Career Path Clustering via Sequential Job Embedding and Mixture Markov Models

Hao Zhong
ESCP Business School, hzhong@escp.eu

Chuanren Liu
University of Tennessee, cliu89@utk.edu

Career Path Clustering via Sequential Job Embedding and Mixture Markov Models

Completed Research Paper

Hao Zhong
Information & Operations Management
ESCP Business School
Paris, France
hzhong@escp.eu

Chuanren Liu
Business Analytics and Statistics
The University of Tennessee, Knoxville
Knoxville, TN, United States
cliu89@utk.edu

Abstract

Extracting typical career paths from large-scale and unstructured talent profiles has recently attracted increasing research attention. However, various challenges arise in effectively analyzing self-reported career records. Inspired by recent advances in neural networks and embedding models, we develop a novel career path clustering approach with two major components. First, we formulate an embedded Markov framework to learn job embeddings from longitudinal career records and further use them to compute dynamic embeddings of career paths. Second, to cope with heterogeneous career path clusters, we estimate a mixture of Markov models to optimize cluster-wise job embeddings with a prior embedded space shared by multiple clusters. We conduct extensive experiments with our framework to investigate its algorithmic performance and extract meaningful patterns of career paths in the information technology (IT) industry. The results show that our approach can naturally discover distinct career path clusters and reveal valuable insights.

Keywords: career path clustering, sequential job embedding, mixture Markov models.

Introduction

The wide adoption of employment-oriented online services such as LinkedIn has enabled novel research investigations on professional career patterns with rich career data. The research on professional careers is particularly important for industries where talents change jobs frequently. For instance, an information technology (IT) professional usually changes job position every one or two years and quickly accumulates a career path, i.e., a sequence of job positions. There is substantial interest in extracting typical sequential patterns from a large number of career path records. Such sequential patterns can be used for job recommendation, career planning, and talent management (Joseph et al. 2012).

The problem of career path characterization has been previously studied in the literature. Early studies adopted the optimal matching analysis (OMA) (Sankoff et al. 2000), a sequence analysis method that could be applied for career path clustering. The OMA-based methods rely on two steps. In the first step, the similarity of a sequence pair is assessed by edit distance, i.e., the cost of editing one sequence into another; In the second step, a clustering algorithm is applied to the similarity/distance matrix to construct sequence clusters. For instance, Biemann and Wolf (2009) applied OMA to the field of top management research and developed six career patterns of top management team members based on their international experience, organizational tenure, and professional experience. Further, Biemann et al. (2012) examined which sociodemographic predictors affect whether or not individuals follow particular career path patterns. In a focused review on OMA, Dlouhy and Biemann (2015) pointed out that the study on sequences of occupational states from the longitudinal perspective is important in career research.

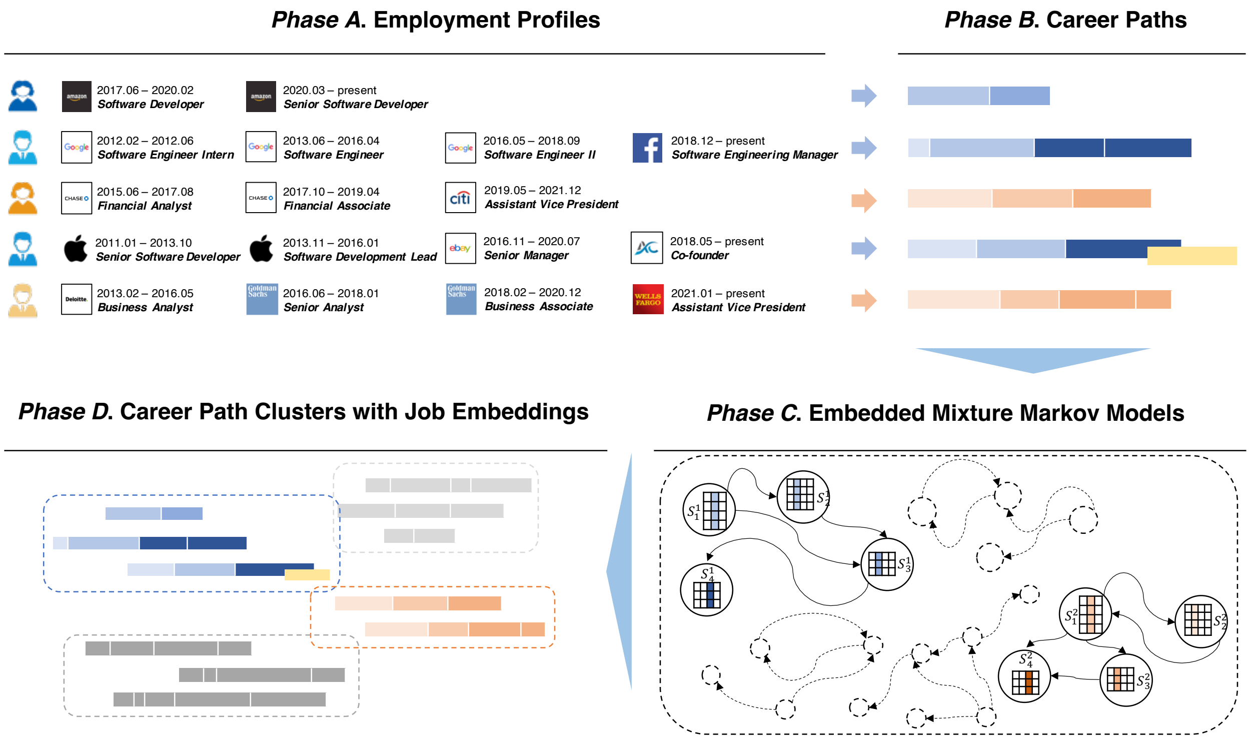


Figure 1. A conceptual diagram of our career path clustering approach

The emergence of online talent networking platforms, e.g., LinkedIn, has again motivated recent studies with large-scale and unstructured career path records (Lappas 2020). However, conventional OMA-based approaches are not applicable to analyze such self-reported data due to several challenges. First, OMA assumes that the elements (e.g., job titles) are equally distinguishable in the sequences. In reality, some job titles are much more similar to each other than other job titles in our data. Conventional OMA doesn't incorporate weighted distances in estimating the cost to edit one sequence into another. Second, the weighted distances between sequence elements are not available to extend OMA or apply other sequence analysis techniques. This problem is particularly challenging in analyzing raw career records where self-reported job titles on online talent networking platforms are much more complex than those in the aforementioned focused studies. For instance, our data includes over 48K distinct titles in the IT industry. Third, as discussed by Lappas (2020), clusters of career paths are naturally fuzzy and cannot be meaningfully simplified into prototype paths. Therefore, clustering approaches based on the aggregated centroids and producing hard clustering labels may ignore fine-grained information in the large-scale career path records. Essentially, we need a quantitative measurement that can objectively estimate distances between job titles and support clustering analysis with a large amount of job sequences.

To address these challenges, in this paper, we develop an approach based on sequential job embedding and mixture Markov models. There are two unique differences in our approach compared with the conventional OMA-based approaches. First, we use optimization and statistical approaches to learn latent embeddings of job titles. The latent embeddings can capture both *functionality* and *responsibility* relationship (Liu et al. 2019) among raw job titles by mapping each term in job titles to a Euclidean space where career paths are modeled with a Markov assumption. Second, we replace conventional cluster centroids with a mixture of probabilistic models that can effectively cope with the heterogeneous nature of real-world career paths. Overall, with a mixture of Markov models, our approach can simultaneously address the two daunting challenges in analyzing large-scale career path records.

We illustrate the procedure of our proposed method for career path characterization and clustering in Figure 1. In essence, we aim to discover coherent career path clusters, where 1) the clusters are distinctive;

2) each cluster is followed by many talents along with their career paths; and 3) all clusters taken together can characterize most talents' career paths. We start with talents' employment profiles (*Phase A*). A typical employment profile consists of the talent's past and present job records, each of which usually includes the employer's name, a job title, and the corresponding employment period. Given each profile, we chain the job records chronologically to form a *career path* (*Phase B*). Note that the tenure length of each employment record is reflected in our coding scheme owing to its revelation of job importance in the regarded career. In *Phase C*, we employ our proposed Embedded Mixture Markov Models (EMMMs) to simultaneously learn cluster-wise job embeddings and construct career path clusters. The likelihood of each career path's association with a cluster is computed through the proximity measurement of sequentially-adjacent job titles in the embedded space. The optimal assignments of career paths are then earned by iteratively picking the cluster with the highest estimated likelihood. Meanwhile, cluster-wise job title/term embeddings are tuned to reflect career clusters' idiosyncratic characteristics. Unlike other categorical sequence clustering methods (e.g., the conventional OMA-based approaches), our embedding-based model can cope with career paths with sequentially-overlapping job positions (e.g., the fourth talent's career in our example), which appears common among the individuals who take concurrent professional roles in their careers. When the model training process converges or reaches an iteration limit, the resultant cluster-wise likelihoods of career paths explicitly quantify their soft bonding with different clusters (*Phase D*). The job embeddings learned from our model reveal implicit connections among different job titles within each cluster.

The rest of our paper is organized as follows. Section *Literature Review* reviews the related literature, focusing on career pattern mining, career path clustering, and general categorical sequence clustering methods. Section *Data Description and Research Challenges* introduces our data sources and summarizes their key statistics. Section *Methodology* elaborates on our method for career path clustering based on sequential job embedding and mixture Markov models. Section *Experimental Analyses and Results* applies our method to investigate typical career path patterns in the IT industry and demonstrates the efficacy of our method. Finally, Section *Conclusion* summarizes our paper, highlighting its limitations and areas for future research.

Literature Review

Our work primarily relates to the broad literature of career path studies. Earlier work in this realm utilized surveys, interviews, and/or narratives to understand careers. Reich and Kaarst-Brown (1999) conducted a case study with 51 former IT professionals using interviews and surveys to understand their career transitions from IT to non-IT business unit roles. Later, Cohen (2006) reviewed and accredited the storytelling or narrative methodologies used in career path studies and, using the narratives collected from a group of research scientists, identified four typical career discourses of scientists. However, the early methods rely heavily on subjective self-expressions of identity and retrospective sense-making and, unfortunately, have not been widely adopted in recent career path studies (Vinkenburg and Weber 2012). Instead, researchers shifted their focus to more analytics-driven strategies. For instance, by targeting similar study subjects as (Cohen 2006), Agarwal and Ohyama (2013) used multiple regression analysis approaches to understand the career trajectories of scientists (basic vs. applied science and industry vs. academic paths). They aimed to understand how the scientists' different careers, combined with their ability and preferences, affect their career choices and subsequent earnings evolution.

Among the numerous studies on career paths that have been conducted, the problem of career path clustering has garnered considerable interest from the research community. Given the sequential nature of career trajectories, most researchers have relied on sequence analysis and clustering techniques. Abbott and Hrycak (1990) was one of the initial contributors to this strand. Their paper pioneered the use of the optimal matching algorithm (OMA) to identify prototype careers of German musicians. Later, Blair-Loy (1999) used the optimal matching techniques and the complete linkage-based clustering method to categorize the careers of female finance executives into four career types. On the other hand, numerous studies focused on the careers of senior managers. Biemann and Wolf (2009) employed the optimal matching analysis (OMA) to generate six career patterns for members of the top management team (TMT) from five nations, taking into account their international experience, organizational tenure, and professional experience. More recently, Koch et al. (2017) studied the career patterns of Fortune 100 CEOs. Similarly, they used the optimal matching analysis (OMA) followed by a hierarchical cluster analysis with Ward's linkage to discover unique groupings of CEOs' career paths and to gain a better understanding of what propelled these individuals to the top management level. In general, we can see that these approaches cluster career paths in a two-step manner. The first is to compute pairwise distances between career sequences primarily using the optimal matching analysis (OMA) method. Then, several clustering techniques are applied to the distance matrix to group job sequences into disjoint clusters. This two-step approach is often effective, especially when job titles are of low diversity.

More recently, as the IT industry has grown rapidly, we have witnessed an increasing interest in understanding the career patterns of the IT workforce. By adopting the aforementioned two-step framework, Joseph et al. (2012) performed an in-depth analysis on characterizing the IT workforce's careers using the optimal matching analysis (OMA) approach and a hierarchical agglomerative clustering technique. Their study gives a detailed picture of IT professionals' careers and career experiences and, in the meantime, highlights a research difficulty posed by the great heterogeneity of these individuals' profiles. Similarly, Lappas (2020) also noted that careers of the IT workforce are more diverse than the traditional view of a dual IT career path (technical versus managerial) and are growing in heterogeneity due to a volatile environment. To address these issues, Lappas (2020) proposed a novel two-step approach to discover the prototype career paths of the IT workforce. The author first clustered job titles by applying an Agglomerative Clustering (AC) algorithm to job description embeddings learned via the Distributed Memory Model (DMM). Then, several career prototypes are discovered via a graph-based clique finder algorithm. Note that our study relates to Lappas (2020)'s work in the following ways. First, our study shares a similar focus on career paths clustering of the IT workforce. Second, we also employ embedding-based techniques to determine the "similarity" between job titles. Our method, however, does not require an explicit second phase of sequence clustering; we learn job term/title embeddings and sequence clusters simultaneously.

Apart from career path studies, our work also contributes to the literature of general categorical sequence clustering (Aggarwal and Reddy 2013). In comparison to conventional item clustering, the categorical sequence clustering problem presents several notable challenges (Aghabozorgi et al. 2015). First, categorical sequences are often of high dimensionality, which makes data handling rather complicated. Second and more crucially, computing similarity/distance between categorical sequences is not straightforward as they are often noisy and with varying length. According to (Bicego et al. 2003), there are three categories of sequential data clustering methods. The first is *proximity-based*, which focuses on devising similarity or distance measures between sequences, followed by a standard clustering method based on the sequence distances. They are generally labelled as raw-data-based approaches as they typically work directly with raw sequence data. Owing to this trait, they have to face the challenges such as sensitivity to noise, amplitude scaling, longitudinal scaling, discontinuities and temporal drifts (Aghabozorgi et al. 2015). The OMA-based two-step approach outlined above falls into this category. The second category is *feature-based*, which aims to extract temporal features from each sequence and then downgrades sequence clustering into a more manageable item/point clustering problem. Different from *proximity-based* approaches, *feature-based* approaches do not directly work with raw sequences, but their transformed features. Therefore, they are more computationally efficient and better for large-scale data sets. The final category is the *model-based* approaches, which presuppose an analytical model for each cluster and try to determine the optimal set of such models that best fits the data. Typical models include time series models, spectral models, and finite state machines, such as hidden Markov models (HMM). Despite popularity of model-based approaches due to the adaptability to a wide range of applications, they may suffer from scalability issues, and performance deterioration when data and clusters become more complex (Aghabozorgi et al. 2015).

Note that model-based approaches have grown in popularity over the years, with hidden Markov models (HMMs) garnering the most attention for sequence modeling. An early study by Smyth (1996) viewed sequence clustering problem as a generalization of the standard mixture model approach to clustering in feature space and developed an HMMs-based solution. Likewise, Bicego et al. (2003) also employed the hidden Markov models (HMMs) to cluster sequence data. They quantified the similarity between sequences as the proximity of the represented vectors in a new representation space generated using HMMs. Moreover, Xiong et al. (2011) proposed a non-HMMs Markov model to approximate the conditional probability distribution (CPD) model, which was then used to create a novel two-tier Markov model to represent a sequence cluster. Built upon this model, a divisive hierarchical algorithm was employed to cluster categorical sequences.

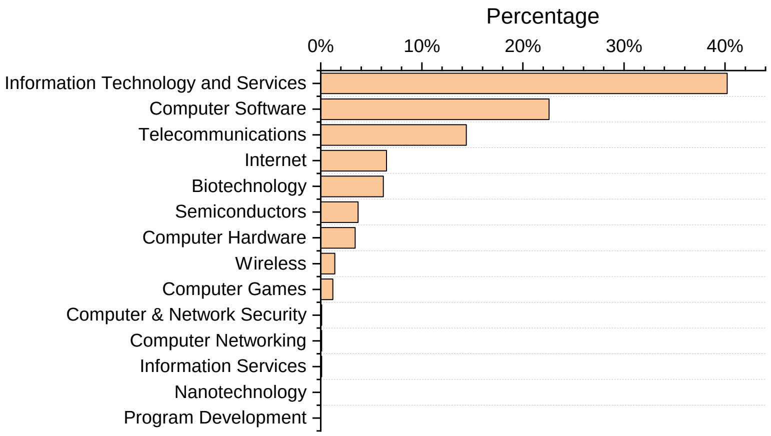


Figure 2. Percentage distribution of talents by industry

Another pertinent study was found at (Chen et al. 2012). Rather than focusing on categorical sequence clustering, the authors attempted to tackle the sequence (playlist) generation problem using a Latent Markov Embedding (LME) model. In their approach, sequences (playlists) are treated as Markov chains in a latent space, and each element (song) is represented as a point in the same space. Their LME algorithm does not treat sequence elements as atomic units devoid of metric properties, but rather gives generalized representations in a Euclidean space. Notably, this idea of sequence modeling can be adopted for the sequence clustering problem. Essentially, our proposed solution in this paper extends their idea. When modeling sequences in our setting, we treat each job term (rather than job title) as a point in the latent space. We propose that this design better captures the semantic similarity of job titles, especially when they contain shared and/or similar terms.

To summarize, our approach has the following advantages over previous work when it comes to categorical sequence clustering problems. First, our solution, unlike OMA-based methods, does not require a job title coding scheme and provides greater flexibility through the transformation of job titles into meaningful latent representations. Second, our model integrates the process of sequence clustering with the learning of job term/title embeddings. Our clustering is *soft* in the sense that no job sequence is *hard-assigned* but *fuzzily affiliated* with clusters to the degree of cluster-wise model likelihoods. Third, with the embedding technique, our model can handle more sophisticated career path clustering scenarios, such as job sequences with overlapping employment records. Meanwhile, the resulting job title embeddings that contain rich semantic information are useful for additional research purposes.

Data Description and Research Challenges

Our data are collected from *LinkedIn*, the world's largest professional networking platform, which enables members to create occupational profiles and network with other professionals. As of November 2021, it had 800 million members from more than 200 countries and territories worldwide[1]. Our study focuses on IT-related job trajectories by purposefully limiting the *industry keywords* to a predefined list as shown in Figure 2. We find that around 80% of talents are concentrated in these four industry sectors: *Information Technology and Services*, *Computer Software*, *Telecommunications*, and *Internet*. To eliminate irregular profiles, we maintain only employment profiles with a tenure of less than 30 years and an employment period between 1990 and 2018. As a result, we end up with a data sample of roughly 200K talents.

In our research, we investigate three distinct types of career-related constructs, i.e., *job sequence*, *job title*, and *job term*. A *job sequence* is a term that refers to a talent's career path, which consists of an ordered list of job titles. A *job title* is a meaningful phrase that clearly specifies the functionality and responsibility of a position, for example, 'software engineer' and 'project manager'. And *job terms* are the words that comprise a job title. They are classified into two genres (Liu et al. 2019): *functionality (FUN)* and *responsibility*

[1] https://about.linkedin.com/

RES	manager, engineer, senior, analyst, consultant, developer, director, lead, specialist, assistant, technician, administrator, president, architect, executive, owner, programmer, principal, associate, founder
FUN	software, technical, support, business, project, systems, development, network, operations, system, service, services, technology, product, engineering, marketing, web, customer, team, program

Table 1. Top 20 most frequent terms in *RES* and *FUN*

Variable	Count	Variable	Mean	Std	Min	Max
# Talents (career paths)	200,878	Employment year			1990	2018
# Distinct titles	48,359	# Terms per job title	2.47	1.16	1	12
# Distinct terms	825	# Titles per job sequence	5.04	3.58	1	25
# Distinct *FUN* & *RES* terms	572 & 253	Tenure length (number of years)	13.15	7.00	1	29

Table 2. Descriptive statistics for our dataset

(RES). For instance, 'software' and 'engineer' are two terms that refer to the job title 'software engineer', where 'software' is a *FUN* term and 'engineer' is a *RES* term. The job titles of all employment records in our data sample are quite diverse, with a total of 48K different phrases. These job titles are composed of 825 distinct terms, 572 of which are *FUN* terms and 253 of which are *RES* terms. We report the top twenty most frequently occuring terms in the *RES* and *FUN* genres in Table 1. Examples of *RES* terms include 'manager', 'engineer', 'senior' and 'analyst' while *FUN* terms include 'software', 'technical', 'systems' and 'product'.

Table 2 summarizes our data sample's descriptive statistics in detail. A job title, on average, contains 2.47 terms whereas a job sequence contains 5.04 titles. In Figure 3, we display the percentage distribution of job sequence length (number of job titles) and discover that the majority of job sequences contain ten or fewer job titles. Figure 4 illustrates how career tenure is distributed, and we observe that the majority of careers endure 5 to 15 years. Talents have an average career tenure of 13.15 years. For illustrative purpose, we construct a *Word Cloud* of job titles in Figure 5. The most frequently used titles are 'software engineer', 'software developer', 'consultant', and 'manager', which aligns with our study's concentration on the IT industry.

Using the data presented above, we intend to uncover significant career path prototypes using an effective and adaptable clustering approach. By capturing job title semantics and fine-grained information in raw career records, our methodology aims to overcome the limits of conventional sequence analysis approaches. The following section will define our research problem and then delve into the methodological details.

Methodology

Problem Formulation

We consider a set of career path sequences $\{S_n : n = 1, 2, \cdots, N\}$, where $S_n =< s_0^n, s_1^n, s_2^n, \cdots, s_{|S_n|}^n >$ is the n-th sequence, and $s_i^n \in \mathcal{S}$ is the i-th job title in the sequence and $\mathcal{S}$ is the set of unique job titles from all career path sequences. Note that the length of each sequence $|S_n|$ varies for difference job sequences. Since job titles can be collected on a regular basis (e.g., quarterly or yearly), or just when a job change is detected, the sequence length can reflect the duration of a career or the frequency of job changes, respectively. Given a large number of such career path sequences, we seek to construct M career path clusters $\{C_1, C_2, \cdots, C_M\}$, where each cluster can capture similar career path dynamics for a set of talents.

Sequence Clustering with Embedded Markov Models

Our method is designed to construct M clusters of career paths while simultaneously estimates the Embedded Markov Model (EMM) parameters for each cluster. To cluster career paths into M clusters, we randomly assign all paths to M disjoint subsets as initial clusters and then iterate the following two steps until convergence:

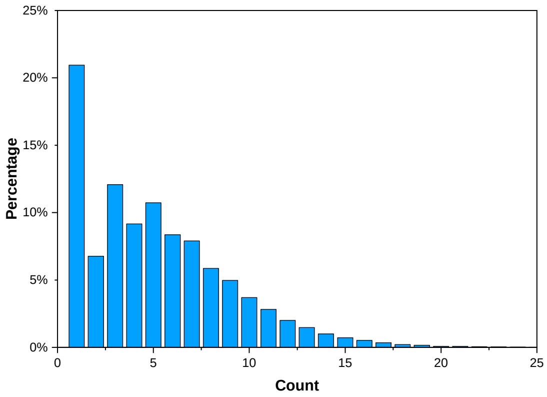


Figure 3. Distribution of job sequence length

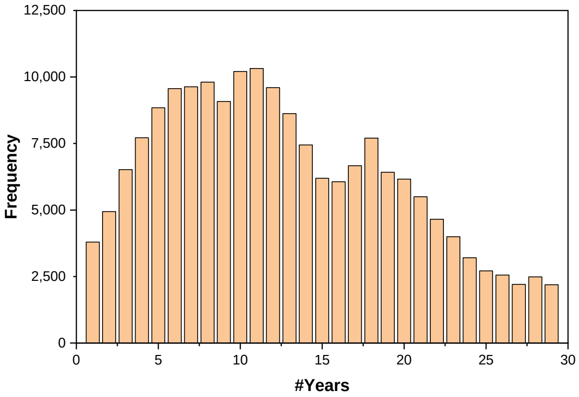


Figure 4. Distribution of career tenure length

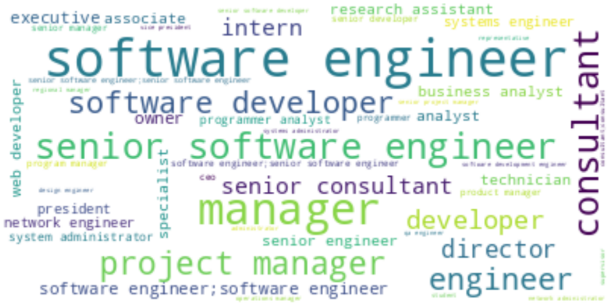


Figure 5. A Word Cloud visualization of all job titles in our data sample

Step 1: Estimate the cluster-wise EMM parameters θ_m for all paths in cluster C_m, for $m = 1, 2, \cdots, M$. The cluster C_m is defined as:

$$C_m \leftarrow \{S_n : \arg\max_{m'} \Pr(S_n|\theta_{m'}) = m\}. \tag{1}$$

Step 2: Assign each path S_n to the cluster C_m with the highest likelihood, i.e.:

$$m \leftarrow \arg\max_{m'} \Pr(S_n|\theta_{m'}). \tag{2}$$

For better illustration purpose, Figure 6 provides a holistic view of our methodology. We start by discussing the Embedded Markov Model and the probabilistic model $\Pr(S_n|\theta_m)$ (*(A) Transition Probability Computation* in Figure 6). Note that θ_m represents the parameters of cluster-wise EMM, which is estimated to maximize the likelihood:

$$\Pr(\theta_m|C_m) \propto \Pr(C_m|\theta_m) \Pr(\theta_m), \tag{3}$$

where

$$\Pr(C_m|\theta_m) = \prod_{S_n \in C_m} \Pr(S_n|\theta_m). \tag{4}$$

The parameterization with $\{\theta_m : m = 1, 2, \cdots, M\}$ and the consequent optimization problem are the primary challenges in our methodology development.

Modeling career paths and job transitions with the Markov assumption is a natural approach, where the probability of the career sequence S_n is decomposed into the product of the transition probabilities $\Pr(s_{i-1}^n \rightarrow s_i^n)$:

$$\Pr(S_n|\theta_m) = \prod_{i=1}^{|S_n|} \Pr(s_{i-1}^n \rightarrow s_i^n|\theta_m). \tag{5}$$

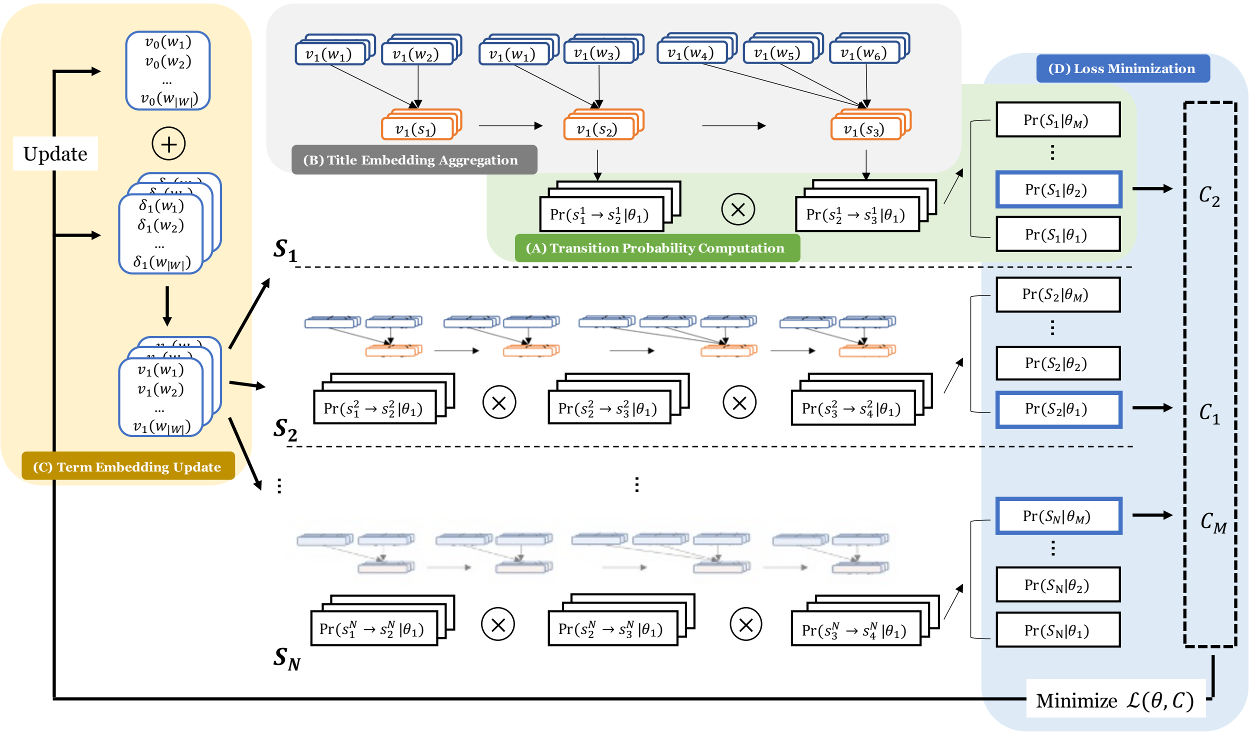


Figure 6. An illustrative diagram of our EMMMs model

A straightforward next step to estimate transition probabilities is to count transition frequencies between job titles of each cluster's career sequences. However, this naive estimation will deem distinct job titles completely independent. As explained in Section *Introduction*, there are important semantic links among complex job titles. To capture these semantic links among job titles, we attempt to learn their embeddings in a Euclidean metric space. To be specific, we define:

$$\Pr(s_{i-1}^n \to s_i^n|\theta_m) = \frac{\exp\left(-\|v_m(s_i^n) - v_m(s_{i-1}^n)\|_2^2\right)}{Z_m(s_{i-1}^n)}, \tag{6}$$

where

$$Z_m(s_{i-1}^n) = \sum_{s \in \mathcal{S}} \exp\left(-\|v_m(s) - v_m(s_{i-1}^n)\|_2^2\right). \tag{7}$$

By this definition, job title embeddings are the optimal locations of job titles in a Euclidean metric space such that the probabilistic likelihood of the sequence cluster C_m in our data is maximized. Finally, using our EMM approach, we have:

$$\Pr(S_n|\theta_m) = \left(\prod_{i=1}^{|S_n|} \frac{\exp\left(-\|v_m(s_i^n) - v_m(s_{i-1}^n)\|_2^2\right)}{Z_m(s_{i-1}^n)}\right)^{1/|S_n|}. \tag{8}$$

Note that the probability is normalized by the sequence length $|S_n|$, ensuring that our model will not favor short sequences. The following subsections further elaborate on the details of the job title embedding component to address unique challenges of career path clustering tasks.

Embedding of Job Titles via Job Terms

One difficulty in assessing self-reported career profiles is navigating the inherent complexity and ambiguity in raw job titles. Our EMM architecture utilizes a metric embedding approach to address this difficulty. To further reduce the modeling complexity, our method is capable of embedding job titles at the term level. For instance, given job titles 'software engineer' and 'hardware engineer', our method learns embeddings of the terms 'software', 'hardware', and 'engineer'. The embedding of a job title is then computed by aggregating the embeddings of the job title's terms, e.g., 'software engineer' is embedded by aggregating the embeddings of 'software' and 'engineer'. The key idea behind this type of term-level embedding is that 'engineer' can share the same metric embedding, provided that both job titles indicate semantically similar skills and professional positions. Meanwhile, the embeddings of 'software' and 'hardware', respectively, capture the distinctions between the two job positions.

Specifically, we consider each job title s_i^n as a set of terms $\{w \in s_i^n\}$ (*(B) Title Embedding Aggregation* in Figure 6). To compute the embedding $v_m(s_i^n)$ of job title s_i^n in cluster m, we learn and aggregate the term-level embeddings:

$$v_m(s_i^n) = \frac{1}{|s_i^n|} \sum_{w \in s_i^n} v_m(w), \tag{9}$$

where $v_m(w) \in \mathbb{R}^D$ is trainable embedding vector in the EMM learning process and D is the dimensionality of the metric embedding space.

According to Table 2, using term-level embeddings can reduce the number of trainable embeddings from $48,359$ (distinct titles) to merely 825 (distinct terms), which is a considerable reduction in modeling complexity. In our empirical studies, we will investigate the performance of the term-level embedding approach from various aspects.

Additionally, our term-level embedding enables us to investigate the responsibility and functionality of professionals, which would be impossible with a title-level embedding approach. As we will discuss in Section Section *Experimental Analyses and Results*, *RES* term embeddings computed using our approach can reveal prominent semantic meanings in contrast to other contemporary pre-trained model-based word embeddings. In a nutshell, job terms with similar *responsibilities* appear closer in the embedded space.

Shared Embedding for Multiple EMM Clusters

The term-level embeddings $\theta_m = \{v_m(w) : w \in \mathcal{W}\}$, where $\mathcal{W}$ is the set of unique job terms, are designed to capture the job transition probabilities via a Euclidean metric space for the job sequences in the m-th cluster. On the one hand, the cluster-specific embedding space enables the model to better fit the cluster-level job transition dynamics. However, on the other hand, the cluster-independent estimates of the embedding vectors add complexity to the modeling and deteriorate the learning efficiency. To achieve a good balance between the modeling capacity and complexity, we resort to the intuition of Hierarchical Bayes (Arora et al. 1998). Specifically, we assume that the cluster-level embeddings $v_m(w)$ follow a Gaussian distribution with the global-level embeddings $v_0(m)$ as the distribution mean. We thus define (*(C) Term Embedding Update* in Figure 6):

$$v_m(w) = v_0(w) + \delta_m(w), \tag{10}$$

where $\delta_m(w)$ is the difference between the term embeddings at the cluster and global levels, respectively. By assuming $\delta_m(w) \sim \mathcal{N}(0, \sigma_m)$, our model penalizes discrepancies in the embeddings of the same term w across different clusters, essentially allowing the clusters to share information for parameter estimation in the clustering phase.

Formally, let $\theta = \{\theta_m : m = 1, 2, \cdots, M\}$, our shared embedding approach adopts the prior distribution of the embedding vectors where the probability density is:

$$\Pr(\theta) \propto \prod_{m=1}^{M} \exp\left(-\frac{\|v_0(w)\|^2}{2\sigma_0^2}\right) \cdot \exp\left(-\frac{\|\delta_m(w)\|^2}{2\sigma_m^2}\right), \tag{11}$$

where the first term assuming $v_0(w) \sim \mathcal{N}(0, \sigma_0)$ is used to control the overall modeling complexity, and the second term assuming $\delta_m(w) \sim \mathcal{N}(0, \sigma_m)$ is used to learn shared embeddings for multiple clusters.

```
Algorithm 1: Embedded Mixture Markov Models (EMMMs)
Input: The set of sequences $\{S_n : n = 1, 2, \cdots, N\}$; the number of clusters $M$; the embedding
       dimensionality $D$; the number of iterations $T$.
Output: Cluster-wise term embeddings $\theta_m$ and probability log-likelihoods $\Pr(S_n|\theta_m)$.
Determine the initial clusters of all career sequences and randomly initialize $v_0(w)$ and
  $\{\delta_m(w) : m = 1, 2, ..., M\}$, where $w \in \mathcal{W}$;
for i=1 to $T$ do // $T$ is the number of iterations
    for m=1 to M do // $M$ is the number of clusters
        if i=1 then
            Initialize an EMM model with $v_0(w)$ and $v_m(w)$;
        else
            Update $v_0'(w) \leftarrow v_0(w) + \frac{1}{M}\sum_m \delta_m(w)$ and $\delta_m'(w) \leftarrow \delta_m(w) - \frac{1}{M}\sum_m \delta_m(w)$;
        end
        for e=1 to E do // $E$ is the number of epochs
            Randomly generate a batch;
            Compute the embedding of career sequence $S_n$ by aggregating $v_m(w) = v_0(w) + \delta_m(w)$ for
              any job term $w$ in the sequence;
            Compute the overall loss $\mathcal{L}(\theta, C)$ according to Equation 12;
            Optimize $v_0(w)$ and $\delta_m(w)$ using backpropagation;
        end
    end
end
Compute $\Pr(S_n|\theta_m)$ for each career sequence $S_n$ in any cluster $C_m$ using Equation 8.
```

Overall Optimization and Implementation Details

With shared embeddings among a mixture of EMM clusters, we denote our method as **Embedded Mixture Markov Models (EMMMs)**. To estimate the term-level embeddings $\theta = \{\theta_m = \{v_m(w) : w \in \mathcal{W}\}, m = 1, 2, \cdots, M\}$, in the EMMMs, we use the *Negative Log Likelihood* to define the *loss* function as (*(D) Loss Minimization* in Figure 6):

$$
\begin{aligned}
\mathcal{L}(\theta, C) &= -\sum_{m=1}^{M} \log\left(\Pr(C_m|\theta_m)\right) - \log(\Pr(\theta)) \\
&= -\sum_{m=1}^{M} \sum_{S_n \in C_m} \log(\Pr(S_n|\theta_m)) - \log(\Pr(\theta)) \\
&= \sum_{m=1}^{M} \sum_{S_n \in C_m} \frac{1}{|S_n|} \sum_{i=1}^{|S_n|} \left(\|v_m(s_i^n) - v_m(s_{i-1}^n)\|_2^2 + \log(Z_m(s_{i-1}^n)) \right) - \log(\Pr(\theta))
\end{aligned}
\tag{12}
$$

where we have $\log(\Pr(\theta)) = -\left(\frac{M}{2\sigma_0^2}\|v_0(w)\|^2 + \sum_{m=1}^{M} \frac{1}{2\sigma_m^2}\|\delta_m(w)\|^2\right)$ according to Equation 11 and after dropping a constant term. The hyper-parameters in the loss function include the number of clusters M, the embedding dimensionality D, and two regularization parameters σ_0 and σ_m.

We provide our algorithm's pseudocode (Algorithm 1) to optimize the loss function in Equation 12 with stochastic gradient descent procedures implemented using PyTorch (Paszke et al. 2019) and the Adam optimizer (Kingma and Ba 2015). Following the common practice in using these tools, we employ an automatic learning rate scheduler and meanwhile tune the weight decay parameter in Adam optimizer to achieve a similar effect of tuning our hyper-parameters θ_0 and θ_m.

Although the optimization problem in Equation 12 is not convex, it is straightforward to show that Algorithm 1 will converge to a local optima. At each iteration, we first update $\theta \leftarrow \theta' = \{\theta_m'\}$ with a fixed sequence assignment $C_m = \{S_n : \arg\max_{m'} \Pr(S_n|\theta_{m'}) = m\}$ and current model parameters $\theta = \{\theta_m\}$,

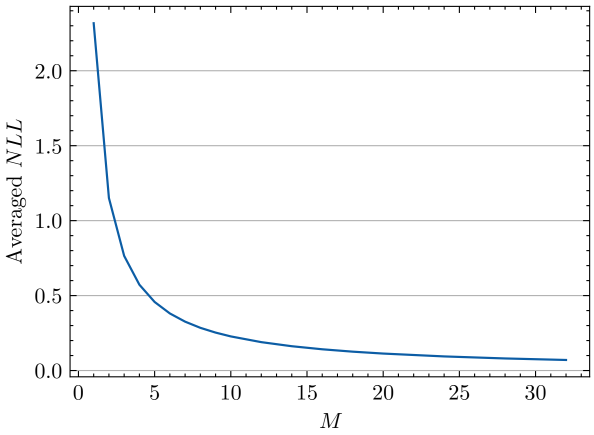


Figure 7. ***Averaged NLL* with varying M**

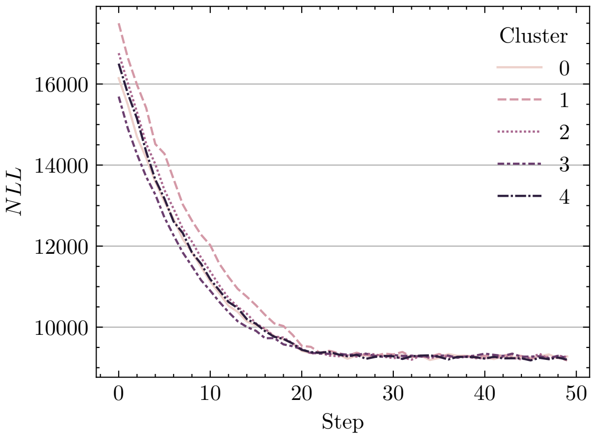


Figure 8. Learning curves of five EMM models

where $m = 1, 2, \cdots, M$. Note that, given the fixed sequence assignment, our model parameters are learned using the maximum log-likelihood method. Thus, we have $\mathcal{L}(\theta', C) \leq \mathcal{L}(\theta, C)$. In the second step, given the fixed embeddings θ', each sequence S_n in cluster C_m will be assigned to a new (and possibly different) cluster $C_{m'}$ with the maximal likelihood:

$$\Pr(S_n|\theta'_{m'}) \geq \Pr(S_n|\theta'_m). \tag{13}$$

Thus, we have:

$$\begin{aligned}\mathcal{L}(\theta', C') =& -\sum_{m=1}^{M} \sum_{S_n \in C_{m'}} \log(\Pr(S_n|\theta'_{m'})) - \log(\Pr(\theta')) \\ \leq& -\sum_{m=1}^{M} \sum_{S_n \in C_m} \log(\Pr(S_n|\theta'_m)) - \log(\Pr(\theta')) \\ =& \mathcal{L}(\theta', C) \leq \mathcal{L}(\theta, C).\end{aligned} \tag{14}$$

We will provide more empirical convergence analysis in the next section.

Experimental Analyses and Results

We present our experimental analysis and findings in this section. We validate our method on a representative sample of real-world IT professionals and discussing numerous noteworthy findings.

Determining the number of clusters. Similar to many other conventional clustering models (e.g., K-means), one critical hyper-parameter of our model is the number of clusters. Typically, people use Silhouette Coefficient (SC) or Akaike's Information Criterion (AIC) to find the appropriate number of clusters (Tan et al. 2016). While the calculation of SC depends on a predefined distance metric (e.g., euclidean distance), the "distances" of the clusters generated from our model are ill-defined, which unfortunately rules out SC as a viable choice. Meanwhile, AIC is also unfavorable in our case as the magnitudes of the resulting log-likelihood are in significantly different scale of M. We thus propose an AIC-alike criteria, the negative log-likelihood (NLL) averaged by M, i.e., *Averaged NLL* = NLL/M. Similar to AIC, this criterion seeks to balance between the goodness-of-fit and simplicity of the model. The clustering model with the lowest *Averaged NLL* is preferred. We display the resultant curve from our model training in Figure 7, in which we find that the model's *Averaged NLL* drops drastically as the number of clusters approaches $M = 5$ and then gradually decreases to zero. Following the Elbow method (Thorndike 1953), we set $M = 5$ as it is the elbow point which sets apart the diminishing trends of *Averaged NLL*. Our model is then trained to group career paths into $M = 5$ clusters with five corresponding EMM models. Note that the embedding dimensionality

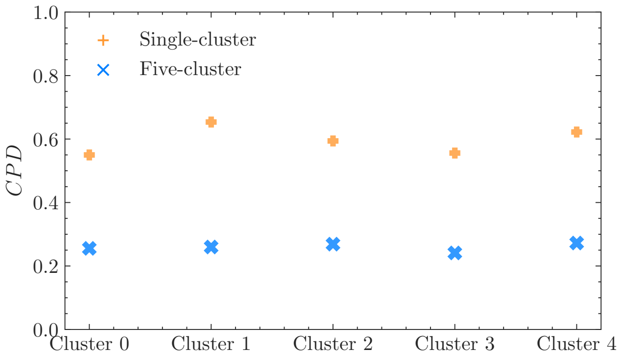


Figure 9. Divergence scores from the five-cluster and single-cluster models

is set $D = 32$. We further show the learning curves of these models in Figure 8. As can be seen, NLLs of all five EMM models decrease efficiently as the iteration step increases.

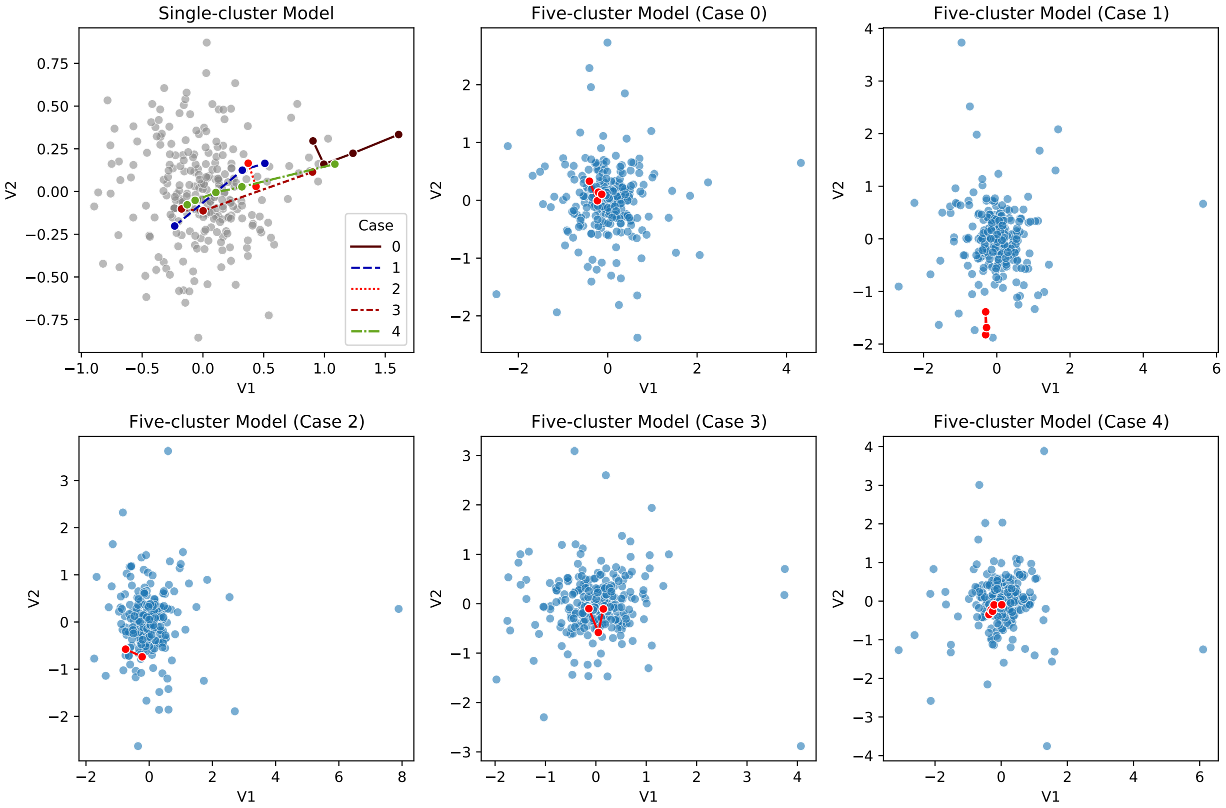
Divergence of clusters. One significant advantage of our method is the flexibility of job title representations earned through embedding techniques. Based upon the right assignment of career paths to clusters, our model tend to bring closer the job titles from the same sequence in the high-dimensional embedded space. In other words, if our clustering approach is effective, the length (spanning distance) of a career path in its proper cluster should be shorter in comparison with the length of the same path assigned to a different cluster. In this regard, we develop the following model evaluation metric. Based on the clustering results, we compute job title embeddings in each cluster using Equation 9. Then we employ PCA (Abdi and Williams 2010) dimensionality reduction technique to convert high-dimensional embeddings into two-dimensional vectors and visualize career paths as trajectories in the embedding space. Next, we compute the "length" of a job sequence in the two-dimensional plane as the sum of *euclidean distances* between successive job titles. The "average length" of job sequences is thus defined as L_m in cluster C_m. On the other hand, for each cluster C_m, we estimate the "dispersion" of job titles on the two-dimensional plane as average distance of each point (representing a job term) to the cloud center, denoted as D_m. Finally, we define the cluster's *career path divergence score* (CPD_m) as the average sequence length normalized by the job title dispersion, i.e.:

$$CPD_m = \frac{L_m}{D_m}. \tag{15}$$

It is worth noting that a lower divergence score suggests more effective clustering process, as the career paths assigned to each cluster are *shorter* in the embedded space.

As a benchmark, we construct another model that clusters job sequences into a *single* cluster C^s and generates new embeddings for the same job titles. Our objective is to illustrate our clustering method's superiority in grouping job sequences into proper clusters when compared to the unclustered (i.e., single-cluster) case. This can be affirmed by substantially shorter length of job sequences in the two-dimensional plane of our method's resultant embedded spaces. Specifically, we calculate the *career path divergence score* of the same sequence assignments using the five-cluster embeddings (denoted as CPD_m) and the single-cluster embedding (denoted as CPD^s_m), respectively. Then, we compare CPD_m and CPD^s_m for each cluster in Figure 9. The figure reveals that grouping job sequences into appropriate clusters significantly reduces divergence scores, demonstrating our model's effectiveness.

To further illustrate, we select one representative career paths (with a high log-likelihood) from each cluster and plot their projections on the two-dimensional plane in Figure 10. Each dot represents a job title projected onto the plane using *PCA*. These career paths are depicted on two distinct sets of two-dimensional planes, corresponding to the results from i) the single-cluster model and ii) the five-cluster model. The top-left figure illustrates the results of the single-cluster model for the five representative career paths, whereas the remaining five figures show the disjoint clusters discovered using the five-cluster model. As can be seen, the connected dots in the first figure are notably more dispersed than those in the other five figures. This


Single-cluster Model

Five-cluster Model (Case 0)

Five-cluster Model (Case 1)

Five-cluster Model (Case 2)

Five-cluster Model (Case 3)

Five-cluster Model (Case 4)

Figure 10. Visualization of career paths on two-dimensional planes

example demonstrates yet again that our model is capable of learning cluster-wise job title representations while also generating proper job sequence assignments.

Cluster-wise job title distribution. To better understand the unique characteristics of career paths inside each cluster, we generate two charts of cluster-wise job titles. To begin, we visualize the majority of job titles for each cluster using *Word Cloud* plots in Figure 11 (in contrast to the holistic view in Figure 5). Second, we use a heat map to visualize the distributions of *starting* job titles (the first title in sequence) of career paths within each cluster in Figure 12. By examining both the word clouds and the heat map, we can see that *Cluster 0* is more focused on job positions like ‘system administrator’ and ‘system engineer’ while *Cluster 1* gathers more ‘consultant’-alike titles. *Cluster 3* contains career paths that begin with ‘interns’ and end with ‘project managers’. Additionally, *Cluster 2* & *4* are similar in that the majority of their titles are ‘software engineers’ and/or ‘senior software engineers’, which is foreseeable considering the popularity of ‘software engineers’ in our data. To further examine their distinctions, we construct a *Sankey* diagram in Figure 13 to visualize *all* careers started as ‘software engineer’. We notice that the majority of careers that begin as *software engineer* fall into *Cluster 2* & *4*, which is consistent with the pattern in Figure 12. A deep dive reveals that *Cluster 2* identifies individuals who remain in a regular ‘software engineer’ position for an extended period of time, whereas *Cluster 4* identifies those who receive continuous promotions from ‘software engineer’ to ‘senior software engineer’ or even ‘principal software engineer’. Our method distills the nuanced distinctions across career paths, even when similar job titles are shared among them.

Job term embeddings. Along with job sequence clustering, our model is expected to produce global-level job term embeddings $v_0(w)$ that contain shared knowledge of job titles among clusters. Here, we investigate *RES* job terms in more detail. As common sense, jobs title like ‘intern’ and ‘assistant’ are considered *junior-level* positions; ‘associate’ and ‘manager’ are *senior-level* ones; and ‘CEO’, ‘CFO’, ‘CTO’ and ‘partner’ associate with *top(executive)-level* positions. The purpose of this investigation is to understand whether job

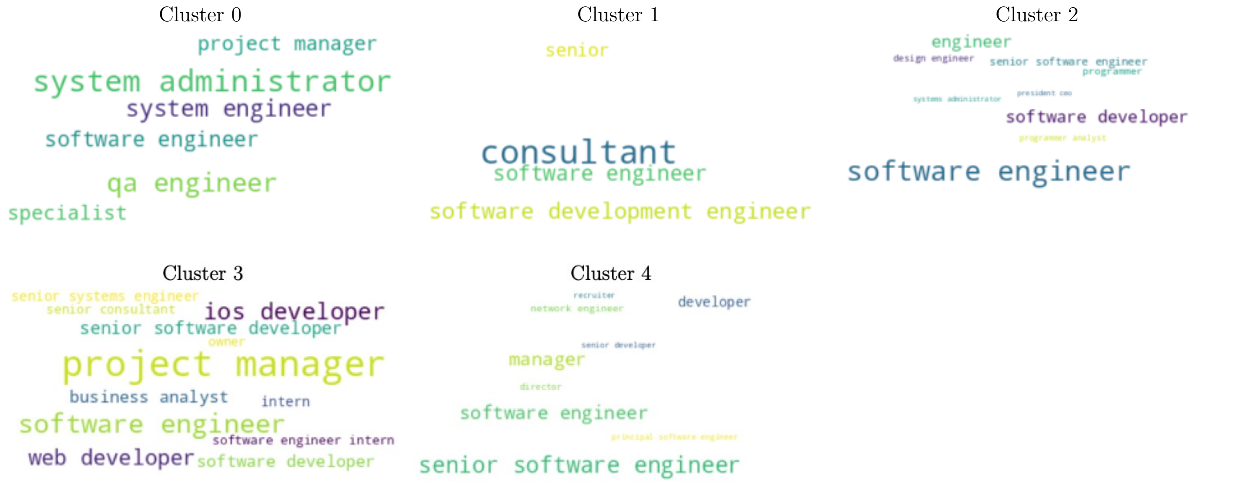


Figure 11. Visualization of job sequences by cluster

embeddings learned from our model can disclose implicit knowledge of job hierarchy. We begin by scattering and labeling several representative job terms according to their PCA-based two-dimensional projections of $v_0(w)$ in Figure 14a. It shows that the *top-level* terms are centralized in the plot and are surrounded by the *junior-level* terms. The distribution of these job terms reveals their hierarchical levels. In comparison, we use a pre-trained word embedding model *glove-wiki-gigaword-50* to produce new embeddings of these same job terms. This model is based on GloVe (Pennington et al. 2014) and was trained using the Wikipedia 2014 + Gigaword 5 dataset with an embedding dimensionality of 50[2]. Similarly, we scatter these job terms on another two-dimensional plane based on their PCA transformations in Figure 14b. Despite the fact that several close terms bear similar semantic meanings (e.g., CTO vs. technology), we fail to discern a notable hierarchical pattern in the distribution of these job terms.

Discussions

Contributions to IS Literature Our study contributes to the IS literature of IT-related career studies (Joseph et al. 2012; Lappas 2020; Reich and Kaarst-Brown 1999). We advance the study of IT-related careers by developing a novel career path clustering model, which is a non-trivial improvement over the case-based studies (Reich and Kaarst-Brown 1999) and conventional sequence similarity-based clustering methods (Joseph et al. 2012; Lappas 2020). Compared with other methods, our model can better address the high-heterogeneity issue of self-reported career records and perform career path clustering in a *fuzzy* manner. Furthermore, our method is applicable to other important sequence-related IS research, such as digital trace analysis (Pentland et al. 2021) and business process mining (Breuker et al. 2016).

Our work also fits into the computational design science research (Padmanabhan et al. 2022; Rai 2017). The computational design science research is concerned with "solving business and societal problems by developing computational models and algorithms" (Rai 2017). Our study develops a novel sequence clustering algorithm using sequential embedding and mixture Markov models to address career path clustering problem. Our contributions are rooted in algorithm-centric advancement and application-oriented solutions.

Methodological Implications for IS The first methodological implication of our study for IS is the embedding-based design in modeling career paths. This technique stems from the recent advancement of representation learning methods which are the core of deep learning research (Bengio et al. 2013). The embedding-based design enables our model to distill nuanced subtleties of job terms/titles which can further serve for other research purposes. Our work is another innovative example of this useful technique applied to IS research problems. Another important implication relates to the use of mixture Markov mod-

[2] https://nlp.stanford.edu/projects/glove/

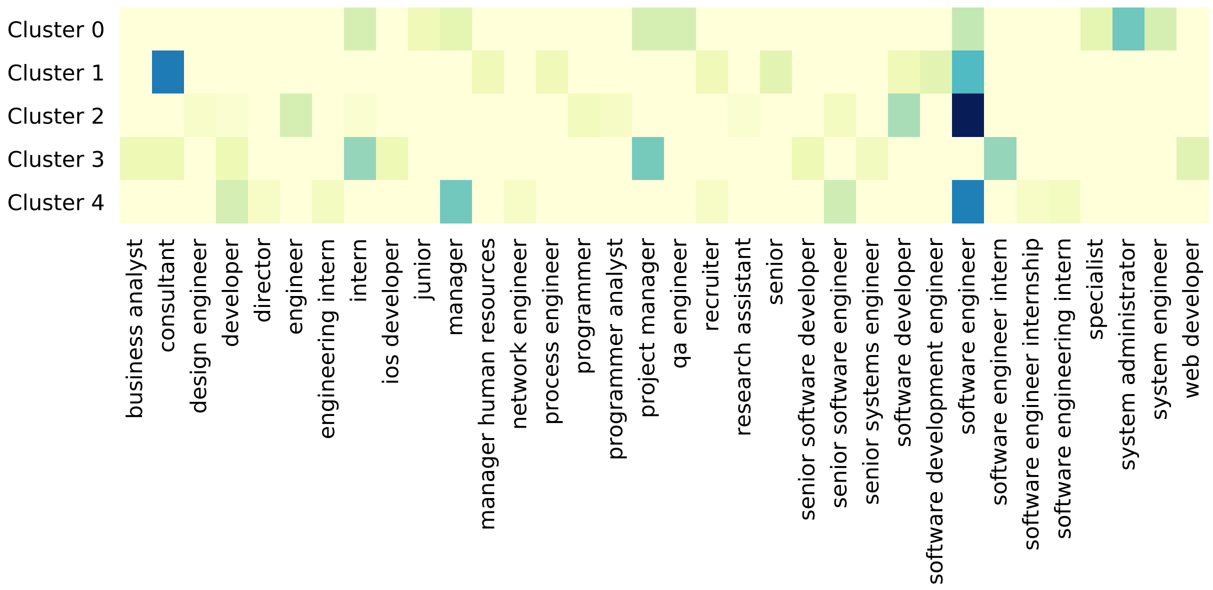


Figure 12. Distribution of *starting* job titles in different clusters

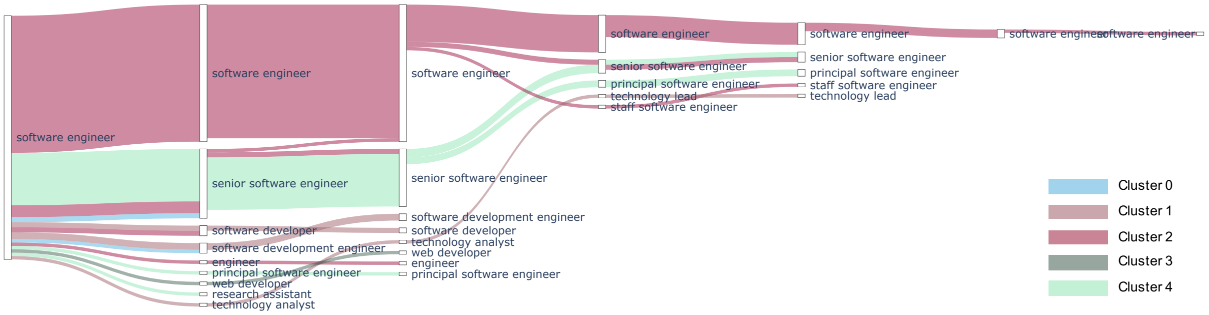
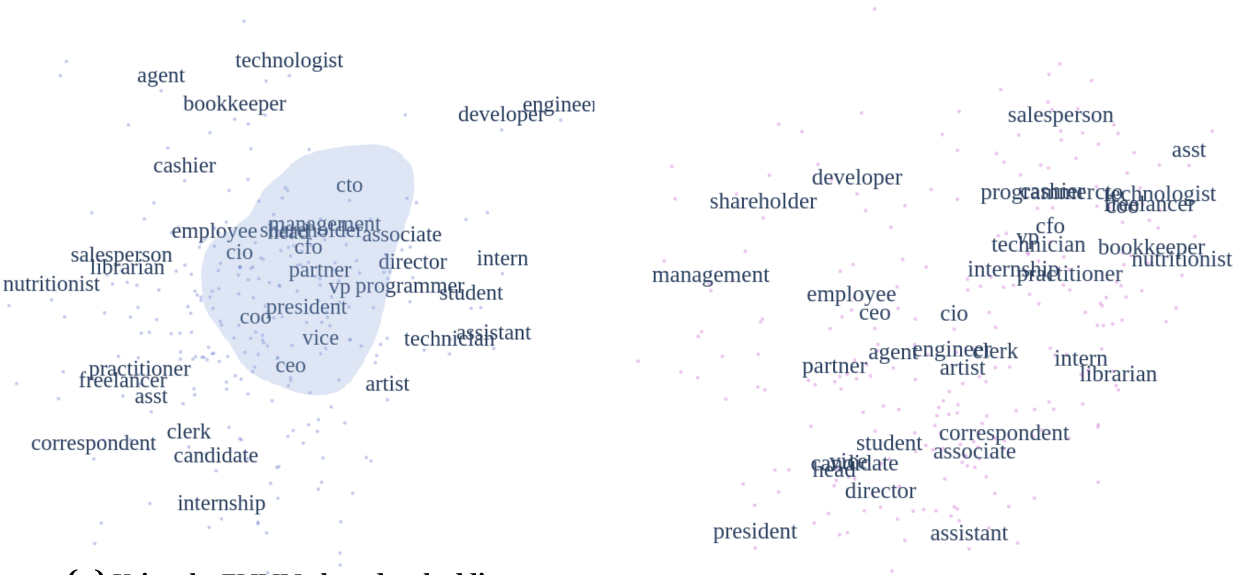


Figure 13. Typical career paths started as 'software engineer' in different clusters

els for sequence clustering. This method is proved advantageous by generating fuzzy clusters, which can capture more fine-grained information among different clusters. Our study demonstrates the superiority of such models and thus opens more doors to the application of probabilistic mixture models for unsupervised learning problems in IS research.

Limitations and Future Research In the meantime, we are aware of several limitations of our study, which can serve as a springboard for future research. First, our clustering model, like others, requires several user-specified inputs. While users are able to determine the number of clusters using the Elbow method, future research could investigate more automatic and principled choice. Meanwhile, the final clustering output is dependent on initial clusters, which are best generated using pilot models (e.g., OMA). Second, our model does not directly incorporate or differentiate between job term genres, i.e., *functionality* and *responsibility*. The development of algorithms for genre-aware career path clustering is an intriguing and promising area of research. Third, our current method overlooks other job variables when learning career patterns, such as employers, industry sectors, talents' demographics, and geographical locations. We believe that a more in-depth study and understanding of career patterns through the use of heterogeneous information has notable benefits for both academics and practitioners.



(a) Using the EMMMs-based embeddings **(b) Using the pre-trained embeddings from GloVe**

Figure 14. Scatter plots of the *RES* terms on two-dimensional planes

Conclusion

Career paths clustering plays a vital role in understanding career prototypes and patterns. In this study, we built an Embedded Mixture Markov Models (EMMMs) to address the unique challenges of career path clustering and applied our method to large-scale career records of IT professionals. Given a pre-determined number of clusters, our method attempts to cluster career paths by creating an embedded Markov model for each cluster. The optimal assignment of career paths is accomplished by simultaneously learning cluster-wise embeddings of job terms/titles and optimizing the mixture Markov model's overall log-likelihood. Experiments on a real-world dataset of extensive career records from Linkedin demonstrated our model's efficacy and revealed interesting career patterns of the IT workforce. The improved career path clusterings are valuable for tasks like talent recruiting and career planning. The acquired insights are also useful for educational institutes to design appropriate training programs and provide people with the skills they require to successfully navigate through the long and complex career paths.

References

Abbott, A. and Hrycak, A. 1990. "Measuring resemblance in sequence data: An optimal matching analysis of musicians' careers". *American journal of sociology* (96:1), pp. 144–185.

Abdi, H. and Williams, L. J. 2010. "Principal component analysis". *Wiley interdisciplinary reviews: computational statistics* (2:4), pp. 433–459.

Agarwal, R. and Ohyama, A. 2013. "Industry or academia, basic or applied? Career choices and earnings trajectories of scientists". *Management Science* (59:4), pp. 950–970.

Aggarwal, C. and Reddy, C. 2013. *Data Clustering: Algorithms and Applications*. Chapman & Hall/CRC Data Mining and Knowledge Discovery Series. Taylor & Francis.

Aghabozorgi, S., Shirkhorshidi, A. S., and Wah, T. Y. 2015. "Time-series clustering–a decade review". *Information Systems* (53), pp. 16–38.

Arora, N., Allenby, G. M., and Ginter, J. L. 1998. "A hierarchical Bayes model of primary and secondary demand". *Marketing Science* (17:1), pp. 29–44.

Bengio, Y., Courville, A., and Vincent, P. 2013. "Representation learning: A review and new perspectives". *IEEE Transactions on Pattern Analysis and Machine Intelligence* (35:8), pp. 1798–1828.

Bicego, M., Murino, V., and Figueiredo, M. A. 2003. “Similarity-based clustering of sequences using hidden Markov models”. In: *International Workshop on Machine Learning and Data Mining in Pattern Recognition*. Springer, pp. 86–95.

Biemann, T. and Wolf, J. 2009. “Career patterns of top management team members in five countries: An optimal matching analysis”. *The International Journal of Human Resource Management* (20:5), pp. 975–991.

Biemann, T., Zacher, H., and Feldman, D. C. 2012. “Career patterns: A twenty-year panel study”. *Journal of Vocational Behavior* (81:2), pp. 159–170.

Blair-Loy, M. 1999. “Career patterns of executive women in finance: An optimal matching analysis”. *American Journal of Sociology* (104:5), pp. 1346–1397.

Breuker, D., Matzner, M., Delfmann, P., and Becker, J. 2016. “Comprehensible predictive models for business processes”. *MIS Quarterly* (40:4), pp. 1009–1034.

Chen, S., Moore, J. L., Turnbull, D., and Joachims, T. 2012. “Playlist prediction via metric embedding”. In: *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (SIGKDD)*, pp. 714–722.

Cohen, L. 2006. “Remembrance of things past: Cultural process and practice in the analysis of career stories”. *Journal of Vocational Behavior* (69:2), pp. 189–201.

Dlouhy, K. and Biemann, T. 2015. “Optimal matching analysis in career research: A review and some best-practice recommendations”. *Journal of Vocational Behavior* (90), pp. 163–173.

Joseph, Boh, Ang, and Slaughter 2012. “The career paths less (or more) traveled: a sequence analysis of it career histories, mobility patterns, and career success”. *MIS Quarterly* (36:2), p. 427.

Kingma, D. P. and Ba, J. 2015. “Adam: A Method for Stochastic Optimization”. In: *Proceedings of the International Conference on Learning Representations (ICLR)*.

Koch, M., Forgues, B., and Monties, V. 2017. “The way to the top: Career patterns of Fortune 100 CEOs”. *Human Resource Management* (56:2), pp. 267–285.

Lappas, T. 2020. “Mining Career Paths from Large Resume Databases: Evidence from IT Professionals”. *ACM Transactions on Knowledge Discovery from Data (TKDD)* (14:3), pp. 1–38.

Liu, J., Guo, C., Ng, Y. C., Wood, K. L., and Lim, K. H. 2019. “IPOD: Corpus of 190, 000 Industrial Occupations”. *CoRR* (abs/1910.10495). arXiv: 1910.10495.

Padmanabhan, B., Fang, X., Sahoo, N., and Burton-Jones, A. 2022. “Machine Learning in Information Systems Research”. *MIS Quarterly* (46:1), pp. iii–xix.

Paszke, A. et al. 2019. “PyTorch: An Imperative Style, High-Performance Deep Learning Library”. In: *Advances in Neural Information Processing Systems*. Ed. by H. Wallach, H. Larochelle, A. Beygelzimer, F. d’Alché-Buc, E. Fox, and R. Garnett. Curran Associates, Inc., pp. 8024–8035.

Pennington, J., Socher, R., and Manning, C. D. 2014. “Glove: Global vectors for word representation”. In: *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1532–1543.

Pentland, B. T., Vaast, E., and Wolf, J. R. 2021. “Theorizing Process Dynamics with Directed Graphs: A Diachronic Analysis of Digital Trace Data”. *MIS Quarterly* (45:2).

Rai, A. 2017. “Editor’s comments: Diversity of design science research”. *MIS quarterly* (41:1), pp. iii–xviii.

Reich, B. H. and Kaarst-Brown, M. L. 1999. “Seeding the Line: Understanding the Transition From IT to Non-IT Careers”. *MIS Quarterly* (23:3), pp. 337–364.

Sankoff, D., Kruskal, J., and Nerbonne, J. 2000. *Time Warps, String Edits, and Macromolecules: The Theory and Practice of Sequence Comparison*. The David Hume Series. Cambridge University Press.

Smyth, P. 1996. “Clustering sequences with hidden Markov models”. *Advances in Neural Information Processing Systems* (9).

Tan, P.-N., Steinbach, M., and Kumar, V. 2016. *Introduction to data mining*. Pearson Education India.

Thorndike, R. L. 1953. “Who belongs in the family”. In: *Psychometrika*. Citeseer.

Vinkenburg, C. J. and Weber, T. 2012. “Managerial career patterns: A review of the empirical evidence”. *Journal of Vocational Behavior* (80:3), pp. 592–607.

Xiong, T., Wang, S., Jiang, Q., and Huang, J. Z. 2011. “A new markov model for clustering categorical sequences”. In: *Proceedings of the IEEE International Conference on Data Mining (ICDM)*. IEEE, pp. 854–863.