Prospecting the Career Development of Talents: A Survival Analysis Perspective*

Huayu Li[1,2], Yong Ge[3], Hengshu Zhu[2], Hui Xiong[4], and Hongke Zhao[5]
[1]UNC Charlotte, [2]Baidu Talent Intelligence Center, [3]University of Arizona,
[4]Rutgers University, [5]University of Sci. & Tech. of China, zhhk@mail.ustc.edu.cn
hli38@uncc.edu,yongge@email.arizona.edu,zhuhengshu@baidu.com,hxiong@rutgers.edu

ABSTRACT

The study of career development has become more important during a time of rising competition. Even with the help of newly available big data in the field of human resources, it is challenging to prospect the career development of talents in an effective manner, since the nature and structure of talent careers can change quickly. To this end, in this paper, we propose a novel survival analysis approach to model the talent career paths, with a focus on two critical issues in talent management, namely turnover and career progression. Specifically, for modeling the talent turnover behaviors, we formulate the prediction of survival status at a sequence of time intervals as a multi-task learning problem by considering the prediction at each time interval as a task. Also, we impose the ranking constraints to model both censored and uncensored data, and capture the intrinsic properties exhibited in general lifetime modeling with non-recurrent and recurrent events. Similarly, for modeling the talent career progression, each task concerns the prediction of a relative occupational level at each time interval. The ranking constraints imposed on different occupational levels can help to reduce the prediction error. Finally, we evaluate our approach with several state-of-the-art baseline methods on real-world talent data. The experimental results clearly demonstrate the effectiveness of the proposed models for predicting the turnover and career progression of talents.

KEYWORDS

Multi-task Learning; Ranking; Career Path Modeling; Career Development; Survival Analysis

1 INTRODUCTION

Due to the intensive competition for talents globally, many companies strive not only to attract the right talents, but also provide sound career development guidance for retaining their skilled and competent employees. Thus, the study of career development has been gaining more importance in strategic human resource management [2, 11, 13, 18, 24, 27, 31].

Recently, the newly available big talent data provide unparalleled opportunities for business leaders to understand talent behaviors and gain tangible knowledge about career paths for assisting talents to plan their future career development. For example, the management team can provide more timely decisions to promote their employees whenever there is a need.

To this end, in this paper, we provide a data-driven solution for modeling talent career paths, and thus help talents better plan their career development. However, the complex and dynamic nature of talent career data, and the presence of large censored and recurrent events can impose significant challenges to model the talent career paths in an effective manner. First, the event of interests, e.g., turnover or promotion, may not be observed during the study period due to the limited observation time. In real-world scenarios, more than 65% career data exhibits this censoring phenomenon. This indicates that simple regression approaches [1, 20] may not be suitable for career path modeling. Second, with the high mobility of talents between organizations, an individual's career path may comprise a sequence of changing events, which can no longer be regarded as non-recurrent. Finally, the change of talent's career path is highly affected by a variety of dynamic factors, such as time-varying performance ratings and superior-and-subordinate relationships. Therefore, how to incorporate these dynamic factors into the modeling process presents another crucial challenge.

In the literature, some survival models can be adapted to career path modeling by predicting employee's survival time at an event of interest [3, 10]. For example, the Cox proportional hazards model [3] calculates the hazard in a multiplicative manner, which is associated with a baseline hazard function and the observed covariates. However, its assumption that the survival curves of all instances share a similar shape is not always true for all application scenarios [10]. Parametric survival model [6] is another popular technique in survival analysis, which assumes that the survival times of all instances share a particular distribution, such as log-logistic, log-normal, weibull, and exponential distribution. However, this high dependence on the choice of the distribution leads to a more critical weakness. Recently, a multi-task learning based method is developed to predict survival time by modeling the non-increasing list structure [10]. However, this approach only works on non-recurrent events, and thus is not suitable for modeling talent career paths.

To address the aforementioned issues, in this paper, we propose a novel survival analysis approach for modeling talent career paths, which is based on multi-task learning with ranking constraint formulation. In particular, we focus our study on two critical issues in talent management, namely turnover and career progression.

*Hui Xiong and Hengshu Zhu are corresponding authors.

KDD'17, August 13–17, 2017, Halifax, NS, Canada.

DOI: http://dx.doi.org/10.1145/3097983.3098107

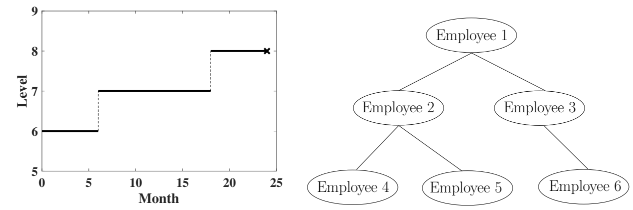
Specifically, for modeling the talent turnover behaviors, we formulate the prediction of survival status at a sequence of time intervals as a multi-task learning problem. The ranking constraint with different margins imposed on each pair of different survival status labels is used to distinguish censored and uncensored data, and simultaneously captures the unique characteristic in general survival analysis with both non-recurrent and recurrent events. Similarly, for career progression modeling, each task only focuses on the prediction of a relative level at a single time interval. Also, we impose ranking constraints between different levels, effectively ensuring that different levels are well separated, and thus, levels are estimated in a more reliable fashion. In addition, the observed time-varying factors are embedded into dynamic feature space and then incorporated into these two models. Finally, we evaluate our approach with several state-of-the-art baseline methods and different validation metrics on real-world talent data. The experimental results firmly demonstrate the effectiveness of the proposed models for predicting the turnover and career progression of employees. The major contributions of this paper can be summarized as follows.

- We study the problem of within-firm career path modeling for talents, with a focus on two issues in talent management, namely, turnover and career progression.
- We propose a novel survival analysis approach for modeling the career paths of employees, which is based on multi-task learning with ranking constraint formulation.
- We conduct extensive evaluations with real-world talent data to demonstrate the effectiveness of our survival models in terms of predicting turnover and career progression of employees.

2 DATA DESCRIPTION

We collected a set of anonymized employee career records from a high tech company across a timespan of 48 months from January 1st, 2011 to December 31st, 2014. Specifically, each employee has a date of joining the company[1], and a time duration (or tenure) until leaving the company or reaching the end of the timespan. The career progression of employees within a company is reflected by a series of observed occupational levels and their corresponding time durations. Each level is denoted by an integer and the change of level indicates whether an employee gets promoted or demoted. For example, Figure 1(a) shows an employee's career path within a company. Specifically, she stayed at level 6 for six months, and spent twelve months getting promotion from level 7 to level 8. Finally she left this company six months later after the last promotion. In addition, our dataset contains both static profile (e.g., gender and age) and other dynamic information of each employee. The dynamic information of each employee includes a numeric performance rating, ranging from one star to five stars, and a tree structure-like report chain, recording employee's superior and subordinate relationship, at different time stamps. Figure 1(b) is an example of report chain. Different from static profile, the dynamic information changes over time.

[1]To guarantee the effectiveness of our model, in this paper we only study the employees who joined the company on or after January 1st, 2011.



(a) An example of career path (b) An example of report chain

Figure 1: (a) An example of employee's career path within a company, where the end marker indicates that she leaves the company. (b) An example of report chain, where employee 1 is the superior of 2 and 3; Employee 4 and 5, and employee 6 are the subordinates of 2 and 3, respectively.

3 METHOD

In this section, we first state the problem of within-firm career path modeling for talents and two essential prediction tasks, namely, turnover and career progression. We then present our methods for each task in details. Finally, we discuss the feature space representation for our models and the optimization algorithm.

3.1 Problem Formulation

We focus on the analysis of employee's within-firm career path, which is mainly reflected by the status of employment and occupational level. Suppose $E_i^{(r)} = 1$ (or $E_i^{(p)} = 1$) indicates a turnover (or level changing) event of the i-th employee, and $E_i^{(r)} = 0$ (or $E_i^{(p)} = 0$) denotes staying at current company (or level). Given a time duration t from the date of joining the company, the joint probability of the i-th employee's employment status and level status can be denoted as,

$$P(E_i^{(r)}, E_i^{(p)}|t) = \begin{cases} P(E_i^{(r)} = 1|t) \\ P(E_i^{(r)} = 0|t) \times P(E_i^{(p)}|E_i^{(r)} = 0, t), \end{cases} \tag{1}$$

where the equation holds due to that an employee's level will never change after she leaves the company, i.e., $P(E_i^{(p)} = 1|E_i^{(r)} = 1, t) = 0$ and $P(E_i^{(p)} = 0|E_i^{(r)} = 1, t) = 1$. *The essential task for modeling an employee's career path within a company is to predict two critical events, i.e., when she will leave this company and when her occupational level will get changed.* In other words, given a time interval t, the task is to infer employee's turnover probability $P(E_i^{(r)} = 1|t)$, and level changing probability $P(E_i^{(p)} = 1|E_i^{(r)} = 0, t)$ if she is staying at this company[2]. In the following, we will introduce turnover behavior prediction in §3.2 and career progression modeling in §3.3 after giving all notations used in this paper.

Notations. Scalars, vectors and matrices are denoted by lower case letters, bold face lower case letters and bold face capital letters, respectively. Sets and lists are represented by calligraphic capital letters, where the i-th element of list $\mathcal{S}$ is denoted by $\mathcal{S}[i]$. $\mathbf{b}_i$ ($\mathbf{b}_{i,.}$) denotes the i-th column (row) of matrix $\mathbf{B}$. Euclidean and Frobenius norms are denoted by $||\cdot||$ and $||\cdot||_F$. $\mathbb{N}_n$ is defined as the set $\{1, \cdots, n\}$. A predicted value is denoted with a ˆ(hat) over it.

[2]Note that due to $P(E_i^{(r)} = 0|t) + P(E_i^{(r)} = 1|t) = 1$, once one of them is estimated, the other one will be known. The same goes for $P(E_i^{(p)}|E_i^{(r)} = 0, t)$.

3.2 Turnover Behavior Modeling

Boomerang employees who leave a company and return in the future have recently received attentions in human resource research [19]. Such turnover behavior may even occur more than once for an employee in a company. We propose to model employee's turnover behavior as a general survival problem which is applicable for both non-recurrent and recurrent events, and introduce a multi-task learning model with ranking based constraints to solve this problem.

Suppose we have n employees, each of who has either a career lifetime o_i for staying at a company or a censoring time c_i, not both. For each employee i in our data, we observe $t_i = \min(o_i, c_i)$, the minimum of the censoring and lifetime. A censoring indicator δ_i is introduced to describe whether observation is terminated by turnover event or censoring, i.e., $\delta_i = 1$ for an uncensored instance, and $\delta_i = 0$ for a censored instance. The observed time t_i is then defined as:

$$t_i = \begin{cases} o_i & \text{if } \delta_i = 1, \\ c_i & \text{if } \delta_i = 0. \end{cases} \tag{2}$$

We transform turnover behavior modeling into a multi-task learning problem by decomposing the classification problem into a series of related tasks by the reason of its popularity [10, 13]. Employee's observed time t_i is considered as countable time intervals with granularity as day, week or month. Let $m = \max\{t_i\}, \forall i = 1, \cdots, n$, be the maximum observation time of all employees. All employee's observation time is translated into a lifetime matrix $\mathbf{R} \in \mathbb{R}^{n \times m}$. Each element in the matrix is a binary value, where $r_{i,j} = 1$ if employee i stays at time interval j and $r_{i,j} = 0$ otherwise. Suppose we have a feature tensor $\mathbf{X}^{n \times p \times m}$, where each feature matrix $\mathbf{X}^j \in \mathbb{R}^{n \times p}$ is observed at the start of the j-th time interval (which will be introduced in details in Section 3.4). The target vector $\mathbf{r}_j$ is approximated using coefficients $\mathbf{B} \in \mathbb{R}^{p \times m}$ as:

$$\hat{\mathbf{r}}_j = \mathbf{X}^j \mathbf{b}_j. \tag{3}$$

As we do not know whether turnover will occur or not for censored instance, an indicator matrix $\mathbf{W}^r \in \mathbb{R}^{n \times m}$ is introduced to denote the missing values, where $w^r_{i,j} = 0$ if $r_{i,j}$ is unknown and $w^r_{i,j} = 1$ otherwise. The sum-of-squared error based loss function then can be achieved by only modeling those observed values as:

$$\min \ell^r(\mathbf{B}, \mathbf{X}) = \min \sum_{j=1}^{m} ||\mathbf{w}^r_j \odot (\mathbf{r}_j - \mathbf{X}^j \mathbf{b}_j)||^2 + \theta^r(\mathbf{B}, \mathbf{X}), \tag{4}$$

where $\odot$ denotes the element-wise multiplication. The left part of Figure 2 shows an example of transformation from original data into lifetime matrix $\mathbf{R}$ and indicator matrix $\mathbf{W}^r$, where the time unit of duration is month. $\theta^r(\mathbf{B}, \mathbf{X})$ incorporates regularization term that avoids overfitting and additional constraints. Different from [10], we capture the following two unique properties in general turnover modeling designed for $\theta^r(\mathbf{B}, \mathbf{X})$:

- *Ranking Relationship.* Three types of ranking relationships are observed in lifetime matrix as follows: (1) The values during career lifetime are larger than those after turnover event, (2) and are *probably* larger than those uncensored data; (3) All values are larger than or equal to zero.
- *Temporal Smoothness.* Due to the temporal consecutiveness, most adjacent tasks are similar.

There are two advantages to relax non-increasing property proposed in [10] into pairwise ranking. First, the ranking constraints can be used to handle both non-recurrent and recurrent events in lifetime modeling. Second, it places more strict constraints on modeling censored data and uncensored data. Specifically for censored instance i, unknown data is actually a mixture of surviving and turnover data. In other words, the observed values should be larger than or equal to unknown values, i.e., $\hat{r}_{i,j} \geqslant \hat{r}_{i,k}, \forall j \in \mathcal{M}^r_{i,1}, k \in \mathcal{M}^r_{i,-1}$, and $\mathcal{M}^r_*$ is defined as Eq.(5). For uncensored instance i, the values during career lifetime are strictly larger than those after turnover, i.e., $\hat{r}_{i,j} \geqslant \hat{r}_{i,k} + \xi^r, \forall j \in \mathcal{M}^r_{i,1}, k \in \mathcal{M}^r_{i,0}$. $\xi^r \in (0, 1]$ is a margin label used for uncensored data [14].

$$\mathcal{M}^r_{i,h} = \begin{cases} \{j | r_{i,j} = 1\} & \text{if } h = 1 \textit{ (before observed turnover)}, \\ \{j | r_{i,j} = 0\} & \text{if } h = 0 \textit{ (after observed turnover)}, \\ \{j | r_{i,j} = '?'\} & \text{if } h = -1 \textit{ (for unknown value)}. \end{cases} \tag{5}$$

Therefore, the ranking relationship can be formulated as follows:

$$\hat{r}_{i,j} \geq \hat{r}_{i,k} + \Delta^r_i, \quad \forall (i,j,k) \in \mathcal{U}^r, \tag{6}$$

$$\hat{r}_{i,j} \geq 0, \quad \forall i \in \mathbb{N}_n, j \in \mathbb{N}_m, \tag{7}$$

where $\mathcal{U}^r = \{i,j,k | \forall i \in \mathbb{N}_n, j \in \mathcal{M}^r_{i,1}, k \in \mathcal{M}^r_{i,\delta_i - 1}\}$ denotes all comparison tuples, and Δ^r_i is the margin label used to distinguish censored data and uncensored data as follows:

$$\Delta^r_i = \begin{cases} \xi^r & \text{if } \delta_i = 1, \\ 0 & \text{if } \delta_i = 0. \end{cases} \tag{8}$$

Consequently, $\theta^r(\mathbf{B}, \mathbf{X})$ is obtained by penalizing those violated constraints shown in Eq.(6) and Eq.(7) as,

$$\theta^r(\mathbf{B}, \mathbf{X}) = \lambda^r_1 \sum_{(i,j,k) \in \mathcal{U}^r} \left(\hat{r}_{i,k} + \Delta^r_i - \hat{r}_{i,j}\right)_+ + \lambda^r_2 \sum_{i,j} \left(-\hat{r}_{i,j}\right)_+ + \frac{\lambda^r_3}{2} ||\mathbf{B}||^2_F + \lambda^r_4 ||\mathbf{B}||_{2,1}, \tag{9}$$

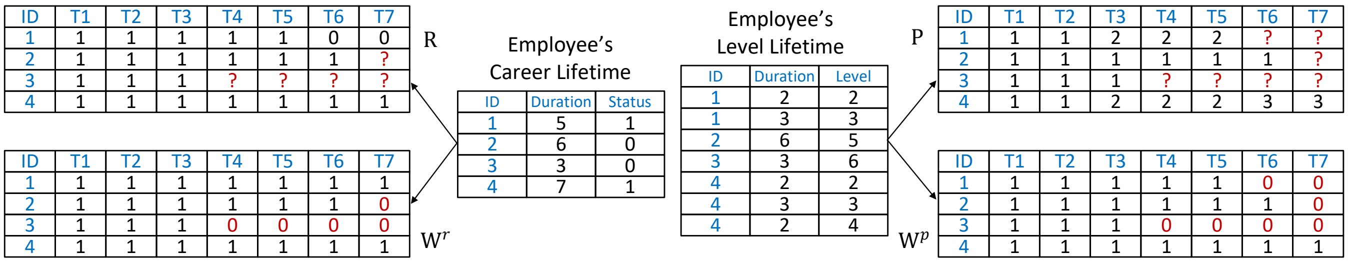
where $(x)_+ = \max\{x, 0\}$ is the plus function. λ^r_* are regularization parameters. Frobenius norm is used to avoid overfitting, and $\ell_{2,1}$-norm is used to capture temporal smoothness property by selecting a set of common features across all tasks.

Once the probability that employee i stays at the company at time interval j, i.e., $P(E^{(r)}_i = 0 | t_j) \propto \hat{r}_{i,j}$, is estimated, the corresponding turnover probability can be easily derived.

3.3 Career Progression Modeling

Given feature tensor $\mathbf{X}$ and historical career data, our goal is to predict the career progression measured by the occupational level. During the career lifetime, an employee's level possibly changes over time. Let $\mathcal{AS}_i$ be a list of levels that employee i has stayed at, where $\mathcal{AS}_i[x]$ is the x-th element in the list and $t_{i,l}$ denotes the duration that she stays at level $\mathcal{AS}_i[l]$ with $t_i = \sum_l t_{i,l}$. Therefore, the observed career progression of the i-th employee can be represented as follows:

$$\{< t_{i,1}, \mathcal{AS}_i[1] >, < t_{i,2}, \mathcal{AS}_i[2] >, \cdots, < t_{i,|\mathcal{AS}_i|}, \mathcal{AS}_i[|\mathcal{AS}_i|] >\},$$



R

ID	T1	T2	T3	T4	T5	T6	T7
1	1	1	1	1	1	0	0
2	1	1	1	1	1	1	?
3	1	1	1	?	?	?	?
4	1	1	1	1	1	1	1

W^r

ID	T1	T2	T3	T4	T5	T6	T7
1	1	1	1	1	1	1	1
2	1	1	1	1	1	1	0
3	1	1	1	0	0	0	0
4	1	1	1	1	1	1	1

Employee's Career Lifetime

ID	Duration	Status
1	5	1
2	6	0
3	3	0
4	7	1

Employee's Level Lifetime

ID	Duration	Level
1	2	2
1	3	3
2	6	5
3	3	6
4	2	2
4	3	3
4	2	4

P

ID	T1	T2	T3	T4	T5	T6	T7
1	1	1	2	2	2	?	?
2	1	1	1	1	1	1	?
3	1	1	1	?	?	?	?
4	1	1	2	2	2	3	3

W^p

ID	T1	T2	T3	T4	T5	T6	T7
1	1	1	1	1	1	0	0
2	1	1	1	1	1	1	0
3	1	1	1	0	0	0	0
4	1	1	1	1	1	1	1

Figure 2: An example of transformation from original career path data into multi-task learning settings.

where $|\mathcal{AS}_i|$ is the length of list $\mathcal{AS}_i$. In the example of Figure 2, the career progression of employee with $ID = 4$ can be described as $\{<2,2>, <3,3>, <2,4>\}$. As the observed time t_i is considered as countable time intervals (§3.2), the observed level of employee i at j-th time interval is denoted as $l_{i,j}$. The career progression is then converted to multi-task learning setting and rewritten as,

$$\{|\mathcal{M}^p_{i,h}|, \;\; \mathcal{AS}_i[h] \;\mid\; h \in \mathbb{R}_{|\mathcal{AS}_i|}\}, \tag{10}$$

where $\mathcal{M}^p_{i,h}$ denotes a set of continuous time intervals for level $\mathcal{AS}_i[h]$. In the example of Figure 2, for employee with $ID = 4$, we have $\mathcal{M}^p_{i,1} = \{T_1, T_2\}$, $\mathcal{M}^p_{i,2} = \{T_3, T_4, T_5\}$, and $\mathcal{M}^p_{i,3} = \{T_6, T_7\}$.

To model employee's career progression, we transform the original level lifetime data into an n-by-m relative level matrix $\mathbf{P}$. Each element $p_{i,j}$ of matrix $\mathbf{P}$ is the relative difference between level $l_{i,j}$ observed at j-th time interval and the first level observed when employee i enters the company,

$$p_{i,j} = l_{i,j} - l_{i,1} + 1, \tag{11}$$

where we add 1 as an offset. The purpose of adding this offset is to guarantee the relative levels in our data to be non-negative for better model learning. Please note that each employee's first observed level $l_{i,1}$ will still be incorporated to $\mathbf{X}$ as a static feature because it is very important for career progression. The reason that we choose to model the relative level is to decrease the matrix bias because most employees' levels get changed often less than five times during their career lifetimes. As the level changing only occurs during an employee's career lifetime, we only model the observed level data [3]. To do this, we use another indicator matrix $\mathbf{W}^p \in \mathbb{R}^{n\times m}$ to indicate the observed data, where $w^p_{i,j} = 1$ if employee i is staying at a level at the j-th time interval, and $w^p_{i,j} = 0$ otherwise. The right part of Figure 2 shows an example of generating a relative level matrix $\mathbf{P}$ and its corresponding indicator matrix $\mathbf{W}^p$. For example, an employee with $ID = 1$ has labels as "1" for two intervals (T_1, T_2) and as "2" for three intervals (T_3, T_4, T_5) in the corresponding row of relative level matrix $\mathbf{P}$, where the remaining unobserved values are labeled as "?". Accordingly, the values in the corresponding row of indicator matrix $\mathbf{W}^p$ are "1" during the observed career lifetime $(T_1 \sim T_5)$ and "0" in the remaining time intervals (T_6, T_7).

The relative levels for all employees at time interval j can be approximated using feature matrix $\mathbf{X}^j$ as,

$$\hat{\mathbf{p}}_j = \mathbf{X}^j \mathbf{a}_j, \tag{12}$$

where $\mathbf{a}_j$ is the j-th column of coefficient matrix $\mathbf{A} \in \mathbb{R}^{p\times m}$. Thus, the loss function in terms of squared error can be achieved as,

$$\min \ell^p(\mathbf{A}, \mathbf{X}) = \min \sum_{j=1}^{m} ||\mathbf{w}^p_j \odot (\mathbf{p}_j - \mathbf{X}^j \mathbf{a}_j)||^2 + \theta^p(\mathbf{A}, \mathbf{X}), \tag{13}$$

where $\theta^p(\mathbf{A}, \mathbf{X})$ is the regularizer incorporating the regularization term and additional constraints. The proposed two properties in Section 3.2, i.e., ranking relationship and temporal smoothness, also can be incorporated into career progression modeling. Distinct from turnover prediction, the ranking constraint designed for career progression is revised as follows [4]:

- Suppose $\mathcal{S}_i$ is a list of relative levels for employee i, i.e., $\mathcal{S}_i = \{p_{i,j} | j \in \mathcal{M}^p_{i,h}, h \in \mathbb{R}_{|\mathcal{AS}_i|}\}$. The pairwise ranking holds for each pair of adjacent levels, shown as,

$$\hat{p}_{i,j} \geq \hat{p}_{i,k} + \xi^p, \quad \forall (j,k) \in \mathcal{U}^p \tag{14}$$

  where $\xi^p \in [0,1]$, and $\mathcal{U}^p$ includes all possible adjacent comparison pairs and are defined as follows:

$$\mathcal{U}^p = \{i,j,k | i \in \mathbb{R}_n, j \in \mathcal{M}^p_{i,h_l}, k \in \mathcal{M}^p_{i,h_s}, \\ (h_s, h_l) = min\_max(h, h+1, \mathcal{S}_i), h \in \mathbb{R}_{|\mathcal{S}_i|-1}\}, \tag{15}$$

$$min\_max(h, h+1, \mathcal{S}_i) = \begin{cases} (h, h+1) & \text{if } \mathcal{S}_i[h] < \mathcal{S}_i[h+1], \\ (h+1, h) & \text{if } \mathcal{S}_i[h+1] < \mathcal{S}_i[h]. \end{cases}$$

- Non-negative property, i.e., $\hat{p}_{i,j} \geq 0, \forall i \in \mathbb{R}_n, j \in \mathbb{R}_m$. We have this constraint because the relative levels in our data are always larger than or equal to zero.

Finally $\theta^p(\mathbf{A}, \mathbf{X})$ in Eq.(13) is obtained by penalizing those violated ranking constraints and placing Frobenium norm and $\ell_{2,1}$-norm on $\mathbf{A}$ as follows:

$$\theta^p(\mathbf{A}, \mathbf{X}) = \lambda^p_1 \sum_{(i,j,k)\in\mathcal{U}^p} (\hat{p}_{i,k} + \xi^p - \hat{p}_{i,j})_+ + \\ \lambda^p_2 \sum_{i,j} (-\hat{p}_{i,j})_+ + \frac{\lambda^p_3}{2} ||\mathbf{A}||^2_F + \lambda^p_4 ||\mathbf{A}||_{2,1}, \tag{16}$$

Given the relative level $p_{i,s}$ observed at time interval s, once the relative level $\hat{p}_{i,j}$ of employee i at time interval j is estimated, the probability that her level will get changed can be calculated and proportional to the difference between values at time interval j and s, i.e., $P(E^{(p)}_i = 1 | E^{(r)}_i = 0, t_j) \propto |\hat{p}_{i,j} - p_{i,s}|$.

Discussion about Two Models. Here we would further clarify the relationship between turnover behavior modeling and career progression modeling. Indeed, the career status of an employee

[3]Please note that we do not fuse $\mathbf{R}$ and $\mathbf{P}$ together due to (1) We want to enable our method to handle demotion scenario. If fusing them together, it cannot distinguish zero from demotion event and turnover event. (2) They have different properties.

[4]If only promotion or demotion is considered, censored data can be modeled like §3.2.

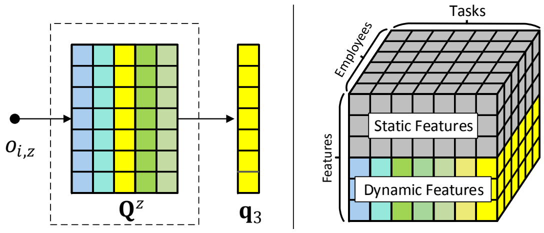


Figure 3: Left is mapping an observed feature value $o_{i,z}$ to a vector through a lookup table $\mathbf{Q}^z$, where $idx(o_{i,z}) = 3$. Right is the visualization of both static and dynamic features.

is determined by her employment status and occupation level status together. Thus, modeling the career path of an employee is essentially to model her turnover behavior and career progression, which are interconnected from the following two aspects. From problem formulation perspective, both can be treated as the general survival problem with recurrent events, aiming at estimating the career lifetime and level lifetime, respectively. From technical solution perspective, both are solved by a multi-task learning with ranking constraints, where different particular ranking constraints and prediction targets are involved.

3.4 Feature Space Representation

In this section, we will discuss the feature representation $\mathbf{X}$. As employee's career path is affected by the time-varying factors including performance rating and report chain, we incorporate them as dynamic features. Specifically, we transform employee's time-varying report chain into features by counting the number of changed superiors and subordinates. Figure 6 in Appendix provides some statistics about the impact of these dynamic features on career development. As we also have static information about employee, we totally have two types of features, i.e., static and dynamic features, as shown in Figure 3. Suppose we observe a set of features $\{o_{i,z}\}_{z=0}^{d}$ at the j-th time interval for employee i, including both static and dynamic features, where $o_{i,0} = 1$ is used for bias term. To address the low feature dimension issue of our data, we map original feature space into a higher dimensional space where data is more separable. Thus, each observed-and-quantized feature $o_{i,z}$ is embedded into a p_z-dimensional vector $\mathbf{q}_{idx(o_{i,z})}$ through the lookup matrix $\mathbf{Q}^z \in \mathbb{R}^{p_z \times range_z}$. The i-th row $\mathbf{x}^j_{i,\cdot}$ of feature matrix $\mathbf{X}^j \in \mathbb{R}^{n \times p}$ at j-th time interval is represented as,

$$\mathbf{x}^{\mathbf{j}}_{\mathbf{i},\cdot} = (\mathbf{q}^T_{idx(o_{i,0})}, \mathbf{q}^T_{idx(o_{i,1})}, \cdots, \mathbf{q}^T_{idx(o_{i,d})}), \quad (17)$$

where $\sum_z p_z = p$, $idx(\cdot)$ is the corresponding index. An example is provided in Figure 3. We will study two commonly used embedding representations in this paper [23]: one is using one-hot vector (i.e., a binary vector with all 0s and only a 1 for the corresponding index), denoted as sparse embedding; the other one is dense embedding by mapping each feature $o_{i,z}$ into a p_z-dimensional latent vector which then can be optimized under an appropriate loss function.

3.5 Optimization Algorithm

The models for turnover behavior and career progression prediction can be trained both separately and jointly. For the convenience of presentation, we provide their joint objective function based on Eq.(4) and Eq.(13) as follows:

$$\ell = \min \beta \ell^r(\mathbf{B}, \mathbf{X}) + (1-\beta)\ell^p(\mathbf{A}, \mathbf{X}) + \theta^x(\mathbf{X}), \quad (18)$$

where $\beta \in [0,1]$ is a controlling parameter, and $\theta^x(\mathbf{X}) = \frac{\lambda_x}{2} \sum_j ||\mathbf{X}^j||_F^2$ is the regularization term designed for feature learning with dense embedding. In Eq.(9) and Eq.(16), plus function $(x)_+$ is not twice differentiable and can be smoothly approximated using the integral to a smooth approximation of the sigmoid function [7, 9]:

$$(x)_+ \approx \sigma(x) = x + \frac{1}{\alpha}\log(1 + \exp(-\alpha x)), \quad (19)$$

where $\alpha > 0$ is a constant parameter. $\theta^r(\mathbf{B}, \mathbf{X})$ and $\theta^p(\mathbf{A}, \mathbf{X})$ then can be refined as follows:

$$\theta^r(\mathbf{B}, \mathbf{X}) = \lambda_1^r \sum_{(i,j,k)\in\mathcal{U}^r} \sigma(\hat{r}_{i,k} + \Delta_i^r - \hat{r}_{i,j}) + \lambda_2^r \sum_{i,j} \sigma(-\hat{r}_{i,j}) + \frac{\lambda_3^r}{2}||\mathbf{B}||_F^2 + \lambda_4^r||\mathbf{B}||_{2,1},$$

$$\theta^p(\mathbf{A}, \mathbf{X}) = \lambda_1^p \sum_{(i,j,k)\in\mathcal{U}^p} \sigma(\hat{p}_{i,k} + \xi^p - \hat{p}_{i,j}) + \lambda_2^p \sum_{i,j} \sigma(-\hat{p}_{i,j}) + \frac{\lambda_3^p}{2}||\mathbf{A}||_F^2 + \lambda_4^p||\mathbf{A}||_{2,1}.$$

Based on Eq.(18), the optimal solution of $\mathbf{B}$, $\mathbf{A}$ and $\mathbf{X}$ (for dense embedding) are given as follows:

$$\mathbf{B}^{t+1} = \underset{\mathbf{B}}{\operatorname{argmin}} \sum_{j=1}^{m} ||\mathbf{w}_j^r \odot (\mathbf{r}_j - \mathbf{X}^j \mathbf{b}_j)||^2 + \theta^r(\mathbf{B}, \mathbf{X}), \quad (20)$$

$$\mathbf{A}^{t+1} = \underset{\mathbf{A}}{\operatorname{argmin}} \sum_{j=1}^{m} ||\mathbf{w}_j^p \odot (\mathbf{p}_j - \mathbf{X}^j \mathbf{a}_j)||^2 + \theta^p(\mathbf{A}, \mathbf{X}), \quad (21)$$

$$\mathbf{X}^{t+1} = \underset{\mathbf{X}}{\operatorname{argmin}}\, \beta \ell^r(\mathbf{B}, \mathbf{X}) + (1-\beta)\ell^p(\mathbf{A}, \mathbf{X}) + \theta^x(\mathbf{X}). \quad (22)$$

The optimization problems associated with $\mathbf{B}$ and $\mathbf{A}$ can be regarded as the following $\ell_{2,1}$-norm regularization problem:

$$\min_{\mathbf{B}} \; loss(\mathbf{B}) + \lambda||\mathbf{B}||_{2,1}. \quad (23)$$

As $loss(\mathbf{B})$ and $loss(\mathbf{A})$ are convex, $\mathbf{B}$ and $\mathbf{A}$ then can be solved by Nesterov's method with efficient Euclidean projection [12]. The proposed optimization algorithm is summarized in Algorithm 1, where $\mathbf{X}$ will be optimized by gradient descent method if dense embedding is used and not optimized otherwise.

Algorithm 1: Optimization of the Proposed Method

Input: Job lifetime matrix $\mathbf{R}$, Relative level matrix $\mathbf{P}$, Indicator matrices $\mathbf{W}^r$, $\mathbf{W}^p$, and parameters λ_*^r, λ_*^p, λ_x, ξ^r, ξ^p, α, β
Output: $\mathbf{B}$, $\mathbf{A}$, $\mathbf{X}$

```
Randomly initialize B and A, X, t ← 1
while t ⩽ maxIter and not convergence do
    Compute B^{t+1} by solving Eq.(20);
    Compute A^{t+1} by solving Eq.(21);
    Update X^{t+1} by using the gradient of Eq.(22) with respect to X;
    t ← t + 1.
end
return B, A, X
```

Complexity Analysis. In each iteration, the worst case is to model all values in the matrix for the empirical loss, so time complexity is $O(nmp)$. For the pairwise ranking constraints, the running time is $O(m^2 p)$ for each instance. Sampling technique can be also used to reduce the comparison number [8, 17], such as sampling mk comparison pairs for each instance. It leads to the time complexity as $O(mkp)$, which approximates to $O(mp)$ because of the small value $k \ll m$. Suppose $\#iter$ is the number of iterations, the

Table 1: The statistics of datasets.

Dataset		#Employees	#Censored	#Tasks
Data 1	Training	7,343	5,323	47
	Testing	6,564	5,688	
Data 2	Training	6,986	4,561	47
	Testing	1,746	1,127	

Table 2: The description of features.

Type	Description
Static Feature	gender, age, year of start date, month of start date, initial level, and initial subordinate number
Dynamic Feature	performance rating, number of changed superiors, and number of changed subordinates

total running time is $O(nm^2p\#iter)$, and reduces to $O(nmp\#iter)$ when sampling method is applied to optimization.

4 EXPERIMENTS

In this section, we evaluate the proposed model with the baseline methods on the real-world dataset.

4.1 Experimental Setup

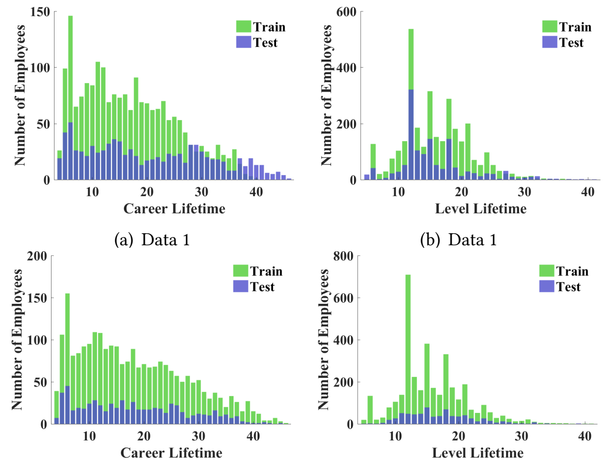
Datasets. In this paper, we use a real-world dataset collected from a high tech company (more details can be found in Section 2) to evaluate the performance of the proposed models. Before fitting our models with the data, we conduct the following preprocessing on our data: 1) removing employees who do not have profiles and performance ratings, 2) filtering out employees whose levels have changed more than five times, 3) removing those employees who have stayed at the company (or level) less than four (or three) months. After preprocessing, we totally have 8,732 employees. Then we split the data into two parts (i.e., training and testing sets) in two different ways. The first way is to split data in chronological order and denote as *Data 1*, where the earlier data ranging from January 2011 to May 2014 is used as training and the remaining half-year data is used as testing. The second way follows standard survival analysis evaluation method, i.e., splitting the data randomly, which is denoted as *Data 2*, where we randomly select 80% employees as training and use the rest as testing. The statistics of two datasets are shown in Table 1, and their histograms about the durations of staying at a company and a level are shown in Figure 4. We divide the data with month-based granularity, and regard each month as a learning task. In addition, we totally have seven static features and three dynamic features, which are summarized in Table 2.

Parameters. In the experiments, the parameters λ_1^r and λ_1^p are set as $2/\#Tasks$. Parameters λ_2^*, λ_3^* λ_4^*, λ_x, ξ^r, ξ^p, and α are set as 0.1, 0.01, 0.01, 0.01, 0.4, 0.1, and 5, respectively.

4.2 Evaluation Metrics

Due to the presence of censored data, we adopt widely-used evaluation metric, i.e., the concordance index (C-index), to measure the performance of prediction models in survival analysis [5, 10]. Suppose there are N_{test} employees in testing data, r_i is the groundtruth career (or level) lifetime time of the i-th employee, and $\hat{r}_i$ is the corresponding predicted lifetime. C-index is defined as,

$$C-index = \frac{1}{|\Omega|} \sum_{i \in \mathbb{R}_{N_{test}} \& \delta_i = 1} \sum_{r_j > r_i} I[\hat{r}_j > \hat{r}_i], \tag{24}$$



(a) Data 1 (b) Data 1 (c) Data 2 (d) Data 2

Figure 4: Histograms of the career lifetime (left) and level lifetime (right) for different training and testing data.

where $I[x]$ is an indicator function that equals to 1 if x is true, and equals to 0 otherwise. $|\Omega|$ is the total comparison number, i.e., $\Omega = \{i, j | i \in \mathbb{R}_{N_{test}}, \delta_i = 1, r_j > r_i\}$.

In addition, we also evaluate model's performance in terms of weighted average AUC (WAUC), indicating whether an employee survives at a company (or a level) at each time interval [10]. WAUC is defined as follows,

$$WAUC = \frac{\sum_i^k AUC^{(i)} n_{\bar{c}}^{(i)}}{\sum_{i=1}^k n_{\bar{c}}^{(i)}}, \tag{25}$$

where k is the total task number, $AUC^{(i)}$ is the AUC value of the i-th task, and $n_{\bar{c}}^{(i)}$ is the number of instances in the testing data which have an observed survival status in the i-th time interval.

Table 3: Settings of the proposed models and various baseline methods. 'Y' indicates the presence of censorship, static features ('StatFea') or dynamic features ('DynFea') in the model. 'N' indicates the absence.

Method		Censoring	StatFea	DynFea
COX	COX	Y	Y	N
Parametric	Log-Logistic	Y	Y	N
	Log-gaussian	Y	Y	N
	Weibull	Y	Y	N
	Exponential	Y	Y	N
Multi-task	M-LASSO	N	Y	N
	M-L2,1	N	Y	N
	MTLSAV2	Y	Y	N
	MTLSA	Y	Y	N
	M-LASSO+DF	N	Y	Y
	M-L2,1+DF	N	Y	Y
	MTLSAV2+DF	Y	Y	Y
	MTLSA+DF	Y	Y	Y
	CDT+SE	Y	Y	Y
	CDT+DE	Y	Y	Y

4.3 Baseline Methods

To comprehensively demonstrate the effectiveness of our model, we compare it with the following models:

- **COX** [3], which models the hazard function in *exp* proportional fashion and relates to a baseline hazard function;
- **Log-Logistic**, **Log-gaussian**,**Weibull**, and **Exponential** [6]: which are popular parametric survival models with logistic, gaussian, weibull, and exponential distributions, respectively;

Table 4: Performance comparison for turnover prediction based on two datasets.

Method	Data 1		Data 2	
	C-index	WAUC	C-index	WAUC
COX	0.81067	0.50032	0.09807	0.49929
Log-logistic	0.31255	0.44432	0.50886	0.51108
Log-gaussian	0.68455	0.71556	0.51044	0.51822
Weibull	0.31567	0.28023	0.55407	0.70118
Exponential	0.51138	0.23764	0.49783	0.39084
M-LASSO	0.82848	0.53221	0.51157	0.64466
M-L2,1	0.82757	0.62762	0.57956	0.73304
MTLSA.V2	0.82449	0.59780	0.58465	0.74269
MTLSA	0.82921	0.62838	0.55323	0.74289
M-LASSO+DF	0.83553	0.54542	0.55477	0.65854
M-L2,1+DF	0.83038	0.65879	0.60272	0.74581
MTLSAV2+DF	0.83794	0.60208	0.59443	0.72357
MTLSA+DF	0.83941	0.68641	0.61111	0.76684
CDT+SE	0.87942	0.79187	0.63170	0.78157
CDT+DE	0.88679	0.82214	0.63891	0.78241

- **M-LASSO** [21], which is a standard multi-task learning model with LASSO penalty, and only models uncensored data.
- **M-L2,1** [12], which is a standard multi-task learning model with $\ell_{2,1}$-norm penalty, and only models uncensored data.
- **MTLSAV2** [10], which uses $\ell_{2,1}$-norm multi-task learning to model both censored and uncensored data.
- **MTLSA** [10], which is designed for survival analysis and places non-increasing constraint on MTLSAV2 model.

All above baseline methods use static features to train model. Also, M-LASSO, M-L2,1, MTLSAV2, and MTLSA are extended to incorporate dynamic features as well and denoted as M-LASSO+DF, M-L2,1+DF, MTLSAV2+DF, and MTLSA+DF, respectively. The features for all baseline methods are represented by sparse embedding. In the experiments, our methods with two feature representations are denoted as **CDT+SE** for sparse embedding, and **CDT+DE** for dense embedding with hidden dimension size as 10. The proposed methods and various baseline methods are summarized in Table 3.

4.4 Performance Comparison

We first evaluate the proposed methods for turnover behavior modeling and career progression modeling. Then, we discuss the influence of different dynamic features on the above tasks.

4.4.1 ***Performance of Turnover Behavior Modeling.*** The performance of our models and various baseline methods for predicting employee's turnover in terms of C-index and WAUC over two different datasets are reported in Table 4. As multi-task learning based methods can predict whether an employee will survive at each time interval, they can be used to predict the lifetime and evaluated from C-index. From the results, we conclude the following observations. First, our models CDT+SE and CDT+DE outperform all baseline methods. For example, our method increases 5.6% and 14.9% over the best of all baseline methods in terms of C-index and WAUC on data 1, respectively. The significant improvement is due to the ranking constraints, which help more accurately optimize the sum-of-squared error loss function. The superior performance of our method over MTLSA+DF also demonstrates that the relaxed ranking constraint is better than directly modeling the non-increasing property for survival analysis based on multi-task learning formulation. We also observe that CDT+DE is slightly better than CDT+SE. Although optimizing **X** brings a little improvement, we observe that it easily overfitts and its result is not very stable in the experiments because a large number of data are used to fit only a limited number of features. Second, the performance of methods with dynamic features are much better than the ones without dynamic features. Third, modeling censored data results in more accurate turnover behavior prediction, which reflects on the performance comparison of M-LASSO*, M-L2,1* and other multi-task learning based methods. Fourth, multi-task learning based methods perform slightly better than most parametric survival models. Parametric survival models are quite data-sensitive, where the distribution assumption is specifically designed based on the empirical data. Different from them, multi-task models can convert the global censored classification problem to a series of local classification problem, and thus achieve a comparatively better result. Fifth, the results on data 1 are totally better than the ones on data 2. The possible reason is that most employees in data 1 have historical observations in the training set, which helps generate more accurate prediction on their career path observed in the testing set.

Table 5: Performance comparison for career progression prediction based on two datasets.

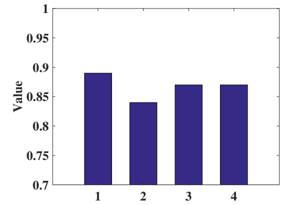
Method	Data 1		Data 2	
	C-index	WAUC	C-index	WAUC
COX	0.55693	0.50500	0.00479	0.49908
Log-logistic	0.65526	0.64517	0.47275	0.54691
Log-gaussian	0.68291	0.72117	0.50275	0.53442
Weibull	0.69249	0.78945	0.4882	0.53868
Exponential	0.61928	0.38461	0.53409	0.53934
M-LASSO	0.7456	0.6727	0.54594	0.07617
M-L2,1	0.77251	0.73323	0.43026	0.48919
MTLSAV2	0.79613	0.76412	0.51892	0.62052
MTLSA	0.58779	0.55589	0.41912	0.69312
M-LASSO+DF	0.74634	0.67259	0.07617	0.53294
M-L2,1+DF	0.79321	0.77225	0.48821	0.49096
MTLSAV2+DF	0.83375	0.84619	0.55520	0.62052
MTLSA+DF	0.58780	0.55314	0.41912	0.69269
CDT+SE	0.88251	0.85912	0.68005	0.78803
CDT+DE	0.91763	0.89012	0.68299	0.81185

4.4.2 ***Performance of Career Progression Modeling.*** In this section, we compare the model's performance on career progression prediction with focus on predicting when employee's level will get changed next time. As employee's level changes over time, conventional survival models (COX, Log-logistic, Log-gaussian, Weibull, Exponential, MTLSA, and MTLSA+DF) cannot be directly used to predict career progression. To do this, we utilize these methods to model each level changing event for each employee, and then use the trained model to predict testing data. For example, in employee's level lifetime records of Figure 2, each level changing record is regarded as an instance and the corresponding level will be used as one static feature. Similar to our model, other multi-task learning based baseline methods (M-LASSO*, M-L2,1*, and MTLSAV2*) can be used to fit the relative level lifetime matrix **P**.

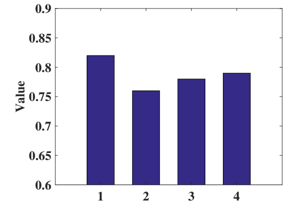
The performance of the proposed models and baseline methods for modeling employee's level changing is shown in Table 5.

Based on the results, we can obtain the following observations. (1) Our models outperform all baseline models. This result clearly validates the effectiveness of our models in two aspects. First, it demonstrates the usefulness of multi-task learning formulation by predicting a relative level at each time interval, which is further justified by the better results of M-LASSO*, M-L2,1*, and MTLSAV2* over other conventional survival baseline methods. Second, the ranking constraint does help to improve the performance, which reflects on the superior result of our model over MTLSAV2+DF method. (2) MTLSA* methods perform even worse than other multi-task learning based models for the sake of its limitation of non-increasing property which is only suitable for non-recurrent events.

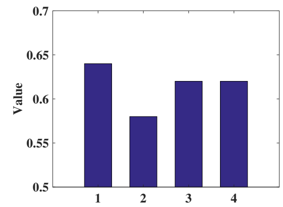
4.4.3 ***Study of Dynamic Features.*** In this section, we study the influence of different dynamic features on employee's career path modeling. We first train CDT+DE model with all dynamic features, and then train the model by removing each of them and check the performance change in terms of C-index and WAUC with both datasets. The performance of the proposed method with different dynamic features is shown in Figure 5. From the results, we can observe that model's performance decreases dramatically without the feature of performance rating. It indicates that an employee's performance rating plays an important role in her career path, and to some extend is able to affect the career status. Compared with the impact of an employee's performance rating, the influence of report chain on her career path is a little weaker. There are two possible reasons for this. First, in our data, there are only a small number of employees whose superiors will change more than one times, and the change of superiors sometimes is caused by the decision of company's strategy rather than the personal reason of employees themselves. Thus, the superior changing has relatively small impact on employee's career path. Second, although the change of subordinates has significant influence on employee's career path as reflected on the observation of Figure 6 in Appendix, most employees do not have subordinates. Therefore, it is reasonable that modeling this type of feature does not bring very significant improvement.



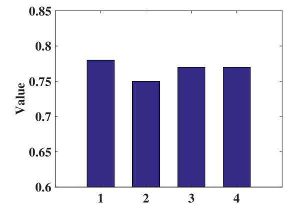
(a) C-index on Data 1 for Turnover



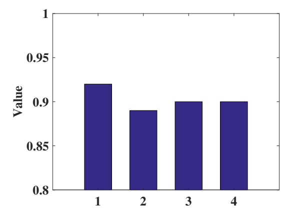
(b) WAUC on Data 1 for Turnover



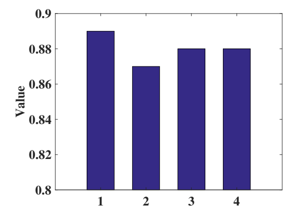
(c) C-index on Data 2 for Turnover



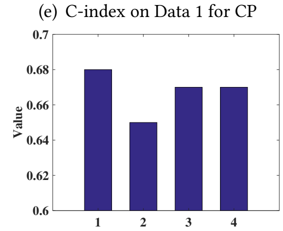
(d) WAUC on Data 2 for Turnover



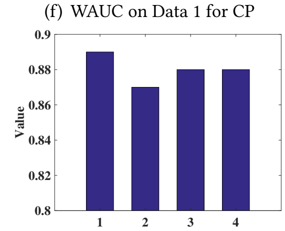
(e) C-index on Data 1 for CP



(f) WAUC on Data 1 for CP



(g) C-index on Data 2 for CP



(h) WAUC on Data 2 for CP

Figure 5: The influence of dynamic features on turnover and career progression predictions, where the results are obtained using CDT+DE model. '1' indicates all three types of dynamic features. '2', '3', and '4' indicate all dynamic features other than performance rating, the changed number of superiors, and the changed number of subordinates, respectively. 'CP' represents career progression.

5 RELATED WORK

The related works about this paper can be grouped into two categories. The first category is about multi-task learning (MTL), which learns multiple related tasks simultaneously to improve generalization performance [29, 32]. Early research work proposes LASSO regression method to shrink some coefficients and require others to be zero in order to retain some good features [16, 21]. MTL with LASSO can be used to select some important features in each single task, but it ignores the relatedness of different tasks. To capture the task relatedness, group lasso regularization based on $\ell_{2,1}$-norm penalty for feature selection is used to select features across all data points with joint sparsity [12, 15, 30]. Totally, different assumptions about how tasks are related lead to different regularization terms.

The second category throws light on survival analysis [22, 25, 26, 28]. Cox proportional hazards model [3, 4] is a popular technique in survival analysis due to its simplicity and assumption-free about the survival time. It determines the hazard in a multiplicative manner associated with baseline hazard and some observed covariates. In addition to cox based survival model, another research line for survival modeling is the parametric survival model. It assumes that the survival times of all instances in the dataset follow a particular distribution, such as log-logistic, log-normal, weibull, and exponential distributions [6]. There is a hypothesis for above models, i.e., the survival curve of all instances shares a similar shape. To overcome this limitation, MTL based survival model [10] is developed to convert original non-recurrent survival problem into a series of related binary classification problems, where the non-negative and non-increasing list constraints are imposed on the modeling process at the same time.

In this paper, we propose a novel survival analysis approach for modeling the career paths of employees, with a focus on two critical issues, i.e., turnover and career progression. Different from the aforementioned methods, we formulate survival analysis into a MTL problem with ranking constraints, which is suitable for general survival problem with non-recurrent and recurrent events. Specifically, based on the proposed framework, each task in turnover behavior modeling concerns the prediction of employment status at each time interval, and each task in career progression prediction focuses on the prediction of a relative level at each time interval.

6 CONCLUSION

In this paper, we proposed a novel survival analysis approach for modeling the career paths of employees, which is based on multi-task learning with ranking constraint formulation. With different ranking constraints and prediction targets, it is capable of modeling two critical issues in talent management, i.e., turnover and career progression. Specifically, for modeling the turnover behaviors of employees, we formulated the prediction of survival status at a sequence of time intervals as a multi-task learning problem by considering the prediction at each time interval as a task. To model both censored and uncensored data, and capture the intrinsic properties exhibited in general lifetime modeling with non-recurrent and recurrent events, we imposed the ranking constraints on each pair of different survival status labels. For modeling career progression, we formulated the prediction of the relative occupational level at each time interval as a task, where the ranking constraints imposed on different levels are used to improve the performance accuracy. Finally, extensive experimental results on real-world talent data clearly validated the effectiveness of our models compared with several state-of-the-art baseline methods in terms of various evaluation metrics.

ACKNOWLEDGMENTS

This work is partially supported by the NIH (1R21AA023975-01) and NSFC (61602234, 61572032, 91646204, 61502077).

REFERENCES

[1] Jonathan Buckley and Ian James. 1979. Linear Regression with Censored Data. *Biometrika* 66, 3 (1979), 429–436.
[2] Kenneth Burdett, Nicholas M. Kiefer, and Sunil Sharma. 1985. Layoffs and duration dependence in a model of turnover. *Journal of Econometrics* (1985).
[3] D. R. Cox. 1972. Regression Models and Life-Tables. *Royal Statistical Society* 34, 2 (1972), 187–220.
[4] Komal Kapoor, Mingxuan Sun, Jaideep Srivastava, and Tao Ye. 2014. A Hazard Based Approach to User Return Time Prediction. In *SIGKDD*. 1719–1728.
[5] Faisal M. Khan and Valentina Bayer Zubek. 2008. Support Vector Regression for Censored Data (SVRc): A Novel Tool for Survival Analysis. In *ICDM*.
[6] Elisa T. Lee and John Wenyu Wang. 2003. *Statistical Methods for Survival Data Analysis*. Wiley.com.
[7] YUH-JYE Lee and O. L. Mangasarian. 2001. SSVM: A Smooth Support Vector Machine for Classification. *Computational Optimization and Applications* (2001).
[8] Huayu Li, Yong Ge, Richang Hong, and Hengshu Zhu. 2016. Point-of-Interest Recommendations: Learning Potential Check-ins from Friends. In *KDD*. 975–984.
[9] Huayu Li, Richang Hong, Defu Lian, Zhiang Wu, Meng Wang, and Yong Ge. 2016. A Relaxed Ranking-Based Factor Model for Recommender System from Implicit Feedback. In *IJCAI*. 1683–1689.
[10] Yan Li, Jie Wang, Jieping Ye, and Chandan K. Reddy. 2016. A Multi-Task Learning Formulation for Survival Analysis. In *SIGKDD*. 1715–1724.
[11] Hao Lin, Hengshu Zhu, Yuan Zuo, Chen Zhu, Hui Xiong, and Junjie Wu. 2017. Collaborative Company Profiling: Insights from an Employee's Perspective. In *AAAI*. 1417–1423.
[12] Jun Liu, Shuiwang Ji, and Jieping Ye. 2012. Multi-Task Feature Learning Via Efficient l2,1-Norm Minimization. *CoRR* abs/1205.2631 (2012).
[13] Ye Liu, Luming Zhang, Liqiang Nie, Yan Yan, and David S. Rosenblum. 2016. Fortune Teller: Predicting Your Career Path. In *AAAI*. 201–207.
[14] Taesup Moon, Alex Smola, Yi Chang, and Zhaohui Zheng. 2010. IntervalRank: Isotonic Regression with Listwise and Pairwise Constraints. In *WSDM*. 151–160.
[15] Feiping Nie, Heng Huang, Xiao Cai, and Chris Ding. 2010. Efficient and Robust Feature Selection via Joint ℓ2,1-norms Minimization. In *NIPS*. 1813–1821.
[16] Guillaume Obozinski, Ben Taskar, and Michael Jordan. 2006. Multi-task feature selection. In *Technical Report*.
[17] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. 2009. BPR: Bayesian personalized ranking from implicit feedback. In *UAI*. 452–461.
[18] James E. Rosenbaum. 1979. Tournament Mobility: Career Patterns in a Corporation. *Administrative Science Quarterly* 24, 2 (1979), 220–241.
[19] Abbie J. Shipp, Stacie Furst-Holloway, T. Brad Harris, and Benson Rosen. 2014. Gone today but here tomorrow: Extending the unfolding model of turnover to consider boomerang employees. *PERSONNEL PSYCHOLOGY* 67 (2014), 421–462.
[20] Pannagadatta K. Shivaswamy, Wei Chu, and Martin Jansche. 2007. A Support Vector Approach to Censored Targets. In *ICDM*. 655–660.
[21] Robert Tibshirani. 1994. Regression Shrinkage and Selection Via the Lasso. *Journal of the Royal Statistical Society* 58 (1994), 267–288.
[22] William Trouleau, Azin Ashkan, Weicong Ding, and Brian Eriksson. 2016. Just One More: Modeling Binge Watching Behavior. In *SIGKDD*. 1215–1224.
[23] Joseph Turian, Lev Ratinov, and Yoshua Bengio. 2010. Word Representations: A Simple and General Method for Semi-supervised Learning. In *ACL*. 384–394.
[24] Kush R. Varshney, Vijil Chenthamarakshan, Scott W. Fancher, Jun Wang, Dongping Fang, and Aleksandra Mojsilović. 2014. Predicting Employee Expertise for Talent Management in the Enterprise. In *SIGKDD*. 1729–1738.
[25] Jian Wang, Yi Zhang, Christian Posse, and Anmol Bhasin. 2013. Is It Time for a Career Switch?. In *WSDM*. 1377–1388.
[26] Shu-Chen Wu. 1982. A semi-Markov model for survival data with covariates. *Mathematical Biosciences* 60, 2 (1982), 197–206.
[27] Huang Xu, Zhiwen Yu, Jingyuan Yang, Hui Xiong, and Zhu Hengshu. 2016. Talent Circle Detection in Job Transition Networks. In *SIGKDD*. 655–664.
[28] Jianfei Zhang, Lifei Chen, Alain Vanasse, Josiane Courteau, and Shengrui Wang. 2016. Survival Prediction by an Integrated Learning Criterion on Intermittently Varying Healthcare Data. In *AAAI*. 72–78.
[29] Liang Zhao, Qian Sun, Jieping Ye, Feng Chen, Chang-Tien Lu, and Naren Ramakrishnan. 2015. Multi-Task Learning for Spatio-Temporal Event Forecasting. In *SIGKDD*. 1503–1512.
[30] Jiayu Zhou, Lei Yuan, Jun Liu, and Jieping Ye. 2011. A Multi-task Learning Formulation for Predicting Disease Progression. In *SIGKDD*. 814–822.
[31] Chen Zhu, Hengshu Zhu, Hui Xiong, Pengliang Ding, and Fang Xie. 2016. Recruitment Market Trend Analysis with Sequential Latent Variable Models. In *SIGKDD*. 383–392.
[32] Hengshu Zhu, Hui Xiong, Fangshuang Tang, Qi Liu, Yong Ge, Enhong Chen, and Yanjie Fu. 2016. Days on Market: Measuring Liquidity in Real Estate Markets. In *SIGKDD*. 393–402.

A IMPACT OF DYNAMIC FEATURES ON CAREER PATHS OF EMPLOYEES

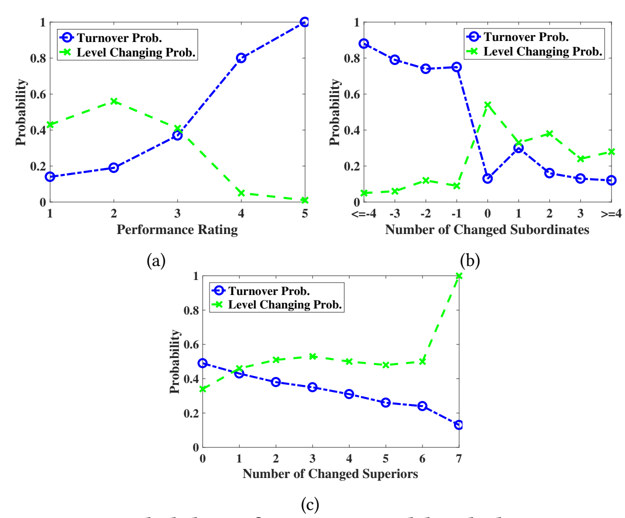


Figure 6: Probability of turnover and level changing as a function of performance rating (a), number of changed subordinates (b) and number of changed superiors (c).