Where Have You Been? Inferring Career Trajectory from Academic Social Network

Kan Wu, **Jie Tang** and **Chenhui Zhang**
Department of Computer Science and Technology, Tsinghua University
wu-k14@mails.tsinghua.edu.cn, jietang@tsinghua.edu.cn, ch-zhang15@mails.tsinghua.edu.cn

Abstract

A person's career trajectory is composed of her/his past work or educational affiliations (institutions) at different points of times. Knowing people's, especially scholars', career trajectories can help the government make more scientific strategies to allocate resources and attract talent and help companies make smart recruiting plans. It could also support individuals find appropriate co-researchers or job opportunities. The paper focuses on inferring career trajectories in the academic social network. For about 1/3 of authors not having any affiliations in the dataset, we need to infer the missings at various years. Traditional affiliation/location inferring methods focus on inferring a stationary location (one and only) for a person. Nevertheless, people won't stay at a place all their lives. We propose a Space-Time Factor Graph Model (STFGM) incorporating spatial and temporal correlations to fulfill the challenging and new task of inferring temporal locations. Experiments show our approach significantly outperforms baselines. At last, as case study, we develop several applications based on our approach which demonstrate the effectiveness further.

1 Introduction

The tough competition on personalized information services in a variety of domains such as personalized search engine, intelligent recommendation for TV programs, merchandise, jobs etc. has driven the demand for more precise user profiling [Iguchi, 2007; Park and Chang, 2009; Sugiyama *et al.*, 2004; Yu *et al.*, 2006; Abel *et al.*, 2011; Xue, 2010]. A user's affiliation is an important part of her/his profile. Just as the saying goes: "You cannot judge a man till you know his whole story". Exploring the past affiliations of a person has been studying or working at different times can help better profile her/him, knowing the career trajectories of many people in a specific research area can better understand the development of the area. These can benefit many applications. For instance, the government could better grasp the transitions of talents and accordingly make more scientific policies relating resources allocating, talent attracting. In addition, companies can design smart recruiting plans or just find candidates with specific experience and individuals can find appropriate co-researchers or job opportunities to extend the academic or professional network.

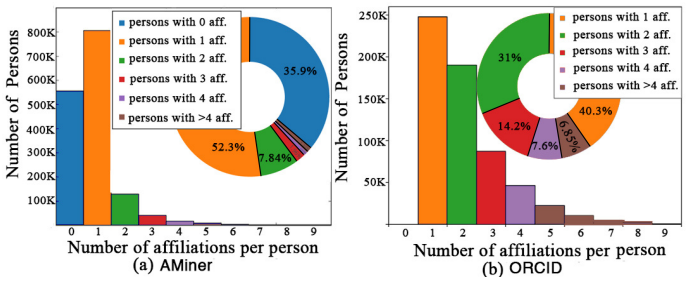


Figure 1: Affiliation statistics on AMiner vs ORCID.

There are some efforts to collect the affiliations of researchers, e.g. ORCID, which requires the user to add her/his profile manually. Unfortunately, the information obtained solely from users themselves is sometimes incomplete. To automatically get the affiliation of a specified user, a usual way is to find her/his home page and then use machine learning techniques to extract from it. Nevertheless, extracting the formatted temporal affiliations from an unstructured biography paragraph is non-trivial, let alone many even don't have home pages [Ceglowski *et al.*, 2003; Tang *et al.*, 2010].

Thanks to the development of the academic social networks such as AMiner[1] and MAG[2] [Tang *et al.*, 2008; Sinha *et al.*, 2015], we can relatively easier get authors' affiliations in published papers. However, the problem of inferring affiliations trajectories from academic social networks still poses challenges. The affiliation information is sparse in the dataset. We sampled about 1.5 million authors in AMiner and found that 0.55 million authors (accounting for 35.9%) don't have any affiliations in their papers, and there are only 11.8% of them having more than one affiliation (Figure 1.a). We also investigated a sample of ORCID for comparison. There are 59.7% (about 5 times of previous one) of people with more than one affiliation (Figure 1.b).

Moreover, traditional affiliation/location inferring methods rarely concern about the time, but people do not tend to stay

[1]https://aminer.org/

[2]https://www.microsoft.com/en-us/research/project/microsoft-academic-graph/

in one place all their lives. Inferring temporal affiliations is more challenging. To the best of our knowledge, there has not been any research about this.

To address the novel problem and the obstacles related, we propose a Space-Time Factor Graph Model (STFGM) which directly models the similarity between authors, and incorporates time and space correlations. At last, for the case study, we develop several applications based on our approach which demonstrate the effectiveness further.

2 Problem Definition

Assume $G = \{G^t\}$ is a time-aware academic network with $G^t = (V_L^t, V_U^t, E^t)$ where all the superscripts t denote time, V_L^t is the affiliation-known authors at time t, V_U^t is the affiliation-unknown authors, $V^t = V_L^t \cup V_U^t$, E^t is the coauthoring relations. Suppose A is the affiliation set, the objective is to learn a predictive function, $f : V_U^t \mapsto A$

3 Related Work

There have been a number of research studies on location inference. Generally speaking, they can be classified into 3 categories. Studies in the first one tried to predict a user/object's location through the content of the user/object. For instance, some tried to predict a user's location from her/his tweets. Cheng et al. explored words' different distributions over regions to identify words in tweets with a strong local geoscope [Cheng *et al.*, 2010]. Eisenstein et al. used a geographic topic model to find topic-specific regional distinctions [Eisenstein *et al.*, 2010]. Chen et al. used a topic model to determine user's interests which were then mapped to locations [Chen *et al.*, 2013]. Wing et al. leveraged language model and information retrieval technology to infer users' locations [Wing and Baldridge, 2011]. Some others attempted to predict webpage's geographical region via exploring its content based on heuristic rules [Amitay *et al.*, 2004]. Ikawa et al. predicted each microblog's location instead of user's through learning out location-relevant keywords from past messages [Ikawa *et al.*, 2012]. One of the limitations of these methods is that they could not get high-resolution locations such as some university or some corporation, because language style won't change significantly in small scale of an area, let alone the more objective academic language in research papers.

The second category leverages the network correlations of users. For example, Davis Jr et al. predicted the user's location with the highest frequent one in the friends, and confirmed friend number would influence precision [Davis Jr *et al.*, 2011]. Jurgens tried to use the social network to predict the user's location. He used Spatial Label Propagation to select a known neighbors' label for unlabeled users and propagate the inferred mappings until convergence [Jurgens, 2013]. Backstrom et al. modeled friendship probability as a function of distance, and they selected the predicted location which maximizes the joint likelihood[Backstrom *et al.*, 2010]. McGee et al. incorporated social tie strengths between users to improve location prediction [McGee *et al.*, 2013].

The third category uses both users' content and network connections. Li et al. treated users and user-tweeted venues as nodes in a network and used respective gaussian distributions to model the nodes' influence scopes. The model tends to select the neighbor with the smallest influence scope [Li *et al.*, 2012]. Another work leveraged users' content to infer their locations if the content contains local words and used friendship to infer locations of users without local words [Ryoo and Moon, 2014].

The general focus of previous methods using network connections is how to select out the nearest neighbor. Most used an indirect metric of neighbors such as the highest frequent, geometric median or the smallest influence scope. In fact, it often may not be the best choice because few took into account the features of the target person and the neighbor pair, which function together to determine the distance. Previous efforts inspired us to build our model and our work was motivated most by McGee et al.'s with the following differences: We directly model the tuple of the user, neighbor and time simultaneously to allow social tie strengths varying with time. In addition, we incorporate the correlations of the tuples in space and time which boost our performance further.

4 Proposed Method

Before proceeding, we first introduce some baseline solutions for this problem.

Time Stretch. If we know a person's affiliations at some discrete years, then the missing affiliations between the years can be predicted with the ones near the affiliation-known years. The approach cannot be adopted if none of the affiliations of a person are known.

Statistics-Based Model. The idea is that the opportunities to work with someone in the same affiliation are much larger than that in different affiliations. Our survey of 2 million randomly selected papers confirmed the assumption, which showed that about 71.3% papers have two or more coauthors from the same affiliation. To predict a missing affiliation of an author at a time t, we can use the affiliation that the most coauthors belong to.

Influence Model. We borrow the idea from [Li *et al.*, 2012], the affiliation-known author i's influence at different locations can be modeled by a gaussian distribution with mean L_i and covariance $\Sigma_i = \begin{pmatrix} \sigma_i & 0 \\ 0 & \sigma_i \end{pmatrix}$, where L_i is the author's location, σ_i reflects the influence ability and can be estimated by affiliation-known coauthors. A person with a wider influence collaborates with many people geographically far apart, while a person with small influence tends to collaborate with people nearby. To predict a missing affiliation of an author in a year, we compute that year's affiliation-known coauthors' influence abilities and choose the one with the smallest.

Space Label Propagation. When the affiliation-missing author's affiliations are inferred, they can be used as known nodes to infer other unknown ones. When inferring a missing affiliation, we choose the affiliation that the most coauthors belong to rather than their locations' simplex median used in [Jurgens, 2013]. Because simplex median often results in wield or institution-free locations such as seas or coast.

Rank-SVM. The problem of inferring affiliation can be formulated as a multiclass classification where the classes are the set of available affiliations. However, the number of dif-

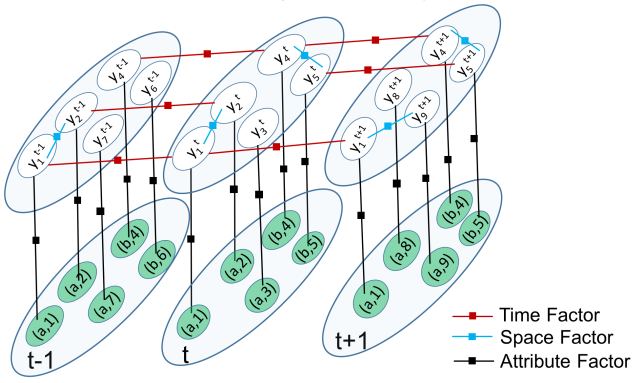


Figure 2: Space-Time Factor Graph Model(STFGM).

ferent affiliations exceeds 10 thousand in our dataset. The 1 vs. 10k+ classification model will fail with limited features. Thus, we change the multiclass classification problem into a binary classification problem. At time t, each author a is associated with individual features $vec(a)^t$ while each coauthorship $< a_{i1}, a_{i2} >$ is associated with coauthoring features $vec(a_{i1}, a_{i2})^t$ and a binary label y_i^t to indicate whether a_{i1} and a_{i2} belong to the same affiliation. Then given a training data, we can build a two-class rankSVM model based on MLE [Scholz, 1985]

$$y_i^t = \arg\max P(y_i^t | \boldsymbol{x}_i^t) \tag{1}$$

where $\boldsymbol{x}_i^t \triangleq [vec(a_{i1}), vec(a_{i2}), vec(a_{i1}, a_{i2})]^t$ is the combination of individual and coauthoring features. The approach predicts the coauthor with the highest similarity score.

Some of the methods above use some temporal connection, some leverage the spatial connection and some capture the features of the node pairs. But none can model them all in a unified way. We now introduce our model in detail.

4.1 Space-Time Factor Graph Model(STFGM)

The general idea of our model is trying to find the affiliation-known coauthors who share the same affiliations as the target authors with missing affiliations.

Figure 2 gives a simplified picture of our STFGM. Each green point with common t outside, representing a tuple of <Time t, Author a_{i1}, Author a_{i2} >, is an observation instance where a_{i1} is the target author and a_{i2} is a coauthor with known affiliation at t. For example, the green point $(b, 6)$ circled by t-1 represents target author b coauthored with person 6 at time $t-1$. We have $a_{i1} \in V_U^t$, $a_{i2} \in V_L^t$, $(a_{i1}, a_{i2}) \in E^t$. Associated with each observation instance is a hidden binary-valued variable representing the affiliation similarity between the two authors. If they belong to the same affiliation at that time, the hidden value is 1, otherwise 0.

Attribute Factor Function: captures the features of each tuple $< t, a_{i1}, a_{i2} >$ and characterizes how the observed tuple features (the respective features of the two authors and the concurrent features between them) contribute to the similarity of the authors in the tuple. The function is defined as an exponential-linear function:

$$f(\boldsymbol{x}_i^t, y_i^t) \triangleq \frac{1}{Z_{\boldsymbol{\omega}}} \exp\left\{\boldsymbol{\omega}^T \boldsymbol{\Phi}(\boldsymbol{x}_i^t, y_i^t)\right\} \tag{2}$$

where $\boldsymbol{x}_i^t \triangleq [vec(a_{i1}), vec(a_{i2}), vec(a_{i1}, a_{i2})]^t$ is the feature vector embracing a_{i1} and a_{i2} 's respective and shared common features at time t; $y_i^t \in \{0, 1\}$ denotes whether the two authors in tuple $< t, a_{i1}, a_{i2} >$ are in the same affiliation at time t ; $\boldsymbol{\omega} \triangleq (\omega_0, \omega_1)$ is the weighting vector; $\boldsymbol{\Phi} \triangleq (\boldsymbol{\Phi}_0, \boldsymbol{\Phi}_1)^T$ is the vector of feature functions with $\boldsymbol{\Phi}_k(\boldsymbol{x}_i^t, y_i^t) \triangleq 1_{y_i^t = k} \boldsymbol{x}_i^t, k \in \{0, 1\}$ defined as indicator function.

Given any a tuple $< t, a_{i1}, a_{i2} >$, the attributes extracted in our system include the features of target author a_{i1} (consisting of the number of coauthors of author a_{i1} at time t, the number of all the coauthors of author a_{i1}, the number of papers published by author a_{i1} at time t, the number of all papers published by author a_{i1}), the features of the coauthor a_{i2} (made up of the number of coauthors of author a_{i2} at time t, the number of all the coauthors of author a_{i2}, the number of papers published by author i at time a_{i2}, the number of all papers published by author a_{i2}), and the shared features of author a_{i1} and a_{i2} (comprised by the number of times author a_{i1} and a_{i2} collaborated at time t, the number of all the times author a_{i1} and a_{i2} collaborated ever).

Space Factor Function: captures the correlation between the hidden variables at the same time (We refer to this correlation as **spatial correlation**). It's defined as an exponential-linear function:

$$\mathcal{S}(y_i^t, \mathcal{N}_S(y_i^t)) \triangleq \frac{1}{Z_{\boldsymbol{\beta}}} \exp\left\{\sum_{y_j^t \in \mathcal{N}_S(y_i^t)} \boldsymbol{\beta}^T \boldsymbol{\Psi}(y_i^t, y_j^t)\right\} \tag{3}$$

where $\mathcal{N}_S(y_i^t)$ denotes neighbors which have spatial correlations with y_i^t, $\boldsymbol{\Psi} \triangleq (\boldsymbol{\Psi}_1, ..., \boldsymbol{\Psi}_C)^T$, C is the number of types of spatial correlations, $\boldsymbol{\Psi}_c \triangleq (\Psi_c^{00}, \Psi_c^{01}, \Psi_c^{10}, \Psi_c^{11})$, $\Psi_c^{kl}(y_i^t, y_j^t) \triangleq 1_{(y_i^t = k, y_j^t = l)}, 1 \le c \le C, 0 \le k, l \le 1$.

We extract two kinds of spatial correlation edges in our implemented system. Suppose the hidden variables corresponding to instances $< t, a, i >$ and $< t, a, j >$ are y_i^t and y_j^t respectively,

1) If author i and j have the same affiliation at time t, we add a space-correlation edge between y_i^t and y_j^t.

2) If author i and j have the same affiliation at any time, we add another space-correlation edge between y_i^t and y_j^t.

Time Factor Function: captures the temporal correlation between different times on the same author pair (We call the correlation **temporal correlation**). It's defined as an exponential-linear function:

$$\mathcal{T}(y_i^t, \mathcal{N}_T(y_i^t)) \triangleq \frac{1}{Z_{\gamma}} \exp\left\{\sum_{y_i^{t'} \in \mathcal{N}_T(y_i^t)} \boldsymbol{\gamma}^T \boldsymbol{\Omega}(y_i^t, y_i^{t'})\right\} \tag{4}$$

where $\mathcal{N}_T(y_i^t)$ denotes neighbors who have temporal correlations with y_i^t, $\boldsymbol{\Omega} \triangleq (\boldsymbol{\Omega}_1, ..., \boldsymbol{\Omega}_{C'})^T$, C' is the number of types of temporal correlations, $\boldsymbol{\Omega}_c \triangleq (\Omega_c^{00}, \Omega_c^{01}, \Omega_c^{10}, \Omega_c^{11})$, $\Omega_c^{kl}(y_i^t, y_i^{t'}) \triangleq 1_{(y_i^t = k, y_i^{t'} = l)}, 1 \le c \le C', 0 \le k, l \le 1$.

We extract two kinds of temporal correlation edges also.

1) If the two instances $< t, a, i >$ and $< t+1, a, i >$ are observed, suppose the corresponding hidden variables are y_i^t and y_i^{t+1} respectively, we add an one-order time-correlation edge between y_i^t and y_i^{t+1}.

2) If the two instances $< t, a, i >$ and $< t+2, a, i >$ are observed, suppose the corresponding hidden variables are y_i^t and y_i^{t+2} respectively, we add another two-order time-correlation edge between y_i^t and y_i^{t+2}.

Model Learning: Now we combine all the factors, observation instances and hidden variables into a unified model. We reuse $\mathcal{N}_S$ and $\mathcal{N}_T$ to denote all the space and time relations. Define $X \triangleq \{\boldsymbol{x}_i^t\} \cup \mathcal{N}_S \cup \mathcal{N}_T$ and $Y \triangleq \{y_i^t\}$ which are two sets representing all the observation instances and the hidden variables respectively.

$$\begin{aligned} P(Y|X,\boldsymbol{\theta}) &= \prod_t \prod_i f(\boldsymbol{x}_i^t, y_i^t)\mathcal{S}(y_i^t, \mathcal{N}_S(y_i^t))\mathcal{T}(y_i^t, \mathcal{N}_T(y_i^t)) \\ &= \frac{1}{Z_{\boldsymbol{\omega}} Z_{\boldsymbol{\beta}} Z_{\boldsymbol{\gamma}}} \exp\left\{(\boldsymbol{\omega}^T, \boldsymbol{\beta}^T, \boldsymbol{\gamma}^T) \sum_t \sum_i g(y_i^t)\right\} \\ &= \frac{1}{\mathcal{Z}_{\boldsymbol{\theta}}} \exp\left\{\boldsymbol{\theta}^T \mathcal{G}(Y)\right\} \end{aligned} \tag{5}$$

where $g(y_i^t) \triangleq (\boldsymbol{\Phi}(\boldsymbol{x}_i^t, y_i^t)^T, \sum_{y_j^t} \boldsymbol{\Psi}(y_i^t, y_j^t), \sum_{y_i^{t'}} \boldsymbol{\Omega}(y_i^t, y_i^{t'}))^T$, $\mathcal{G}$ is an aggregation of the factor functions over all the hidden variables. $\boldsymbol{\theta} \triangleq (\boldsymbol{\omega}^T, \boldsymbol{\beta}^T, \boldsymbol{\gamma}^T)^T$ is the parameter configuration of the model, $\mathcal{Z}_{\boldsymbol{\theta}} \triangleq Z_{\boldsymbol{\omega}} Z_{\boldsymbol{\beta}} Z_{\boldsymbol{\gamma}}$ is the normalization term. For Y is partially labeled, we define our log-likelihood objective function ($\mathcal{O}(\boldsymbol{\theta}) \triangleq logP(Y^L|X,\boldsymbol{\theta})$) on the labeled data Y^L. We use $Y'|Y^L$ to denote the label configuration Y' that satisfies all the known labels Y^L.

Learning the STFGM model is to estimate a parameter configuration $\boldsymbol{\theta}^*$, so that the log-likelihood objective function is maximized.

$$\boldsymbol{\theta}^* = \arg\max_{\boldsymbol{\theta}} \mathcal{O}(\boldsymbol{\theta}) = \arg\max_{\boldsymbol{\theta}} logP(Y^L|X,\boldsymbol{\theta}) \tag{6}$$

We use gradient ascent algorithm to solve.

$$\frac{\partial \mathcal{O}(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} = \sum_{Y'|Y^L} P_{\boldsymbol{\theta}}(Y'|Y^L, X)\mathcal{G}(Y') - \sum_{Y} P_{\boldsymbol{\theta}}(Y|X)\mathcal{G}(Y) \tag{7}$$

The computing of $P_{\boldsymbol{\theta}}(Y'|Y^L, X)$ and $P_{\boldsymbol{\theta}}(Y|X)$ can be approximated using Loopy Belief Propagation (LBP)[Murphy *et al.*, 1999]. We perform LBP twice, one time giving the labeled data and another time not giving it. Denoting η as the learning rate, finally, the $\boldsymbol{\theta}^*$ can be iteratively updated until convergence by:

$$\boldsymbol{\theta}_{new} = \boldsymbol{\theta}_{old} + \eta \frac{\partial \mathcal{O}(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} \tag{8}$$

Inferring Missing Affiliations: After learning out parameter set $\boldsymbol{\theta}$, we can compute the similarity probability of each tuple instance. Given $a \in V_U^t$, for each candidate $i \in V_L^t$ satisfying $(a, i) \in E^t$, suppose the corresponding hidden variable for the tuple $< t, a, i >$ is y_i^t, then the most likely person with the same affiliation at t is:

$$i^* = \arg\max_i P_{\boldsymbol{\theta}}(y_i^t = 1|X) \tag{9}$$

Finally, the affiliation of author a at t can be predicted as:

$$Predict(a,t) = Affiliation(i^*, t) \tag{10}$$

5 Experiments

5.1 Experimental Setup

Datasets

We evaluate our method on datasets constructed from two famous academic networks: AMiner and MAG.

AMiner: (ArnetMiner) is an expertise search and mining service for researcher social networks. Currently, the academic network includes more than 231,832,378 publications and 127,513,531 researchers.

MAG: (Microsoft Academic Graph) is a heterogeneous graph containing scientific publication records, citation relationships between those publications, as well as authors, institutions, journals, conferences, and fields of study, which has more than 126,909,021 papers and 114,698,044 authors.

Datasets Construction: To simultaneously construct our training and testing datasets from AMiner and MAG respectively, we first randomly choose 1336 target authors with at least ten years of affiliations available in their papers and present in both AMiner and MAG. We then collect all their and their coauthors' information in AMiner and MAG respectively. The total authors involved in our datasets constructed from AMiner and MAG are 57037, 147543 respectively. After that, in order to generate training and testing datasets, we introduce two split methods: First one, we randomly split the target authors into training and testing datasets, with the testing authors not knowing any affiliations in training datasets. We call this splitting method **Inter-person train-test split**. Another one, we randomly select some affiliation-known years of every author into training datasets and other years into testing. We call this **Intra-person train-test split**.

According to the testing-dataset "ground truth", we can compute the algorithms' **precision in the dataset** (abbreviated to P1). The way of getting "ground truth" in the last paragraph is very common in machine learning, especially when getting the real ground truth is impossible or the cost is huge. One shortcoming of the above setting is that the ground truth extracted from the dataset could be inaccurate, e.g., the affiliations of a paper are ascribed to the wrong coauthors, or a paper is ascribed to a wrong author due to name ambiguity. Therefore, we further evaluate the performance in terms of the true **precision in the real world** (abbreviated to P2). To get the real world ground truth, we organized a group of researchers to search out the target authors' curricula vitae on the Internet and to organize the information searched out into <when, who, where> format. The granularity of time is year. Each result item was rechecked by two people. Finally, 722 curricula vitae were found along with 21544 high-quality information items. We released the dataset at the site[3].

Evaluation Metrics

We use Ratcliff & Obershelp pattern matching algorithm to compute the affiliation strings' similarity scores [Ratcliff and Metzener, 1988]. If the score is greater than 0.6 (maximum is 1), we treat them the same; otherwise, the different. Because many affiliations can be written in different ways by different people, we have tested other thresholds and found

[3] https://www.aminer.cn/careerMap

Method	Precision in dataset(P1)		Precision in real world(P2)	
	AMiner	MAG	AMiner	MAG
Inter-person train-test split				
Statistics-based	67.48	70.31	56.61	64.49
Influence model	68.98	74.22	55.97	64.68
Space label propagation	71.00	79.23	54.58	60.18
SVM-Rank	73.91	76.47	55.56	66.48
STFGM	**82.98**	**87.32**	**66.87**	**76.89**
Intra-person train-test split				
Statistics-based	67.07	71.86	55.35	64.31
Influence model	62.77	70.01	50.93	61.49
Space label propagation	70.77	80.18	54.51	59.31
SVM-Rank	69.35	78.16	51.02	68.51
Time-Stretch	82.02	86.83	68.57	75.98
STFGM	**90.43**	**92.42**	**72.18**	**82.48**

Table 1: Performance comparison of different methods

0.6 can get most positive pairs with a less than 5% false positive ones. When the inferred <author, time, affiliation> tuple is the same as the ground truth, the right items increase 1. The precision is the ratio of the right items to the total items. The train-test ratio is set to 6:4. For each splitting, we run 10 times and report average results in the following section. It's worth noting that this is a "hit at one problem", so we don't compute recall and F1.

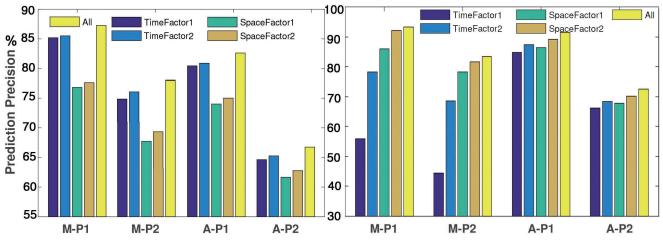
5.2 Experiment Results

Performance Analysis

We compare the performance of all methods in two datasets and on two precision evaluation metrics. Table 1 shows the performance. We find that precisions in the real world have a 6% - 17% drop than ones in the dataset. The performance drop in the real world is inevitable and easy to understand because we never use any information in the real world ground truth to train our models. For STFGM, the Inter-person splits have lower performance than their Intra-person counterparts, because it's harder to predict a person's affiliations without knowing any her/his affiliations in any years. However, the models other than Time-Stretch and STFGM don't achieve significant improvement even with some known affiliations in the Intra-person split, because these models cannot capture the temporal connections. The Time-Stretch method can beat methods other than STFGM because a person's temporal connections with himself are stronger than space connections with other coauthors. Overall, STFGM achieves the best performance because it incorporates spatial and temporal connections together as well as personal attributes.

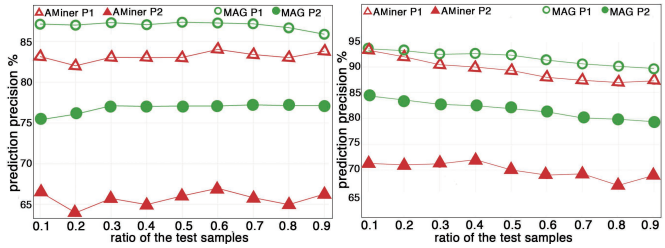
Factor Contribution Analysis

We now give an in-depth analysis of the effects of different factors. We examine the contribution of different factors by removing each of them. Figure 3(a) shows the results in two datasets with the Inter-person split. Figure 3(b) is the counterpart with intra-person split. In Figure 3(a), we can see clearly that the performance drops significantly about 7% - 11% without space factors. The time factors also have 1% - 3% performance boost contribution. While in Figure 3(b), time factors contribute more than space factors. Especially in MAG, time factors contribute 13% - 39 %. This is because with the Intra-person split there are more time connections from training instances to testing instances in target persons, while with Inter-person split all the affiliations of the target persons are unknown.



(a) (b)

Figure 3: Factor contribution analysis. A-P and M-P stand for precision in AMiner and MAG respectively. P1 stands for precision in the dataset and P2 stands for precision in the real world. TimeFactor1, TimeFactor2, SpaceFactor1 and SpaceFactor2 each stands for removing the corresponding correlations in factor graph. (a) Performance comparison of Inter-person train-test split. (b)Performance comparison of Intra-person train-test split.



(a) (b)

Figure 4: Train/test Ratio Analysis. P1 stands for precision in the dataset and P2 stands for precision in the real world. (a) Performance comparison of Inter-person train-test split. (b) Performance comparison of Intra-person train-test split.

Train/Test Ratio Analysis

From Figure 3(a), we can see the performance with the Inter-person split keeps relatively stable as test sample ratio increases. While with the Intra-person split, Figure 3(b) demonstrates the performance goes down slightly with the increasing of test sample ratio. This is because the time connections from training instances to testing instances in target persons decreases as train sample ratio drops. However, even with a low training ratio of 0.1 (test ratio 0.9), the performance is still highly acceptable.

Complexity Analysis

Suppose the number of instances and edges in STFGM is $|V|$ and $|E|$, the time complexity is $O(|V| + |E|)$. Generally, our model takes 50 - 100 iterations to converge. For a typical configuration of 80719 instances and 133947 edges, it takes 4 minutes to converge in a MacBook with 2.5 GHz Intel Core i7 and 16G 1600MHz DDR3 Memory, while the other methods take less than a minute. Good news is this can be done offline when precision is the priority concern, and the learning process of the model can be further accelerated by graph partition and parallel computing technologies.

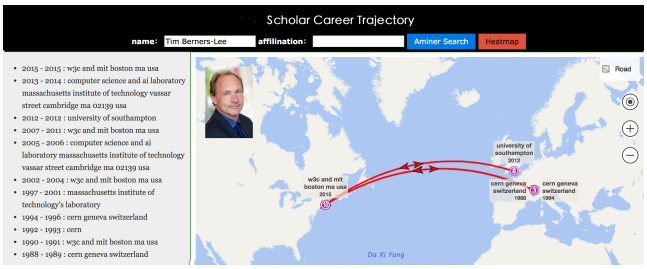


Figure 5: Scholar Career Trajectory for Tim Berners-Lee.

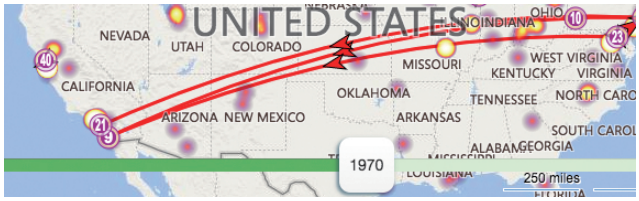


Figure 6: Scholars Group Migration Heatmap in 1970.

6 Case Study

Based on our proposed STFGM model, we then developed several applications as the case study.

6.1 APP 1: Scholar Career Trajectory

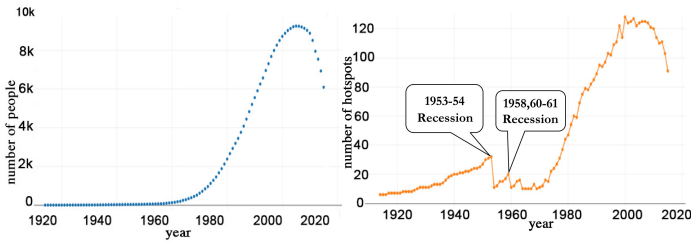
Given an author in the academic social network, this application can automatically list out the author's career experiences and draw the trajectory path on the map. Figure 5 is the results generated automatically for "Tim Berners-Lee", 2016 Turing Award winner. The results basically accord with the bio on his homepage.

6.2 APP 2: Scholars Group Migration Heatmap

In this application, we collect top 10000 scientists in academic network sorted by h-index, then we infer and draw all their trajectories covering one century (from 1913 to 2016) into one dynamic map. The reason for selecting top scientists is they often attract a bunch of other scientists working with them, so their migrations are more representative and may reflect some trends. We then merge nearby persons into hotspots with radius 50km and take hotspots as our research objects in order that their group behavior can be more robust. For a vivid example of the application, as is well-known, Los Angeles became a focus for intellectual discourse in the 1970s after 10 years of urban decay in 1960s.[4] Figure 6 certificate the history, clearly showing that many from the east coast of the U.S. migrated to Los Angeles in 1970.

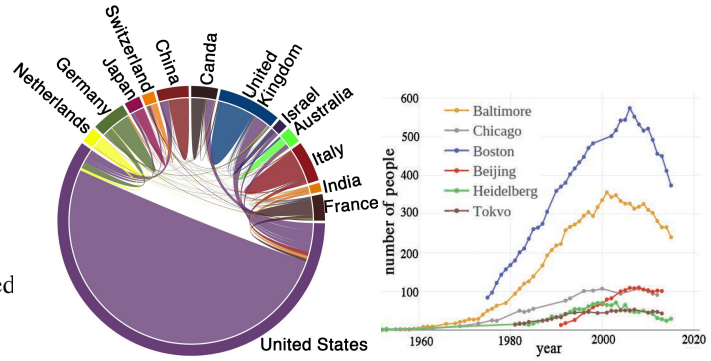
On further analysis, we find some interesting phenomena. Figure 7(a) plots the distribution of the active scholars over years. "Active" indicates that they still published some paper(s). We see most of the top scholars are still active in recent ten years. Figure 7(b) plots the number of hotspots distributed over years. The phenomenon revealed is even more amazing that the biggest and the second biggest drops coincide astonishingly well with two historic economic recessions at those

[4]https://www.laconservancy.org/explore-la/curating-city/modern-architecture-la/history-la-modernism/1970-1980-los-angeles



(a) (b)

Figure 7: (a) People distribution. (b) Hotspots distribution.



(a) (b)

Figure 8: (a) Transitions between/in countries. (b) Dynamics of several big cities.

periods. [5] Figure 8(a) depicts all the transitions between or in countries, we can see most transitions happened domestically. The most frequently emigrating terminal of other countries is the U.S. and the top 3 inter-country mutual transitions are (U.S., U.K.), (U.S., Canada), and (U.S., Germany). This is reasonable because the U.K. has a deep historical connection with the U.S., Canada is the biggest neighbor of the U.S. and Germany has many world-famous scientists. Figure 8(b) shows the dynamics of some big cities. We can see cities at East coast of the United States such as Boston and Baltimore enjoy a considerably higher growth rate than other places, but the uptrend turns to the downside after the year 2005. Beijing exhibits a very different growth pattern – starting up very late near the year 1990, but the immigration trend rises significantly in recent years.

7 Conclusions

The paper focuses on temporal location inferring in the academic social network. It proposed a Space-Time Factor Graph Model (STFGM) incorporating spacial and temporal correlations to fulfill the task. Experiments on two datasets (AMiner and MAG) demonstrate that STFGM outperforms baselines significantly by 5%-27%. The model developed is generic and can be adopted to other datasets with time and space dimensions. At last, two applications were developed to demonstrate the effectiveness of our model.

[5]https://en.wikipedia.org/wiki/Recession_of_1953

References

[Abel *et al.*, 2011] Fabian Abel, Qi Gao, Geert-Jan Houben, and Ke Tao. Semantic enrichment of twitter posts for user profile construction on the social web. In *Extended Semantic Web Conference*, pages 375–389. Springer, 2011.

[Amitay *et al.*, 2004] Einat Amitay, Nadav Har'El, Ron Sivan, and Aya Soffer. Web-a-where: geotagging web content. In *Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 273–280. ACM, 2004.

[Backstrom *et al.*, 2010] Lars Backstrom, Eric Sun, and Cameron Marlow. Find me if you can: improving geographical prediction with social and spatial proximity. In *WWW'10*, pages 61–70. ACM, 2010.

[Ceglowski *et al.*, 2003] Maciej Ceglowski, Aaron Coburn, and John Cuadrado. Semantic search of unstructured data using contextual network graphs. *National Institute for Technology and Liberal Education*, 10, 2003.

[Chen *et al.*, 2013] Yan Chen, Jichang Zhao, Xia Hu, Xiaoming Zhang, Zhoujun Li, and Tat-Seng Chua. From interest to function: Location estimation in social media. In *AAAI'13*, 2013.

[Cheng *et al.*, 2010] Zhiyuan Cheng, James Caverlee, and Kyumin Lee. You are where you tweet: a content-based approach to geo-locating twitter users. In *Proceedings of the 19th ACM international conference on Information and knowledge management*, pages 759–768. ACM, 2010.

[Davis Jr *et al.*, 2011] Clodoveu A Davis Jr, Gisele L Pappa, Diogo Rennó Rocha de Oliveira, and Filipe de L Arcanjo. Inferring the location of twitter messages based on user relationships. *Transactions in GIS*, 15(6):735–751, 2011.

[Eisenstein *et al.*, 2010] Jacob Eisenstein, Brendan O'Connor, Noah A Smith, and Eric P Xing. A latent variable model for geographic lexical variation. In *EMNLP'10*, pages 1277–1287. Association for Computational Linguistics, 2010.

[Iguchi, 2007] Makoto Iguchi. User-profile based web page recommendation system and user-profile based web page recommendation method, June 29 2007. US Patent App. 11/772,071.

[Ikawa *et al.*, 2012] Yohei Ikawa, Miki Enoki, and Michiaki Tatsubori. Location inference using microblog messages. In *WWW'12*, pages 687–690. ACM, 2012.

[Jurgens, 2013] David Jurgens. That's what friends are for: Inferring location in online social media platforms based on social relationships. *ICWSM'13*, 13:273–282, 2013.

[Li *et al.*, 2012] Rui Li, Shengjie Wang, Hongbo Deng, Rui Wang, and Kevin Chen-Chuan Chang. Towards social user profiling: unified and discriminative influence model for inferring home locations. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1023–1031. ACM, 2012.

[McGee *et al.*, 2013] Jeffrey McGee, James Caverlee, and Zhiyuan Cheng. Location prediction in social media based on tie strength. In *CIKM'13*, pages 459–468. ACM, 2013.

[Murphy *et al.*, 1999] Kevin P Murphy, Yair Weiss, and Michael I Jordan. Loopy belief propagation for approximate inference: An empirical study. In *Proceedings of the Fifteenth conference on Uncertainty in artificial intelligence*, pages 467–475. Morgan Kaufmann Publishers Inc., 1999.

[Park and Chang, 2009] You-Jin Park and Kun-Nyeong Chang. Individual and group behavior-based customer profile model for personalized product recommendation. *Expert Systems with Applications*, 36(2):1932–1939, 2009.

[Ratcliff and Metzener, 1988] John W Ratcliff and David E Metzener. Pattern matching:the gestalt approach. *Dr Dobbs Journal*, 13(7):46, 1988.

[Ryoo and Moon, 2014] KyoungMin Ryoo and Sue Moon. Inferring twitter user locations with 10 km accuracy. In *WWW'14*, pages 643–648. ACM, 2014.

[Scholz, 1985] FW Scholz. Maximum likelihood estimation. *Encyclopedia of statistical sciences*, 1985.

[Sinha *et al.*, 2015] Arnab Sinha, Zhihong Shen, Yang Song, Hao Ma, Darrin Eide, Bo-june Paul Hsu, and Kuansan Wang. An overview of microsoft academic service (mas) and applications. In *Proceedings of the 24th international conference on world wide web*, pages 243–246. ACM, 2015.

[Sugiyama *et al.*, 2004] Kazunari Sugiyama, Kenji Hatano, and Masatoshi Yoshikawa. Adaptive web search based on user profile constructed without any effort from users. In *Proceedings of the 13th international conference on World Wide Web*, pages 675–684. ACM, 2004.

[Tang *et al.*, 2008] Jie Tang, Jing Zhang, Limin Yao, Juanzi Li, Li Zhang, and Zhong Su. Arnetminer: extraction and mining of academic social networks. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 990–998. ACM, 2008.

[Tang *et al.*, 2010] Jie Tang, Limin Yao, Duo Zhang, and Jing Zhang. A combination approach to web user profiling. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 5(1):2, 2010.

[Wing and Baldridge, 2011] Benjamin P Wing and Jason Baldridge. Simple supervised document geolocation with geodesic grids. In *ACL'11*, pages 955–964. Association for Computational Linguistics, 2011.

[Xue, 2010] Ding Xue. Research on the book intelligent recommendation system based on data mining [j]. *Information Studies: Theory & Application*, 5:029, 2010.

[Yu *et al.*, 2006] Zhiwen Yu, Xingshe Zhou, Yanbin Hao, and Jianhua Gu. Tv program recommendation for multiple viewers based on user profile merging. *User modeling and user-adapted interaction*, 16(1):63–82, 2006.