

Fortune Teller: Predicting Your Career Path

Ye Liu[†], Luming Zhang[‡], Liqiang Nie[†], Yan Yan[§], David S. Rosenblum[†]

[†] School of Computing, National University of Singapore, Singapore

[‡] Department of Electric Engineering and Information System, Hefei University of Technology, China

[§] Department of Information Engineering and Computer Science, University of Trento, Italy
 {liuye, david}@comp.nus.edu.sg, {zglumg, nieliqiang}@gmail.com, yan@disi.unitn.it

Abstract

People go to fortune tellers in hopes of learning things about their future. A future career path is one of the topics most frequently discussed. But rather than rely on “black arts” to make predictions, in this work we scientifically and systematically study the feasibility of career path prediction from social network data. In particular, we seamlessly fuse information from multiple social networks to comprehensively describe a user and characterize progressive properties of his or her career path. This is accomplished via a multi-source learning framework with fused lasso penalty, which jointly regularizes the source and career-stage relatedness. Extensive experiments on real-world data confirm the accuracy of our model.

Introduction

With the proliferation of social network services, an increasing number of individuals are involved in multiple social networks at the same time. This trend has been witnessed by a recent survey: multi-platform use is on the rise, and 52% of online adults now use multiple social media services (Cohen 2014). Following this trend, multi-platform applications have attracted many researchers’ attention due to the fact that multiple social networks can characterize the same user from different perspectives. For example, Twitter reflects users’ casual activities and personal opinions; Facebook exposes users’ social connections and daily events explicitly; and LinkedIn uncovers users’ professional skills and career paths. The heterogeneous information distributed across those diverse social networks is usually complementary rather than conflicting. Hence, as compared to single social network, appropriate aggregation of multiple social networks could provide a better way to understand users comprehensively, and consequently facilitate many applications, such as the inference of users’ age, gender, race, occupation, personality and political orientation.

With the help of those multiple social networks, in this paper, we scientifically study an ancient problem, the *fortune teller*, specifically in the field of career path prediction. A user’s career path, in this work, refers to the user’s occupational growth in his or her career life. It comprises sev-

eral distinct career stages, and each stage contains a set of equivalent occupational titles. The objective of this work is to predict the future career stages of a given user, the so-called *career path modeling*, which can provide potential benefits for employees, employers and headhunters. For employees, they can get information about their current career stages, the time point for their next job-hopping, as well as the whole picture of their own career paths. For employers, they will be informed of the career progressions of their employees and decide what would be the best time to promote their employees or increase their salaries. When it comes to headhunters, they could be advised of the appropriate time to talk to their target customers as well as the proper job positions for their customers. These efficient and accurate job hunting and recommendation processes will greatly facilitate headhunting and reduce their efforts considerably. As a consequence, career path modeling is a research topic with high potential and has many real-world applications.

Despite its significant value, career path modeling from multiple social networks is a non-trivial task due to the following reasons. (1) **Source Fusion**. The information from multiple social networks of the same user describes his or her characteristics from various views, but it should reflect his or her career progression *consistently*. Thus, how to seamlessly and effectively fuse such heterogeneous information is a tough challenge. (2) **Temporal Relatedness Modeling**. A user’s career path normally comprises a sequence of occupations. Instead of mutual independence, they are correlated with each other in chronological order. Therefore, how to temporally characterize such relatedness poses another challenge. (3) **Influential Factors Identification**. Different career paths have different influential factors. For example, education background may play an important role in the *academic* career path, but it might not be so crucial to an *acting* career. Furthermore, even within the same career path, the influential factors for different career stages also vary. For instance, publications might be key factors to research fellows, while research community services may be a more significant consideration for full professors. Hence, learning the stage-sharing and stage-specific features in each career path presents another crucial challenge.

To tackle the above challenges, we present a multi-source learning framework with a fused lasso penalty (MSLFL). It co-regularizes the following factors: (1) **Source Consis-**

tency. In particular, the predicted results from individual sources should be the same or similar. Thus their disagreements should be penalized. (2) **Temporal Smoothness.** The career path is equally split into multiple time intervals, and each of them is treated as a task. A career path is generally a gradual process, and hence sudden changes of career stages between neighboring time points should be penalized. For instance, it is much more smooth for a research fellow to become an assistant professor rather than a full professor in their next position. (3) **Feature Learning.** Features extracted from multiple sources are in high-dimension spaces. We employ a fused lasso to control sparsity and identify the task-sharing and task-specific features, which will identify the influential factors that affect a user’s career progression at different stages. This enhances the interpretation of influential factors.

We summarize the contributions as follows:

- As far as we know, this is the first work on career path modeling and prediction by exploring multiple social network sources. It can serve as a lifelong career mentor for every professional.
- We present a novel multi-source learning framework with fused lasso penalty to integrate multiple social network sources and model a career path. In addition, this model is able to identify the influential factors of career paths.

Related Work

It is worth mentioning that several research efforts have been dedicated to occupation analysis from social networks (Preoțiuc-Pietro, Lampos, and Aletras 2015; Filatova and Prager 2012; Sloan et al. 2015). For example, Preoțiuc-Pietro *et al.* (2015) inferred the occupation of a user based upon user profiles and social contents. However, they mainly used the information from one single source, which makes it difficult to comprehensively characterize a user’s personality from various aspects. Instead of learning from a single source, multi-source learning has been proposed and has demonstrated its success in user modeling, profiling and behavior analysis with the assumption that information extracted about the same user from different sources may complement one another (Abel et al. 2011; 2013; Meo et al. 2013; Xiang et al. 2013; Huang et al. 2014; Song et al. 2015a; 2015b). For instance, Huang *et al.* (2014) proposed a multi-source integration framework to infer a user’s occupation by combining both content and network information from Sina Weibo. However, as compared to occupation inference, career path modeling is much more complex, since it exhibits dual heterogeneities. In particular, besides comprehensive user description, a career path comprises a sequence of occupations and progressively develops from junior to senior stages. Multi-source learning fails to consider these progressions, and thus their performance is far from satisfactory.

Multi-task learning is a learning paradigm that jointly learns multiple related tasks and has demonstrated its advantages in handling dynamic progression problems in many domains, such as medical science and transportation (Zhang and Yeung 2010; Chang and Yang 2014; Zhou et al. 2011;

2012; Liu et al. 2015; Zheng and Ni 2013). In this framework, the prediction at each time point is treated as a task, and the intrinsic correlations among different time points are automatically learned, which could capture the dynamics effectively. For example, Zhou *et al.* (2011) proposed a multi-task learning model to capture the intrinsic temporal relatedness for disease progression prediction. However, most existing efforts on multi-task learning failed to consider the appropriate source fusion, which usually leads to suboptimal performance.

Thus, multi-view multi-task learning is proposed to explore both source relatedness and task relatedness simultaneously (Zhang and Huan 2012; He and Lawrence 2011; Jin et al. 2013). For instance, He *et al.* (2011) proposed a graph-based iterative framework (*Gram*²) for multi-view multi-task learning and obtained impressive results in text categorization applications. However, as far as we know, the literature on multi-view multi-task learning is relatively sparse, and very few efforts have been applied to career path modeling. In contrast, our MSLFL model provides a natural way to fuse information from different sources by penalizing their disagreements. Moreover, MSLFL can better capture the dynamic progressions of career paths and learn the stage-sharing and stage-specific features via fused lasso penalty.

Data Collection and Preprocessing

Social Accounts Alignment. To the best of our knowledge, there is no available benchmark dataset suitable for career path modeling. We thus created new datasets by crawling four popular career paths, namely *software engineer*, *sales*, *consultant* and *marketing*. Each career path is an individual dataset. Social accounts assignment is the key challenge in this data collection, which aims to build the links among different social network accounts of the same user (Abel et al. 2011; 2013; Meo et al. 2013). Towards this end, we employed the social service About.me¹, which encourages its users to list their multiple social accounts explicitly in their personal profiles. We collected the data from About.me by searching keywords corresponding to the career paths. Considering the *software engineer* dataset as an example, we used “software engineer”, “programmer” and “program developer” as the keywords to search from About.me and got 6,284 candidates. Then we retained only those candidates who provided their Twitter (Tw), Facebook (Fb) and LinkedIn (Lk) accounts as the *software engineer* dataset. The dataset statistics² are presented in Table 1.

Career Stages and Occupation Variants. Based on prior knowledge, we roughly split each career path in our collected datasets into four stages, where each stage represents a milestone within the whole career path. Take *software engineer* as an example. We define it to compose four stages, namely *software developer*, *senior software developer*, *manager* and *CEO*. It is worth noting that due to vocabulary vari-

¹<https://about.me/>

²The size of dataset depends on two factors: the total number of users returned from about.me and the number of returned users with three accounts (Twitter, Facebook and LinkedIn) available at the same time.

Table 1: Statistics of our collected benchmark dataset.

| career path | # of users crawled | # of matching users with Lk, Fb, Tw |
|--------------------------|--------------------|-------------------------------------|
| <i>software engineer</i> | 6,284 | 2,478 |
| <i>sales</i> | 4,359 | 1,412 |
| <i>consultant</i> | 13,067 | 3,567 |
| <i>marketing</i> | 8,974 | 4,884 |

ations, different users utilize different occupational titles to refer to the same career stage. For instance, in the *marketing* career, *product managers* and *marketing coordinators* can be used to refer to the same stage as *marketing executive*, and we thus manually mapped all these semantically similar occupations to the same career stage. Table 2 demonstrates the career stage partition and occupational title variants for the same career stage. Due to limited space, we are unable to list all title variants.

Table 2: Career stage partitions and term variants for the same career stage.

| | Career stages |
|--------------------------|--|
| <i>software engineer</i> | software engineer/developer/programmer senior software engineer/developer principal/manager/leader/director CEO/CTO/founder/owner |
| <i>sales</i> | sales executive senior sales executive/team lead/assistant manager sales director/manager founder/CEO/vice president/president |
| <i>consultant</i> | analyst/associate/consultant senior associate/engagement manager/senior consultant principal/senior manager/director partner/founder/co-founder |
| <i>marketing</i> | marketing executive/product managers/marketing coordinators senior marketing executive/marketing manager/brand manager marketing director/CMO founder/owner/CEO |

Ground Truth Construction. To simplify the computation, we quantized the four career stages in different career paths from junior to senior as 1 to 4. For instance, *software developer*, *senior software developer*, *manager* and *CEO* in the *software engineer* path are mapped to 1, 2, 3 and 4, respectively. In addition, for each career path, we defined several time stamps with equivalent time period. In particular, we chose the start time of a user’s first job as the first time stamp. For example, for a given user u , his/her first work is a *software developer*; we thus labeled his/her career stage at time t_0 as 1. Assuming that the gap between two neighboring time stamps in the *software engineer* path is three years, then four years later the occupational title in his/her LinkedIn is *senior software developer*. We thus label his/her career stages at time t_1 as 2. On the other hand, from users’ LinkedIn profiles, we can obtain their working experiences, including occupational titles and corresponding time periods (i.e., the career stage and time stamp). Figure 1 depicts the ground truth construction for the career datasets.

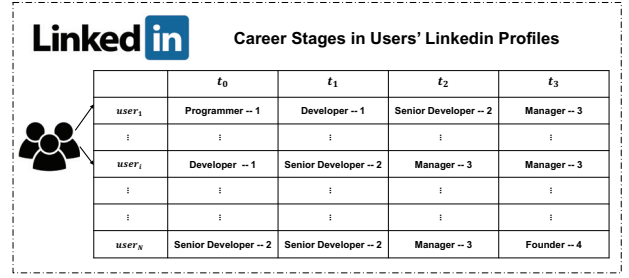


Figure 1: Illustration of ground truth construction.

Career Path Modeling

Notation

We first define some notation. In particular, we use bold capital letters (e.g., \mathbf{X}) and bold lowercase letters (e.g., \mathbf{x}) to denote matrices and vectors, respectively. We employ non-bold letters (e.g., x) to represent scalars, and Greek letters (e.g., λ) as parameters. Unless stated, otherwise, all vectors are in column form.

Let us assume that we have N labeled users for a given career path. This career path can be split into M time points and each time point is aligned with a task. Meanwhile, each user is described by $S \geq 2$ sources. $\mathbf{X}_s \in \mathbb{R}^{N \times D_s}$ denotes the feature matrix extracted from the s -th source, where D_s is the feature dimension of the corresponding source. The whole data matrix can be written as $\mathbf{X} = \{\mathbf{X}_1, \mathbf{X}_2, \dots, \mathbf{X}_S\} \in \mathbb{R}^{N \times D}$, where $D = \sum_{s=1}^S D_s$. The label matrix is denoted as $\mathbf{Y} = \{\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_M\} \in \mathbb{R}^{N \times M}$, and $\mathbf{y}_m = (\mathbf{y}_m^1, \mathbf{y}_m^2, \dots, \mathbf{y}_m^N)^T \in \mathbb{R}^N$ is the label vector of the m -th task.

Problem Formulation

The career status of users at the m -th time point can be linearly predicted from the s -th source as follows:

$$f_m^s(\mathbf{X}_s) = \mathbf{X}_s \mathbf{w}_m^s, \quad (1)$$

where $\mathbf{w}_m^s \in \mathbb{R}^{D_s}$ denotes the linear mapping function for the task m with source s . Without prior knowledge on the contributions of different sources, we assume that all sources contribute equally. Thus, the final prediction model of all sources for task m is obtained by the following late fusion:

$$f_m(X) = \frac{1}{S} \sum_{s=1}^S f_m^s(\mathbf{X}_s). \quad (2)$$

Information distributed in various sources in fact describes the inherent characteristics of the same user from various views, and hence their predicted results should be forced to be similar. In a sense, we can reinforce the learning performance of individual sources. Considering the least-squares loss function, we can define the following objective function:

$$\min_{\mathbf{w}_m^s} \frac{1}{2} \sum_{m=1}^M \|\mathbf{y}_m - \frac{1}{S} \sum_{s=1}^S \mathbf{X}_s \mathbf{w}_m^s\|_2^2 + \frac{\lambda}{2} \sum_{m=1}^M \sum_{s,s'=1}^S \|\mathbf{X}_s \mathbf{w}_m^s - \mathbf{X}_{s'} \mathbf{w}_m^{s'}\|_2^2. \quad (3)$$

To consider the temporal smoothness of career progressions and learn descriptive features, we expand the model in Eqn. (3) to incorporate a fused lasso penalty. This penalty ensures a small deviation between two tasks at successive time stamps and automatically selects task-specific and task-sharing features for career path modeling. In particular, the fused lasso penalty comprises a temporal lasso and a regular lasso. Let $\mathbf{W} = \{\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_M\} \in \mathbb{R}^{D \times M}$ denote the overall weight matrix, where $\mathbf{w}_m = \{\mathbf{w}_m^1, \mathbf{w}_m^2, \dots, \mathbf{w}_m^S\}^T \in \mathbb{R}^D$. The overall objective function can be restated as

$$\begin{aligned} \min_{\mathbf{w}_m^s} \frac{1}{2} \sum_{m=1}^M \|\mathbf{y}_m - \frac{1}{S} \sum_{s=1}^S \mathbf{X}_s \mathbf{w}_m^s\|_2^2 + \gamma \sum_{m=1}^{M-1} \|\mathbf{w}_{m+1} - \mathbf{w}_m\|_1 \\ + \frac{\lambda}{2} \sum_{m=1}^M \sum_{s, s'=1}^S \|\mathbf{X}_s \mathbf{w}_m^s - \mathbf{X}_{s'} \mathbf{w}_m^{s'}\|_2^2 + \theta \|\mathbf{W}\|_1, \end{aligned} \quad (4)$$

where λ, γ and θ are regularization parameters. $\|\cdot\|_1$ denotes the entry-wise matrix ℓ_1 -norm.

Optimization

The optimization of our overall objective function is not easy due to the two non-smooth terms: temporal lasso and regular lasso. To solve this problem, we first rewrite the second term in Eqn. (4) as,

$$\begin{aligned} TL &= \sum_{m=1}^{M-1} \|\mathbf{w}_{m+1} - \mathbf{w}_m\|_1 \\ &= \|\mathbf{W}\mathbf{H}\|_1 \equiv \max_{\|\mathbf{A}\|_{\infty} \leq 1} \langle \mathbf{A}, \mathbf{W}\mathbf{H} \rangle, \end{aligned} \quad (5)$$

where $\mathbf{H} \in \mathbb{R}^{M \times (M-1)}$ is defined as follows: $H_{ij} = -1$ if $i = j$, $H_{ij} = 1$ if $i = j + 1$, and $H_{ij} = 0$ otherwise. $\langle \mathbf{U}, \mathbf{V} \rangle \equiv \text{tr}(\mathbf{U}^T \mathbf{V})$ denotes a matrix inner product; $\mathbf{A} = \{\mathbf{A} \mid \|\mathbf{A}\|_{\infty} \leq 1, \mathbf{A} \in \mathbb{R}^{D \times (M-1)}\}$ is an auxiliary matrix associated with $\|\mathbf{W}\mathbf{H}\|_1$; and $\|\cdot\|_{\infty}$ is the matrix entry-wise ℓ_{∞} -norm. Eqn. (5) is still a non-smooth term. We approximate it by the following smooth one (Chen et al. 2011),

$$TL_{\mu}(\mathbf{W}) = \max_{\|\mathbf{A}\|_{\infty} \leq 1} \langle \mathbf{A}, \mathbf{W}\mathbf{H} \rangle - \mu d(\mathbf{A}), \quad (6)$$

where $d(\mathbf{A})$ is defined as $d(\mathbf{A}) = \frac{1}{2} \|\mathbf{A}\|_F^2$. We can obtain an analytical solution of \mathbf{A} in Eqn. (6). It is obvious that TL_{μ} in Eqn. (6) is a lower bound of TL in Eqn. (5), and the parameter μ controls the gap between the two:

$$G = \frac{1}{\mu} \max_{\|\mathbf{A}\|_{\infty} \leq 1} d(\mathbf{A}) = \frac{1}{2\mu} \|\mathbf{A}\|_F^2 = \frac{1}{2\mu} D(M-1). \quad (7)$$

The gradient of $TL_{\mu}(\mathbf{W})$ is computed by

$$\nabla TL_{\mu}(\mathbf{W}) = \mathbf{A}^* \mathbf{H}^T, \quad (8)$$

where \mathbf{A}^* is the optimal solution of Eqn. (6) and is computed by

$$\mathbf{A}^* = \Psi\left(\frac{\mathbf{W}\mathbf{H}}{\mu}\right), \quad (9)$$

where Ψ is a defined as follows: For $x \in \mathbb{R}$, $\Psi(x) = x$ if $-1 < x < 1$; $\Psi(x) = 1$, if $x \geq 1$; and $\Psi(x) = -1$, if $x \leq -1$. For matrix \mathbf{A} , $\Psi(\mathbf{A})$ applies Ψ on each entry of \mathbf{A} .

Then the overall objective function can be approximated by

$$\tilde{\Gamma} = L + \lambda C + \gamma TL_{\mu} + \theta L_1. \quad (10)$$

With the discussion above, it is easy to show that Eqn. (10) is a convex function with three smooth terms and one non-smooth lasso penalty. We define

$$\begin{aligned} h(\mathbf{W}) &= L + \lambda C + \gamma TL_{\mu}, \\ g(\mathbf{W}) &= \theta L_1. \end{aligned} \quad (11)$$

We can thus use the Fast Iterative Shrinkage-Thresholding Algorithm (FISTA) (Beck and Teboulle 2009) to solve it. One of the key steps in using FISTA is to solve the proximal step:

$$\begin{aligned} \mathbf{W}^{(k)} &= \underset{\mathbf{W}}{\text{argmin}} \{g(\mathbf{W}) + \\ &\frac{L_k}{2} \|\mathbf{W} - (\mathbf{V}^{(k)} - \frac{1}{L_k} \nabla h(\mathbf{V}^{(k)}))\|_F^2\}, \end{aligned} \quad (12)$$

where $\mathbf{V}^{(k)}$ is the search point and is defined by the affine combination of $\mathbf{W}^{(k-1)}$ and $\mathbf{W}^{(k-2)}$; and L_k is a scalar that can be determined by the line search method (Beck and Teboulle 2009). The gradient $\nabla h(\mathbf{W})$ in Eqn. (12) can be computed as

$$\nabla h(\mathbf{W}) = \frac{1}{S} \mathbf{X}^T \left(\frac{1}{S} \mathbf{X} \mathbf{W} - \mathbf{Y} \right) + \mathbf{P} \mathbf{W} + \gamma \mathbf{A}^* \mathbf{H}^T, \quad (13)$$

where $\mathbf{P} \in \mathbb{R}^{D \times D}$ is a sparse block matrix with $S \times S$ blocks, and its entries are defined as,

$$\begin{cases} \mathbf{P}_{ss} = \lambda(S-1) \mathbf{X}_s^T \mathbf{X}_s, \\ \mathbf{P}_{ss'} = -\lambda \mathbf{X}_s^T \mathbf{X}_{s'}. \end{cases} \quad (14)$$

As Eqn.(12) is computed in every FISTA iteration, it needs to be solved efficiently. Specifically, it can be reformulated as

$$\mathbf{W}^{(k)} = \underset{\mathbf{W}}{\text{argmin}} \left\{ \frac{1}{2} \|\mathbf{W} - \mathbf{B}\|_F^2 + \beta \|\mathbf{W}\|_1 \right\}, \quad (15)$$

where $\mathbf{B} = \mathbf{V}^{(k)} - \frac{1}{L_k} \nabla h(\mathbf{V}^{(k)})$ and $\beta = \frac{\theta}{L_k}$. It is easy to show that Eqn.(15) has a closed-form solution (Wright, Nowak, and Figueiredo 2009),

$$\mathbf{W}^{(k)} = \max(0, 1 - \frac{\beta}{\|\mathbf{B}\|_1}) \mathbf{B}. \quad (16)$$

We thus can solve Eqn.(12) quite efficiently.

Experiments

Experimental Settings

To validate our model, we first need to define the time stamps. In particular, we treated the start time of a user's first job as the first time stamp (time t_0), and set three years as the time window between two neighboring time stamps, since three years appears to be a typical period of time between transitions in a person's career path. In this work, we examined four successive time stamps for each user, since we believe that nine years should be long enough to reflect a person's career progressions effectively (Veiga 1981). Meanwhile, each time stamp is aligned with a task. We kept the same settings for all careers we considered. In addition, we employed the average classification accuracy over these four tasks in each career as our performance metric. The experimental results reported in this paper are based on 10-fold cross-validation. The parameters were selected using grid search on each career dataset.

Feature Extraction

We extracted some career-oriented features from multiple social networks to informatively represent each user.

- **Demographic features.** As reported previously (Stacy 2003), some demographic characteristics, such as education and participation, are important factors that influence the speed of users' career progression. We thus extracted a rich set of demographic features including *gender*, *education level*, *relationship status*, and *the number of social connections* from users' LinkedIn and Facebook profiles.
- **LIWC features.** LIWC is a widely-used psycholinguistic analysis tool for investigating the relationship between word use and psychological variables, and it has been successfully applied to identify personality and social traits of users in many social applications (Paek et al. 2010). It has been reported that personality and social traits are strongly correlated to a user's career path (Kafetsios et al. 2009; Judge and Kammeyer-Mueller 2007). The main component of LIWC is a dictionary that contains mapping from words to 72 psychologically meaningful categories.³ For a given document, LIWC calculates the frequency of words related to the particular category and represents the document as a 72-dimensional feature vector.
- **User topic features.** According to our observation, the topics discussed by users are strong indicators of their career stages. For example, *software engineers* may frequently talk about programming skills, while *CEOs* may be interested in the topic of company management or business. This drives us to explore the topic distributions of their social posts. In particular, we employed Latent Dirichlet Allocation (LDA) (Blei, Ng, and Jordan 2003) to generate topic distributions, which has been proven to be effective in latent topic modeling (Hu et al. 2012; Li, Huang, and Zhu 2010). With the assistance of this tool⁴, we ultimately obtained 85-, 45- and 130-dimensional topic-level features from users' Twitter, Facebook and LinkedIn profiles, respectively. The number of topics over each source were separately tuned by optimizing the perplexity metric (Blei, Ng, and Jordan 2003).

Learning Model Comparison

To demonstrate the effectiveness of our MSLFL model, we compared our model with the following five baselines:

- **SVM:** The first baseline is the Support Vector Machine (SVM), which is a mono-source mono-task learning method that concatenates the feature vectors from all sources to form a single feature vector and then learns each task individually. We selected a linear kernel and implemented this method based on LIBSVM (Chang and Lin 2011).
- **RLS:** Regularized least square (RLS) is a multi-source mono-task learning method, which learns each task separately by minimizing $\frac{1}{2N} \|\mathbf{y}_m - \frac{1}{S} \sum_{s=1}^S \mathbf{X}_s \mathbf{w}_m^s\|_2^2 + \frac{\lambda}{2} \|\mathbf{w}_m\|_2^2$.

³<http://www.liwc.net/>

⁴<http://nlp.stanford.edu/software/tmt/tmt-0.4/>

- **MTL:** Multi-task learning (MTL) is a mono-source multi-task learning approach (Zhang and Yeung 2010), and it is able to capture pairwise task relatedness.
- **FL:** Fused Lasso (FL) (Tibshirani et al. 2005) is a mono-source multi-task approach, which aims to capture dynamic progressions by minimizing the objective function $\frac{1}{2} \|\mathbf{Y} - \mathbf{X}\mathbf{W}\|_F^2 + \alpha \sum_{m=1}^{M-1} \|\mathbf{w}_{m+1} - \mathbf{w}_m\|_1 + \beta \|\mathbf{W}\|_1$.
- **regMVMT:** The regularized multi-view multi-task learning model (regMVMT) (Zhang and Huan 2012) jointly regularizes source consistency and uniform task relatedness.

The experimental results are summarized in Table 3. From this table, we have the following observations: 1) The last four multi-task learning methods stably outperform the first two mono-task learning methods, which verifies that the tasks are not independent and that jointly learning them can boost learning performance. Moreover, it is not unexpected that SVM achieves the worst performance. This may be due to two reasons. First, appending the features from different sources directly may lead to suboptimal performance, since they may belong to different feature spaces. Second, certain tasks may hold insufficient training samples. 2) As compared to MTL, our model and regMVMT achieve higher accuracies due to the fact that MSLFL and regMVMT can incorporate heterogeneous information from different sources, which may help to improve overall performance. 3) FL performs better than MTL and regMVMT. This is because MTL and regMVMT assume uniform task relatedness, while FL captures the temporal relatedness between neighboring time points and is more suitable for progressive applications. In addition, FL automatically discovers task-sharing and task-specific features for a better representation. This further confirms the correctness of our hypothesis that there exist certain temporal patterns in career path progression and that the fused lasso penalty is effective for capturing these dynamic progressions. 4) The MSLFL model significantly outperforms regMVMT in *sales* and *marketing* and shows superiority in *software engineer* and *consultant*, which underscores the complex career progressions in various careers paths. For instance, *sales* and *marketing* may exhibit stronger progressive trends than *software engineer* and *consultant*. In addition, our model outperforms FL. This demonstrates that our model is able to leverage source relatedness and further improve performance.

Source Comparison

To demonstrate the descriptiveness of multiple social networks, we compared the performance of our model over individual social networks and their various combinations.

The results are presented in Table 4. From this table, we observe that the combinations of two distinct social networks outperform each individual one, and that the "Twitter+Facebook+LinkedIn" combination achieves the best performance over the others. This observation reveals that the more sources fed to our model, the better the performance that can be achieved. This verifies that information from various sources are complementary to each other instead of conflicting, and that information fusion from multiple sources

Table 3: Performance comparison among various approaches over different career paths. The p-values are the pairwise significance test between the MSLFL model and each of the baselines based on 10-fold cross-validation results.

| Career paths | <i>software engineer</i> | | <i>sales</i> | | <i>consultant</i> | | <i>marketing</i> | |
|--------------|--------------------------|---------|---------------------|---------|---------------------|---------|---------------------|---------|
| Approaches | Accuracy (%) | P-value | Accuracy (%) | P-value | Accuracy (%) | P-value | Accuracy (%) | P-value |
| SVM | 56.94 ± 1.08 | 4.8e-5 | 64.11 ± 1.21 | 2.0e-6 | 50.95 ± 1.21 | 8.0e-8 | 55.45 ± 1.07 | 1.5e-7 |
| RLS | 57.08 ± 2.28 | 8.0e-4 | 67.86 ± 2.32 | 3.9e-4 | 56.76 ± 0.76 | 1.6e-6 | 60.03 ± 0.74 | 4.3e-6 |
| MTL | 59.90 ± 1.38 | 4.6e-3 | 69.78 ± 1.91 | 1.2e-3 | 59.58 ± 0.43 | 3.8e-4 | 62.45 ± 0.38 | 1.0e-4 |
| FL | 61.19 ± 1.59 | 5.0e-2 | 72.43 ± 1.57 | 3.8e-2 | 61.66 ± 0.76 | 4.7e-2 | 63.70 ± 0.59 | 3.9e-2 |
| regMVM | 60.68 ± 2.08 | 4.2e-2 | 70.84 ± 1.76 | 3.9e-3 | 60.72 ± 0.58 | 1.3e-3 | 62.63 ± 0.98 | 4.6e-3 |
| MSLFL | 63.43 ± 1.48 | - | 74.56 ± 1.11 | - | 62.78 ± 0.77 | - | 64.60 ± 0.57 | - |

Table 4: Performance illustration of the MSLFL model over different source combinations on various career paths. The p-values are the pairwise significance test between the outputs of different source combinations.

| Career paths | <i>software engineer</i> | | <i>sales</i> | | <i>consultant</i> | | <i>marketing</i> | |
|-----------------------------|--------------------------|---------|---------------------|---------|---------------------|---------|---------------------|---------|
| Social network combinations | Accuracy (%) | P-value | Accuracy (%) | P-value | Accuracy (%) | P-value | Accuracy (%) | P-value |
| Twitter | 60.42 ± 1.16 | 7.3e-3 | 71.12 ± 0.98 | 8.3e-4 | 61.62 ± 0.67 | 3.4e-2 | 63.69 ± 0.51 | 2.9e-2 |
| Facebook | 60.98 ± 1.71 | 4.1e-2 | 72.36 ± 1.33 | 2.2e-2 | 61.42 ± 0.67 | 1.6e-2 | 63.66 ± 0.64 | 3.9e-2 |
| LinkedIn | 57.81 ± 0.98 | 1.0e-4 | 68.32 ± 1.71 | 1.3e-4 | 48.85 ± 0.43 | 4.0e-10 | 53.90 ± 1.13 | 6.0e-8 |
| Twitter+Facebook | 62.27 ± 1.30 | 2.2e-1 | 73.62 ± 1.18 | 2.3e-1 | 62.21 ± 0.84 | 2.9e-1 | 64.13 ± 0.38 | 1.6e-1 |
| Twitter+LinkedIn | 61.61 ± 0.82 | 4.3e-2 | 72.63 ± 1.32 | 3.6e-2 | 62.00 ± 1.12 | 2.3e-1 | 64.08 ± 0.77 | 2.6e-1 |
| Facebook+LinkedIn | 61.79 ± 1.09 | 8.2e-2 | 72.86 ± 1.96 | 1.2e-1 | 61.91 ± 0.59 | 7.9e-2 | 63.90 ± 0.84 | 1.6e-1 |
| Twitter+Facebook+LinkedIn | 63.43 ± 1.48 | - | 74.56 ± 1.11 | - | 62.78 ± 0.77 | - | 64.60 ± 0.57 | - |

can comprehensively capture users’ characteristics. In addition, we observed that the performance of MSLFL on Twitter and Facebook is better than that on LinkedIn. This may be caused by the data sparsity in LinkedIn, since most of the users update their LinkedIn profiles less frequently. Moreover, even on the same source, the predictive performance of the model over different career paths is completely different, which implies that the capability of source description varies from career to career.

Computational Complexity Analysis

In this section, we discuss the computational complexity for solving the MSLFL model. For the optimization of \mathbf{W} , the main computational cost comes from calculating the gradient $\nabla h(\mathbf{W})$ in each FISTA iteration. In particular, as $\mathbf{X}^T \mathbf{X}$, $\mathbf{X}^T \mathbf{Y}$ and \mathbf{P} remain unchanged in every FISTA iteration, we can pre-compute and store them with time complexity of $O(ND^2 + NMD)$. For each FISTA iteration, computing the gradient in Eqn. (8) takes $O(M^2D)$ time, and therefore the total time complexity for calculating the gradient $\nabla h(\mathbf{W})$ in Eqn. (13) is $O((D+M)DM)$. On the other hand, computing the closed-form solution in Eqn.(16) takes $O(DM)$ time for $\mathbf{W}^{(k)}$. Hence, the complexity for each iteration in the FISTA algorithm is $O((D+M)DM)$. Meanwhile, the FISTA algorithm converges within $O(1/\epsilon^2)$ iterations, and the total time cost of FISTA for solving MSLFL is $O(\frac{(D+M)DM}{\epsilon^2})$, where ϵ is the desired accuracy. Thus, the MSLFL model can be solved efficiently. Moreover, the per-iteration complexity of FISTA for solving MSLFL is independent of the sample size N , which demonstrates that our model has the potential to scale to large-scale Web data.

Conclusion and Future Work

This work has presented a scientific fortune teller for career path prediction. The algorithm behind it is a novel multi-source multi-task learning model. This model fuses information distributed over multiple social networks to characterize users from multiple views. Meanwhile, it jointly penalizes disagreements among sources and the sudden changes between tasks on two neighboring time points. In addition, it is able to learn the task-sharing and task-specific features simultaneously. Extensive experimental results have demonstrated its superiority over the state-of-the-arts competitors.

In the future, we will extend our model to consider the source descriptiveness and learn the source confidence adaptively. Moreover, we will explore model-based approaches for title standardization, which may help our approach scale.

Acknowledgments

This research was supported in part by grants R-252-000-473-133 and R-252-000-473-750 from the National University of Singapore.

References

- Abel, F.; Araújo, S.; Gao, Q.; and Houben, G.-J. 2011. Analyzing cross-system user modeling on the social web. In *Proceedings of the International Conference on Web Engineering*, 28–43.
- Abel, F.; Herder, E.; Houben, G.-J.; Henze, N.; and Krause, D. 2013. Cross-system user modeling and personalization on the social web. *User Modeling and User-Adapted Interaction* 23(2-3):169–209.
- Beck, A., and Teboulle, M. 2009. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM Journal on Imaging Sciences* 2(1):183–202.

- Blei, D. M.; Ng, A. Y.; and Jordan, M. I. 2003. Latent dirichlet allocation. *Journal of Machine Learning Research* 3:993–1022.
- Chang, C.-C., and Lin, C.-J. 2011. Libsvm: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology* 2(3):27.
- Chang, X., and Yang, Y. 2014. Semi-supervised feature analysis by mining correlations among multiple tasks. *CoRR* abs/1411.6232.
- Chen, X.; Lin, Q.; Kim, S.; Carbonell, J. G.; and Xing, E. P. 2011. Smoothing proximal gradient method for general structured sparse learning. In *Proceedings of the Conference on Uncertainty in Artificial Intelligence*, 105–114.
- Cohen, H. 2014. Are you ready for multi-platform social media use? <http://heidicohen.com/multi-platform-social-media-use/>. Accessed on 24 November 2015.
- Filatova, E., and Prager, J. 2012. Occupation inference through detection and classification of biographical activities. *Data & Knowledge Engineering* 76:39–57.
- He, J., and Lawrence, R. 2011. A graph-based framework for multi-task multi-view learning. In *Proceedings of the International Conference on Machine Learning*, 25–32.
- Hu, Y.; John, A.; Wang, F.; and Kambhampati, S. 2012. Et-lda: Joint topic modeling for aligning events and their twitter feedback. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 59–65.
- Huang, Y.; Yu, L.; Wang, X.; and Cui, B. 2014. A multi-source integration framework for user occupation inference in social media systems. *World Wide Web* 18(5):1247–1267.
- Jin, X.; Zhuang, F.; Wang, S.; He, Q.; and Shi, Z. 2013. Shared structure learning for multiple tasks with multiple views. In *Proceedings of the European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases*, 353–368.
- Judge, T. A., and Kammeyer-Mueller, J. D. 2007. Personality and career success. *Handbook of career studies* 59–78.
- Kafetsios, K.; Maridaki-Kassotaki, A.; Zammuner, V. L.; Zampetakis, L. A.; and Vouzas, F. 2009. Emotional intelligence abilities and traits in different career paths. *Journal of Career Assessment* 17(4):367–383.
- Li, F.; Huang, M.; and Zhu, X. 2010. Sentiment analysis with global topics and local dependency. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 1371–1376.
- Liu, Y.; Nie, L.; Han, L.; Zhang, L.; and Rosenblum, D. S. 2015. Action2activity: Recognizing complex activities from sensor data. In *Proceedings of the International Joint Conference on Artificial Intelligence*, 1617–1623.
- Meo, P. d.; Ferrara, E.; Abel, F.; Aroyo, L.; and Houben, G.-J. 2013. Analyzing user behavior across social sharing environments. *ACM Transactions on Intelligent Systems and Technology* 5(1):14.
- Paek, T.; Gamon, M.; Counts, S.; Chickering, D. M.; and Dhesi, A. 2010. Predicting the importance of newsfeed posts and social network friends. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 1419–1424.
- Preoțiuc-Pietro, D.; Lampos, V.; and Aletras, N. 2015. An analysis of the user occupational class through Twitter content. In *Proceedings of the Conference of the Association for Computational Linguistics*, 1754–1764.
- Sloan, L.; Morgan, J.; Burnap, P.; and Williams, M. 2015. Who tweets? deriving the demographic characteristics of age, occupation and social class from twitter user meta-data. *PLoS ONE* 10(3):e0115545.
- Song, X.; Nie, L.; Zhang, L.; Akbari, M.; and Chua, T.-S. 2015a. Multiple social network learning and its application in volunteerism tendency prediction. In *Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval*, 213–222.
- Song, X.; Nie, L.; Zhang, L.; Liu, M.; and Chua, T.-S. 2015b. Interest inference via structure-constrained multi-source multi-task learning. In *Proceedings of the International Joint Conference on Artificial Intelligence*, 2371–2377.
- Stacy, M. E. W. 2003. *Influences of selected demographic variables on the career decision-making self-efficacy of college seniors*. Ph.D. Dissertation, Northwestern State University.
- Tibshirani, R.; Saunders, M.; Rosset, S.; Zhu, J.; and Knight, K. 2005. Sparsity and smoothness via the fused lasso. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 67(1):91–108.
- Veiga, J. F. 1981. Plateaued versus nonplateaued managers: Career patterns, attitudes, and path potential. *Academy of Management Journal* 24(3):566–578.
- Wright, S. J.; Nowak, R. D.; and Figueiredo, M. A. T. 2009. Sparse reconstruction by separable approximation. *IEEE Transactions on Signal Processing* 57(7):2479–2493.
- Xiang, S.; Yuan, L.; Fan, W.; Wang, Y.; Thompson, P. M.; and Ye, J. 2013. Multi-source learning with block-wise missing data for alzheimer’s disease prediction. In *Proceedings of the ACM International Conference on Knowledge Discovery and Data Mining*, 185–193.
- Zhang, J., and Huan, J. 2012. Inductive multi-task learning with multiple view data. In *Proceedings of the ACM International Conference on Knowledge Discovery and Data Mining*, 543–551.
- Zhang, Y., and Yeung, D.-Y. 2010. A convex formulation for learning task relationships in multi-task learning. In *Proceedings of the Conference on Uncertainty in Artificial Intelligence*, 733–742.
- Zheng, J., and Ni, L. M. 2013. Time-dependent trajectory regression on road networks via multi-task learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 1048–1055.
- Zhou, J.; Yuan, L.; Liu, J.; and Ye, J. 2011. A multi-task learning formulation for predicting disease progression. In *Proceedings of the ACM International Conference on Knowledge Discovery and Data Mining*, 814–822.
- Zhou, J.; Liu, J.; Narayan, V. A.; and Ye, J. 2012. Modeling disease progression via fused sparse group lasso. In *Proceedings of the ACM International Conference on Knowledge Discovery and Data Mining*, 1095–1103.