Effective Job-market Mobility Prediction with Attentive Heterogeneous Knowledge Learning and Synergy

Sida Lin*
The Chinese University of Hong Kong, Shenzhen
Shenzhen, China
sidalin1@link.cuhk.edu.cn

Zhouyi Zhang*
The Hong Kong University of Science and Technology (Guangzhou)
Guangzhou, China
zzhang932@connect.hkust-gz.edu.cn

Yankai Chen†
Cornell University
Ithaca, United States
yankaichen@acm.org

Chenhao Ma†
The Chinese University of Hong Kong, Shenzhen
Shenzhen, China
machenhao@cuhk.edu.cn

Yixiang Fang, Shan Dai
The Chinese University of Hong Kong, Shenzhen
Shenzhen, China
{fangyixiang,daishan}@cuhk.edu.cn

Guangli Lu
School of Management and Economics and Shenzhen Finance Institute, The Chinese University of Hong Kong, Shenzhen
Shenzhen, China
luguangli@cuhk.edu.cn

Abstract

Job-market mobility prediction plays a crucial role in optimizing human capital usage for both employees and employers. Most conventional methods primarily focus on learning sequential career sequences while ignoring the sufficient information extraction of mutual entity correlations in the job market. In this work, we push forward to exploit the heterogeneous relational knowledge among the job market structures by proposing a model namely *Attentive Heterogeneous Knowledge Learning and Synergy (AHKLS)*. Equipped with the subsequent module of time-aware perception, AHKLS achieves effective career trajectory encoding for job-market mobility prediction. To evaluate the AHKLS performance, we conduct extensive experiments on three real-world datasets with different sizes. The empirical analyses demonstrate not only the performance superiority of AHKLS over several competing methods, but also the module effectiveness and model compatibility with other methods in enhancing the mobility prediction tasks accordingly.

CCS Concepts

• **Information systems → Recommender systems**; • **Computing methodologies → Supervised learning**; **Neural networks**.

Keywords

Job-Market Mobility Prediction, Heterogeneous Graphs, Knowledge Synergy

ACM Reference Format:
Sida Lin*, Zhouyi Zhang*, Yankai Chen†, Chenhao Ma†, Yixiang Fang, Shan Dai, and Guangli Lu. 2024. Effective Job-market Mobility Prediction with Attentive Heterogeneous Knowledge Learning and Synergy. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management (CIKM '24), October 21–25, 2024, Boise, ID, USA.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3627673.3679906

*Equal Contribution. † Corresponding Authors.

CIKM '24, October 21–25, 2024, Boise, ID, USA.

ACM ISBN 979-8-4007-0436-9/24/10
https://doi.org/10.1145/3627673.3679906

1 Introduction

In the dynamic labor markets, effective mobility prediction and recommendation play an important role in facilitating convenience for both individuals and companies. Traditional studies of job-market mobility prediction primarily focus on discovering the influential factors from a macro perspective of economics [1, 9] and/or a micro view of job market matching state analysis [2]. With the rapid development of recommendation techniques [5, 7, 24], online talent acquisition platforms, such as LinkedIn and Indeed, serve as the key intermediaries to offer unprecedented career opportunities. This motivates the study of data mining and learning from vast amounts of career trajectory data and job information [12, 14, 20, 22].

• **Technical Challenges.** Conventional methods [6, 12, 14] leverage deep learning architectures, e.g., encoder-decoder structures, to integrate and learn personal profiles and/or career trajectories to predict the next career positions. Despite the abundant data usage, these methods may not fully extract the inherent relational knowledge within these data. This is because their learning frameworks usually work on individual-level sequential data. Recent recommender work has introduced Graph Neural Networks (GNNs) [11, 23, 25] to better extract information contained in the talent flow between companies and jobs at a macro-level. For instance, the state-of-the-art model Ahead [22] proposes to leverage an attention-based graph embedding framework; The transformer-based structure is also considered to cooperate with the Dual-GRU module for further enhancement of Ahead [20]. These efforts directly follow the meta-path-based methodology [4, 15] to extract multiple subgraphs for separate learning and subsequent fusion. However, the major concern is that they usually split original heterogeneous graphs into several subgraphs, according to the manually selected meta-paths; this may lead to insufficient structural learning, particularly for large-scale heterogeneous graphs, which thus limits the model capability for real-world labor market analyses.

• **Our Contributions.** To address these issues, we introduce the framework of AHKLS for job-market mobility prediction. Specifically, AHKLS consists of two major designs: (1) *Heterogeneous*

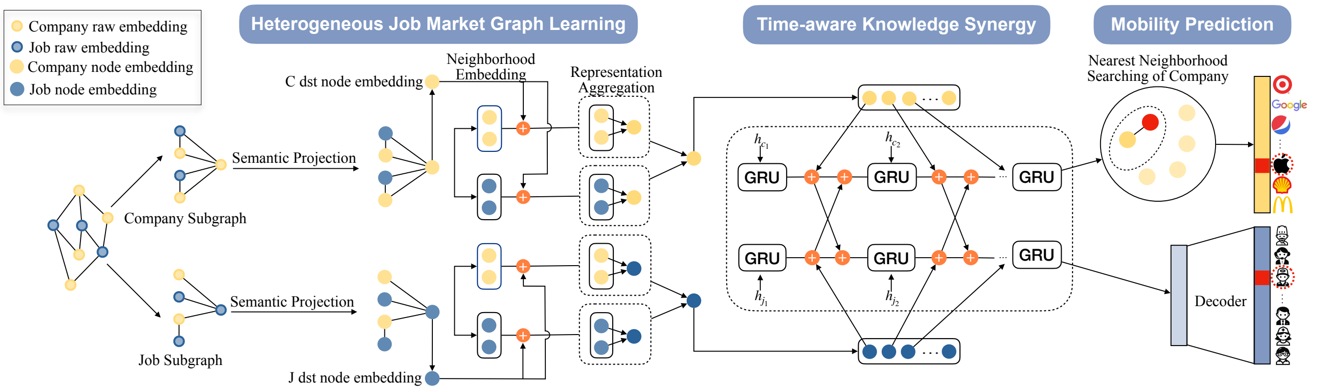

Figure 1: Illustration of our proposed AHKLS framework (best view in color).

Job-Market Graph Learning. We focus on exploiting the semantic learning and unification of heterogeneous nodes. Without exhaustive subgraph partition and ensembling, we achieve attentive knowledge aggregation with rich informativeness. This design eventually enables our model to be easily applicable to large-scale heterogeneous graphs. (2) *Time-aware Knowledge Synergy.* Based on the learned mutual knowledge of market entities, we move forward to encode the career trajectories. We explicitly implement the time information encoding for both company and job transition representations, which are of ease for downstream mobility prediction.

• **Experimental Findings.** To evaluate the model performance, in this concise work, we include three real-world datasets with 1K to 1M trajectory data. The experimental results show that our AHKLS achieves performance superiority over competing methods with 0.30% to 2.99% and 0.78% to 21.20% improvements of the next job and next company prediction, respectively. Furthermore, the extensive empirical studies suggest the effectiveness of each proposed module as well as the compatibility of our method with several existing models in enhancing their prediction capability.

2 Our Methodology

2.1 Problem Formulation

Career trajectory is denoted by an ordered sequence of job and company pairs, i.e., $\{T_1, T_2, ..., T_n\}$, where $T_i = (j_i, c_i)$ with j_i being the i-th job and c_i denoting the affiliated company. Then the problem to study is, given a user u's career trajectory $S_u = \{T_1, T_2, ..., T_{|S_u|}\}$, we aim to construct a predictive function f, which takes the S_u as input and outputs the future potential job and company transitions.

2.2 Heterogeneous Job Market Graph Learning

2.2.1 ***Semantic Projection of Heterogeneous Nodes.*** Since node types have distinct semantics, we first use type-specific transformation matrices to project these input node features into a unified space for computation. Specifically, we use "c" and "j" to distinguish the "company" and "job" type, respectively. Given an edge from the source node x_α to the destination node x_β, where $\alpha, \beta \in \{c, j\}$, we have their initial embeddings as $\boldsymbol{h}^{src}_{x_\alpha} \in \mathbb{R}^{d_\alpha}$ and $\boldsymbol{h}^{dst}_{x_\beta} \in \mathbb{R}^{d_\beta}$. d_α and d_β denote the type-specific embedding dimensions. We first have the following projection processes:

$$\boldsymbol{Q}^{src}_{x_\alpha} = \boldsymbol{W}^{src}_{\mathrm{Q}_\alpha} \cdot \boldsymbol{h}^{src}_{x_\alpha}, \quad \boldsymbol{Q}^{dst}_{x_\beta} = \boldsymbol{W}^{dst}_{\mathrm{Q}_\beta} \cdot \boldsymbol{h}^{dst}_{x_\beta}, \tag{1}$$

$$\boldsymbol{V}^{src}_{x_\alpha} = \boldsymbol{W}^{src}_{\mathrm{V}_\alpha} \cdot \boldsymbol{h}^{src}_{x_\alpha}, \quad \boldsymbol{V}^{dst}_{x_\beta} = \boldsymbol{W}^{dst}_{\mathrm{V}_\beta} \cdot \boldsymbol{h}^{dst}_{x_\beta}, \tag{2}$$

where the transformation matrices are defined as: $\{\boldsymbol{W}^{src}_{Q_\alpha}, \boldsymbol{W}^{src}_{V_\alpha}, \boldsymbol{W}^{dst}_{Q_\alpha}, \boldsymbol{W}^{dst}_{V_\alpha}\} \in \mathbb{R}^{d_{hid} \times d_\alpha}$ and $\{\boldsymbol{W}^{src}_{Q_\beta}, \boldsymbol{W}^{src}_{V_\beta}, \boldsymbol{W}^{dst}_{Q_\beta}, \boldsymbol{W}^{dst}_{V_\beta}\} \in \mathbb{R}^{d_{hid} \times d_\beta}$. Then we integrate the source and destination node embeddings to have the combined representations of destination node x_β as:

$$\boldsymbol{Q}_{x_\beta} = \boldsymbol{Q}^{src}_{x_\alpha} + \boldsymbol{Q}^{dst}_{x_\beta}, \quad \boldsymbol{V}_{x_\beta} = \boldsymbol{V}^{src}_{x_\alpha} + \boldsymbol{V}^{dst}_{x_\beta}. \tag{3}$$

2.2.2 ***Self-attentive Representation Aggregation.*** Based on the destination node, we consider each meta-path ϕ of all necessary meta-paths Φ_β that finally arrive at node x_β. For each meta-path $\phi \in \Phi$, we have its corresponding learnable transformation matrix $\boldsymbol{K}_\phi \in \mathbb{R}^{d_{hid}}$. Then we calculate the normalized attention scores:

$$w^\phi_\beta = \mathrm{Softmax}\left(\boldsymbol{Q}^\phi_{x_\beta} \cdot \boldsymbol{K}_\phi\right), \tag{4}$$

where $\boldsymbol{Q}^\phi_{x_\beta}$ is meta-path ϕ specified representation that follows the processes from Eqn. (1) to Eqn. (3). Then w^ϕ_β balances the contribution of path ϕ's information to aggregate into $\widehat{\boldsymbol{h}}_{x_\beta}$:

$$\widehat{\boldsymbol{h}}_{x_\beta} = \frac{1}{|\Phi|} \sum_{\phi \in \Phi}^{|\Phi|} \left(w^\phi_{x_\beta} \cdot \boldsymbol{V}^\phi_{x_\beta}\right). \tag{5}$$

2.3 Time-aware Knowledge Synergy

2.3.1 ***Career Trajectory Encoding.*** Given the career trajectory $S_u = \{T_1, T_2, ..., T_{|S_u|}\}$, its each element, e.g, $x^{(i)}_\beta \in T_i$, the element embedding is $\boldsymbol{h}_{x^{(i)}_\beta}$. We use β to denote either c or j, generalizing the node type of $\{c, j\}$, as each trajectory element is a tuple of job and company nodes. Given the transformation matrix $\boldsymbol{W}_\beta \in \mathbb{R}^{d_{\mathrm{hid}} \times d_\beta}$, we then introduce the following time-aware encoding:

$$\boldsymbol{l}_{x^{(i)}_\beta} = \boldsymbol{W}_\beta \cdot \boldsymbol{h}_{x^{(i)}_\beta} + \boldsymbol{t}_{\boldsymbol{i}}. \tag{6}$$

$\boldsymbol{t}_{\boldsymbol{i}}$ is the global cosine time embedding of time step i encoded from [18] with the shape of $\mathbb{R}^{d_\beta}$. Subsequently, $\boldsymbol{l}_{x^{(i)}_\beta} \in \mathbb{R}^{d_{hid}}$ synergizes both trajectory element information and time information.

2.3.2 ***Time-aware Mobility Perception.*** To capture the market mobility, we adopt the recent structure Dual-GRU [22]. Furthermore, during its step-by-step forward computation, we additionally leverage the graph information to adjust the hidden state as follows:

$$(\boldsymbol{l}'_{x^{(t)}_c}, \boldsymbol{l}'_{x^{(t)}_j}) = \text{Dual-GRUCell}(\boldsymbol{h}_{x^{(t-1)}_c}, \boldsymbol{h}_{x^{(t-1)}_j}, \boldsymbol{l}''_{x^{(t-1)}_c}, \boldsymbol{l}''_{x^{(t-1)}_j}). \tag{7}$$

In practice, we have the hidden states for both company and job nodes simultaneously. These hidden states, e.g., $\boldsymbol{l}'_{x_\beta^{(t)}}, \boldsymbol{l}''_{x_\beta^{(t-1)}}$ are initialized by zeros and iteratively updated with:

$$\boldsymbol{l}''_{x_\beta^{(t)}} = \boldsymbol{l}'_{x_\beta^{(t)}} + \widehat{\boldsymbol{h}}_{x_\beta^{(t-1)}}. \tag{8}$$

$\widehat{\boldsymbol{h}}_{x_\beta^{(t-1)}}$ is encoded from Eqn. (5) at the t-th step. Let $\boldsymbol{W}_c^{\text{decode}} \in \mathbb{R}^{d_c \times d_{hid}}$ and $\boldsymbol{W}_j^{\text{decode}} \in \mathbb{R}^{d_j \times d_{hid}}$ denote the decoding transformation matrices, we finally output the trajectory representations regarding company and job information, denoted by $\boldsymbol{u}_c$ and $\boldsymbol{u}_j$, as:

$$\boldsymbol{u}_c = \boldsymbol{W}_c^{\text{decode}} \cdot \boldsymbol{l}''_{x_c^{(t)}}, \quad \boldsymbol{u}_j = \boldsymbol{W}_j^{\text{decode}} \cdot \boldsymbol{l}''_{x_j^{(t)}}. \tag{9}$$

2.3.3 Mobility Prediction and Model Optimization. Following the recent work [22], we predict mobility for the next job and the next company, respectively.

- **Next Company Prediction.** We use the learned latest company trajectory embedding, e.g., $\boldsymbol{u}_c$, to perform the nearest neighbor searches over all other company representations, e.g., x'_c. The scoring function is $1 - \frac{\boldsymbol{u}_c \cdot \boldsymbol{h}_{x'_c}^{\top}}{|\boldsymbol{u}_c||\boldsymbol{h}_{x'_c}|}$. The loss term is defined as:

$$\mathcal{L}_1 = \frac{\boldsymbol{u}_c \cdot \boldsymbol{h}_{x_c^-}^{\top}}{|\boldsymbol{u}_c||\boldsymbol{h}_{x_c^-}|} - \frac{\boldsymbol{u}_c \cdot \boldsymbol{h}_{x_c^+}^{\top}}{|\boldsymbol{u}_c||\boldsymbol{h}_{x_c^+}|}, \tag{10}$$

  x_c^+, x_c^- denote positive and negative candidates, respectively.
- **Next Job Prediction.** For future job node prediction, we directly adopt the classification formulation on the job trajectory representation regarding job information, i.e., $\boldsymbol{u}_j$. Let $\boldsymbol{y}_i$ denote the ground-truth one-hot vector of job occupation. Then its loss term is defined with the cross-entropy as follows:

$$\mathcal{L}_2 = -\boldsymbol{y}_j \cdot \log(\boldsymbol{u}_j)^{\top}. \tag{11}$$

Based on $\mathcal{L}_1$ and $\mathcal{L}_2$, we adopt the following function as final objective function to optimize our model:

$$\mathcal{L} = \mathcal{L}_1 + \mathcal{L}_2. \tag{12}$$

3 Experiments

We aim to answer the following research questions:

- RQ1: How does our model perform for Job Mobility Prediction?
- RQ2: How do proposed modules contribute to final performance?
- RQ3: Can our model be compatible with other existing methods?

3.1 Setups

• **Datasets.** We include three real-world datasets in Table 1. For the first dataset Ahead-1K, we directly use it from the state-of-the-art work Ahead [22]. For the others, we collect data from the largest online employment-focused social media platform LinkedIn, and follow the data processing steps outlined in Ahead [22]. To evaluate the model performance under different data sizes, we processed two datasets and named them LK-31K and LK-1M, respectively.

• **Baselines.** We compare AHKLS with several representative models as follows. Firstly, we include two classical sequential models such as LSTM [17] and GRU [8], and modify them for job mobility prediction tasks by denoting them as LSTM+ and GRU+. Then for general heterogeneous graph embedding models, we modify HAN [16] and HGAT [21] to HAN+ and HGAT+ for comparison. For the end-to-end job mobility prediction model, we include the best model Ahead [22] that jointly combines sequential modeling and GNNs to achieve state-of-the-art performance. Notice that we exclude other related methods [3, 8, 10, 12, 14, 19], as Ahead [22] has already presented performance superiority over them.

Table 1: Dataset statistics. # denotes the number; T, J, C denote "Trajectory", "Job", and "Company", respectively. We use the pairwise notation, e.g., C–C, to represent the specific relation.

Name	#T	#J	#C	#C-C	#C-J	#J-J
Ahead-1K	1,000	2,099	1,381	95,230	57,128	121,522
LK-31K	31,217	2,076	1,385	5,497	23,926	90,388
LK-1M	1,010,501	20,080	20,101	928,181	1,281,200	828,267

• **Evaluation Metrics.** In this work, we randomly split the data for training and testing with a ratio of 8:2. After five times of experiments, we use the averaged accuracy (ACC) as the primary evaluation metric. We respectively assess the model prediction regarding the target companies and positions and report results of ACC@1, ACC@15, and ACC@30 accordingly.

• **Experiment Configurations.** We implement with Python 3.8 and PyTorch 1.13.1 with non-distributed training. The experiments are run on a Linux machine with 1 NVIDIA A100 GPU and 6 Intel(R) Xeon(R) Platinum 8350C CPU with 2.60GHz. We adhere to all baselines' officially reported hyper-parameter settings and conduct a grid search to models without prescribed configurations. For a fair comparison, we fix the embedding dimension at 512. The learning rate is tuned in the range $\{10^{-4}, 10^{-3}, 10^{-2}\}$. Optimization for all models is performed using the default AdamW optimizer [13].

3.2 Overall Performance (RQ1)

Based on the prediction target, we respectively report the results in Tables 2 and 3. On one hand, for the next job prediction, as shown in Table 2, we have twofold observations. (1) For different datasets, the classical sequential models, ie., LSTM+ and GRU+, present surprisingly good performance for the task, indicating their capability in capturing the knowledge of job mobility in sequential data. In addition, the state-of-the-art job mobility prediction model Ahead runs out of computational memory for LK-1M dataset, showing its limitation of practical usage for large-scale data. (2) While for our model, AHKLS consistently achieves superior performance across all datasets over competitive methods with a range of 0.3% to 21.2% performance gain. On the other hand, for the next company prediction, we notice that the performance of all included competing methods is more variance on different datasets, where Ahead and HAN+ are two representative examples. And our model AHKLS is also competitive on this task, mainly thanks to the design of joint learning both sequential job transition knowledge and heterogeneous relations in the graph format among all these entities.

3.3 Ablation Study (RQ2)

To study the empirical effect of our proposed modules, we conduct ablation study on the LK-31K dataset and report the results in Table 4. Specifically, we respectively disable the design of heterogeneous relational knowledge learning, denoted by "w/o HRK", and hidden-state sharing with dual GRU, denoted by "w/o HS". We observe that, the variant of w/o HRK leads to a dramatic performance decay ranging from 3.30% to 26.31%, showing the importance of learning heterogeneous relational knowledge to the model. Furthermore, the performance degradation of variant w/o HS, i.e., from 0.03% to 4.61%, also demonstrates the effectiveness of our introduced

Table 2: Performance comparison across different datasets on "Next Job Prediction". (1) We use the bold and the underline to denote the best and second-best performance. (2) "OOM" means the model runs out of memory.

Dataset	Ahead-1K			LK-31K			LK-1M		
Metrics	ACC@1	ACC@15	ACC@30	ACC@1	ACC@15	ACC@30	ACC@1	ACC@15	ACC@30
LSTM+	0.1233 ± 0.0137	0.2850 ± 0.0264	**0.3518** ± 0.0290	0.0995 ± 0.0008	0.3631 ± 0.0040	0.4479 ± 0.0033	0.1353 ± 0.0004	0.3268 ± 0.0008	0.3842 ± 0.0006
GRU+	0.1225 ± 0.0137	0.2741 ± 0.0363	0.3276 ± 0.0381	0.0984 ± 0.0018	0.3592 ± 0.0039	0.4428 ± 0.0050	0.1372 ± 0.0007	0.3324± 0.0009	0.3890 ± 0.0010
HAN+	0.0427 ± 0.0093	0.2122 ± 0.0132	0.3161 ± 0.0240	0.0956 ± 0.0008	0.3640 ± 0.0046	0.4501 ± 0.0061	0.1350 ± 0.0054	0.3592 ± 0.0021	0.4179 ± 0.0063
HGAT+	0.1108 ± 0.0141	0.2926 ± 0.0080	0.3419 ± 0.0134	0.0480 ± 0.0024	0.2554 ± 0.0021	0.3311 ± 0.0046	OOM	OOM	OOM
Ahead	0.0893 ± 0.0072	0.2479 ± 0.0055	0.3241 ± 0.0220	0.0893 ± 0.0010	0.3444 ± 0.0036	0.4222 ± 0.0040	OOM	OOM	OOM
AHKLS	**0.1255** ± 0.0123	**0.2996** ± 0.0230	0.3482 ± 0.0348	**0.0998** ± 0.0008	**0.3694** ± 0.0036	**0.4562** ± 0.0061	**0.1413** ± 0.0007	**0.3645** ± 0.0011	**0.4289** ± 0.0009
Gain	1.78%	2.39%	-1.02%	0.30%	1.48%	1.36%	2.99%	1.48%	2.63%

Table 3: Performance comparison across different datasets on "Next Company Prediction".

Dataset	Ahead-1K			LK-31K			LK-1M		
Metrics	ACC@1	ACC@15	ACC@30	ACC@1	ACC@15	ACC@30	ACC@1	ACC@15	ACC@30
LSTM+	0.2824 ± 0.0157	0.3729 ± 0.0132	0.3942 ± 0.0097	0.1478 ± 0.0062	0.3037 ± 0.0087	0.3693 ± 0.0076	0.2070 ± 0.0003	0.3182 ± 0.0003	0.3540 ± 0.0025
GRU+	0.2790 ± 0.0234	0.3672 ± 0.0206	0.3813 ± 0.0203	0.1391 ± 0.0055	0.3033 ± 0.0076	0.3695 ± 0.0076	0.2112 ± 0.0010	0.3224 ± 0.0010	0.3556 ± 0.0025
HAN+	0.2606 ± 0.0161	0.3919 ± 0.0214	0.4178 ± 0.0451	0.1336 ± 0.0067	0.1818 ± 0.0089	0.2681 ± 0.0069	0.2086 ± 0.0081	0.2236 ± 0.0038	0.3015 ± 0.0011
HGAT+	0.2872 ± 0.0042	0.3654 ± 0.0051	0.3818 ± 0.0044	0.0244 ± 0.0018	0.1316 ± 0.0060	0.1893 ± 0.0095	OOM	OOM	OOM
Ahead	0.2826 ± 0.0128	0.3959 ± 0.0287	0.4499 ± 0.0145	0.1352 ± 0.0048	0.2932 ± 0.0091	0.3564 ± 0.0087	OOM	OOM	OOM
AHKLS	**0.3036** ± 0.0191	**0.4008** ± 0.0075	**0.4534** ± 0.0091	**0.1581** ± 0.0066	**0.3164** ± 0.0063	**0.3832** ± 0.0057	**0.2468** ± 0.0030	**0.3892** ± 0.0042	**0.4310** ± 0.0045
Gain	5.71%	1.24%	0.78%	6.97%	4.18%	3.71%	16.86%	20.72%	21.20%

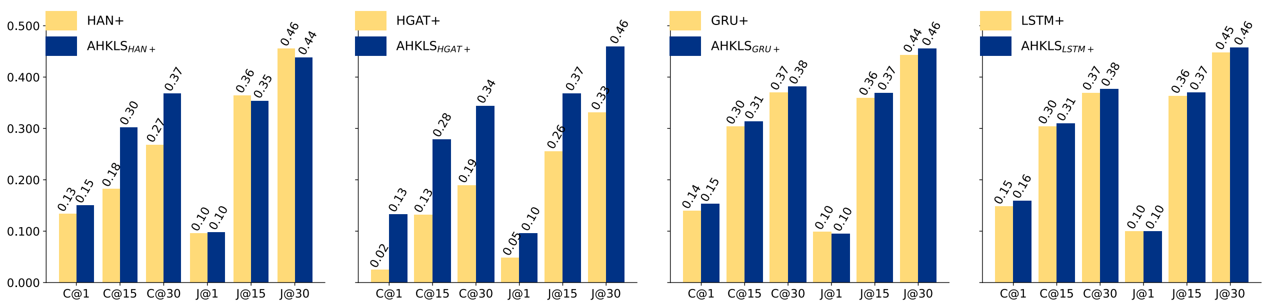

Figure 2: Compatibility study of integrating AHKLS into the general learning framework for job market mobility prediction.

Table 4: Ablation study results on LK-31K dataset.

Design	Job			Company		
	ACC@1	ACC@15	ACC@30	ACC@1	ACC@15	ACC@30
w/o HRK	0.0961	0.3572	0.4402	0.1165	0.2877	0.3582
	-3.71%	-3.30%	-3.52%	-26.31%	-8.56%	-6.53%
w/o HS	0.0952	0.3693	0.4558	0.1529	0.3135	0.3811
	-4.61%	-0.03%	-0.09%	-3.29%	-0.35%	-0.55%
AHKLS	**0.0998**	**0.3694**	**0.4562**	**0.1581**	**0.3146**	**0.3832**

technical designs of sharing hidden-states, especially for the Top-1 metrics of ACC@1 in both Job and Company Mobility Prediction.

3.4 Compatibility Study (RQ3)

In addition to studying the internal modules, we also explore how our model can be adapted to enhance existing models. Specifically, we integrate our unique designs with existing models, either supplementing or replacing components as necessary. For example, in the case of HAN+, we replace its heterogeneous graph learning with our relational knowledge synergy design and other elements in AHKLS. We denote such variant as AHKLS$_{HAN+}$ and visually compare it with the original HAN+. As shown in Figure 2, the comparison across five model modifications demonstrates that our methodology is highly compatible with these models. More importantly, such integration significantly boosts their original performance.

4 Conclusion

In this paper, we introduce AHKLS for the job market mobility prediction tasks. AHKLS is equipped with designs of specific knowledge learning and synergy on heterogeneous graphs. The extensive experiments on three real-world datasets demonstrate the performance superiority of AHKLS over several competing methods and the module effectiveness and compatibility with other methods.

Acknowledgement

This work was partially supported by NSFC under Grant 62302421, Basic and Applied Basic Research Fund in Guangdong Province under Grant 2023A1515011280, Shenzhen Research Institute of Big Data under grant SIF20240004, and the Guangdong Provincial Key Laboratory of Big Data Computing, The Chinese University of Hong Kong, Shenzhen.

References

[1] Regis Barnichon and Andrew Figura. 2015. Labor market heterogeneity and the aggregate matching function. *American Economic Journal: Macroeconomics* 7, 4 (2015), 222–249.

[2] Manudeep Bhuller, Andreas Kostol, and Trond Vigtel. 2019. How broadband internet affects labor market matching. *Available at SSRN 3507360* (2019).

[3] Leo Breiman. 2001. Random Forests. *Machine Learning* 45, 1 (2001), 5–32. https://doi.org/10.1023/A:1010933404324

[4] Yankai Chen, Yixiang Fang, Qiongyan Wang, Xin Cao, and Irwin King. 2024. Deep Structural Knowledge Exploitation and Synergy for Estimating Node Importance Value on Heterogeneous Information Networks. In *AAAI*, Vol. 38. 8302–8310.

[5] Yankai Chen, Quoc-Tuan Truong, Xin Shen, Jin Li, and Irwin King. 2024. Shopping Trajectory Representation Learning with Pre-training for E-commerce Customer Understanding and Recommendation. *Proceedings of the 30th SIGKDD Conference on Knowledge Discovery and Data Mining* (2024).

[6] Yankai Chen, Quoc-Tuan Truong, Xin Shen, Ming Wang, Jin Li, Jim Chan, and Irwin King. 2023. Topological Representation Learning for E-commerce Shopping Behaviors. *Proceedings of the 19th International Workshop on Mining and Learning with Graphs.*

[7] Yankai Chen, Yifei Zhang, Menglin Yang, Zixing Song, Chen Ma, and Irwin King. 2023. WSFE: wasserstein sub-graph feature encoder for effective user segmentation in collaborative filtering. In *SIGIR*. 2521–2525.

[8] Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using RNN encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078* (2014).

[9] James S Coleman. 1991. Matching processes in the labor market. *Acta sociologica* 34, 1 (1991), 3–12.

[10] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. 2009. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction.* Springer Science & Business Media.

[11] Xiangnan He, Kuan Deng, Xiang Wang, Yan Li, Yongdong Zhang, and Meng Wang. 2020. Lightgcn: Simplifying and powering graph convolution network for recommendation. In *SIGIR*. 639–648.

[12] Liangyue Li, How Jing, Hanghang Tong, Jaewon Yang, Qi He, and Bee-Chung Chen. 2017. NEMO: Next career move prediction with contextual embedding. In *Proceedings of the 26th International Conference on World Wide Web Companion*. 505–513.

[13] Ilya Loshchilov and Frank Hutter. 2019. Decoupled Weight Decay Regularization. In *Proceedings of the 7th International Conference on Learning Representations.* International Conference on Learning Representations, New Orleans, LA, USA.

[14] Qingxin Meng, Hengshu Zhu, Keli Xiao, Le Zhang, and Hui Xiong. 2019. A hierarchical career-path-aware neural network for job mobility prediction. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*. 14–24.

[15] Yizhou Sun, Jiawei Han, Xifeng Yan, Philip S Yu, and Tianyi Wu. 2011. Pathsim: Meta path-based top-k similarity search in heterogeneous information networks. *Proceedings of the VLDB Endowment* 4, 11 (2011), 992–1003.

[16] Xiao Wang, Houye Ji, Chuan Shi, Bai Wang, Yanfang Ye, Peng Cui, and Philip S Yu. 2019. Heterogeneous Graph Attention Network. In *The World Wide Web Conference*. ACM, New York, NY, USA. https://doi.org/10.1145/3308558.3313562

[17] Svante Wold, Kim Esbensen, and Paul Geladi. 1987. Principal component analysis. *Chemometrics and intelligent laboratory systems* 2, 1-3 (1987), 37–52.

[18] Da Xu, Chuanwei Ruan, Evren Korpeoglu, Sushant Kumar, and Kannan Achan. 2020. Inductive Representation Learning on Temporal Graphs. In *International Conference on Learning Representations.*

[19] Huang Xu, Zhiwen Yu, Jingyuan Yang, Hui Xiong, and Hengshu Zhu. 2016. Talent Circle Detection in Job Transition Networks. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. Association for Computing Machinery, San Francisco, CA, USA, 655–664. https://doi.org/10.1145/2939672.2939732

[20] Michiharu Yamashita, Yunqi Li, Thanh Tran, Yongfeng Zhang, and Dongwon Lee. 2022. Looking further into the future: Career pathway prediction. *WSDM Computational Jobs Marketplace 2022* (2022).

[21] Tianchi Yang, Linmei Hu, Chuan Shi, Houye Ji, Xiaoli Li, and Liqiang Nie. 2021. HGAT: Heterogeneous graph attention networks for semi-supervised short text classification. *ACM Transactions on Information Systems (TOIS)* 39, 3 (2021), 1–29.

[22] 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 *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery amp; Data Mining*. ACM, New York, NY, USA. https://doi.org/10.1145/3447548.3467388

[23] Xinni Zhang, Yankai Chen, Cuiyun Gao, Qing Liao, Shenglin Zhao, and Irwin King. 2022. Knowledge-aware Neural Networks with Personalized Feature Referencing for Cold-start Recommendation. *arXiv preprint arXiv:2209.13973* (2022).

[24] Xinni Zhang, Yankai Chen, Chenhao Ma, Yixiang Fang, and Irwin King. 2024. Influential Exemplar Replay for Incremental Learning in Recommender Systems. In *AAAI*, Vol. 38. 9368–9376.

[25] Yifei Zhang, Hao Zhu, Yankai Chen, Zixing Song, Piotr Koniusz, and Irwin King. 2023. Mitigating the popularity bias of graph collaborative filtering: a dimensional collapse perspective. In *NeurIPS*. 67533–67550.