% for testing. In other words, we will test the performance for $T = 10$ times that the percents of the *training* parts are the first $\alpha_t = 35\%, 40\%$ to 80%, for $t = 1, 2, \ldots, T$.

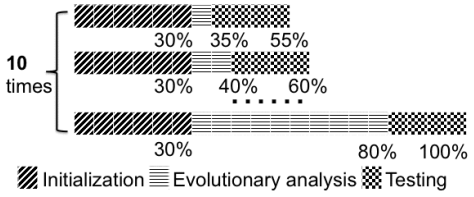


Figure 2: Experimental settings: we use the first 30% data for initialization, and set up 10 times of predictions that each time we train 5% more data, predict and test with the next 20%.

5.2 Competing Algorithms

We evaluate the effectiveness and efficiency of the proposed method. To evaluate the effectiveness, we implement 3 versions of FEMA:

- FEMA models academic research behavior with tensors of authors, affiliations and keywords, and uses the co-authorship regularization. Similarly, it models the behavior of mentioning in tweets with tensors of source users, target users and words, and uses the social regularization.
- EMA models with multi-way tensors but does not utilize regularization. It does not use the co-authorship and social network information.
- EA models the behavior with author-keyword and source-target matrix and uses the standard matrix factorization to predict the missing values. It does not use the multi-faceted information and will only be used for 2W prediction.

To compare with the state-of-the-art methods, we implement the following popular methods:

- CP (CANDECOMP/PARAFAC) [16] decomposes the updated tensor as a sum of rank-one tensors every time. It requires a unified value for the number of groups in each dimension: $r^{(1)} = r^{(2)} = r^{(3)} = R$.
- HOSVD (high-order SVD) [8] is Tucker decomposition of the updated tensor, which is a high-order form of principal component analysis.
- DTA (Dynamic tensor analysis) [26] updates the covariance matrices for quick tensor dimensionality reduction. It does not store any historical tensor but still has to decompose the huge covariance matrix. We will test the online processing ability of our method with it.

To evaluate the approximation quality and efficiency, we also implement an offline learning version of FEMA:

- FMA utilizes the same knowledge as FEMA, however, it merges increments with previous data and processes the decomposition with the updated tensor every time.

We implement our framework in MATLAB and perform the experiments on a single machine with Intel Xeon CPU at 2.40GHz and 32GB RAM, running Windows Server 2008.

By default, the parameters are $r^{(i)} = 50$ and $\mu^{(i)} = 0.3$, for $i = 1, 2, 3$. The discussion for the performances of different parameter settings is given later in Section 5.6.

5.3 Evaluation Methods

For the first task, complex behavior prediction, we use the standard evaluation metrics Mean Absolute Error (*MAE*) and Root Mean Square Error (*RMSE*) [3] defined as:

$$MAE = \frac{\sum_{(u,v,w)\in D} |r_{u,v,w} - \hat{r}_{u,v,w}|}{|D|}$$

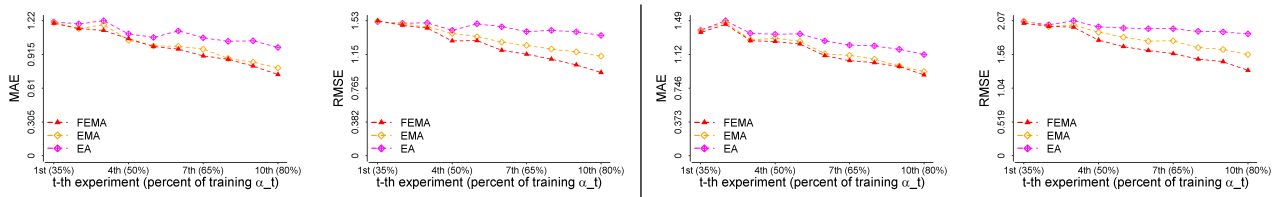
$$RMSE = \frac{\sum_{(u,v,w)\in D} (r_{u,v,w} - \hat{r}_{u,v,w})^2}{|D|}$$

where D denotes the testing set; $r_{u,v,w}$ is the predicted probability of the behavior that author u publishes keyword v in affiliation w or user u mentions user v in tweets of word w; and $\hat{r}_{u,v,w}$ is the frequency of the behaviors in the testing set and 0 if not. Small *MAE* and *RMSE* will be a better model.

Also we use two frequently used metrics, *Precision* and *Recall* [7], to evaluate the quality of ranking for prediction values. Let $T(u, v, w)$ be the set of behaviors in the testing set and let $P(u, v, w)$ be the set of the predicted behaviors. *Precision* considers the positively predicted entries within all the predictions, and *Recall* considers the positively predicted entries within all the positive ones in the testing set, so that we can plot the *Precision-Recall* curves by changing the lower limit of the predicted values for $P(u, v, w)$:

$$Precision = \frac{|P(u,v,w) \cap T(u,v,w)|}{|P(u,v,w)|}$$

$$Recall = \frac{|P(u,v,w) \cap T(u,v,w)|}{|T(u,v,w)|}$$



(a) MAE on MAS data (b) RMSE on MAS data (c) MAE on WEIBO data (d) RMSE on WEIBO data

Figure 3: Multi-faceted analysis improve the 2W prediction performance: FEMA and EMA perform much better than EA that formulate human behavior as matrices instead of high-order tensors. The model is better if the MAE and RMSE are smaller.

High *Precision* and *Recall* will be a better model.

When make prediction on the mentioned (target) users in the tweet data, we generate a list of N ($N = 5$) target users named $R_{u,w}$ for each source user u to mention in his/her tweets of a given word w. If the target user v appears in the list, we call it a hit. The *Hit Ratio* [14] is calculated in the following way:

$$Hit\ Ratio = \frac{\sum_{u,v,w} I(v \in R_{u,w})}{|U|}$$

where $I(\cdot)$ is an indicator function, $R_{u,w}$ is a set of top-N mentioned users to user u in tweet of word w, and v is the hold-off user in the testing set that u posts a tweet with "@v". A high *Hit Ratio* will be a better model.

For the second task, simplex behavior prediction, we sum the entries of the tensor along with the dimension of affiliation or word in tweet w as the result of prediction. Then we define similar definitions of *MAE*, *RMSE*, *Precision*, *Recall*, and *Hit Ratio*, using (u, v) as subscripts instead of (u, v, w) and (u) instead of (u, w).

5.4 Experiments on Behavior Prediction

In this section, we conduct three different experiments to demonstrate the effectiveness and efficiency of the model settings of our FEMA. First, we present the usefulness of leveraging multi-faceted information with 2W prediction tasks on learning behavioral patterns on the academic research data MAS and tweet data WEIBO. Second, we present the usefulness of leveraging flexible regularizations with 3W prediction tasks. And finally, we show the effectiveness and efficiency of our evolutionary analysis.

5.4.1 *Usefulness of Leveraging Multi-faceted Information: 2W Prediction*

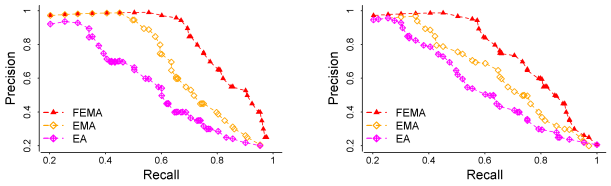
In this subsection, we show the results of 2W prediction: predicting author-keyword behaviors on MAS, and source-target behaviors on WEIBO. We compare our FEMA with EMA and EA, while EA uses matrix instead of high-order tensor to formulate the human behaviors. In other words, EA does not learn from the information of affiliation and word in tweets. Fig. 3 shows MAE and RMSE of the above methods on the 10 experiments varying the percent of training data α_t from 35% to 80% by 5%: Fig. 3a and 3b plot the results of MAS, while Fig. 3c and 3d plot the results of WEIBO. FEMA has the smallest MAE and RMSE, while even EMA is much better than EA. Furthermore, we show the numbers of MAE and RMSE in Tab. 2, when we set the percents of the training part as $\alpha_t = 80\%$. FEMA decreases the RMSE of EA by 30.8% on MAS and 30.0% on WEIBO.

Fig. 4 plots the precision-recall curves to test the ranking results of predicted human behaviors. We show that the tensor-based method EMA performs much better than the matrix-based one EA, and the FEMA performs the best on both MAS and WEIBO data.

EA uses author-keyword matrix to model the academic research behavior, while EMA and FEMA use author-affiliation-keyword tensors to model it, and perform better than EA. The information of affiliation has strong impacts in the keywords: when an author changes his/her affiliation, his/her research topics may change because he/she has got new collaborators and new projects. For example, in Fig. 1 we know that when Professor Jiawei Han moves from Simon Fraser University to University of Illinois at Urbana-Champaign, his main research topics change from the area of database systems to data mining. The methods of multi-faceted analysis EMA and FEMA learn the affiliation information from the MAS dataset and better predict what topic an author will study.

	MAS data		WEIBO data	
	MAE	RMSE	MAE	RMSE
FEMA (+flexible)	**0.735**	**0.944**	**0.894**	**1.312**
EMA (tensor)	0.794	1.130	0.932	1.556
EA (matrix)	0.979	1.364	1.120	1.873

Table 2: The tensor-based methods FEMA and EMA have the smaller MAE and RMSE than the matrix-based method EA on 2W Prediction tasks. FEMA models the behavior as tensors, learns from flexible regularizers, and reaches the smallest errors. The model is better if the MAE and RMSE are smaller.



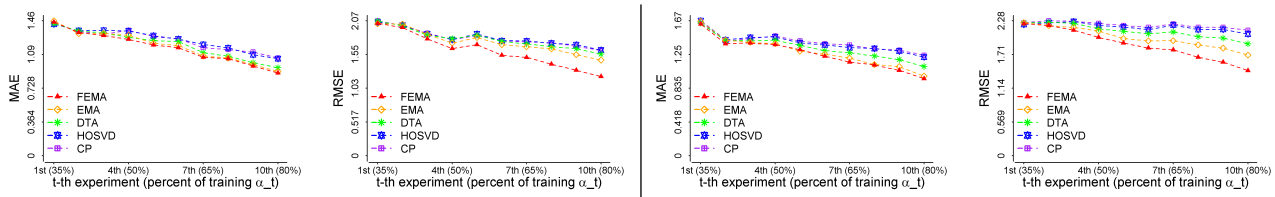
(a) Precision-recall on MAS (b) Precision-recall on WEIBO

Figure 4: FEMA and EMA that use high-order tensors to model human behavior performs the better than the matrix-based EA on 2W Prediction tasks when $t = 10$ and $\alpha_t = 80\%$. The model is better if the precision and recall are higher.

Similarly, EMA and FEMA use the *words in tweets* as the third facet to model the mentioning behavior on the microblogging dataset WEIBO. Weibo users usually mention different accounts in their tweets of different content. For example, sports fans usually mention their favorite players when they post messages to send their congratulations, comforts or best wishes; they mention their friends in life when they hear some interesting news like marriage, graduation, travelling and shopping discounts. Multi-faceted analysis can better model this kind of user behavior and predict who will be mentioned later for a Weibo user.

5.4.2 *Usefulness of Leveraging Flexible Regularizations: 3W Prediction*

As mentioned before, here we predict author-affiliation-keyword



(a) MAE on MAS data (b) RMSE on MAS data (c) MAE on WEIBO data (d) RMSE on WEIBO data

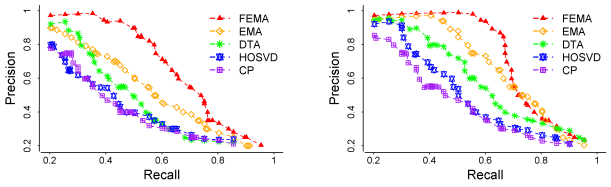
Figure 5: Flexible regularizers alleviate the sparsity problem: on 3W prediction tasks, FEMA performs better than other methods that do not use regularizers. The model is better if the MAE or RMSE is smaller.

behaviors on the academic research dataset MAS, and source (user)-target (user)-word (in tweet) behaviors on the social dataset WEIBO. We compare our FEMA with EMA and three related methods DTA, HOSVD and CP that do not use flexible regularizers on the decompositions. Similarly with Fig. 3, Fig. 5 shows MAE and RMSE of the methods on the 10 experiments varying the percent of training data α_t from 35% to 80% by 5%: Fig. 5a and 5b plot the results of MAS, while Fig. 5c and 5d plot the results of WEIBO. With the size of the training data increasing, the models can learn more from it and thus the MAE and RMSE often decrease by the size. FEMA often reaches the smallest values of MAE and RMSE, which shows that flexible regularizers can alleviate the sparsity problem and thus can help in the prediction task. Furthermore, we show the numbers of MAE and RMSE in Tab. 3, when we set up the experiments with the largest piece of training part $\alpha_t = 80\%$. FEMA decreases the RMSE of the best of the other methods by 17.1% on MAS and 15.4% on WEIBO.

	MAS data		WEIBO data	
	MAE	RMSE	MAE	RMSE
FEMA	**0.893**	**1.215**	**0.954**	**1.437**
EMA	0.909	1.466	0.986	1.698
DTA [26]	0.950	1.556	1.105	1.889
HOSVD [8]	1.047	1.618	1.220	2.054
CP [16]	1.055	1.612	1.243	2.117

Table 3: Flexible FEMA has the smallest MAE and RMSE on 3W Prediction tasks when $t = 10$ and $\alpha_t = 80\%$. The model is better if the MAE and RMSE are smaller.

Similarly with Fig. 4, we also plot the precision-recall curves to test their abilities of ranking the predicted probabilities of human behaviors. In Fig. 6, we show FEMA performs the best when we operate all the algorithms on both MAS data and WEIBO data.



(a) Precision-recall on MAS (b) Precision-recall on WEIBO

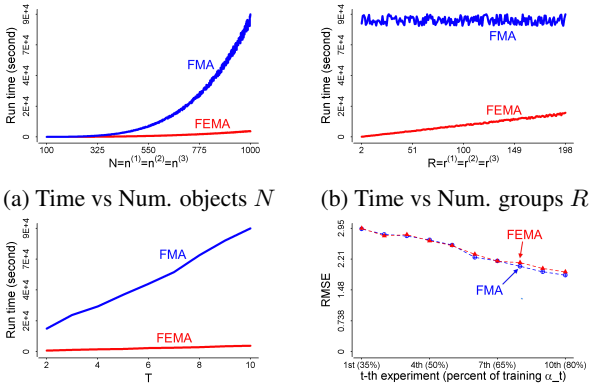
Figure 6: FEMA that uses flexible regularizers performs the best on 3W Prediction tasks when $t = 10$ and $\alpha_t = 80\%$. The model is better if the precision and recall are higher.

FEMA uses the *co-authorship* information to constrain the projection matrices on the dimension of *author*. The co-author network is a complementary graph to the authors' affiliation network. It also has strong impacts in determining the topics of authors' academic research. Though the author-affiliation-keyword tensor is too sparse, learning the co-authorship matrix can better understand and predict the authors' behaviors.

Similarly, on the social dataset WEIBO, FEMA uses the *social network* information to constrain the grouping of both the *source users* and *target users*. When a source user u looks for an appropriate target user v from millions of accounts to mention in his/her tweets, u has often already connected to v and followed up v's messages. Therefore, learning the social information can help predict the users' behaviors of mentioning in tweet.

5.4.3 *Effectiveness and Efficiency*

Here we first test the run time of FEMA by changing the following three factors: (1) the number of objects in each dimension, i.e., the scale of tensors $N = n^{(1)} = n^{(2)} = n^{(3)}$; (2) the number of groups in each dimension $R = r^{(1)} = r^{(2)} = r^{(3)}$; (3) the number of tensor increments T. For convenience, let the number of objects/groups be the same in all the dimensions. Second, we show that the loss of FEMA from FMA is quite small, while FEMA saves lots of time.

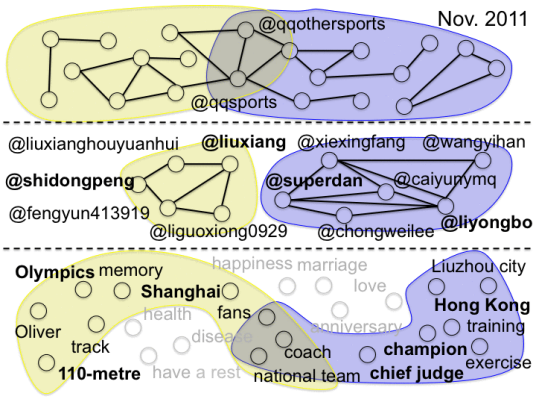


(a) Time vs Num. objects N (b) Time vs Num. groups R

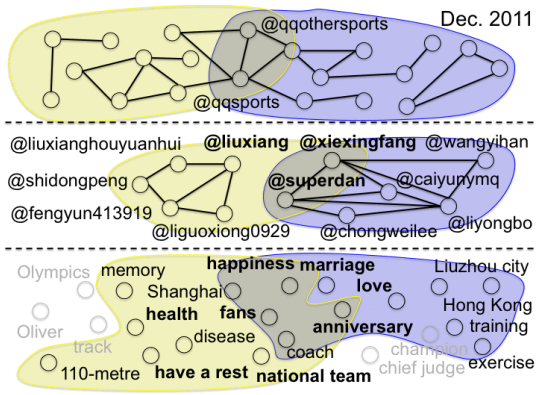
(c) Time vs Num. increments T (d) The loss is small.

Figure 7: FEMA saves lots of time and the loss is small: FEMA runs much faster than FMA.

Fig. 7a shows how much time FEMA and FMA cost by changing the number of objects N in each dimension. We random sample $N \times N \times N$ tensors from WEIBO data, for $N = 100, \ldots, 1000$, so that the density of tensors is stable. We set $R = 50$ and $T = 10$ as default. The run time of FEMA goes up much slower than that of FMA. When FMA takes 25 hours (more than one day) to process, FEMA needs only 51 minutes (less than one hour). Fig. 7b shows the time cost by changing the number of groups R from 2 to 100 in each dimension. We use the $1000 \times 1000 \times 1000$ sample tensors and let T be 10. Though the run time of FEMA is proportional to R, it is still much smaller than that of FMA. Fig. 7c shows that the time cost is linear to the number of tensor increments T. The evolutionary analysis method FEMA updates the projection matrices with sparse increments, saving lots of time on decomposition.



(a) In November 2011, Xiang Liu's fans talk about his championship of Olympics 2004, and a badminton match ended in Hong Kong



(b) In December 2011, Xiang Liu reported his health problem, while Dan Lin and Xingfang Xie celebrated their first marriage anniversary

Figure 8: The temporal pattern of Weibo users' mentioning to someone in tweets: FEMA discovers the groups of hurdle fans and badminton fans in China who use "@" to mention their idols in different words of topics at different times.

In Fig. 7d, we check the loss of FEMA using 3W prediction tasks on the $1000 \times 1000 \times 1000$ sampled tensors and find that FMA achieves smaller RMSE than FEMA but the loss is quite small. Since high-order terms in Eq. 8 are small, though FEMA omits the terms, the result is close to that of FMA.

5.5 Discovery of Behavioral Patterns

In this section, we present interesting discovery from the behaviors of both academic research and mentions in tweet. In the Introduction, we have shown the temporal behavioral pattern of the research groups led by Prof. Jiawei Han in Fig. 1.

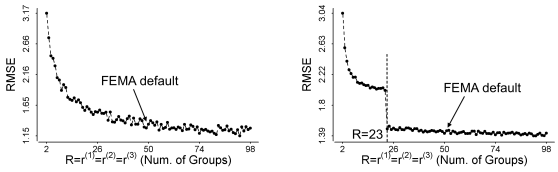
Similarly, in Fig. 8, we give a showcase of our discovery from WEIBO data, where the three layers are source users, target (mentioned) users and words in tweets. Fig. 8a shows that the left yellow groups in the three layers are fans of 110-metre hurdle, hurdle runners including the Olympics 2004 champion Xiang Liu (*@liuxiang*), and words about the sport of hurdle and the runners. The right blue groups are fans of badminton, Chinese famous badminton players and the related words. In November 2011, Xiang's fans mentioned Xiang and his friends like Dongpeng Shi (*@shidongpeng*), talking about their sweet memories of welcoming them back to "Shanghai" in 2004. At the same time, the badminton team of China has just finished their matches in Hong Kong. Their most famous player Dan Lin (*@superdan*) and related words like "champion" get high weights in their corresponding groups.

Fig. 8b shows the temporal behavioral patterns of these two group clusters in December 2011. Xiang Liu reported his sickness, and his fans sent their best wishes by mentioning to him. Dan Lin posted a message, saying it was the first anniversary of marriage with Xingfang Xie (*@xiexingfang*) who is also a badminton player. Therefore, we can see the words "love", "marriage" get higher weights than "training" and "exercise". Note that first, the weights of *@qqsports* and *@qqothersports* increase in the two source user groups. Second, the weights of Dan and Xingfang increase in the two target user groups. We examine the data and find out the reason that *@qqsports*, *@qqothersports* and even some of Xiang's fans congratulated to Dan and Xingfang for their good news.

5.6 Parameter Settings

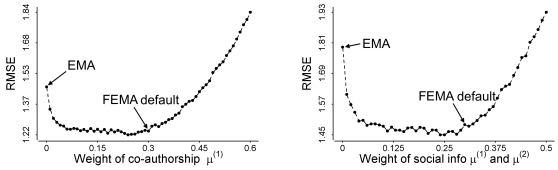
In this section, we discuss how we set the parameters in our experiments: one is the number of groups R and the other is the weight of regularizers μ.

On both datasets, we change the number of groups in each dimension R from 2 to 100 and observe that the RMSE decreases and reaches the bottom when R is larger than 30. In WEIBO, we spot the RMSE decreases much when $R = 23$. As mentioned before, FEMA updates the eigenvectors with the other eigenvectors multiplied by some weights. When R is smaller than 23, if the testing entries have objects in the 23-th cluster, the values of objects are 0 in all the eigenvectors. Thus, the entries will be always zero. We let the default number of groups R to be 50 for the trade-off between better accuracy (smaller RMSE) and faster computing.



(a) Academic data MAS (b) Tweet data WEIBO

Figure 9: FEMA sets the default number of groups R in each dimension as 50: RMSE decreases when R changes from 2 to 100. In WEIBO, the RMSE decreases much when $R = 23$.



(a) Academic data MAS (b) Tweet data WEIBO

Figure 10: FEMA sets the default weight of regularizers μ as 0.3: the performance of our FEMA is insensitive to μ.

Next, we change the weight of the regularizers in our FEMA μ from 0 to 1: when $\mu = 0$, FEMA is actually EMA that does not use the flexible regularizers; when $\mu = 1$, FEMA uses only the regularizers but none information from the tensors. On both MAS and WEIBO datasets, we observe that the RMSE first decreases and reaches the bottom when μ is from 0.1 to 0.3. The RMSE increases fast when μ is larger than 0.3. Note that our FEMA is insensitive to the weight μ. For convenience, we set the default value of the weight as 0.3. We demonstrate that it would be a better model to understand human behavior if it learns from both the sparse high-order tensors and dense flexible regularizers.

6. CONCLUSION

In this work, we present a novel tensor factorization based framework FEMA for temporal multi-faceted behavior prediction and behavioral pattern mining. The model uses flexible regularizers to alleviate the sparsity problem and gives approximation algorithms to fast process the increments with a theoretical guarantee. Extensive experiments performed on real world datasets demonstrate that our framework is effective and efficient in behavior prediction tasks. The fast speed can support real-time applications of behavior prediction and pattern mining.

7. ACKNOWLEDGEMENT

This work is supported by National Natural Science Foundation of China, No. 61370022, No. 61210008, and No. 61303075; International Science and Technology Cooperation Program of China, No. 2013DFG12870; National Program on Key Basic Research Project, No. 2011CB302206; NExT Research Center funded by MDA, Singapore, WBS:R-252-300-001-490.

8. REFERENCES

[1] E. Acar, D. M. Dunlavy, T. G. Kolda, and M. Mørup. Scalable tensor factorizations for incomplete data. *Chemometrics and Intelligent Laboratory Systems*, 106(1):41–56, 2011.

[2] W. Chen, W. Hsu, and M. L. Lee. Making recommendations from multiple domains. In *KDD'13*, pages 892–900.

[3] W. Chen, W. Hsu, and M. L. Lee. Modeling user's receptiveness over time for recommendation. In *SIGIR'13*, pages 373–382.

[4] A. Cichocki and R. Zdunek. Regularized alternating least squares algorithms for non-negative matrix/tensor factorization. In *Advances in Neural Networks–ISNN 2007*, pages 793–802.

[5] P. Cui, S. Jin, L. Yu, F. Wang, W. Zhu, and S. Yang. Cascading outbreak prediction in networks: a data-driven approach. In *KDD'13*, pages 901–909.

[6] P. Cui, F. Wang, S. Liu, M. Ou, S. Yang, and L. Sun. Who should share what? item-level social influence prediction for users and posts ranking. In *SIGIR'11*, pages 185–194.

[7] J. Davis and M. Goadrich. The relationship between precision-recall and roc curves. In *ICML'06*, pages 233–240.

[8] L. De Lathauwer, B. De Moor, and J. Vandewalle. A multilinear singular value decomposition. *SIAM journal on Matrix Analysis and Applications*, 21(4):1253–1278, 2000.

[9] D. M. Dunlavy, T. G. Kolda, and E. Acar. Temporal link prediction using matrix and tensor factorizations. *TKDD*, 5(2):10, 2011.

[10] L. Hu, J. Cao, G. Xu, L. Cao, Z. Gu, and C. Zhu. Personalized recommendation via cross-domain triadic factorization. In *WWW'13*, pages 595–606.

[11] M. Jiang, P. Cui, R. Liu, Q. Yang, F. Wang, W. Zhu, and S. Yang. Social contextual recommendation. In *CIKM'12*, pages 45–54.

[12] M. Jiang, P. Cui, F. Wang, Q. Yang, W. Zhu, and S. Yang. Social recommendation across multiple relational domains. In *CIKM'12*, pages 1422–1431.

[13] U. Kang, E. Papalexakis, A. Harpale, and C. Faloutsos. Gigatensor: scaling tensor analysis up by 100 times-algorithms and discoveries. In *KDD'12*, pages 316–324.

[14] G. Karypis. Evaluation of item-based top-n recommendation algorithms. In *CIKM'01*, pages 247–254.

[15] T. G. Kolda. Orthogonal tensor decompositions. *SIAM Journal on Matrix Analysis and Applications*, 23(1):243–255, 2001.

[16] T. G. Kolda and B. W. Bader. Tensor decompositions and applications. *SIAM review*, 51(3):455–500, 2009.

[17] T. G. Kolda and J. Sun. Scalable tensor decompositions for multi-aspect data mining. In *ICDM'08*, pages 363–372.

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

[19] L. Liu, J. Tang, J. Han, M. Jiang, and S. Yang. Mining topic-level influence in heterogeneous networks. In *CIKM'10*, pages 199–208.

[20] K. Narang, S. Nagar, S. Mehta, L. Subramaniam, and K. Dey. Discovery and analysis of evolving topical social discussions on unstructured microblogs. In *Advances in Information Retrieval*, pages 545–556. 2013.

[21] H. Rachlin. The value of temporal patterns in behavior. *Current Directions in Psychological Science*, 1995.

[22] K. Radinsky, K. M. Svore, S. T. Dumais, M. Shokouhi, J. Teevan, A. Bocharov, and E. Horvitz. Behavioral dynamics on the web: Learning, modeling, and prediction. *TOIS*, 31(3):16, 2013.

[23] S. Rendle and L. Schmidt-Thieme. Pairwise interaction tensor factorization for personalized tag recommendation. In *WSDM'10*, pages 81–90.

[24] S. B. Roy, M. De Cock, V. Mandava, S. Savanna, B. Dalessandro, C. Perlich, W. Cukierski, and B. Hamner. The microsoft academic search dataset and kdd cup 2013. In *Proceedings of the KDD Cup 2013 Workshop*.

[25] G. W. Stewart and J.-g. Sun. Matrix perturbation theory.

[26] J. Sun, D. Tao, S. Papadimitriou, P. S. Yu, and C. Faloutsos. Incremental tensor analysis: Theory and applications. *TKDD*, 2(3):11, 2008.

[27] J.-T. Sun, H.-J. Zeng, H. Liu, Y. Lu, and Z. Chen. Cubesvd: a novel approach to personalized web search. In *WWW'05*, pages 382–390.

[28] Y. Sun, J. Tang, J. Han, C. Chen, and M. Gupta. Co-evolution of multi-typed objects in dynamic star networks. *TKDE*, 2013.

[29] P. Symeonidis, A. Nanopoulos, and Y. Manolopoulos. Tag recommendations based on tensor dimensionality reduction. In *RecSys'08*, pages 43–50.

[30] F. Wang, J. Sun, J. Hu, and S. Ebadollahi. imet: interactive metric learning in healthcare applications. In *SDM'11*, pages 944–955.

[31] F. Wang, H. Tong, and C.-Y. Lin. Towards evolutionary nonnegative matrix factorization. In *AAAI'11*, pages 501–506.

[32] X. Wang, C. Zhai, and D. Roth. Understanding evolution of research themes: a probabilistic generative model for citations. In *KDD'13*, pages 1115–1123.

[33] L. Xiang, Q. Yuan, S. Zhao, L. Chen, X. Zhang, Q. Yang, and J. Sun. Temporal recommendation on graphs via long-and short-term preference fusion. In *KDD'10*, pages 723–732.

[34] Q. Yuan, G. Cong, Z. Ma, A. Sun, and N. M. Thalmann. Who, where, when and what: Discover spatio-temporal topics for twitter users. In *KDD'13*, pages 605–613.

[35] X. Zheng, H. Ding, H. Mamitsuka, and S. Zhu. Collaborative matrix factorization with multiple similarities for predicting drug-target interactions. In *KDD'13*, pages 1025–1033.