Looking Further into the Future: Career Pathway Prediction

Michiharu Yamashita
The Pennsylvania State University
University Park, PA, USA
michiharu@psu.edu

Yunqi Li
Rutgers University
New Brunswick, NJ, USA
yunqi.li@rutgers.edu

Thanh Tran
Worcester Polytechnic Institute
Worcester, MA, USA
tdtran@wpi.edu

Yongfeng Zhang
Rutgers University
New Brunswick, NJ, USA
yongfeng.zhang@rutgers.edu

Dongwon Lee
The Pennsylvania State University
University Park, PA, USA
dongwon@psu.edu

ABSTRACT

The problem of predicting one's next career move (also known as job mobility prediction) is one of the most fundamental tasks in a computational job marketplace. The problem is helpful not only for recruiters to find potential talented labors but also for job seekers to understand and plan their future pathways. While there exist multiple studies to this problem, however, they mainly focus on how to predict the "immediate" next career move (i.e., one's next company or job), thus lacking the information about one's long-term career movement. To address this gap, we propose a unique task of predicting one's future career pathway as a "sequence" (i.e., one's next N steps of career movement). Toward this challenge, we develop a new model, NAOMI, that uses: (1) multi-view graph embeddings and BERT embeddings from job titles and companies extracted from resumes, (2) job duration masking to adjust the job experience, and (3) neural collaborative reasoning to represent the multi-factors available among job seekers' resume graph. Based on the multi-factor encoding, NAOMI predicts one's next N steps of job titles and companies. When evaluated with our large-scale real-world dataset with more than 300K job titles, NAOMI outperforms state-of-the-art baselines on predicting one's career pathway of N steps, more so as N increases.

CCS CONCEPTS

• **Information systems → Data mining**.

KEYWORDS

Career Pathways; Job Mobility; Multi-view Embedding; Graph Embedding; Collaborative Reasoning

1 INTRODUCTION

The past decades of career technology have witnessed the rapid growth of online professional networks, wherein millions of users have posted their career trajectories and resumes [1, 15]. In fact, as of 2021, LinkedIn, Indeed, and CareerBuilder have career trajectories of nearly 800 million users[1], 175+ million resumes[2], and 140+ million resumes[3], respectively. They have a wide range of career features: university, company, job title, job level, year, skills, location, awards, certifications, etc. Thanks to the wealth of the resume dataset, computational career analysis has been developed to help job seekers to get their ideal job opportunities, and companies to find and recruit their fitting workforce [6, 16, 19, 24].

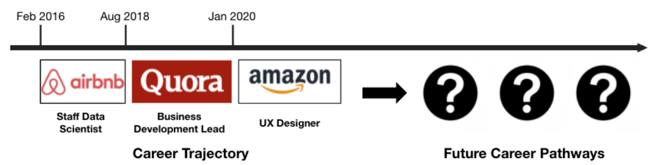

Figure 1: An illustration of career pathway prediction.

However, the problem of predicting *long-term* career movement is still not well explored. As career trajectories and career planning are rapidly diversifying and changing, prior models cannot support job seekers to plan their long-term career pathways because the models only focused on and are trained for predicting one *immediate* next step of career movement. Therefore, we want to develop long-term pathway recommendation system for helping both job seekers and recruiters. In prior studies, since job mobility models predicted only one next job [8, 12, 27], users cannot make their long-term future plans, and job recruiters cannot estimate the long-term potential of each candidate. If recruiters can find job seekers who are not yet valued in the market but have a high potential for future success, they would be able to hire talented people at a lower cost. In this paper, therefore, we propose a unique task, sequential future career pathway prediction, which predicts the next N steps of one's career movement. Figure 1 shows an illustraion of this task.

To achieve accurate career pathway predictions, we emphasize the importance of reasoning instead of just similarity matching for career prediction and job recommendation due to the cognitive reasoning nature of job transitions. Usually, people do not make career decisions just based on job similarity but instead based on careful reasoning and career planning due to the significant effort and cost of making career changes. For example, the best candidate for a Data Analyst (DA) position in FinTech industry would better have *either* Computer Science (CS) *or* Data Science (DS) *and* Business (BIS) *or* Economics (Econ) backgrounds or experiences. Such patterns can be represented as neural logical reasoning formulations [4, 20] such as $(CS \vee DS) \wedge (BIS \vee Econ) \rightarrow DA$, thus, a candidate who has prepared experiences in both (CS or DS) and (BIS or Econ) would have a better chance to take a DA position and be successful in that career. As a result, the ability of reasoning beyond similarity matching is particularly important for career pathway prediction and such reasoning patterns hidden in data can help us to make more accurate career predictions and job recommendations.

[1] https://about.linkedin.com/
[2] https://www.indeed.com/about
[3] https://hiring.careerbuilder.com/resume-search

Considering the above motivations, we propose a model for seque**N**tial career p**A**thway predicti**O**n from **M**ulti-view embedd**I**ngs (NAOMI). First, we use multi-view graph embeddings and BERT embeddings from job titles and companies to represent each job title and company from topological and semantic views. Second, we use job duration weight masking to normalize the job title/company graph embeddings. Intuitively, a user works as a data scientist in a small company in few years can have a higher chance to get a data scientist position in a top-tech company than a user with only few working months. Thus, the job duration can be served as a mask to down weight the user's jobs with less working duration. Third, we adapt neural collaborative reasoning [4] over multi-view embeddings to find reasoning patterns in the representation space for prediction. Then, our model predicts the next N steps of job titles and companies via sequential neural models. We conduct comprehensive experiments on our large-scale real-world dataset with more than 300K resumes. Our results show the effectiveness of NAOMI against state-of-the-art baselines, especially more accurate in prediction as N increases. In short, we make the following contributions in this paper:

- We propose a unique task, sequential future pathway prediction, which predicts the long-term sequential future career pathways for talents.
- To solve the sequential future pathway prediction problem, we propose a novel model NAOMI that employs multi-view graph embeddings and neural collaborative reasoning accounting for *topological* and *semantic* aspects of jobs.
- We conduct extensive experiments and show the effectiveness of NAOMI against baseline models on real-world dataset.

2 RELATED WORK

2.1 Representation Learning from Resume

Representation learning approaches have been popularly developed in multiple domains [3]. In graph representation, we can convert the original graph data into a high-dimensional vector space while preserving the graph structure and node relationship information [3]. Several studies consider graph representation learning in the career domain. Take resume datasets for example, job transition graphs have been frequently used for representation learning. Chen et al. [5] used a person's company transition data as a graph to learn company embeddings, and then they inferred the company similarities. Zhang et al. [26] proposed talent flow representation using the job transition graph for company competitive analysis, wherein they created the attraction vectors from the talent flow network. Zhang et al. [25] also proposed job title representations based on career trajectory. They predicted job link based on the representations, though they only focused on the IT and finance domain. As argued in [6], relational graph convolutional network can generate deep representations for the effects of multiple company relations, and they measured the competition preferences using a real-world enterprise dataset. Luo et al. [13] learned job transition representations based on Random Walk [7]. Nonetheless, all of those representation methods just used the resume data as a graph, but did not leverage the textual information such as job titles. Hence, in this paper, we aim to build a multi-aspect-aware model to improve the performance.

2.2 Job Mobility Prediction

The recent proliferation of online resumes has led to the possibility of predicting the user's next career. To the best of our knowledge, there are only few studies on this topic. Liu et al. [12] is one of the first work on machine learning-based career path prediction. In this work, authors considered career path prediction by exploring multiple social media: Twitter, Facebook, and LinkedIn, and they integrated multiple social network features such as demography, LIWC [21], and user discussion topics for prediction. In this work, authors manually defined the career patterns, however, manually defining the career patterns is time consuming and may not be necessarily realistic. For instance, the ultimate goal of "software developer" is not necessarily "CEO/CTO". Someone might want to be a product manager or data scientist. Furthermore, the above approach cannot predict career paths across different areas. Nowadays, many people change their career into totally different areas such as marketing to software engineer [2, 9]. As a result, advanced techniques are highly needed to make more accurate predictions.

Li et al. [8] proposed a unique problem formulation to predict an employee's next career move, where they developed contextual embedding using an LSTM model from the career trajectories on the LinkedIn dataset. Zhang et al. [27] proposed an enhanced approach to job mobility prediction based on a heterogeneous company-position network from massive career trajectory dataset. They also developed Dual-GRU to model job titles and companies simultaneously. However, both works predicted only the single next job title and the effectiveness of the models for long-term pathway prediction is still unknown. Therefore, we develop models for long-term sequential future pathway predictions against existing methods.

3 PROBLEM FORMULATION

We define the future *career pathway prediction* task as follows:

> Denote a user's job T as a tuple of a job title j and a company name c, i.e. $T = (j, c)$. Given a user's sequence of job trajectory $S = \{T_1, T_2, ..., T_n\}$, where n refers to the length of the job sequence, the goal of the **career pathway prediction** task is to build a function $f(\cdot)$: $\{T_1, T_2, ..., T_n\} \rightarrow \{T_{n+1}, T_{n+2}, ..., T_{n+m}\}$, that inputs the sequence of job trajectory S and predicts next m future jobs for the user.

In this paper, we set the career pathway prediction length $m \in \{2, 3, 4, 5\}$. We do not consider cases of $m > 5$, as most of users in our dataset have 5 historical jobs. Table 1 presents notations that we mainly refer to in this paper.

Table 1: Definition of symbols in our paper.

Symbol	Definition
j	Job title from a user's resume.
c	Company name from a user's resume.
T	A user's job, consisting of a job title and a company name.
$S = \{T_1, T_2, ..., T_n\}$	Job trajectory of a user, with n is the length of the job sequence.
m	Length of the career pathway prediction.
$\boldsymbol{g_j}, \boldsymbol{g_c}$	hyperbolic graph embeddings of job and company, respectively.
$\boldsymbol{b_j}$	BERT embeddings of a job title.

4 THE PROPOSED MODEL: NAOMI

In this section, we present how to build our career pathway prediction model, NAOMI. NAOMI is made of 1) multi-view embeddings, 2) job duration masking, and 3) neural collaborative reasoning.

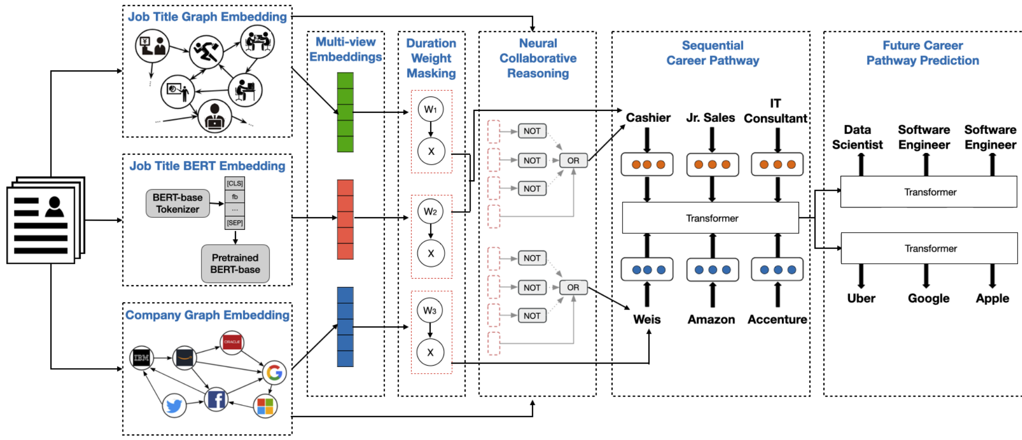

Figure 2: The illustration of the architecture of NAOMI.

4.1 Multi-View Embeddings

4.1.1 Job Title and Company Hyperbolic Graph Embeddings. Given all sequences of job trajectory of all users in the system, we can represent each unique job T as a node, and an edge connects two jobs $T_i \longrightarrow T_k$ if there exists a user who has job T_i as a past job before job T_k. However, such representation can make the job graph extremely sparse, leading to ineffective graph representations. In fact, different jobs can share a same job title, but different company names, i.e. *(data scientist, Google) vs (data scientist, Linkedin)*, or share a same company but different job titles, i.e. *(data scientist, Linkedin) vs (research scientist, Linkedin)*. To further feature and enrich these shared information, we build two separated graphs: (i) job title transition graph, and (ii) company transition graph. From each graph, we learn the graph embeddings for each job title j_i and each company name c_i for each job $T_i = (j_i, c_i)$.

We create a directed and asymmetric job title transition graph $G_j = (V_j, E_j, W_j)$ from all sequences of job trajectory of all users in the system, where V_j is a set of all unique job titles, E_j is a set of directed edges connecting two nodes in the graph, and W_j is a list of edge weights. A directed edge connects two job titles j_i and j_k if there exists a user who moved directly from $T_i = (j_i, c_i)$ to $T_k = (j_k, c_k)$. We set an edge weight $w_{i,k}$ connecting two jobs j_i and j_k as $w_{i,k} = e_{i,k} / \sum_{i=1}^{n} \sum_{k=1}^{n} e_{i,k}$, where $e_{i,k}$ is the number of transitions from node j_i to node j_k.

Next, we learn hyperbolic graph embeddings for all job titles in the job title transition graph G_j. Our goal is to learn hyperbolic representations of graph nodes such that the hyperbolic distances between a target node and its neighbors are minimal, while the hyperbolic distances between the target node and its unrelated/unconnected nodes are maximal. Following [14], we embed the nodes on hyperbolic space adopting Poincare embedding [14] as a hyperbolic embedding, and train Poincare ball model from the relations of nodes. Since the Poincare ball is a Riemannian manifold, the Riemannian metric tensor is represented in the d-dimentional ball $B^d = \{x \in \mathbb{R}^d \big| ||x|| < 1\}$, where $||x||$ is the Euclidean norm. Then, the Riemannian metric tensor r_x is defined as:

$$r_x = \left(\frac{2}{1 - ||x||^2}\right)^2 r^E \tag{1}$$

where $x \in B^d$ and r^E is the Euclidean metric tensor. Then, the distance of two node embeddings $j_i, j_k \in B^d$ is defined as:

$$d(j_i, j_k) = arcosh\left(1 + 2\frac{||j_i - j_k||^2}{(1 - ||j_i||^2)(1 - ||j_k||^2)}\right) \tag{2}$$

We output $\boldsymbol{g_j}$ as hyperbolic graph embeddings for each job title $j \in V_j$. In the same manner, we create company transition graph $G_c = (V_c, E_c, W_c)$ from all sequences of job trajectory of all users in the system. Following the same method for learning job title hyperbolic graph embeddings, we obtain $\boldsymbol{g_c}$ as hyperbolic graph embeddings for each company $c \in V_c$.

4.1.2 Job Title BERT Embedding. Different job titles may refer to a same job position, such as *Data Analyst* vs *Data Scientist*. Thus, we adopt Sentence-BERT [17] which follows the training procedure of DistilBERT [18] with RoBERTa [11] to obtain semantic embeddings for job titles. For the sake of speeding up the model training process, we freeze Sentence-BERT and extract only the embeddings of the [CLS] token as final representations of the input job title. As a result, for each input job title j, we obtain a Sentence-BERT based job title embeddings $\boldsymbol{b_j}$. Note that we only extract semantic embeddings for job titles but not for company names as company names do not necessarily require semantic information. For instance, "apple" is a fruit, but "Apple" company is a high-tech company.

4.2 Job Duration Masking

Intuitively, if the user quit his/her job quickly, indicating that (i) he/she is not interested in either the current job title, or the company itself, and (ii) he/she has not gained much experience on the *early-quit* job. Thus, effect of this job toward the next job should be smaller compared to his/her other jobs with more working years. As such, we measure job duration weight for each user's job duration and implement it as a masking. Our idea is when the job duration is small enough, the job duration weight will be close to *zero* and will *mask out* both job title and company embeddings, letting our model focus more on user's longer experienced jobs. The job duration weight for the job $T_i = (j_i, c_i)$ of user u_p is calculated as follows:

$$w_i^{(p)} = \frac{l}{\sum_{q=1}^{l} t_{u_q, T_i}} t_{u_p, T_i} \tag{3}$$

where $t(u_p, T_i)$ is the T_i's job duration of the user u_p, l is the total number of users. Afterward, we perform *broadcast multiplication*

Table 2: Neural Logical Regularizations.

	Logical Rule	Equation	Neural Logical Regularization.
NOT	Negation	$\neg T = F$	$r_1 = \sum_{i=1}^{n} sim(\boldsymbol{g_j}_i, NOT(\boldsymbol{g_j}_i)) + \sum_{i=1}^{n} sim(\boldsymbol{g_c}_i, NOT(\boldsymbol{g_c}_i))$
	Double Negation	$\neg(\neg \boldsymbol{g}) = \boldsymbol{g}$	$r_2 = \sum_{i=1}^{n} \left(1 - sim(\boldsymbol{g_j}_i, NOT(NOT(\boldsymbol{g_j}_i)))\right) + \sum_{i=1}^{n} \left(1 - sim(\boldsymbol{g_c}_i, NOT(NOT(\boldsymbol{g_c}_i)))\right)$
OR	Identity	$\boldsymbol{g} \vee F = \boldsymbol{g}$	$r_3 = \sum_{i=1}^{n} \left(1 - sim(OR(\boldsymbol{g_j}_i, F), \boldsymbol{g_j}_i)\right) + \sum_{i=1}^{n} \left(1 - sim(OR(\boldsymbol{g_c}_i, F), \boldsymbol{g_c}_i)\right) +$
	Annihilator	$\boldsymbol{g} \vee T = T$	$r_4 = \sum_{i=1}^{n} \left(1 - sim(OR(\boldsymbol{g_j}_i, T), T)\right) + \sum_{i=1}^{n} \left(1 - sim(OR(\boldsymbol{g_c}_i, T), T)\right)$
	Idempotence	$\boldsymbol{g} \vee \boldsymbol{g} = \boldsymbol{g}$	$r_5 = \sum_{i=1}^{n} \left(1 - sim(OR(\boldsymbol{g_j}_i, \boldsymbol{g_j}_i), \boldsymbol{g_j}_i)\right) + \sum_{i=1}^{n} \left(1 - sim(OR(\boldsymbol{g_c}_i, \boldsymbol{g_c}_i), \boldsymbol{g_c}_i)\right)$
	Complementation	$\boldsymbol{g} \vee \neg \boldsymbol{g} = T$	$r_6 = \sum_{i=1}^{n} \left(1 - sim(OR(\boldsymbol{g_j}_i, NOT(\boldsymbol{g_j}_i)), T)\right) + \sum_{i=1}^{n} \left(1 - sim(OR(\boldsymbol{g_c}_i, NOT(\boldsymbol{g_c}_i)), T)\right)$

between $w_i^{(p)}$ and the user u_p's embeddings including $\boldsymbol{g_j}^{(p)}$, $\boldsymbol{g_c}^{(p)}$, and $\boldsymbol{b_j}^{(p)}$, resulting in respective $w_i^{(p)}\boldsymbol{g_j}$, $w_i^{(p)}\boldsymbol{g_c}$, and $w_i^{(p)}\boldsymbol{b_j}$.

4.3 Neural Collaborative Reasoning

Suppose that a user u's job trajectory is $\{T_1, T_2, ..., T_n\}$, where $T_i = (j_i, c_i)$, his/her career trajectory can be naturally represented as a logical reasoning: $(j_1 \vee c_1) \wedge ... \wedge (j_n \vee c_n) \rightarrow (j_{n+1} \vee c_{n+1})$. Thus, we transform our multi-view embeddings into the reasoning space. Specifically we define Horn clauses for job title and company transitions as follows:

$$\begin{aligned} I(u, j_1) \wedge I(u, j_2) \wedge \cdots \wedge I(u, j_n) \rightarrow I(u, j_{n+1}) \\ I(u, c_1) \wedge I(u, c_2) \wedge \cdots \wedge I(u, c_n) \rightarrow I(u, c_{n+1}) \end{aligned} \tag{4}$$

where $I(u, j_i), I(u, c_i)$ are functions for if user u experienced job title j_i and company c_i, i.e. for a personalization purpose. For simpliciy, we use $I(u, j_i) = \boldsymbol{g_j}_i$ and $I(u, c_i) = \boldsymbol{g_c}_i$ in this paper (i.e. graph embeddings of job titles and company names that are learned in Section 4.1.1). Note that it is possible to fuse graph and semantic embeddings of job titles before learning their reasoning embeddings, and we leave this as for our future work. Equation (4) then becomes:

$$\begin{aligned} \boldsymbol{g_j}_1 \wedge \boldsymbol{g_j}_2 \wedge \cdots \wedge \boldsymbol{g_j}_n \rightarrow \boldsymbol{g_j}_{n+1} \\ \boldsymbol{g_c}_1 \wedge \boldsymbol{g_c}_2 \wedge \cdots \wedge \boldsymbol{g_c}_n \rightarrow \boldsymbol{g_c}_{n+1} \end{aligned} \tag{5}$$

Based on De Morgan's Law, we can rewrite Equation (4) using only two basic logical operator *OR* (i.e. $\vee$) and *NOT* (i.e. $\neg$). As such, we can obtain logical reasoning based embeddings for job titles ($\boldsymbol{o_j}$) and company names ($\boldsymbol{o_c}$) by following reasoning designs:

$$\begin{aligned} \boldsymbol{o_j} = (\neg \boldsymbol{g_j}_1 \vee \neg \boldsymbol{g_j}_2 \vee \cdots \vee \neg \boldsymbol{g_j}_n) \vee \boldsymbol{g_j}_{n+1} \\ \boldsymbol{o_c} = (\neg \boldsymbol{g_c}_1 \vee \neg \boldsymbol{g_c}_2 \vee \cdots \vee \neg \boldsymbol{g_c}_n) \vee \boldsymbol{g_c}_{n+1} \end{aligned} \tag{6}$$

In Equation (6), if OR and NOT/NEGATION operators are represented by neural networks, then we can learn $\boldsymbol{o_j}$ and $\boldsymbol{o_c}$ embeddings in an end-to-end neural design. Following [4], we implement OR and NEGATIVE networks as one-layer MLP networks, as well as T (True) and F (False) as two neural embeddings. To explicitly guarantee that these OR and NOT/NEGATION neural modules learn the expected logic operations, we define logical regularizers for job titles and company names and list in Table 2.

In the end, we obtain job title reasoning-based embeddings $\boldsymbol{o_j}$ and company reasoning-based embeddings $\boldsymbol{o_c}$ as outputs. We next perform sequential modelling using these processed embeddings.

4.4 Sequential Prediction

In our career trajectory dataset, we can consider the job on each step as one point, and the consecutive job steps as one sequence. Then, we develop an transformer-based [23] career prediction model from our embeddings. Our inputs are the weighed multi-view embeddings and reasoning embeddings where 1) job title embeddings are built by fusing job title graph embedding and job title BERT embedding (i.e., $w_1\boldsymbol{g_j}$ and $w_2\boldsymbol{b_j}$), 2) company embedding is company graph embedding (i.e., $w_3\boldsymbol{g_c}$), and 3) reaosning embeddings are made of $\boldsymbol{o_j}$ and $\boldsymbol{o_c}$. Based on these embeddings, we use the bi-encoder model to predict the sequential future career pathways. As the outputs, the decoder returns job title pathways or company pathways on the length m.

We use the categorical cross-entropy as the loss function to train NAOMI. The categorical cross-entropy loss function is defined as $L(\theta) = -\sum_{T \in \mathcal{Y}} y_j log(\hat{y}_j) + \sum_{q=1}^{6} r_q$ where θ refers to all the parameters in the entire model and $\sum_{q=1}^{6} r_q$ is the neural logical regularizations defined in Table 2.

5 EXPERIMENTS

5.1 Experimental Settings

Dataset: In this study, we use a real-world and large-scale talent dataset from a career platform, FutureFit AI[4] , which partners with companies and governments globally to help workers navigate career transitions in a world of increasing automation and disruption. The dataset consists of a large number of individual's career trajectories including company, job title, and working duration. From this platform, we randomly selected resumes with at least five valid employment experiences in the United States between 1980 and 2020. To the end, our dataset includes 300K+ resumes and a total of 2M+ of job transition trajectories. All users' privacy information is anonymized. For pre-processing, we remove resumes with no job titles, and remove entities that appear less than 10 times in the dataset as similar to previous works [8].

Baseline: Next career path prediction from career trajectory data is still in a developmental stage, and little is known about this topic. To our best of knowledge, the latest state-of-the-art baselines are NEMO [8] and AHEAD [27], which predict only the single next career. Thus, we use these models repeatedly to predict the sequential future career pathways. Note that although [8] also used the user's skills, school, and location as features, we adjust the conditions of the available features in order to *apple-to-apple* comparisons. In other words, we operate the baseline models using only career trajectory information (i.e., job title, company, and working period). In addition, we also compare our model with several traditional machine learning models (i.e. Logistic Regression, Random Forest, XGBoost, and LSTM) by using them repeatedly to predict the next sequential career pathway.

Evaluation Metrics: We use mean average precision (MAP) to measure the result as MAP is a well-known metric to measure the average exact matching for each prediction result [10, 22].

[4] https://www.futurefit.ai/

Table 3: Mean Average Precision for Job Title Pathways. m indicates the length of predicted career pathway. Last row indicates the *Relative Improvement* (%) of our NAOMI compared to the best baseline.

Method	m=1	m=2	m=3	m=4	m=5
LR	0.0355	0.0354	0.0341	0.0337	0.0314
RF	0.0655	0.0567	0.0487	0.0401	0.0373
XGBoost	0.0734	0.0606	0.0532	0.0487	0.0414
LSTM	0.1746	0.1696	0.1555	0.1332	0.1062
NEMO [8]	0.1894	0.1824	0.1601	0.1329	0.1001
AHEAD [27]	0.2371	0.2179	0.1794	0.1651	0.1485
NAOMI	0.2365	0.2186	0.1893	0.1782	0.1610
Relative Impv (%)	-0.25%	0.32%	**5.52%**	**7.93%**	**8.42%**

Table 4: Mean Average Precision for Company Pathways.

Method	m=1	m=2	m=3	m=4	m=5
LR	0.0690	0.0658	0.0631	0.0607	0.0574
RF	0.1416	0.1367	0.1240	0.1201	0.1183
XGBoost	0.1609	0.1569	0.1437	0.1371	0.1254
LSTM	0.3748	0.3496	0.3269	0.2819	0.2650
NEMO [8]	0.3921	0.3722	0.3629	0.3412	0.3196
AHEAD [27]	0.4007	0.3819	0.3713	0.3414	0.3279
NAOMI	0.4115	0.3949	0.3827	0.3691	0.3417
Relative Impv (%)	**2.70%**	**3.40%**	**3.07%**	**8.11%**	**4.21%**

5.2 Experimental Results

Table 3 and 4 show the performances for sequential future pathway predictions of job title and company, respectively. For both job title and company pathway prediction, we observe that traditional machine learning models do not work as well as NEMO and AHEAD. Among all baselines, AHEAD performs the best.

For the job title pathway prediction, when predicting career pathway with step sizes of m=1 and m=2, NAOMI and AHEAD perform quite similarly. However, when predicting for longer steps (i.e. $m = 3, 4, 5$), NAOMI outperforms all other baselines. For instance, NAOMI significantly improves AHEAD (i.e. the best baseline) by relative margins of 5.52%, 7.93%, and 8.42% for m=3, m=4, and m=5, respectively (*p-value* < 0.05 under the bidirectional chi-squared statistical test). This is obvious, as NAOMI is designed in a sequential modeling manner.

We observe that NAOMI and baselines have higher company pathway prediction results than their performance in the job title pathway prediction. We reason that there are less variations among company names across similar/related jobs (i.e. different job titles but same company names). Overall, NAOMI outperforms all baselines for all step sizes, significantly improving AHEAD by 4.3% relatively on average over all step sizes.

6 CONCLUSION

In this paper, we proposed a new task, future career pathway prediction, which predicts the next N steps of career movements. We presented our model NAOMI, which uses 1) multi-view embeddings, 2) job duration weight masking, and 3) neural collaborative reasoning to solve this task. We conducted comprehensive experiments on our large-scale real-world dataset and our results showed the effectiveness of NAOMI against the state-of-the-art baselines.

ACKNOWLEDGMENTS

The work was in part supported by NSF awards 1934782 and 1909702.

REFERENCES

[1] Michele A Brandão and Mirella M Moro. 2017. Social Professional Networks: A Survey and Taxonomy. *Computer Communications* 100 (2017), 20–31.

[2] Wendy A Campione. 2015. Corporate Offerings: Why Aren't Millennials Staying? *Journal of Applied Business & Economics* 17, 4 (2015).

[3] Fenxiao Chen, Yun-Cheng Wang, Bin Wang, and C-C Jay Kuo. 2020. Graph representation learning: a survey. *APSIPA TSIP* 9 (2020).

[4] Hanxiong Chen, Shaoyun Shi, Yunqi Li, and Yongfeng Zhang. 2021. Neural Collaborative Reasoning. In *Proceedings of the Web Conference 2021*. 1516–1527.

[5] Xi Chen, Yiqun Liu, Liang Zhang, and Krishnaram Kenthapadi. 2018. How LinkedIn Economic Graph Bonds Information and Product: Applications in LinkedIn Salary. In *KDD*. 120–129.

[6] Le Dai, Yu Yin, Chuan Qin, Tong Xu, Xiangnan He, Enhong Chen, and Hui Xiong. 2020. Enterprise Cooperation and Competition Analysis with a Sign-Oriented Preference Network. In *KDD*. 774–782.

[7] Francois Fouss, Alain Pirotte, Jean-Michel Renders, and Marco Saerens. 2007. Random-walk computation of similarities between nodes of a graph with application to collaborative recommendation. *IEEE TKDE* 19, 3 (2007), 355–369.

[8] Liangyue Li, How Jing, Hanghang Tong, Jaewon Yang, Qi He, and Bee-Chung Chen. 2017. Nemo: Next career move prediction with contextual embedding. In *WWW-Companion*. 505–513.

[9] Lei Li, Guangzhi Zheng, Svetlana Peltsverger, and Chi Zhang. 2016. Career trajectory analysis of information technology alumni: A linkedin perspective. In *SIGITE*. 2–6.

[10] Dawen Liang, Jaan Altosaar, Laurent Charlin, and David M Blei. 2016. Factorization meets the item embedding: Regularizing matrix factorization with item co-occurrence. In *RecSys*. 59–66.

[11] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692* (2019).

[12] Ye Liu, Luming Zhang, Liqiang Nie, Yan Yan, and David Rosenblum. 2016. Fortune teller: predicting your career path. In *AAAI*, Vol. 30.

[13] Haiyan Luo, Shichuan Ma, Anand Joseph Bernard Selvaraj, and Yu Sun. 2019. Learning job representation using directed graph embedding. In *DLP-KDD*. 1–5.

[14] Maximilian Nickel and Douwe Kiela. 2017. Poincaré embeddings for learning hierarchical representations. In *NeurIPS*. 6341–6350.

[15] Richard J Oentaryo, Ee-Peng Lim, Xavier Jayaraj Siddarth Ashok, Philips Kokoh Prasetyo, Koon Han Ong, and Zi Quan Lau. 2018. Talent flow analytics in online professional network. *Data Science and Engineering* 3, 3 (2018), 199–220.

[16] Chuan Qin, Hengshu Zhu, Tong Xu, Chen Zhu, Liang Jiang, Enhong Chen, and Hui Xiong. 2018. Enhancing person-job fit for talent recruitment: An ability-aware neural network approach. In *SIGIR*. 25–34.

[17] Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. In *EMNLP*.

[18] Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108* (2019).

[19] Baoxu Shi, Shan Li, Jaewon Yang, Mustafa Emre Kazdagli, and Qi He. 2020. Learning to Ask Screening Questions for Job Postings. In *SIGIR*. 549–558.

[20] Shaoyun Shi, Hanxiong Chen, Weizhi Ma, Jiaxin Mao, Min Zhang, and Yongfeng Zhang. 2020. Neural logic reasoning. In *CIKM*. 1365–1374.

[21] Yla R Tausczik and James W Pennebaker. 2010. The psychological meaning of words: LIWC and computerized text analysis methods. *Journal of Language and Social Psychology* 29, 1 (2010), 24–54.

[22] Thanh Tran, Kyumin Lee, Yiming Liao, and Dongwon Lee. 2018. Regularizing matrix factorization with user and item embeddings for recommendation. In *CIKM*. 687–696.

[23] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *NeurIPS*. 5998–6008.

[24] Tong Xu, Hengshu Zhu, Chen Zhu, Pan Li, and Hui Xiong. 2018. Measuring the popularity of job skills in recruitment market: A multi-criteria approach. In *AAAI*, Vol. 32.

[25] Denghui Zhang, Junming Liu, Hengshu Zhu, Yanchi Liu, Lichen Wang, Pengyang Wang, and Hui Xiong. 2019. Job2Vec: Job title benchmarking with collective multi-view representation learning. In *CIKM*. 2763–2771.

[26] Le Zhang, Tong Xu, Hengshu Zhu, Chuan Qin, Qingxin Meng, Hui Xiong, and Enhong Chen. 2020. Large-Scale Talent Flow Embedding for Company Competitive Analysis. In *TheWebConf*. 2354–2364.

[27] Le Zhang, Ding Zhou, Hengshu Zhu, Tong Xu, Rui Zha, Enhong Chen, and Hui Xiong. 2021. Attentive heterogeneous graph embedding for job mobility prediction. In *KDD*. 2192–2201.