University Evaluation through Graduate Employment Prediction: An Influence based Graph Autoencoder Approach

Yuyang Ye, Hengshu Zhu, *Senior Member, IEEE*, Tianyi Cui, Runlong Yu, Le Zhang, Hui Xiong, *Fellow, IEEE*

Abstract—It is always challenging task for students to select right universities. For students, graduate job placement is the most important component of university quality. However, existing university evaluation methods predominantly depend on either subjective criteria, such as the perceived quality of the learning environment and academic prestige, or on factors like faculty excellence, which may not provide a precise indication of graduate job placement. Indeed, there is still a lack of a data-driven approach to accurately measure university quality based on the employment situation of graduates. Moreover, the inherently unsupervised nature of university evaluation, compounded by the absence of a reasonable ground truth, necessitates the development of a reliable supervised methodology to precisely quantify university quality. Our basic assumption is that highly influential companies would attract graduates from high-ranking universities. To this end, in this paper, we formulate university evaluation problem into the graduate flow prediction problem, and propose an Influence based Graph Autoencoder (IGAE) method to learn the representation of universities based on the employment of their graduates. Specifically, we first build a talent transition graph based on the massive resume information. This graph reveals the flow of talent between institutions. Then, considering the asymmetric and heterogeneous properties of talent flow, an unidirectional aggregation process with a heterogeneous attention mechanism is designed to encode the nodes in the directed graph and preserve the influence terms at the same time. Afterwards, a novel dual self-attention module is exploited to capture the dynamic pattern of institutions to forecast future employment. Furthermore, we design an influence based decoder to predict the existence of talent flows and estimate the frequency of employment, which can be learnt in a joint learning framework. Finally, we conduct extensive experiments on a real-world dataset for performance evaluation. The experimental results clearly validate the effectiveness of our approach compared to the state-of-the-art baselines, and we provide a case study on university influence analysis.

Index Terms—Graph Mining, Network Analysis, University Evaluation, Human Resource Management

1 Introduction

Nowadays, millions of high school seniors face big decisions about enrolling in universities every year [1]. Indeed, it is always a challenging task for students to select the right universities. For students, graduate job placement is the most important component of university quality. For example, a recent survey on freshmen in universities [2] reported that more than 83% of freshmen believe that they can get a better job after graduation, which is the most common reason for them to decide to attend college, but rarely attracts enough attention. Existing university evaluation methods either heavily rely on arbitrarily assessed subjective factors, such as learning environment [3] [4], academic reputation [5], and faculty quality [6], or depend on the arbitrarily assigned combination of them [7], which may not accurately reflect graduate job placement. Indeed, there remains an absence of a data-driven approach to precisely gauge university quality based on the employment trajectories of its graduates.

In the literature, the swift expansion of online professional profile repositories has led to the production of a vast number of digital resumes. This proliferation of data offers an unprecedented resource for scholars, allowing a deeper dive into the educational backgrounds and career trajectories of professionals across institutions. Simultaneously, it paves the way for a data-driven approach to assess universities based on the career progression of their alumni. Although significant efforts have been made in talent trajectory research, including career path prediction [8], [9] and talent flow analysis [10], [11], leveraging sophisticated machine learning meth-

- *Y. Ye is with the Management Science and Information Systems Department, Rutgers University, Newark, NJ 07102, USA. E-mail: yuyang.ye@rutgers.edu.*
- *H. Zhu (corresponding author) is with Career Science Lab, BOSS Zhipin, Beijing, 100013, China. E-mail: zhuhengshu@gmail.com.*
- *T. Cui is with the Department of Network Engineering, Bytedance Inc., Seattle, WA 98004, USA. E-mail: 1997cui@gmail.com.*
- *R. Yu is with the Department of Computer Science, University of Pittsburgh, Pittsburgh, PA 15260, USA. E-mail: ruy59@pitt.edu.*
- *L. Zhang is with Baidu Research, Baidu Inc, Beijing, 100013, China. E-mail: zhangle0202@gmail.com.*
- *H. Xiong (corresponding author) is with the Artificial Intelligence Thrust, Hong Kong University of Science and Technology (Guangzhou), GuangZhou, 510000, China. E-mail: xionghui@ust.hk.*

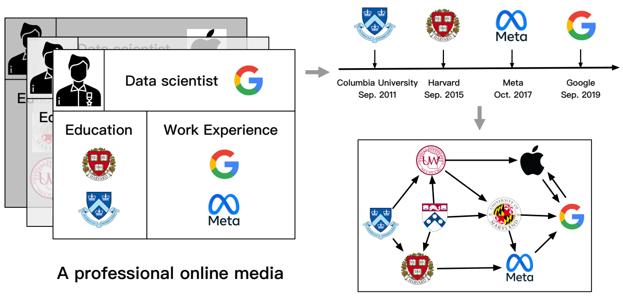


Fig. 1: An illustration of the talent flow graph construction from online professional social network profiles.

ods such as graph mining that adeptly discern patterns in career trajectories, the task of university assessment presents some challenges. First, it is challenging to integrate graduates' employment outcomes with their educational backgrounds. Meanwhile, the absence of dependable ground-truth data makes a quantitative appraisal of universities formidable. Moreover, prevalent graph learning techniques often struggle when faced with the task of modeling unidirectional propagation in expansive talent flow datasets. Additionally, the role and prominence of educational and corporate institutions in shaping career paths are dynamic, necessitating a temporal dependencies modeling to furnish forward-thinking insights. In light of these challenges, it is necessary to develop innovative methodologies that adeptly leverage graduate employment to evaluate universities.

Aligning with these challenges, this paper introduces the Influence based Graph Autoencoder (IGAE) method to evaluate universities via learning representations grounded on the employment trajectories of their alumni. Based on the fundamental hypothesis that highly influential corporations naturally gravitate towards graduates from prestigious universities, our problem can be addressed through a supervised learning paradigm. Specifically, to model the employment circumstances of graduates, illustrated in Figure 1, we initially construct a dynamic heterogeneous university-company transition graph that charts multiple extracted talent career paths based on massive resume information, which reveals talent flows between institutions. Then, for dealing with the integration of career employment with educational background, we propose an influence-based graph encoder. Here, a novel attention mechanism is proposed to learn node embedding by cohesively aggregating insights from heterogeneous neighbor nodes, ensuring the integration of employment and educational facets. Afterwards, for prospective insights, a dual self-attention based temporal dependencies modeling module is exploited for capturing the dynamic pattern of nodes to forecast the future employment. Importantly, our proposed encoder sustains two influence terms inspired by the Hyperlink-Induced Topic Search (HITS) algorithm, revealing the asymmetric of talent flows. Next, we design an influence based graph decoder to predict predict emergent talent flows and gauge imminent employment trends with a link prediction task in our proposed dynamic graph. In particular, our proposed influence terms can address the asymmetric property of directed transition graphs and offer a robust insight to analyze the overarching impact of universities. The main contributions of this paper can be summarized as follows.

- We study the problem of university evaluation by exploring their influence based on the career employment of graduates, which fills the void in existing research which predominantly relies on subjective factors, and provides a data-driven alternative to the extant literature.
- We propose a delicately-designed Influence based Graph Autoencoder (IGAE), which incorporates a heterogeneous information aggregation module and a dual self-attention module to capture the dynamic influence of nodes revealed in the heterogeneous talent transition graph by introducing HITS-inspired influence terms and simultaneously address the challenge presented by the asymmetric property of influence propagation during the graph reconstruction process.
- We conducted extensive experiments on a real-world dataset for performance evaluation. The experimental results clearly validate the effectiveness of our approach compared to the state-of-the-art baselines. Furthermore, a detailed case study on university influence analysis further elucidates the practical application and potential of our method.

2 Related Work

In this section, we will introduce the related work of our paper, which can be divided into two categories, namely network embedding and university evaluation.

2.1 Network Embedding

Network embedding, also known as network representation learning, is a rapidly evolving field that aims to map nodes in a network to low-dimensional, continuous, and dense representations, which can be utilized for a variety of applications including link prediction, node classification, and community detection. These applications are common tasks in talent analysis [12], [13], [14].

One of the first network embedding methods is DeepWalk [15], which learns node representations using short random walks and the skip-gram approach. Node2vec [16] expanded on this by offering flexible random walk sampling. Then, models like LINE [17] focused on preserving the direct and indirect relationships between nodes. Similarly, SDNE [18] used a deep autoencoder to maintain these relationships, and GraRep [19] learns node representations with preserving the k-step proximity between nodes.

Recently, with the development of deep learning techniques, there are also some methods that utilize graph neural networks (GNNs) architectures to learn network representation, such as GCN [20], GraphSAGE [21] and

GAT [22]. Generally speaking, the GNN framework is a message passing and aggregation process between nodes, and the pattern of the graph is captured with the guidance of the given label. Additionally, graph auto-encoders (GAEs) [23] are proposed to reconstruct graphs with the encoding information of its nodes, which were learned by GNN, to enhance the robustness of the node embedding. Since the above methods are not proposed to work with undirected graphs, some research has focused on directed graph neural networks (DiGNNs). For example, Gravity [24] extended GAE by proposing a novel gravity-inspired decoder to construct the directed graph. DiGCN [25] proposed a PageRank based digraph convolution mechanism with preserving kth-order proximity to achieve larger receptive fields and learn higher performance node representation in digraphs.

However, in real-world scenarios, complex networks are generally dynamic and heterogeneous, while above-mentioned methods are all proposed for homogeneous graphs. Thus, there has been a rising trend in the investigation of heterogeneous and dynamic network embedding methods. For example, HetGNN [9] aggregates content embeddings from different neighboring groups, which have different relation types, using bi-LSTM blocks and further combines different types of nodes while considering the contribution of different groups. HAN [26] focuses on neighbors in meta-paths in heterogeneous graph, which respectively designed a node-level GAT and a semantic-level GAT to capture the proximity of node with different types in the information aggregation process. In addition, methods such as HGAT [27] and HGSL [28] address various types of interaction by introducing additional graph structures. Furthermore, several neural network based approaches have been developed specifically for dynamic networks [29], [30]. Most of these methods focus on modeling snapshots derived from dynamic graphs. Besides, there are also some Spatial-temporal graph neural networks excel in capturing spatial and temporal dependencies in stable graph [31], which are especially adept at modeling systems like traffic networks [32], [33] and human activity [34]. Given the inherent characteristics of talent flows, this paper introduces an influence based heterogeneous dynamic graph autoencoder framework, which is designed to model the talent transition between universities and companies while highlighting the influence of institutions.

2.2 University Evaluation

University evaluation is a crucial task in education research, with a goal of accurately assessing the performance and quality of universities, to provide valuable information and recommendations [35], [36].

The commonly used university evaluation approaches are based on a comprehensive assessment of different factors [7], such as academic results, teaching quality, learning facilities and volumes. Most of them rely on the proposal of a weighted combination of these factors [3] [4], where the scores and weights of factors are usually arbitrarily assigned by education experts based on some data, such as some university ranking list (e.g., US News [37] and QS [38]). Moreover, [39] proposed a method to rank universities based on alignment of public ranking lists. Also, there are some other methods that focus only on one single factor, such as academic results [5] and reputation among peers [6], and deep analyzing it with surveys and statistical approaches.

However, the above-mentioned approaches suffer from limitations that highly rely on subjective assessments and opinions, resulting in a lack of statistical robustness. Therefore, many other research tried to utilize some statistical methods and machine learning models to evaluate universities by predicting their student outcomes such as graduation and dropout. For example, [40] used logistic regression to predict student dropout in a large urban school district. Similarly, [41] used decision trees to predict student persistence in a state university. Besides, [42] described how data mining techniques such as decision trees, neural networks, and Bayesian networks have been used to predict student outcomes. [43] proposed a context-aware matrix factorization approach to predict student academic performance based on exploring the history of book loan. [44] proposed a deep learning algorithm based on temporal modeling to predict the drop out probability of the at-risk student.

Nevertheless, we found that there still lack of works related to assess universities based on their graduates employment orientation, which is a common selection criteria in the student enrollment process. Inspired by [45], which incorporate several statistical methods to evaluate universities by exploring their graduates' careers on the talent graph, in this paper, we propose a novel approach to evaluate universities based on the career outcomes of their graduates, which is designed to capture the relationships between universities and companies via talent flow while modeling the influence of universities with a graph structure.

3 Preliminary

In this section, we will introduce the details and some prestudies of the real-world datasets we use, and then formally give the definition of our research problem.

3.1 Data Description

The data used in this paper were collected from a well-known professional website, i.e., LinkedIn [1], where millions of users around the world build resumes to share information about their career. To be specific, the resume of each user consists of education and working experience where each education record shows university, major, degree and studying duration, while working records indicate company name, job position and working duration. Meanwhile, we also collect some attributes of companies and universities, such as their size, type, location, and text description. We would further introduce some details of our dataset in the section of the experiment.

1. https://www.linkedin.com/

3.2 Data Exploration

Next we conduct several data explorations to preliminarily validate our assumptions.

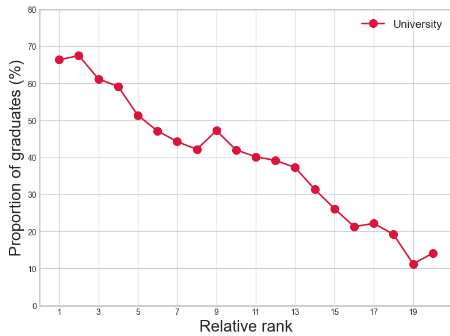
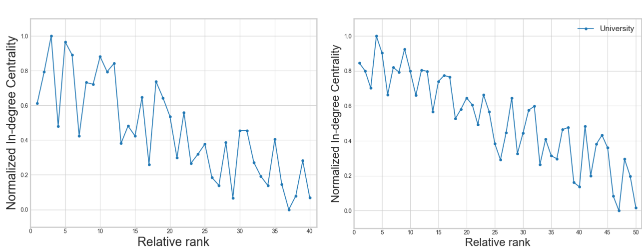


Fig. 2: The relationship between proportion of graduates entering top companies and university ranking.

The first data exploration is designed to analyze the relationship between education quality and job opportunities. We list the top 40 IT companies in the U.S. based on a public list released by Forbes[2] and randomly selected 20 universities whose ranks vary a lot in public university rank [38], and draw a diagram to depict the proportion of graduates who enter these top companies from IT major of these ten universities as Figure 2. Through this task, we can clearly propose our statement that high-quality education would provide better opportunities for their graduates to get good jobs.

In our secondary data exploration, we operate under the assumption that companies with greater influence tend to attract more talent. Initially, we extracted a talent transition graph encompassing the top 40 companies identified earlier, within a given period. Then we compute the normalized PageRank value [46] as an indicator of its influence. Subsequently, the Pearson's correlation coefficient [47] (PCC) is used to gauge the relationship between PageRank and company rankings, shown in Figure 3a. The PCC score is 0.7881 with a p-value of 0.03, which validates our assumption that talent flows between companies could reveal the influence of companies.



(a) Company (b) University

Fig. 3: The Relation between relative ranking and normalized PageRank value in talent transition graph.

Similarly, in the third task, we analyze the relationship between the PageRank value and the university rank within the university transition graph, which is made up of 50 randomly selected universities in the USA as Figure 3b. A PCC score of 0.8615 with 0.01 p-value demonstrates a robust correlation, which is consistent with our common understanding that students are more willing to pursue high degree in better university.

2. https://www.forbes.com/top-digital-companies/

3.3 Problem Definition

In this section, we initially propose the concept of the university-company transition graph. Then we provide a clear definition of our university evaluation problem.

As previously mentioned, our framework aims to understand the employment landscape of universities by investigating the talent flow patterns of their alumni. Essential to achieving this goal is the concurrent assessment of the influence of the companies that hire these graduates. Our basic assumption is that high-influence institutions inherently serve as magnets for top-tier talent. Following the result of the data exploration, we build some graph structures to deal with this influence modeling problem. It is imperative to inform that the influence of a university or a company is usually fast evolving in the real world, which could change a lot over time. For example, a popular AI related company was probably little known 10 years ago. Thus, modelling the influence is the issue with temporal dependency, which should be considered as a dynamic graph modeling problem. Here, we design a dynamic heterogeneous university company transition graph and give its definition as follows.

Definition 1 (University-Company Transition Graph). The University-Company Transition Graph is defined as $G = (V, E, W, X)$, where $V = (V_U, V_C)$ is the set of nodes, V_U and V_C respectively denote the set of University and Company, $E = (E_{UU}, E_{CC}, E_{UC})$ is the set of edges, E_{UU}, E_{CC} respectively denote the talent transition between different universities and companies while E_{UC} denote the talent flow graduated from university to company, W is the frequency of transition records E, and X is the feature matrix. To capture the evolving influence of each company, we construct a dynamic company transition graph, represented by snapshots on T timestamps, denoted as $\mathcal{G} = \{G_1, \cdots, G_T\}$.

As emphasized in previous sections, our objective is evaluating university premised upon employment status of their graduates. However, university evaluation is fundamentally an unsupervised learning problem that lacks a tangible ground truth. Thus this paper endeavors to address this inherent challenge by reiterating our goal as exploring the employment status of university graduates, and concurrently quantifying the influence of universities, where influence is indicative of the status that a school or company maintains within the professional sphere. Therefore, after modeling university-company transition graph, our model would assign each entity a latent representation. These representations would be used to predict the talent flow between entities, which explicitly reveals the employment status of graduates. Based on the above definitions, we can formulate the graduate employment situation based university evaluation problem as follows,

Definition 2 (Problem Definition). Given a dynamic university-company transition graph, we aim to learn the representation for each university and company in order to estimate the existence of talent flows and forecast the employment situation in the future while capturing the influence of universities.

3.4 Graph Autoencoders

As we described above, our university evaluation problem can be categorized as a link prediction problem on a directed graph. Thus, we provide an overview of graph autoencoders (GAE) [18], [23], which is treated as the dominant approach for link prediction, and some variants of it. Generally, GAE usually has two components, namely encoder and decoder.

First, given a graph $G = (V, E, X)$, where V, E respectively denotes the set of verticals and edges, the encoder assigns a latent vector z_i for each node $v_i \in V$ in the graph. We usually leverage graph Convolutional Network as the encoder due to its computation efficiency and great capability on graph modeling. The naive Graph Convolution Network can be formulated as follows,

$$\mathbf{Z} = \hat{\mathbf{D}}^{-1/2}\hat{\mathbf{A}}\hat{\mathbf{D}}^{-1/2}\mathbf{X}\mathbf{W} + b \tag{1}$$

where Z is the embedding matrix, $\hat{\mathbf{A}} = \mathbf{A} + \mathbf{I}$ denotes the adjacency matrix with inserted self-loops, $\hat{D}_{ii} = \sum_{j=0} \hat{A}_{ij}$ is the diagonal degree matrix, X is the feature matrix of nodes, and W, b are parameters to be learned. For the node-wise perspective, the convlotional layer can be viewed as the aggregation process of information contained in neighbouring nodes.

Then, the decoder aims to reconstruct A with the output of the encoder Z where the basic decoding process is based on the inner product of latent representations. For two nodes $v_i, v_j \in V$, the larger the inner product $z_i^T z_j$ means a higher probability that nodes v_i and v_j are connected in the graph. The adjacency matrix of the reconstruction graph $\hat{A}$ decoded from A can be written as $\hat{A} = \sigma(Z^T Z)$, and the reconstruction process demonstrates the intuition behind the autoencoder: If the adjacency matrix can be well reconstructed by the latent vectors, the learned latent vectors should preserve enough information from the original graph. Besides, [23] also proposed Variational Graph Autoencoder (VGAE), which extended VAE on graph with introducing a probabilistic model in the graph modeling.

4 METHODOLOGY

In this section, we introduce the technical details of our proposed IGAE model, shown in Figure 4. As previously discussed, our primary objective is to forecast employment trends by effectively modeling universities and companies within our meticulously designed dynamic transition graph, and subsequently assigning each a fixed-length representation vector. Given the unidirectional property of talent flow, the main challenge of this component lie in assigning an asymmetric representation to each node in the dynamic directed graph while making the representations capable of depicting the Influence of each node.

To tackle this challenge, we take inspiration from the HITS algorithm, renowned for its ability to estimate link values within directed networks. In the HITS algorithm, two metrics, known as Hubs and Authorities, would respectively measure the value of the links from a node and the inherent value of the node itself. By integrating this link analysis approach into a graph autoencoder framework, we can not only gauge the likelihood of links between node pairs but also retain the quantifiable feature of node influence. Thus this section presents the details of our HITS-inspired graph autoencoder methods, followed by an introduction to the encoder and decoder components of the IGAE model, respectively.

4.1 HITS-Inspired graph Autoencoder

This section will introduce the conceptual integration of the HITS algorithm within the graph autoencoder.

4.1.1 HITS Algorithm

Hyperlink-Induced Topic Search (HITS) [48] is a link analysis algorithm devised to rate web pages, considering the World Wide Web (WWW) as a directed graph. The HITS algorithm draws its inspiration from the structure of the Internet, where certain authoritative web pages have a high value due to the intrinsic value of their content. In contrast, some pages, referred to as 'hubs', have notable significance due to their role as extensive resource catalogs that guide users to other high-value pages. Accordingly, HITS assigns two scores to each page: an authority value, which estimates the value of the page's content, and a hub value, which evaluates the worth of links from the page to other pages. To compute these values, we initially set $hub(v) = 1$ and $auth(v) = 1$ for each $v \in V$, as part of the initialization process, and then update the scores as follows,

$$auth(v) = \sum_{i}^{d-in} auth(i), \quad hub(v) = \sum_{j}^{d-out} hub(j) \tag{2}$$

where $d - in$ and $d - out$ are respectively the in-degree and out-degree of node v, i is a node with an edge linked to v, and j is a node with an edge originating from v. After several iterations of updates, we can derive the hub and authority values for all nodes. Given the impressive success of the HITS algorithm in network analysis, we propose integrating its concepts into our graph modeling approach to capture node influence information.

4.1.2 HITS-Inspired Graph Autoencoder

Returning now to our original problem of learning node representations within university and company transition networks, our objective is to determine a latent vector

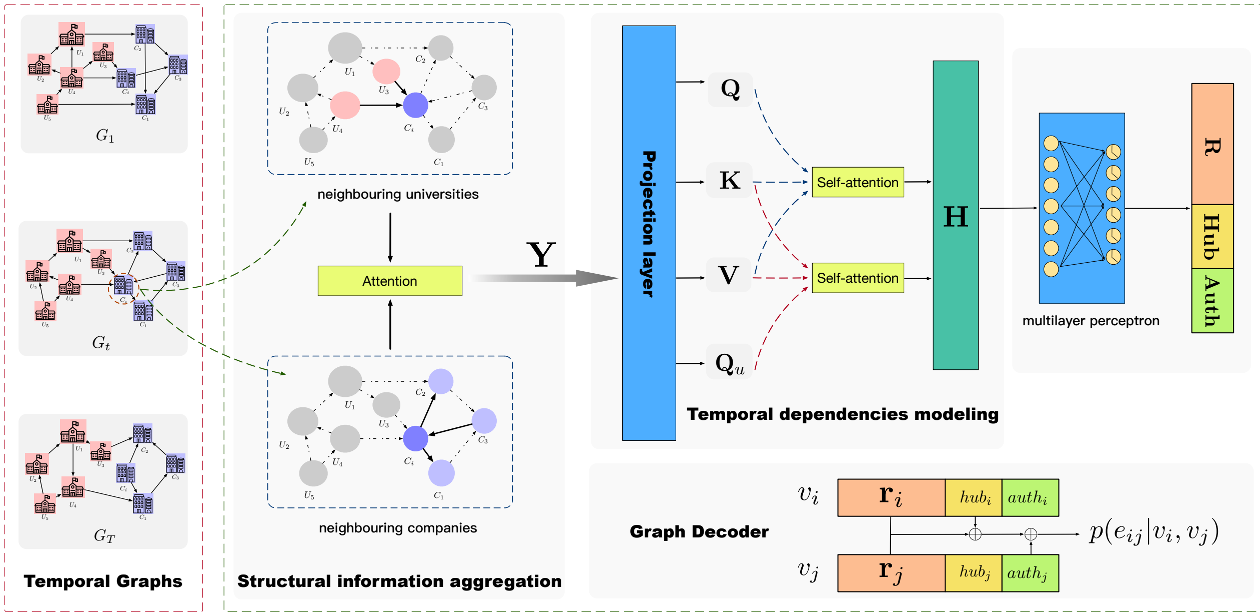


Fig. 4: The diagrammatic sketch of the proposed IGAE framework.

$r_i \in \mathbb{R}^d$ $(d << n)$ for each node. This vector should encapsulate the information of talent flow between nodes of different types and capture the node influence as manifested within the transition networks. More specifically, our objective is to learn two additional parameters, hub_i and $auth_i \in \mathbb{R}^1$, each representing the hub and authority values of the nodes, respectively. Following the logic of the HITS algorithm, we postulate that a node with a high authority score is likely to draw links from nodes with high hub scores. Hence, the reconstruction function can be expressed as follows.

$$p(\widetilde{e}_{ij}|v_i, v_j) = \frac{hub_i \cdot auth_j}{||\mathbf{r}_i - \mathbf{r}_j||_2} \tag{3}$$

where $||\mathbf{r}_i - \mathbf{r}_j||_2$ represents the second-order norm, signifying that nodes with similar learned representations that reflect similar structural proximity, are more likely to establish connections.

4.2 Influence-based Graph Encoder

In this part, we introduce our method to derive node representations and influence-based parameters hub_i and $auth_i$ for all nodes $v_i \in V$, with a graph encoder framework. Specifically, our encoder architecture is meticulously consisted of multiple spatial-temporal blocks. Within each block, a heterogeneous GCN module is dedicated to the extraction of inherent structural nuances, while a dual self-attention module is adeptly designed for temporal dynamics capture. A comprehensive exposition of these modules is presented in the subsequent sections..

4.2.1 Structural Information Aggregation

Given that GCN aggregates information from adjacent nodes, it can emulate the computational approach of the HITS algorithm, thereby capturing the intrinsic node influence within the graph. For each node, the convolutional layer can be viewed as the aggregation process of the information contained in neighboring nodes. Since universities only have the neighbour with the same types, the node-wise representation update of each GCN layer within directed subgraph (V_U, E_{UU}) could be written as,

$$\mathbf{y}_i^{(l)} = \mathbf{W}^\top \sum_{j \in \mathcal{N}(v)} \frac{w_{ji}}{\sqrt{\hat{d}_j \hat{d}_i}} \mathbf{h}_j^{(l-1)} \tag{4}$$

with $\hat{d}_i = 1 + \sum_{j \in \mathcal{N}(i)} e_{ji}$, where w_{ji} denotes the weight of edge e_{ij}, and $\mathbf{h}_i^{(0)} = FC(\mathbf{x}_i)$.

However, based on the definition of university-company graph, each company would probably have connections with universities and other companies. Given the inherent heterogeneity in the feature spaces of these distinct node types, the conventional GCN aggregation methodology proves suboptimal.

To address this limitation, we propose a different convolutional layer for aggregating information from different types. The first step is to design the transformation matrix $\mathbf{M}_\tau = (\mathbf{M}_U, \mathbf{M}_C)$ to project the features of each type of node into the same space. Moreover, the appeal of companies to university talents varies for several reasons, such as the competitiveness of the company and the relevance of the major. Thus we design a type-based attention mechanism to assign the weights to model the impact from different types of nodes. Specifically, for company v_i, the impact embedding of a specific type τ can be defined as the aggregation of neighboring nodes of this type,

$$\mathbf{y}_{\tau i}^{(l)} = \mathbf{W}^\top \sum_{j \in \mathcal{N}_\tau(v)} \frac{w_{j,i}}{\sqrt{\hat{d}_j \hat{d}_i}} \mathbf{M}_\tau \mathbf{h}_j^{(l-1)} \tag{5}$$

Then we need to assign the weight to the different information type τ for node v_i, based on a similarity between node embedding and its type-specific neighbouring embedding, which can be calculated as,

$$a_{\tau i} = \boldsymbol{h}^T ReLU(\boldsymbol{W}[\mathbf{y}_{\tau i}^{(l)} || \mathbf{h}_i^{(l-1)}] \tag{6}$$

$$e_{\tau i} = \frac{exp(a_{\tau i})}{\sum_{\tau} exp(a_{\tau i})} \tag{7}$$

where $\boldsymbol{W}$ and $\boldsymbol{h}$ are parameters to be learnt, $||$ means the concatenation operator. With this type-based attention mechanism, the overall hidden representation of company v_i can be obtained as,

$$\mathbf{y}_i^{(l)} = \mathbf{y}_{ci}^{(l)} + \mathbf{y}_{ui}^{(l)} = \sum_{\tau} \mathbf{e}_{\tau i} \mathbf{h}_{\tau i}^{(l)} \tag{8}$$

where y_{ci} and y_{ui} respectively denote information aggregated from the neighbouring companies and universities for node v_i. Thus, by applying this aggregation mechanism on each snapshot of the dynamic graph $\mathcal{G}$, our heterogeneous information aggregation module can be formulated as,

$$\mathbf{Y}^{(l)} = HetGCN(\mathcal{G}, \mathbf{H}^{(l)}) \tag{9}$$

where $\mathbf{Y}^{(l)} \in \mathbb{R}^{N*T*d}$ is the output of HetGCN module.

4.2.2 Temporal Dependencies Modeling

Considering the inherent temporal characteristics of our proposed transition graph, which delineates the progression from university to company trajectories, it becomes important for our model to provide valuable insights that can assist students in making informed career decisions. To address the dynamic complexities of this graph, we design a self-attention based temporal dependencies modeling approach in order to capture the temporal dependencies and evolving patterns present in the data. Subsequently, this self-attention-based module is meticulously integrated into the encoder, ensuring a comprehensive analysis of the graph dynamic behavior over time.

Upon deploying our proposed heterogeneous information aggregation module on every graph snapshot, we derive a temporal sequence of hidden representations for every node, symbolized as $\mathbf{Y}_i^{(l)} = \{\mathbf{y}_i^1, \cdots, \mathbf{y}_i^T\} \in \mathbb{R}^{T*d}$, where $v_i \in V$ and T represents the number of snapshots, which collate the representation of node v across timestamps, and employ a self-attention mechanism to discern temporal dependencies. To determine the representation at time t, we first incorporate the position information with sinusoidal position encoding as the foundational Transformer model [49] does. The positional encoding matrix $\mathbf{P}_i \in \mathbb{R}^{T*d}$ is then element-wise added to the sequence $\mathbf{Y}_i$, as $\mathbf{Y}_i = \mathbf{Y}_i + \mathbf{P}_i$, to furnish the input representation for the self-attention. We initially project the input $\mathbf{Y}_i$ into queries $\mathbf{Q}_i$, keys $\mathbf{K}_i$, and values $\mathbf{V}_i$ using three distinct projection matrices, $\mathbf{W}_q$, $\mathbf{W}_k$, and $\mathbf{W}_v$. Subsequently, we compute the relevance of the previous information as,

$$\begin{aligned} \mathbf{Z}_i = \beta_i \left(\mathbf{Y}_i \mathbf{W}_v\right), \quad \beta_i^{kl} = \frac{\exp\left(e_i^{kl}\right)}{\sum_{j=1}^{T} \exp\left(e_i^{kj}\right)}, \\ e_i^{kl} = \left(\frac{\left((\mathbf{Y}_i \mathbf{W}_q)(\mathbf{Y}_i \mathbf{W}_k)^T\right)_{kl}}{\sqrt{d'}} + M_{kl}\right) \end{aligned} \tag{10}$$

where d' is dimensionality of projected node embedding, and $\mathbf{M} \in R^{T*T}$ is a masked matrix, encoding the order and defined as,

$$M_{kl} = \begin{cases} 0, & k \leq l \\ -\infty, & k > l \end{cases} \tag{11}$$

When $M_{kl} = -\infty$, the influence of subsequent timestamps on the current timestamp is mitigated.

Within the self-attention paradigm, the updated representations capture temporal dependencies. This update process incorporates both multi-head self-attention mechanisms and position-wise feed-forward operations. Formally, for each module, the latent representation is updated as,

$$\begin{aligned} \mathbf{H}_i^{(l)} &= \text{LayerNorm}\left(\mathbf{Y}_i^{(l)} + \text{MultiHead}(\mathbf{Q}_i, \mathbf{K}_i, \mathbf{V}_i)\right), \\ \mathbf{H}_i^{(l)} &= \text{LayerNorm}\left(\mathbf{H}_i^{(l)} + \text{FFN}(\mathbf{H}_i^{(l)})\right). \end{aligned} \tag{12}$$

where LayerNorm is the layer normalization network and FFN is the position-wise feed-forward network. The multi-head attention mechanism segments the input H into h partitions, concatenating outputs from these individual heads,

$$\text{MultiHead}(\mathbf{Q}_i, \mathbf{K}_i, \mathbf{V}_i) = \text{Concat}(\mathbf{H}_i^1, \ldots, \mathbf{H}_i^H)\boldsymbol{W}_m, \tag{13}$$

where $\mathbf{H}_i^h$ is the output of each head and $\boldsymbol{W}_m$ is another learned projection matrix.

The self-attention mechanism described above can be succinctly represented as $\mathbf{H}^{(l)} = \text{Attn}(\mathbf{Q}, \mathbf{K}, \mathbf{V})$. Here, $\mathbf{H}^{(l)} \in \mathbb{R}^{N \times T \times d}$ denotes the output of current temporal modeling block, and $\mathbf{H}^{(l)}$ signifies the sequence of representation vectors corresponding to node v_i.

Building upon this, we introduce refinements that are suited for our specific scenario. Our definition of the problem, together with the autoencoder architecture, inherently requires that the representation of a company be closely aligned with the dynamics of its associated universities. To further this idea, we propose a dual attention mechanism that not only models temporal dependency but also emphasizes the connection between a company representation and information aggregated from universities. This mechanism integrates an additional information sequence, sourced by using university influence as a query $\mathbf{Q}_{ui} = \mathbf{Y}_{ui}\mathbf{W}_{qu}$ where the goal is to enrich the company representation with more nuanced insights from related universities. In essence, for a company c_i, its representation is meticulously crafted through the subsequent attention process.

$$\begin{aligned}
\mathbf{H}_{ci}^{(l)} &= \text{Attn}(\mathbf{Q}_i, \mathbf{K}_i, \mathbf{V}_i), \\
\mathbf{H}_{ui}^{(l)} &= \text{Attn}(\mathbf{Q}_{ui}, \mathbf{K}_i, \mathbf{V}_i), \\
\mathbf{E}_{ci} &= \boldsymbol{h_d^T} ReLU(\boldsymbol{W_d}[\mathbf{H}_{ci}^{(l)} || \mathbf{Y}_i^{(l)}]), \\
\mathbf{E}_{ui} &= \boldsymbol{h_d^T} ReLU(\boldsymbol{W_d}[\mathbf{H}_{ui}^{(l)} || \mathbf{Y}_i^{(l)}]), \\
a_{i,t} &= \frac{exp(e_{ci,t})}{exp(e_{ci,t}) + exp(e_{ui,t})} \\
\mathbf{A}_i &= [a_{i,1}, a_{i,2}, \ldots, a_{i,T}], \quad t \in 1, \ldots, T \\
\mathbf{H}_i^{(l)} &= \mathbf{A}_i \odot \mathbf{H}_{ci}^{(l)} + (\mathbf{I} - \mathbf{A}_i) \odot \mathbf{H}_{ui}^{(l)}.
\end{aligned} \tag{14}$$

where $\boldsymbol{h_d}$ and $\boldsymbol{W_d}$ are parameters to be learned.

4.2.3 Node Representation Forecasting

Upon deploying multiple spatial-temporal blocks, we derive the culminating node representation sequence, denoted as $\mathbf{H}^L$. Given our objective of advising students on prospective career paths based on historical employment trends, it is imperative to model the next graph snapshot using antecedent ones. To this end, we employ a dense layer to map $\mathbf{H}^L$ to a representation matrix $\mathbf{R}' \in \mathbb{R}^{N*d}$. Inspired by the HITS algorithm, our encoder is designed to encapsulate node representations and quantify node influence. The node representation is obtained as,

$$\mathbf{R}' = (\mathbf{Hub}, \mathbf{Auth}, \mathbf{R}) = \boldsymbol{W}_2 \sigma(\boldsymbol{W}_1 \mathbf{H}^L + \boldsymbol{h}_1) + \boldsymbol{h}_2 \tag{15}$$

Here, $\boldsymbol{W}_1$, $\boldsymbol{W}_2$, $\boldsymbol{h}_1$, and $\boldsymbol{h}_2$ are learnable parameters. The output $\mathbf{R}'$ is the concatenation of $\mathbf{Hub} \in \mathbb{R}^{N*1}$, $\mathbf{Auth} \in \mathbb{R}^{N*1}$, and $\mathbf{R} \in \mathbb{R}^{N*(d-2)}$.

4.3 Influence-based Graph Decoder

In the decoding phase, we utilize the forecasting node representations $\mathbf{R}'$ learned by our proposed Influence-based Graph Encoder to reconstruct the original graph at the next timestamp G_{t+1}, with using the HITS-inspired reconstruction function (Eq. (3)). Specifically, we aim to predict potential talent flows. Our primary objective is to prognosticate potential talent trajectories. Notably, given the typically denser talent flow from universities to companies, our focus narrows to predicting the weight of links engendered by graduate employment.

In practice, we utilize $log(\widetilde{e}_{ij})$ to simplify the training process, as it can mitigate the influence of potentially large values. More formally, we estimate the probability of potential talent flow from v_i to v_j by leveraging the influence terms as follows,

$$p(e_{ij}|v_i, v_j) = \sigma(log(hub_i) + log(auth_j)) - \lambda_e log(||\mathbf{r}_i - \mathbf{r}_j||_2), \tag{16}$$

where $\sigma(\cdot)$ is the sigmoid activation function to regulate the range of possibility, and λ_e is a hyperparameter to balance the influence of the distance between two representation vectors. With this estimated probability, we can capture the asymmetry of talent flows. Then we compute the reconstruction loss with negative log-likelihood of all observed edges in E as following,

$$L_e = -\sum_{e_{ij} \in E} log(p(e_{ij}|v_i, v_j)) - \sum_{e_{ij} \notin E} log(1 - p(e_{ij}|v_i, v_j)). \tag{17}$$

Note that in our training process, we adopt negative sampling strategies proposed by [50]. For each node v_i, we sample some nodes that are not connected to v_i according to the noisy distribution $P_{\text{neg}}(v) \propto d_v^{\frac{3}{4}}$ (d_v is the average out-degree of all nodes), in order to ensure the efficiency of training process.

Subsequently, regarding the connections from universities to companies, due to their representations being aligned within the same latent space while preserving information on edge weights throughout the aggregation process, we propose to forecast the weights of these connections utilizing a similar reconstruction function, omitting the sigmoid function. Indeed, the weight of edge between $v_i \in V_U$ and $v_j \in V_C$ would be calculated as,

$$\widetilde{w}_{ij} = log(hub_i) + log(auth_j) - \lambda_w log(||\mathbf{r}_i - \mathbf{r}_j||_2), \tag{18}$$

where λ_w is another introduced hyperparameter to adjust the importance of distance between latent vectors of two nodes. Afterward, we can utilize the RMSE loss function to calculate the total predicted loss as,

$$L_w = \sqrt{\frac{\sum_{e_{ij} \in E_{UC}} ||\widetilde{w}_{ij} - w_{ij}||^2}{|E_{UC}|}}. \tag{19}$$

4.4 Model Learning

The primary objective of IGAE is to derive a holistic representation for each node within the university-company graph. This representation is instrumental in dynamically forecasting potential talent flows and gauging the weight of graduate employment. Given that both tasks are geared towards bolstering the representational efficacy of node embedding while retaining node influence information, they exhibit intrinsic correlations. Consequently, a joint learning approach for these tasks is both feasible and beneficial. Hence, the comprehensive objective function is thus articulated as:

$$L = L_e + \lambda L_w + ||\Theta||_2^2, \tag{20}$$

where λ serves as a hyperparameter, fine-tuning the balance between the impacts of the two tasks, and $||\Theta||_2^2$ represents the L2-norm regularization term. Parameters within the IGAE framework are optimized by minimizing the reconstruction loss mentioned above.

5 EXPERIMENTS

In this section, we will introduce the details of our experiment on real-world dataset to validate the efficiency of our proposed IGAE [3].

3. Our source code is at https://github.com/YuyangYe/IGAE

5.1 Experiment Setup

5.1.1 Dataset

The dataset for this study was sourced from a prominent online professional network which has been used in several related studies on talent transition analysis [12], [13], [51], as detailed in Section 3.1. To ensure the quality of the data and its compatibility with the requirements of our model, we focused on the individuals who graduated from institutions listed in the QS World University Rankings between 2008 and 2019 [38]. This approach was chosen to prioritize data from top-tier universities, ensuring the relevance and quality of our dataset. Furthermore, any internship or work experience prior to graduation was excluded to maintain a consistent focus on post-graduate trajectories. After these preprocessing steps, our refined dataset comprised 620,087 individual trajectories. Using these data, we constructed a dynamic university-company transition graph by generating annual snapshots, resulting in 12 distinct snapshots. This graph includes 1,305 universities and 5,662 companies, with the average edge counts in the sets (E_{UU}, E_{CC}, E_{UC}) being 12,165, 35,347, and 74,521, respectively. In addition, we incorporated features associated with both universities and companies in our analysis. We list the detail of features in Table 1. Specifically, we aggregate similar disciplines and job types with the selected keywords. Subsequently, categorical attributes, such as location and type, were converted into one-hot encoding. For textual descriptions, we used *doc2vec* [52] to derive fixed-length vector representations.

TABLE 1: Description of the used features.

Description	Type
University ID, Location, Age	Categorical
University Size, Major Disciplines Distribution	Numerical
University Description	Textual
Company ID, Location, Age, Type	Categorical
Company Size, Job Type Distribution	Numerical
Company Description	Textual

5.1.2 Baselines

We compared IGAE with several state-of-the-art methods. Given that our two prediction tasks can be respectively formulated as a link prediction task and an edge weight prediction task, we selected the following methods:

- **Self-Attention(SA)**: this method leverages a self-attention mechanism to model the basic feature sequences over time to predict graduate flows.
- **DeepWalk** [15]: This method learns node embedding by treating random walks obtained from a graph as a training corpus.
- **LINE** [17]: LINE emphasizes the preservation of node proximity within network structures to derive node embedding.
- **VGAE** [23]: An extension of the variational autoencoder, VGAE employs two GCN layers to learn node distributions for undirected graphs.
- **DynGEM** [53]: DynGEM uses a deep autoencoder to incrementally learn the embedding of a dynamic graph in snapshot t by encoding snapshot $t-1$.
- **Gravity** [24]: Inspired by Newton's theory of universal gravitation, Gravity extends VGAE for directed network embedding.
- **DiGCN** [25]: As an extension of GCN for directed graphs, DiGCN simplifies its approach using personalized PageRank.
- **HAN** [26]: HAN employs node-level and semantic-level attention mechanisms to derive representations for heterogeneous graphs.
- **HetGNN** [9]: HetGNN samples a fixed number of neighbors through random walks and utilizes a bi-LSTM structure as an aggregator to integrate heterogeneous neighbors.
- **HTGNN** [33]: HTGNN proposes a hierarchical aggregation mechanism to jointly model heterogeneous spatial dependencies and temporal dimensions in the heterogeneous temporal graph.
- **DSTAGNN** [54]: DSTAGNN designs sophisticated multi-head attentions and multi-scale gated convolutions to model the spatial temporal graph.

Note that for static graph modeling approaches, we also integrate the same self-attention module with them to modify them to solve our dynamic graph modeling problem. Besides, following the setting of [24], we adopt source/target vectors paradigm to handle the asymmetric graph with undirected graph embedding methods.

5.1.3 Evaluation Metrics

We chose the Area Under the Curve (AUC) and the Mean Average Precision (MAP) to evaluate the performance of the talent transition prediction task (Task1) and used the Root Mean Square Error (RMSE) and the Mean Absolute Error (MAE) to evaluate the performance of the employment situation forecasting task (Task2).

5.1.4 Implement Setting

To enhance the robustness of IGAE, given the dynamic nature of our graph with T snapshots, we implement a meticulous training methodology. Specifically, our model is designed to reconstruct the snapshot of $t+1^{th}$ step with training model on graphs of step $t-l$ to t, where l is a hyperparameter and is assigned to 4. For cross-validation, we systematically reconstruct each snapshot from the latter half of the provided sequence. The model is trained on five distinct snapshot sequences and tested on the remaining one. Then we present the average performance metrics of these evaluations. It should be noted that for the test snapshot, we selected an equivalent number of node pairs that remained unconnected to serve as negative samples. Within the IGAE framework, the dimension of node representation is set to 128. Parameters λ_e, λ_w and λ are assigned values of 1, 0.1, and 0.01, respectively. Our model is optimized by the Adam algorithm with a learning rate of 0.01.

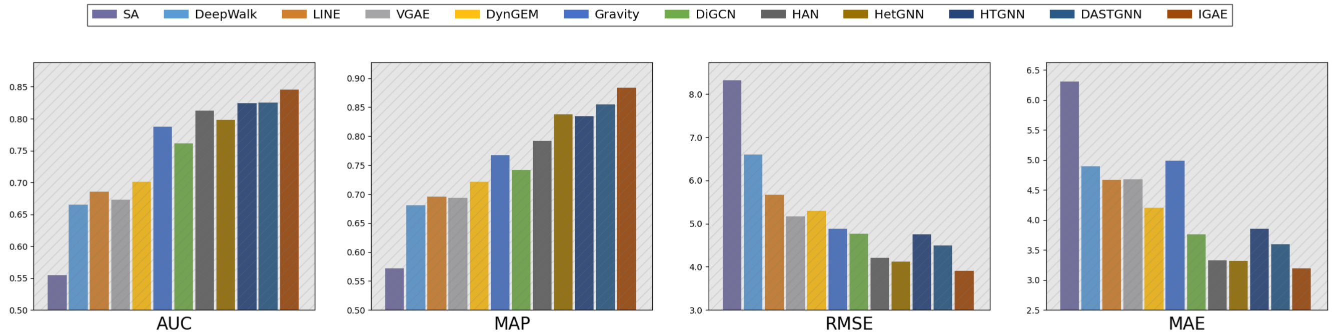


Fig. 5: The performance of IGAE and other baselines under four metrics.

5.2 Performance Evaluation

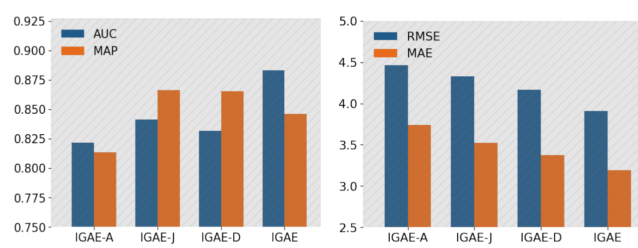
5.2.1 Overall Performance

The comparative evaluation of performance metrics is depicted in Figure 5, highlighting the superior performance of our IGAE model over comparative baseline models across two prediction tasks, which emphatically affirms the effectiveness of the proposed IGAE framework. Initially, it is evident that the Self-Attention (SA) method, which does not use a graph structure to predict talent flow, performs poorly, underscoring the effectiveness of our constructed University-Company Transition Graph. On a detailed analysis, unsupervised network embedding methods including DeepWalk, LINE, and DynGEM underperform significantly. This could be ascribed to their exclusive dependence on node proximity learning, lacking label-guided aggregation mechanisms crucial for delineating the subtle, directional influences in talent transitions modeled by the directed graph. Then, compared to other graph neural network methods, VGAE performs poorly across all metrics, especially in employment situation prediction, which could be attributed to its limited capacity to capture asymmetric node relationships, probably resulting from imbalances in in-degree and out-degree. In contrast, while GNN models designed specifically for directed graphs show promising results in the first task, their performance deteriorates in predicting employment. This observation suggests that these models may struggle to balance the consideration of node influence and node similarity, especially when connecting different types of nodes. For example, Gravity may overestimate the influence of a prominent company, assuming that it forms connections with all universities, leading to skewed weight assignments. Conversely, the performance of modified heterogeneous graph neural networks, such as HAN and HetGNN, aligns more closely with the employment prediction task, emphasizing the critical need to account for the inherent heterogeneity within the transition graph to achieve more accurate and reliable predictions. Moreover, spatial-temporal graph neural networks, such as HTGNN and DSTAGNN, demonstrate more commendable efficacy in the first task, affirming the benefits of parallelly integrating spatial and temporal module. However, their performance also decreases significantly in the second task. The results of the comparative experiment further indicate the robustness of IGAE in modeling unidirectional and heterogeneous properties intrinsic to dynamic talent transition graph.

5.2.2 Ablation Study

To assess the impact of various components within the IGAE framework, we devised several variants of IGAE. Their descriptions and modifications are as follows:

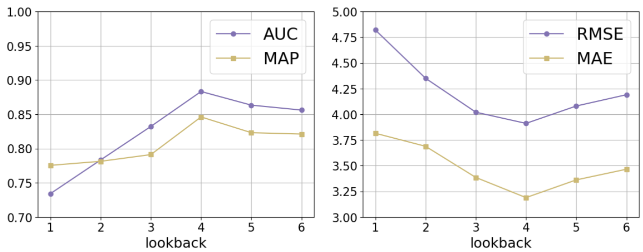
- **IGAE-A**: In this variant, the type-based attention mechanism present in the encoder component of IGAE is omitted.
- **IGAE-J**: Rather than adopting a joint approach to the multitask problem, IGAE-J is designed to learn the two tasks independently.
- **IGAE-D**: This variant replaces the dual self-attention module with the fundamental one.



(a) Task1 (b) Task2

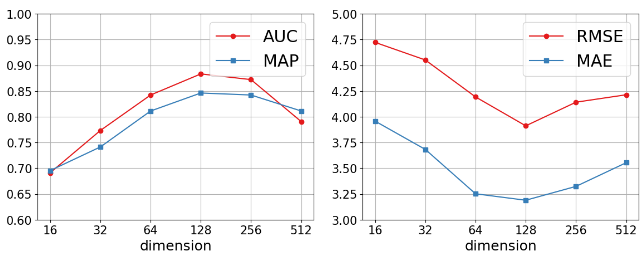
Fig. 6: Performance of IGAE and its variants on two tasks.

As depicted in 6, a comparative evaluation between IGAE and IGAE-A underscores the importance of distinguishing influences from various types of neighboring information during the aggregation process. Additionally, IGAE-J exhibits notably inferior performance relative to a multi-task learning strategy, specifically in employment frequency forecasting task. This finding underscores the advantages of concurrently tackling both tasks. Moreover, when compared with IGAE, the worse performance of IGAE-D validates the efficacy of our proposed dual attention mechanism, which integrates information from associated universities to better understanding the temporal pattern of companies.



(a) Task1 (b) Task2

Fig. 7: Performance of IGAE under different lookback l.



(a) AUC (b) RMSE

Fig. 8: Performance of IGAE under different d.

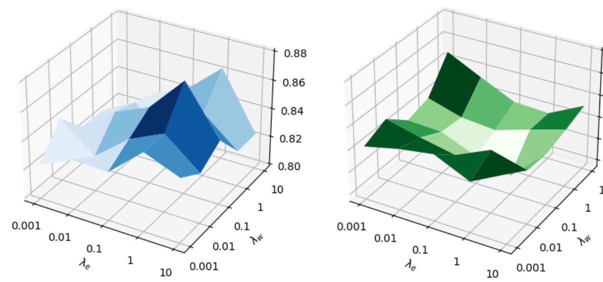
5.2.3 Parameter Sensitivity

Subsequently, we sought to ascertain the robustness of IGAE by examining its performance sensitivity relative to the number of training snapshots, denoted as l. We varied l from 1 to 6 and documented the corresponding outcomes in Figure 7. Notably, as l increases, IGAE's performance exhibits improvement, stabilizing when l exceeds 4. Furthermore, we investigated the influence of the dimension size, d, on the model's efficacy for each task. As depicted in Figure 8, IGAE's performance escalates with an increase in d, suggesting that a higher dimension effectively retains more information from our proposed talent transition graph. However, a discernible decline in performance is observed upon continuous augmentation of the dimension size d. This downturn is likely attributable to the curse of dimensionality coupled with the diminishing impact of influence terms.

Additionally, we assessed the influence of parameters λ_e and λ_w, which serve to calibrate the significance of node proximity in each prediction task. IGAE was trained with these parameters varied from 0.001 to 10. As illustrated in Figure 9, IGAE attains optimal performance when $\lambda_e = 1$ and $\lambda_w = 0.1$. This outcome underscores the heightened importance of node proximity in the initial task. A plausible explanation for this observed behavior is the unidirectional property of links between universities and companies, which amplifies the significance of influence terms during graph reconstruction.

5.2.4 Data Visualization

To validate the efficacy of our approach in modeling universities, we visualized the learned representations. Specifically, we employed K-means clustering [55] to group universities and companies based on their embedding. Subsequently, t-SNE [56] was utilized to project these node representations into a two-dimensional space. The clustering results are depicted in Figure 10. Upon closer analysis of the cluster overlaps, we observe that universities from the same country or region tend to cluster together, while companies predominantly cluster based on category similarities. This behavior in node embedding implies that there is a higher likelihood of talent mobility within the same region for universities and within similar industry sectors for companies. Analyzing deeper into the granularity of the visualization reveals further patterns and inferences. For instance, Carnegie Mellon University is proximate to the IT company cluster, on contrast to University of Michigan, which has the similar location and QS ranking but is closer to the finance company cluster. This spatial distribution implies a possible advantage for Carnegie Mellon University in successfully placing graduates within leading IT companies, while University of Michigan proximity to finance firms underscores its strength in the finance domain. Such observations indicate the capability of our method on providing detailed and profession-specific insights.



(a) AUC (b) RMSE

Fig. 9: Performance of IGAE under different λ.

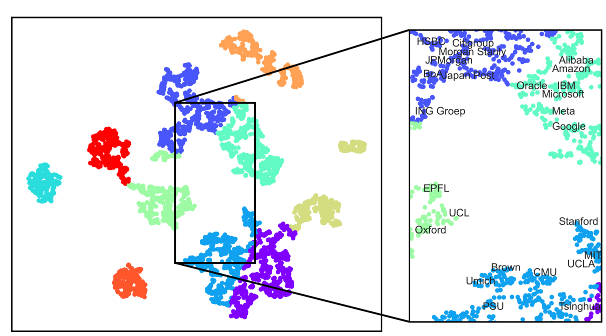


Fig. 10: Visualization of the university and company representation. The left part is the global view of node embedding. The right part is a fine-grained display.

TABLE 2: Comparison of University Ranking Results

Top 20 Universities in QS World University Ranking
Massachusetts Institute of Technology; University of Oxford; Stanford University; University of Cambridge; Harvard University; California Institute of Technology; Imperial College London; ETH Zurich - Swiss Federal Institute of Technology; University College London; EPFL; University of Chicago; National University of Singapore; Nanyang Technological University; University of Pennsylvania; Yale University; The University of Edinburgh; Tsinghua University; Peking University; Columbia University; Princeton University
Top 20 Universities under IGAE Method
Stanford University; Massachusetts Institute of Technology; Harvard University; University of California, Berkeley; California Institute of Technology; University of Cambridge; University of Oxford; Princeton University; University of Chicago; Yale University; University of Pennsylvania; Columbia University; University of California, Los Angeles; University of Toronto; University of Michigan, Ann Arbor; Tsinghua University; University of Illinois at Urbana-Champaign; University of Washington; Carnegie Mellon University

5.3 Case Study

Considering that our IGAE methodology integrates two quantifiable parameters inspired by the HITS algorithm, which are related to influence, during the construction of the network structure, it enables a detailed analysis of influence based on these factors. According to the rules of graph reconstruction in IGAE, universities with higher authority values are considered to be more effective in attracting talent from other academic institutions. On the other hand, high hub values suggest the ability of a university to direct talent to influential companies. Therefore, by comprehensively analyzing these factors, we have developed a unique university ranking system that focuses on graduate employment outcomes. Table 2 presents the top 20 universities according to our ranking methodology, alongside their QS world university rankings [38]. A quick comparison shows a significant overlap between the top universities in both ranking systems. Nevertheless, our evaluation method highlights certain universities, underscoring that our rankings align with the professional reputation captured by well-established public university rankings while also providing a new and reasoned perspective on university evaluation.Additionally, there are noticeable differences in the rankings of specific universities. For example, while the University of Massachusetts Amherst and Georgetown University have similar rankings in the QS system, Georgetown University is ranked higher in our list. This discrepancy suggests that Georgetown University may place a stronger emphasis on student career development, while the University of Massachusetts Amherst may excel in other areas. These detailed insights can offer prospective students more precise and informed support in choosing their university.

6 Conclusion

In this paper, we proposed an Influence based Graph Autoencoder (IGAE) approach to evaluate universities by exploring the employment situation of their graduates, which fills the research voids in quantitatively assessing university quality. Specifically, our basic assumption posits that high-influence institutions would attract top-tier graduates from high-ranking universities, indicating our problem can be addressed through a talent flow prediction paradigm. Along this line, we first conducted a heterogeneous university-company talent transition graph based on massive resume information, which talent flows between different institutions and implicitly reveals the influence of universities and companies. Afterward, considering the asymmetric properties of talent flows, we proposed an unidirectional aggregation process with a type based attention mechanism to encode structural information of our talent transition graph. Concurrently, we integrate an innovative dual self-attention module to effectively capture the dynamic patterns of the talent transition graph and predict the node representation at the subsequent timestamp. Note that the devised encoder, built upon these components, learns node representation while preserving influence of institutions by maintaining the corresponding terms inspired by HITS algorithm. Hence, with designing an influence based graph decoder, we predict the future talent transition based on the node dynamics and influential factors. In particular, IGAE aimed to respectively forecast potential talent flows and estimate employment frequency, which were optimized in a joint learning framework. Finally, extensive experiments on real-world dataset demonstrate the effectiveness of IGAE compared to state-of-the-art baselines, while several additional university influence analysis and case studies are further provided to enhance its credibility.

Acknowledgments

This research was partially supported by the National Science Foundation (NSF) via the grant number IIS-2006387 and IIS-2040799.

References

[1] M. Bond, K. Buntins, S. Bedenlier, O. Zawacki-Richter, and M. Kerres, "Mapping research in student engagement and educational technology in higher education: A systematic evidence map," *International journal of educational technology in higher education*, vol. 17, no. 1, pp. 1–30, 2020.

[2] E. B. Stolzenberg, M. K. Eagan, M. C. Aragon, N. M. Cesar-Davis, S. Jacobo, V. Couch, and C. Rios-Aguilar, "The american freshman: National norms fall 2019," *Higher Education Research Institute, UCLA*, 2019.

[3] F. van Vught and F. Ziegele, Eds., *Multidimensional Ranking: The Design and Development of U-Multirank*, ser. Higher Education Dynamics. Springer Netherlands, 2012, vol. 37.

[4] C. Claassen, "Measuring university quality," *Scientometrics*, vol. 104, no. 3, pp. 793–807, September 2015.

[5] L. Waltman, C. Calero-Medina, J. Kosten, E. C. Noyons, R. J. Tijssen, N. J. van Eck, T. N. van Leeuwen, A. F. van Raan, M. S. Visser, and P. Wouters, "The leiden ranking 2011/2012: Data collection, indicators, and interpretation," *Journal of the American society for information science and technology*, vol. 63, no. 12, pp. 2419–2432, 2012.

[6] "Best undergraduate business programs methodology," US News & World Report, 2022.

[7] F. A. Van Vught and F. Ziegele, *Multidimensional ranking: The design and development of U-Multirank*. Springer Science & Business Media, 2012, vol. 37.

[8] Q. Meng, H. Zhu, K. Xiao, L. Zhang, and H. Xiong, "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*, 2019, pp. 14–24.

[9] C. Zhang, D. Song, C. Huang, A. Swami, and N. V. Chawla, "Heterogeneous graph neural network," in *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, 2019, pp. 793–803.

[10] L. Zhang, T. Xu, H. Zhu, C. Qin, Q. Meng, H. Xiong, and E. Chen, "Large-scale talent flow embedding for company competitive analysis," in *Proceedings of The Web Conference 2020*, 2020, pp. 2354–2364.

[11] L. Li, H. Jing, H. Tong, J. Yang, Q. He, and B.-C. Chen, "Nemo: Next career move prediction with contextual embedding," in *Proceedings of the 26th International Conference on World Wide Web Companion*, 2017, pp. 505–513.

[12] L. Zhang, D. Zhou, H. Zhu, T. Xu, R. Zha, E. Chen, and H. Xiong, "Attentive heterogeneous graph embedding for job mobility prediction," in *Proceedings of the 27th ACM SIGKDD conference on knowledge discovery & data mining*, 2021, pp. 2192–2201.

[13] Y. Ye, Z. Dong, H. Zhu, T. Xu, X. Song, R. Yu, and H. Xiong, "Mane: Organizational network embedding with multiplex attentive neural networks," *IEEE Transactions on Knowledge and Data Engineering*, 2022.

[14] H. Zhu, H. Xiong, F. Tang, Q. Liu, Y. Ge, E. Chen, and Y. Fu, "Days on market: Measuring liquidity in real estate markets," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2016, pp. 393–402.

[15] B. Perozzi, R. Al-Rfou, and S. Skiena, "Deepwalk: Online learning of social representations," in *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2014, pp. 701–710.

[16] A. Grover and J. Leskovec, "node2vec: Scalable feature learning for networks," in *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*, 2016, pp. 855–864.

[17] J. Tang, M. Qu, M. Wang, M. Zhang, J. Yan, and Q. Mei, "Line: Large-scale information network embedding," in *Proceedings of the 24th international conference on world wide web*, 2015, pp. 1067–1077.

[18] D. Wang, P. Cui, and W. Zhu, "Structural deep network embedding," in *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*, 2016, pp. 1225–1234.

[19] S. Cao, W. Lu, and Q. Xu, "Grarep: Learning graph representations with global structural information," in *Proceedings of the 24th ACM international on conference on information and knowledge management*, 2015, pp. 891–900.

[20] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," *arXiv preprint arXiv:1609.02907*, 2016.

[21] W. Hamilton, Z. Ying, and J. Leskovec, "Inductive representation learning on large graphs," *Advances in neural information processing systems*, vol. 30, 2017.

[22] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Lio, and Y. Bengio, "Graph attention networks," *arXiv preprint arXiv:1710.10903*, 2017.

[23] T. N. Kipf and M. Welling, "Variational graph auto-encoders," *arXiv preprint arXiv:1611.07308*, 2016.

[24] G. Salha, S. Limnios, R. Hennequin, V.-A. Tran, and M. Vazirgiannis, "Gravity-inspired graph autoencoders for directed link prediction," in *Proceedings of the 28th ACM international conference on information and knowledge management*, 2019, pp. 589–598.

[25] Z. Tong, Y. Liang, C. Sun, X. Li, D. Rosenblum, and A. Lim, "Digraph inception convolutional networks," *Advances in neural information processing systems*, vol. 33, pp. 17 907–17 918, 2020.

[26] X. Wang, H. Ji, C. Shi, B. Wang, Y. Ye, P. Cui, and P. S. Yu, "Heterogeneous graph attention network," in *The world wide web conference*, 2019, pp. 2022–2032.

[27] H. Linmei, T. Yang, C. Shi, H. Ji, and X. Li, "Heterogeneous graph attention networks for semi-supervised short text classification," in *Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (EMNLP-IJCNLP)*, 2019, pp. 4821–4830.

[28] J. Zhao, X. Wang, C. Shi, B. Hu, G. Song, and Y. Ye, "Heterogeneous graph structure learning for graph neural networks," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 35, no. 5, 2021, pp. 4697–4705.

[29] P. Goyal, S. R. Chhetri, and A. Canedo, "dyngraph2vec: Capturing network dynamics using dynamic graph representation learning," *Knowledge-Based Systems*, vol. 187, p. 104816, 2020.

[30] Y. Ma, Z. Guo, Z. Ren, J. Tang, and D. Yin, "Streaming graph neural networks," in *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2020, pp. 719–728.

[31] K.-H. N. Bui, J. Cho, and H. Yi, "Spatial-temporal graph neural network for traffic forecasting: An overview and open research issues," *Applied Intelligence*, vol. 52, no. 3, pp. 2763–2774, 2022.

[32] B. Yu, H. Yin, and Z. Zhu, "Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting," pp. 263–272, 2018.

[33] Y. Fan, M. Ju, C. Zhang, and Y. Ye, "Heterogeneous temporal graph neural network," in *Proceedings of the 2022 SIAM International Conference on Data Mining (SDM)*. SIAM, 2022, pp. 657–665.

[34] S. Yan, Y. Xiong, and D. Lin, "Spatial temporal graph convolutional networks for skeleton-based action recognition," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 32, no. 1, 2018.

[35] A. Rauhvargers *et al.*, *Global university rankings and their impact*. European University Association Brussels, 2011.

[36] M. A. Fauzi, C. N.-L. Tan, M. Daud, and M. M. N. Awalludin, "University rankings: A review of methodological flaws," *Issues in Educational Research*, vol. 30, no. 1, pp. 79–96, 2020.

[37] U. N. . W. Report, "How u.s. news calculated the 2016 best colleges rankings," 2016.

[38] QS, "Qs world university rankings: Methodology," 2022.

[39] L. Angelis, N. Bassiliades, and Y. Manolopoulos, "On the necessity of multiple university rankings," *COLLNET Journal of Scientometrics and Information Management*, vol. 13, pp. 11 – 36, 2019.

[40] S. W. Raudenbush and A. S. Bryk, *Hierarchical linear models: Applications and data analysis methods*. sage, 2002, vol. 1.

[41] V. Tinto, "Dropout from higher education: A theoretical synthesis of recent research," *Review of educational research*, vol. 45, no. 1, pp. 89–125, 1975.

[42] R. S. Baker, K. Yacef *et al.*, "The state of educational data mining in 2009: A review and future visions," *Journal of educational data mining*, vol. 1, no. 1, pp. 3–17, 2009.

[43] D. Lian, Y. Ye, W. Zhu, Q. Liu, X. Xie, and H. Xiong, "Mutual reinforcement of academic performance prediction and library book recommendation," in *2016 IEEE 16th International Conference on Data Mining (ICDM)*, 2016, pp. 1023–1028.

[44] W. Xing and D. Du, "Dropout prediction in moocs: Using deep learning for personalized intervention," *Journal of Educational Computing Research*, vol. 57, no. 3, pp. 547–570, 2019.

[45] N. Kapur, N. Lytkin, B.-C. Chen, D. Agarwal, and I. Perisic, "Ranking universities based on career outcomes of graduates," in *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, 2016, pp. 137–144.

[46] L. Page, S. Brin, R. Motwani, and T. Winograd, "The pagerank citation ranking: Bringing order to the web," 1998.

[47] K. Pearson, "Vii. note on regression and inheritance in the case of two parents," *proceedings of the royal society of London*, vol. 58, no. 347-352, pp. 240–242, 1895.

[48] J. M. Kleinberg, "Authoritative sources in a hyperlinked environment," *Journal of the ACM (JACM)*, vol. 46, no. 5, pp. 604–632, 1999.

[49] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[50] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," *arXiv preprint arXiv:1301.3781*, 2013.

[51] P. Guo, K. Xiao, H. Zhu, and Q. Meng, "Preference-constrained career path optimization: An exploration space-aware stochastic model," in *2023 IEEE International Conference on Data Mining (ICDM)*. IEEE, 2023, pp. 120–129.

[52] Q. Le and T. Mikolov, "Distributed representations of sentences and documents," in *International conference on machine learning*. PMLR, 2014, pp. 1188–1196.

[53] P. Goyal, N. Kamra, X. He, and Y. Liu, "Dyngem: Deep embedding method for dynamic graphs," *arXiv preprint arXiv:1805.11273*, 2018.

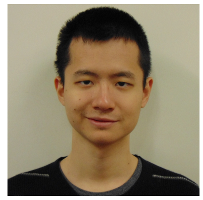
[54] S. Lan, Y. Ma, W. Huang, W. Wang, H. Yang, and P. Li, "Dstagnn: Dynamic spatial-temporal aware graph neural network for traffic flow forecasting," in *International Conference on Machine Learning*, 2022. [Online]. Available: https://api.semanticscholar.org/CorpusID:250340665

[55] C. M. Bishop and N. M. Nasrabadi, *Pattern recognition and machine learning*. Springer, 2006, vol. 4, no. 4.

[56] L. Van der Maaten and G. Hinton, "Visualizing data using t-sne." *Journal of machine learning research*, vol. 9, no. 11, 2008.

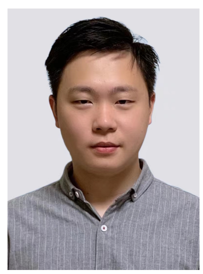
Yuyang Ye received the matser's degree in computer science and technology from the University of Science and Technology of China (USTC), Hefei, China, in 2020. Before that, he received the BEng degree in computer science and technology from USTC, Hefei, China, in 2017. He is currently working toward the Ph.D. degree in the Management Science and Information Systems Department at the Rutgers, the State University of New Jersey. His research interests include representation learning, social network analysis, recommender systems, and their applications in business intelligence. He was the recipient of the KDD CUP 2019 PaddlePaddle Special Award.

Hengshu Zhu (SM'19) is currently the head of BOSS Zhipin Career Science Lab (CSL). He received the Ph.D. degree in 2014 and B.E. degree in 2009, both in Computer Science from University of Science and Technology of China (USTC), China. His general area of research is data mining and machine learning, with a focus on developing advanced data analysis techniques for innovative business applications. He has published prolifically in refereed journals and conference proceedings, such as Nature Communications, IEEE TKDE, IEEE TMC, ACM TOIS, SIGKDD, SIGIR, and NeurIPS. He served as the program co-chair of KDD CUP-2019 Regular ML Track, the industry chair of PRICAI-2022, the area chair of AAAI and IJCAI, and regularly as the (senior) program committee members in numerous top conferences. He was the recipient of the Distinguished Dissertation Award of CAS (2016), the Distinguished Dissertation Award of CAAI (2016), the Special Prize of President Scholarship for Postgraduate Students of CAS (2014), the Best Student Paper Award of KSEM-2011, WAIM-2013, CCDM-2014, and the Best Paper Nomination of ICDM-2014 and WSDM-2022. He is the senior member of ACM, CAAI, CCF and IEEE.



Tianyi Cui received a master's degree in computer science from the University of Washington in 2022. Before that, he received the BEng degree in computer science and technology from USTC, Hefei, China, in 2018. He is currently working as a software engineer at Bytedance.

Runlong Yu received his Ph.D. degree in computer science and technology at the University of Science and Technology of China (USTC) in Hefei, China, in 2023, following his BEng degree in computer science and technology from USTC, in 2017. He is currently a Postdoctoral Associate in the Department of Computer Science at the University of Pittsburgh. His research interests include data mining, machine learning, evolutionary computation, AI for science, and recommender systems. He has published over twenty papers in refereed journals and conference proceedings including IEEE TKDE, ACM SIGKDD, ACM CIKM, ACM SIGIR, PAKDD, IEEE ICDM, IJCAI, AAAI, and DASFAA. He was the recipient of the KDD CUP 2019 PaddlePaddle Special Award and the 2020 CCF BDCI Contest championship.



Le Zhang is currently a researcher at Baidu Research. He received his B.E. degree in Software Engineering from Dalian University of Technology in 2016. And he received his Ph.D. degree in Computer Science from University of Science and Technology of China (USTC) in 2022. His general research interests are data mining and machine learning, with a focus on spatio-temporal modeling and its applications in business intelligence.

Hui Xiong (F'20) is a Chair Professor, Associate Vice President (Knowledge Transfer), and Head of the AI Thrust at Hong Kong University of Science and Technology (Guangzhou). His research interests span Artificial Intelligence, data mining, and mobile computing. He obtained his PhD in Computer Science from the University of Minnesota, USA. Dr. Xiong has served on numerous organization and program committees for conferences, including as Program Co-Chair for the Industrial and Government Track for the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD), Program Co-Chair for the IEEE 2013 International Conference on Data Mining (ICDM), General Co-Chair for the 2015 IEEE International Conference on Data Mining (ICDM), and Program Co-Chair of the Research Track for the 2018 ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. He received several awards, such as the 2021 AAAI Best Paper Award and the 2011 IEEE ICDM Best Research Paper award. For his significant contributions to data mining and mobile computing, he was elected as a Fellow of both AAAS and IEEE in 2020.