Engineering Applications of Artificial Intelligence 126 (2023) 107147

Contents lists available at ScienceDirect

Engineering Applications of Artificial Intelligence

journal homepage: www.elsevier.com/locate/engappai

An improved heterogeneous graph convolutional network for job recommendation

Hao Wang, Wenchuan Yang, Jichao Li *, Junwei Ou, Yanjie Song, Yingwu Chen

College of Systems Engineering, National University of Defense Technology, Changsha, China

ARTICLE INFO

Keywords:
Heterogeneous graph convolutional network
Job recommendation
Resume classification

ABSTRACT

Job recommendation is crucial in online recruitment platforms due to the overwhelming number of job postings. Job seekers spend considerable time and effort searching for suitable employment. With millions of job seekers browsing job postings daily, the demand for accurate and effective job recommendations is more pressing than ever. To address this challenge, we propose IHGCN, an improved semi-supervised heterogeneous graph convolutional network model for job recommendation. IHGCN aims to provide job recommendations for early job seekers based on their resumes. Firstly, we introduce a novel labeling classification standard specifically tailored to early job seeker resumes. Secondly, we construct a heterogeneous resume graph where each resume is represented as a node. Job recommendation is treated as a multi-classification problem. Thirdly, our IHGCN model learns a node representation from the graph to perform effective job recommendations. To evaluate our model, we conduct experiments using a real-world resume dataset obtained from LinkedIn. The results demonstrate that IHGCN outperforms the baselines by around 10%. This study highlights the benefits of leveraging meta-paths within the Graph Convolutional Network model to address the sparsity problem caused by the one-hot representation of nodes.

1. Introduction

In recent years, the rapid advancement of AI-related technologies has led to the emergence of various online recruitment platforms such as linkedin.com ,[1] indeed.com ,[2] maimai.cn ,[3] careerbuilder.com .[4] These platforms aim to provide services for both job seekers and employers. One of the key tasks for recruitment platforms is to recommend suitable job opportunities to job seekers. According to data from the 2022 LinkedIn Pressroom,[5] LinkedIn alone holds a user base of over 850million, with more than 58 million listed companies, over 120,000 schools, and approximately 39,000 skills listed. With such a vast number of job vacancies and job seekers, it becomes challenging for recruiters and recruitment platforms to promptly identify the right match. Intuitively, early job seekers face the initial challenge of determining the specific job they are willing to commit to. In practice, recruiters pay more attention on two crucial aspects of resumes: education background and skill level. The educational background reflects job seekers' commitment to long-term endeavors, while the skill level demonstrates their ability to acquire new knowledge and skills.

The traditional recruitment process heavily relies on subjective experience, making it difficult to quantitatively and comprehensively evaluate resume information. Classical machine learning methods have been used for job recommendations based on work history (Paparrizos et al., 2011), but they cannot be directly applied to early job seekers. Some deep learning related methods, such as Convolutional Neural Networks (CNN) and Long Short-Term Memory (LSTM), have been applied in the recruitment process (Qin et al., 2020; Flambeau and Norbert, 2021; He et al., 2019) but not specifically for job

* Corresponding author. College of Systems Engineering, National University of Defense Technology, No. 109, Deya Road, Kaifu District, Changsha City, Hunan Province, China.

E-mail addresses: crystalwang95@163.com (H. Wang), wenchuanyang97@163.com (W. Yang), ljcnudt@hotmail.com (J. Li), junweiou@163.com (J. Ou), songyj_2017@163.com (Y. Song), ywchen@nudt.edu.cn (Y. Chen).

[1] "LinkedIn." https://www.linkedin.com/feed/.

[2] "Job Search | Indeed." https://www.indeed.com/.

[3] "Maimai-achieve career dreams." https://maimai.cn/.

[4] "Find a Job | CareerBuilder." https://www.careerbuilder.com/.

[5] "About Us." https://news.linkedin.com/about-us.

https://doi.org/10.1016/j.engappai.2023.107147
Received 9 March 2023; Received in revised form 29 July 2023; Accepted 11 September 2023

recommendation (Kavianpour et al., 2021a, 2021b, 2022a). Furthermore, existing researches heavily relies on manual features and expert knowledge, which is costly, difficult to update, and prone to errors (Seveso et al., 2021). In fact, collecting a reliable resume dataset (Zhu et al., 2016) poses challenges due to privacy concerns. Resumes are typically sent directly towards target recruiters and not freely available. Moreover, few datasets contain valid matches between job offers and resumes, making it challenging to obtain ground truth. Additionally, most research focuses on labor market participants (Dave et al., 2018) rather than early job seekers, such as graduates who have not yet entered the labor market. Considering the impact of COVID-19,[6] universities have expanded the enrollment,[7] leading to a significant increase in the number of fresh graduates entering the job market in the coming years. However, fresh graduates have often lack sufficient knowledge about the job market and struggle to make informed decisions regarding job selection. Therefore, it is imperative to develop a personalized and accurate model to recommend suitable jobs for early job seekers. The challenge lies in training a job classifier specifically tailored to early job seekers using unstructured plain text resume data.

To address these challenges, we propose an **I**mproved semi-supervised **H**eterogeneous **G**raph **C**onvolutional **N**etwork model (IHGCN) for job recommendation. Specifically, we first construct a large heterogeneous resume graph by using a dataset of early jobseekers' resume. In this graph, each node represents an individual job seeker's resume, and the edges are formed based on the "also-viewed" attribute, which is derived from LinkedIn's underlying statistics. By formulating our tasks as a node classification problem within this graph, we aim to provide effective job recommendations. To overcome the sparsity of node embeddings (Kipf and Welling, 2016), we introduce meta-paths to the adjacency matrix of the Graph Convolutional Network (GCN) model. By incorporating these meta-paths, we enhance the connectivity and information flow within the graph. This augmentation ultimately enables us to obtain enriched node embeddings, which serve as the foundation for generating accurate job recommendations through the IHGCN model. To summarize, our work makes the following contributions:

- We propose a novel framework for job recommendation problem, consisting of key components. Firstly, we construct a heterogeneous resume graph by preprocessing the job seekers' resume dataset. Secondly, we leverage the IHGCN model to obtain node embeddings, which capture the essential features of the job seekers. Lastly, we utilize the node classification approach to provide suitable job recommendations.
- We propose the IHGCN model, an improved semi-supervised heterogeneous GCN model, specifically designed to provide job recommendations for early jobseekers. By incorporating meta-paths into the GCN model and addressing the sparsity problem caused by one-hot representations, our model offers enhanced performance and accuracy in job recommendations.
- We conducted extensive experiments on real-world datasets to evaluate the effectiveness of our proposed model compared with state-of-the-art baselines.

This paper is organized as follows in the subsequent sections. Section 2 introduces the relevant literature on Human Resource Management (HRM) and Graph Neural Network (GNN). Section 3 presents the necessary preliminaries and provide a formal definition of the problem. Section 4 give the overview and details of IHGCN model. Section 5 describe and analyze the experimental results. Section 6 conclude and propose perspectives.

2. Related works

In this section, we briefly introduce some researches related to this paper. Data-driven HRM and GNN model is related to our work from the application view and the technical view respectively.

2.1. Data-driven Human Resource Management

Recruitment plays a pivotal role in HRM. The emergence of online recruitment platforms has provided large and informative recruitment datasets, opening up new perspectives for recruitment analysis (Xu et al., 2018)– (Kenthapadi et al., 2017). For instance, Boselli et al. (2017) collected online job vacancies over 7 million over 5 EU countries and automatically classified them under ESCO taxonomy via machine learning for labor market intelligence. In addition, Dave et al. (2018) gathered a dataset of 20 million resumes from CareerBuilder.com and proposed a jointly representation learning model to offer better job and skill recommendations. Furthermore, Liu et al. (2019) introduced the Industrial and Professional Occupation Dataset (IPOD), which includes over 192,000 job titles from 56,648 LinkedIn profiles. This dataset serves as a valuable resource for both researchers and professionals in the field. Given the challenges faced by job seekers and recruiters regarding market dynamics, Zhang et al. (2016) proposed a detailed generalized linear mixed model called GLMix. This model demonstrated the ability to generate between 20% and 40% more job applications for job seekers in the LinkedIn job recommender system. At the fine-grained level, Wu et al. (2019) analyzed 2,284,903 job advertisements posted by 147,690 companies on Lagou websites (Zhu et al., 2016) and introduced a trend-aware approach for analyzing skill demand.

Person-job fit is the key task of online talent recruitment. Recently, studies about matching the job requirements and resumes have become a conspicuous topic in intelligent HRM domain. To reduce the reliance on human labor, Qin et al. (2018) proposed a novel end-to-end Ability-aware Person- Job Fit Neural Network model, namely APJFNN, aiming at measuring the matching degree between resume and job requirements. Some researchers try to achieve person-job fit by reducing the skill gap in labor market. Börner et al. (2018) analyzed and visualized skill discrepancies between academic, industry, and education in data science and data engineering (DS/DE) domain, revealing the critical need to provide soft skills for the data economy. Besides, Xu et al. (2018) developed a data-driven model for measuring the popularity of job skills and help talents choosing the right skill to learn and then staying in gainful employment. In fact, recruitment is a two-way selection process, which should meet the bilateral needs of both job seekers and recruiters. Yang et al. (2022a) suggested a method that utilizes dual-perspective graph representation learning to model directed interactions between job seekers and jobs. In order to bridge the semantic gap between textual job postings and textual resumes, Yao et al. (2022) suggested a person-job fit approach that enhances the model with external knowledge fused into the graph representation learning.

Besides, some studies provide better skill and job recommendations based on job seekers' current skill sets and job requirements. Referring to the recommendation questions, there are currently three main methods: content-based filtering (Lops et al., 2011), collaborative filtering (Koren et al., 2021) and hybrid approaches (Basilico and Hofmann, 2004). Content-based methods make recommendations based on the similarity of resumes, but face the challenges of over-specialization and the need for a large amount of labeled data. In addition, collaborative filtering methods (Papadakis et al., 2022), (Alhijawi and Kilani, 2020) try to recommend item based on user behavior similarities, but suffering from new-user problem and sparsity problem. Hybrid approaches (Sun et al., 2021), (Velickovic et al., 2017) is a combination of more than one filtering method into a unified system. Besides, some

[6] "Four-Year Colleges with the Highest Enrollment Growth During COVID-19." https://www.onlineu.com/magazine/college-enrollment-increases-during-covid.

[7] "This year's master's degree, college-education expansion of 511,000 people - Xinhua." http://www.xinhuanet.com/politics/2020-05/13/c_1125976668.htm.

researchers use utility-based (Zihayat et al., 2019), Knowledge-based (Tarus et al., 2018) and demographic approaches (Pazzani, 1999) to solve the recommendation problems. Considering the bilateral satisfaction, reciprocal recommender system define the task as an user-to-user recommendation problem,which is also applied in recruitment (Su et al., 2022). Dave et al. (2018) build three types of information networks, job transition networks, job skill networks, and skill co-occurrence networks from historical job data, and proposed a representation learning model aimed at providing better jobs and skill recommendations. In addition, researches on skill recommendation suggests that due to the dynamics of skill learning and the uncertain benefits, talents lack sustainable guidance in vocational skill learning. Sun et al. (2021) introduced a recommender system based on deep reinforcement learning that is both cost-effective and capable of providing personalized job skill recommendations.

The vast volume of data generated by online recruitment platforms has created an opportunity for intelligent HRM. Numerous studies have focused on areas such as person-job fit, skills and job recommendation. However, to our best knowledge, there is currently a lack of research specifically addressing job recommendation for early job seekers. This knowledge gap underscores the urgency and importance of conducting this study, which aims to fill this void and provide effective job recommendations tailored to the needs of early job seekers.

2.2. Graph Neural Network

Research on GNNs is closely intertwined with the field of graph embedding and has garnered increasing interest in the realm of machine learning. Graph embedding aims to represent nodes in a low-dimensional vector space while preserving the network topology and node information for various tasks such as classification, clustering, and recommendation. One significant GNN model designed specifically for graph data is the GCN. GCN (Kipf and Welling, 2016) serves as a semi-supervised classification framework, particularly useful when only a small subset of node labels are available in the graph. In GCN, the feature vector of each node is updated by aggregating the feature vectors of its neighboring nodes, enabling the refinement of node representations. In the traditional GCN approach, all neighboring nodes are considered equally important. However, in practical scenarios, certain nodes possess greater importance than others within the graph. To address this limitation, Graph attention networks (GAT) (Velickovic et al., 2017) offer a solution by introducing an attention mechanism that assigns weight factors to each connection based on the importance of the respective neighbor. The attention mechanism of GAT enables each node to dynamically adapt its attention weights for neighboring nodes, facilitating a more refined capturing of node relationships.

In recent years, GNNs have gained significant attention and have been widely utilized in various problem domains due to their exceptional performance. In the field of Natural Language Processing (NLP), GNNs have been applied to text classification (Yao et al., 2019a), (Hamilton et al., 2017) referring use inter-relationships between documents or words to infer document labels. Additionally, GNNs have been employed in graph-based recommendation systems (Pradhyumna and Shreya, 2021). This involves learning node-level or graph-level representations and utilizing the relationships between nodes along with their content information to provide personalized recommendations. GNNs have also made substantial contributions to computer vision-related domains. They have been extensively used in tasks such as human-object interaction (Yang et al., 2022b), scene graph generation (He et al., 2022), few-shot image classification (Chen et al., 2022), action recognition (Yao et al., 2019b), (Hao et al., 2021) semantic segmentation (Li et al., 2021), visual inference (Guo et al., 2021) and question answering (Narasimhan et al., 2018), (Yasunaga et al., 2021). GNNS also are applied in program verification (Si et al., 2018), social influence prediction (Qiu et al., 2018), event detection (Nguyen and Grishman, 2018), taxi-demand prediction (Yao et al., 2018) and bearing fault diagnosis (Ghorvei et al., 2023), (Kavianpour et al., 2022b). The application of GNNs in these areas has shown promising results, enhancing the understanding and interpretation of visual data.

As mentioned above, GNN-related variants have been widely used to solve related problems in various fields, except for recruitment, due to the ability to learn without labor-intensive feature engineering. Traditionally, graph structures are used to manage rich data. Surprisingly, deep graph learning models have not been studied for job recommendation problems. Moreover, these studies ignored early job seekers with rarely work experience. In this paper, we propose an improved semi-supervised classification with HGCN model for job recommendation, namely IHGCN, which is well-designed for early job seekers.

3. Dataset description and problem definition

In this section, we will give a brief description about the dataset, and then analyze and define the problem.

3.1. Dataset description

The dataset used is randomly selected from a larger corpus consisting of 2,985,414 publicly available LinkedIn user profiles (Zhang et al., 2015). Each profile within this dataset contains a variety of information written in natural language, including personal details, educational background, skills related information and so on. For our purposes, we consider each profile on the LinkedIn platform as a user resume. To establish connections between resumes, we utilize the “also-viewed” attribute which is a list of other resumes that have been viewed by users who have viewed this resume. The “also-viewed” attribute indicates the existence of a potential relationship between these resumes via LinkedIn’s underlying algorithm. Our research subject is early job seekers, which mainly consist of fresh graduates with barely no work experience who have not yet formally entered the labour market. The resumes of early job seekers typically contain information regarding their educational background and relevant skills. It is the target group that we aim to provide job recommendations for. Before conducting our analysis, we preprocess the resume dataset and obtain a dictionary comprising various entities, including 4,534° entities, 6,402 major entities, 9,307 university entities, 16,299 skill entities, and 832 industry entities. These entities play a crucial role in representing and understanding the information contained in the resumes.

Additionally, we have devised an interpretable labeling method that takes into account various factors such as the dataset’s nature, the insights of HR experts, the classification criteria used by mainstream recruitment platforms, and established classification standards.

To ensure accuracy and consistency in labeling, we engaged the expertise of three human resource domain experts, each possessing more than five years of experience in recruiting. The manual labeling and preprocessing of the entire dataset required over 500 h manual effort. The process of manual labeling and preprocessing of the resume information is visualized in Fig. 1.

In this study, jobs are classifying into six categories, including *Technical, Management, Operational, Marketing, Functional, and Research & Development.* The raw resume data written in natural language contains a wide range of categories for each feature attribute. Thus, we streamlined the labeling process by assigning labels based on specific criteria. We divide all input resume information into two dimensions: Education and Skill. Specifically, the education dimension is divided into three related attributes: Major (*Maj*), Degree (*Deg*), and Uname (*Unr*). The following are the label criteria for different feature attributes:

- Major is classified into six categories: *Arts & Humanities, Engineering & Technology, Life Sciences & Medicine, Natural Sciences, Social Sciences & Management, and Others.*
- Degree is grouped into six classes: *Postdoc, Doctor, Master, Bachelor, Pre-college, and Others.*

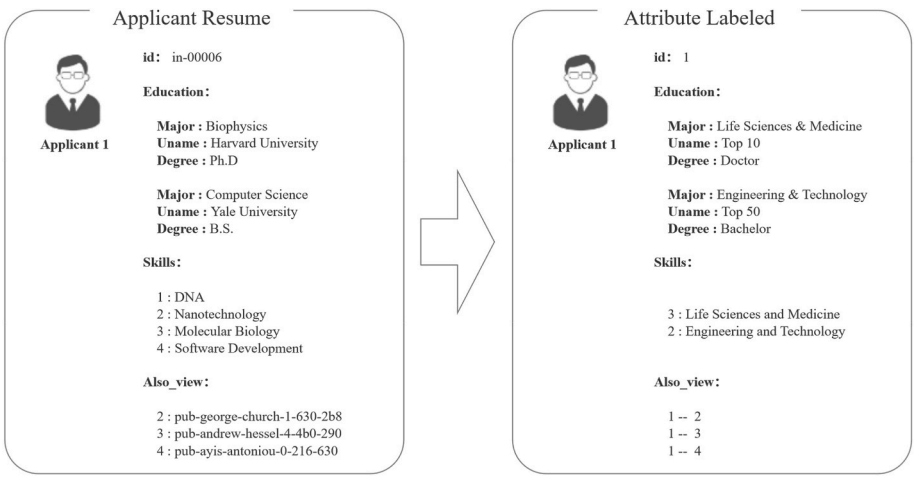


Fig. 1. The schematic diagram of data pre-processing.

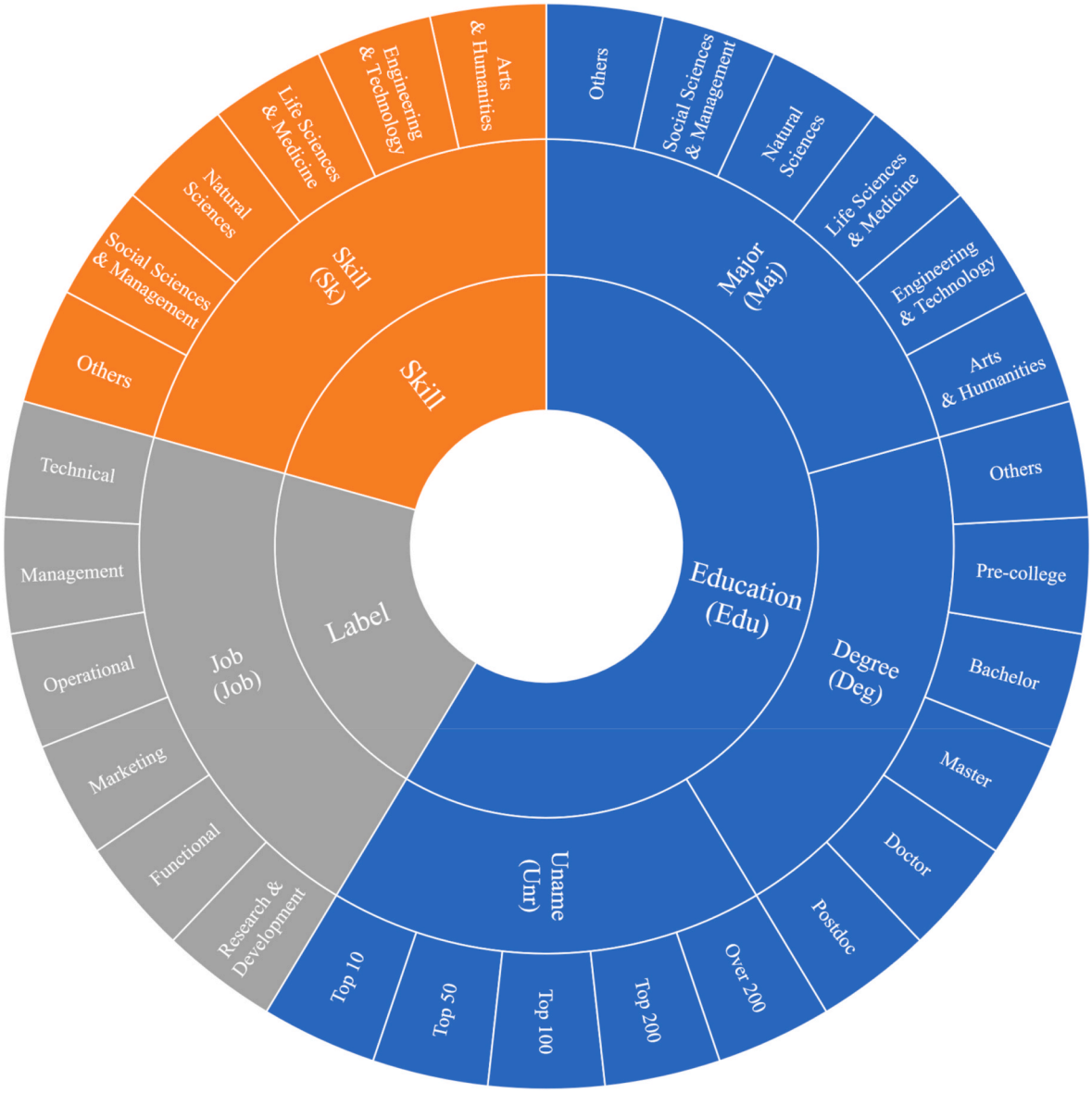


Fig. 2. The detailed description of the input attributes and output classes. The input features are two dimensions: Education and Skill. Specifically, the Education dimension is further divided into three attributes: Major, Degee, and Uname.

- Uname, representing the world rankings of universities, is divided into five levels: *Top 10, Top 50, Top 100, Top 200, and Over 200.*
- Skill is grouped into six types: *Arts & Humanities, Engineering & Technology, Life Sciences & Medicine, Natural Sciences, Social Sciences & Management, and Others.*

For greater clarity, we provide a detailed overview of the specific 23 input attributes and 6 output categories in Fig. 2. In addition, the detailed explanation of each attribute can be found in Table 2. Besides, the detailed descriptive information and abbreviations related to the output job labels are listed in Table 1.

3.2. Problem formulation

In this section, we first analyze the problem, then define several related concepts and symbols, and finally describe the problem studied in this paper.

Early job seekers, or fresh graduates, lack work experience but possess a certain level of education, skills, and learning ability. Due to their limited job market knowledge, they often struggle to make informed job choices. Additionally, the abundance of job opportunities further complicates their decision-making process, requiring them to invest a significant amount of time to find a suitable career path. To address this issue, this study proposes an effective solution that leverages education and skills-related information from resumes to assist early job seekers in quickly identifying suitable job opportunities.

Job recommendation research based on early job applicant resume data aims to provide decision support for job applicants, recruiting companies, and recruitment platforms. For job seekers, recommendations based on real datasets offer greater reliability, reduce the time spent on job search, and enhance the potential for better career development. Recruiting companies can efficiently screen out unsuitable job seekers from a large pool of resumes, thereby saving valuable manpower resources. On the other hand, recruitment platforms can benefit from accurate job recommendations by expediting the person-job matching process, improving the recruiting success rate, and enhancing the overall user experience.

3.2.1. Definition 1. Heterogeneous network

A heterogeneous graph is defined as $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$, where $\mathcal{V}$ is the node set and $\mathcal{E}$ represent the edge set. The network schema of the graph denoted as, $T_G = (\mathcal{A}, \mathcal{L})$, serve as a meta-template for the graph $\mathcal{G}$. The mapping function from the node set to the node type set is represented by $\varphi(v) : \mathcal{V} \rightarrow \mathcal{A}$, while the link set is mapped to the link type set by $\varphi(e) : \mathcal{E} \rightarrow \mathcal{R}$. For each node, $v \in V$ is associated with a typical node type $\varphi(v) \in \mathcal{A}$. Also, for each link $e \in E$ is associated with one typical link type $\varphi(e) \in \mathcal{L}$. The sets of node types and link types are denoted by $\mathcal{A}$ and $\mathcal{L}$, respectively. In Heterogeneous Network, $|\mathcal{A}| + |\mathcal{L}| > 2$.

3.2.2. Definition 2. Meta-path

A meta-path is a path defined on the graph of network schema $T_G = (\mathcal{A}, \mathcal{L})$, and represent a sequence of links between node types: $\mathcal{P} = \mathcal{A}_1 \xrightarrow{L_1} \mathcal{A}_2 \xrightarrow{L_2} \ldots \xrightarrow{L_k} \mathcal{A}_{k+1}$, in which $\mathcal{T}_i \in \mathcal{T}, i \in \{1, 2, \ldots, k+1\}$ and $\mathcal{L}_k \in \mathcal{L}$, $i \in \{1, 2, \ldots, \ell\}$. The meta-path defines a collection of relation $\mathcal{L} = \mathcal{L}_1 \mathcal{L}_2 \cdots \mathcal{L}_k$ between type $\mathcal{A}_1$ and $\mathcal{A}_{k+1}$, in which $\circ$ represents the composition operator on relations. Meta-paths extend the concept of link types in heterogeneous networks and carry distinct semantics depending on their composition.

Table 1
The detailed description of data classification.

Job Category	Abbreviation	Number of samples
Technical	Tech	3833
Management	Mgmt	643
Operational	Ops	577
Marketing	Mktg	4186
Functional	Func	107
Research & Development	R&D	653

Table 2
Important mathematical symbols.

Symbols	Description
Res	Represents a collection of resumes for early job seekers
Edu	Represents educational background information in resume
Deg	Represents academic qualifications information in resume
Unr	Represents school world ranking information in resume
Maj	Represents major information in resume
Sk	Represents skill-related information in resume
Job	Represents job information in resume

The purpose is providing job recommendations for early jobseekers based on their resumes. We use $Res = \{r_1, r_2, \ldots, r_n\}$ to denote n early jobseekers' resumes. As shown in Fig. 1, resumes mainly include two parts: educational background and skill related information, which is denoted as $Res = \{Edu, Sk\}$. Every job applicant may have several educational experiences, denoted as $Edu = \{e_1, e_2, \ldots, e_t\}$. The smaller the t, the earlier the job seeker's educational experience. In other words, e_t represent the most recent educational experience. As shown in Fig. 1, educational background *Edu* on resumes includes three aspects: academic qualifications *Deg*, the world ranking of university *Unr* and major *Maj*. The educational background of the early jobseekers can be denoted as $Edu = \{Deg, Unr, Maj\}$. Skill-related information *Sk* is labeled as j types, defined as $Sk = \{s_1, s_2, \ldots, s_j\}$.

Moreover, we treated the job data as categorical data due to the excessive complexity and scatter, and denoted as $Job = \{T_1, T_2, \ldots, T_o\}$. Thus, resume information can be described by a set of attributes, denoted as $Res = \{Deg, Unr, Maj, Sk\}$, where r_* represents attribute information for each resume. The "also-viewed" data is denoted by $V_{n_1 \times n_2} = \{v_{r_1 r_2} | r_1 \in Res, r_2 \in Res\}$. Overall, this study aims to learn a multi-class classification model F and provide recommendations on job *Job* based on resume information, $\forall_{r \in R}, Job = \underset{n \in R}{\operatorname{argmax}} F(r_n | Deg, Unr, Maj, Sk)$.

For better understanding, we provide a comprehensive list of the significant mathematical symbols employed in Table 2. And, Fig. 3 illustrates the correlation matrix among each input variable.

4. Method

In this section, we introduce the proposed IHGCN model, an

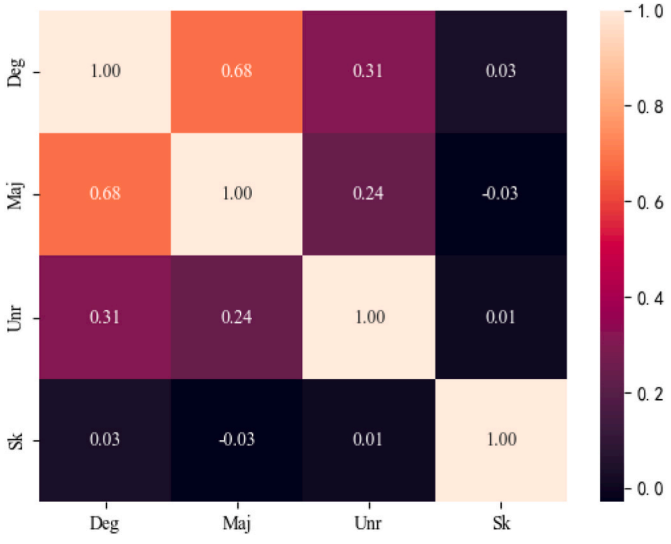


Fig. 3. The correlation matrix between each input variable.

Improved semi-supervised **HGCN** model for Job Recommendation (IHGCN) . The overall framework of IHGCN is presented in Fig. 4. The IHGCN model mainly contains three modules. The first part is to obtain the jobseekers' resume representation. The second part is the heterogeneous graph convolutional network with meta-path, which captures the relations between jobseekers. The third module is to train and predict for job recommendation. This model helps early job seekers select suitable jobs by analyzing their education and skill-related information.

4.1. Heterogeneous resume graph construction

The first phase is constructing the resume graph. In this part, we aim at transforming the original resume data *Res* into a heterogeneous graph. We let $G = (Res, V)$ denote the resume

dataset, where *Res* is the set of resumes, *V* is the set of links. The number of nodes in the graph *N* is the number of resumes (id size) in the dataset. The goal is to learn the function of features on the resume graph $G = (A_G, L_G)$.

4.1.1. Feature matrix construction

In the graph, each vertex corresponds to an early jobseeker's resume, as shown in Fig. 4. The feature matrix $X_\omega \in \mathbb{R}^{N_e \times d_f}$, where d_f is the dimension of feature vector. We use $X_\omega{}^{(i)}$ to denote the feature vector of i_{th} jobseeker. We simply set the feature matrix $X = I$ as an identity matrix. Each resume is denoted as a one-hot vector as the input to IHGCN.

4.1.2. Adjacency matrix construction

As shown in the left bottom of Fig. 4, we combine the adjacency matrix *B* with meta-paths matrix. Firstly, we use attribute "also-viewed" to construct adjacency matrix *B* between users' resumes. The "also-viewed" attribute contains a list of users shown on the user's LinkedIn homepage, which means users who see this user's resume also sees those ones. In short, this user' s resume is similar to the resumes in the also-viewed list based on the LinkedIn's data analyze. In addition, we assume that two resumes with similar specific meta-paths are more comparable in job selection. Specifically, we aggregate the adjacency matrix *B* and different meta-paths $\mathscr{P}$ as a new adjacency matrix. We obtain node embeddings under different meta-paths separately and then aggregate them to get the final embedding. We present a toy example in Fig. 4 to illustrate this process.

Different meta-paths capture the semantic relationships between early jobseekers from different views, as shown in Table 3. For example, the meta-path of "RDR" indicates two early jobseekers have the same academic degree level. And, meta-path "RSR" represents two resumes have the similar skill type. Generally speaking, early jobseekers, who have certain similar meta-paths, are more likely to be qualified for the same job.

Table 3
The semantic meaning of meta-paths designed for datasets.

Meta-path	Semantic meaning
RDR	Two resumes with the same academic degree level are more likely to be qualified for the same job.
RSR	Two resumes with the same skill type are more likely to be qualified for the same job.
RUR	Two resumes with the same rank level of university are more likely to be qualified for the same job.
RMR	Two resumes with the same major category are more likely to be qualified for the same job.

4.1.3. Graph construction

The Heterogeneous Resume Graph *G* proposed in our IHGCN model is a heterogeneous, attributed, unweighted graph for the sake of simplicity. A resume represents a node, and each node has its own features. We set these nodes features to form an $N \times M$-dimensional feature matrix *X*. The relationship between each node is formed as an $N \times N$-dimensional matrix *B*, also known as the adjacency matrix. Overall, feature matrix *X* and adjacency matrix *B* is the input of our IHGCN model.

Define the heterogeneous resume graph as $G = (A_G, L_G)$. A_G is a composition of different types of vertices and L_G is a collection of various links, where each edge $l_G \in L_G$ and each vertex $a_G \in A_G$. Each node is assumed to be connected to itself, i.e., $(a_G, a_G) \in A_G$ for any a_G. Define feature matrix as $X \in \mathbb{R}^{n \times m}$, where m is the dimension of the feature vectors and n is the number of nodes. Each row $x_a \in \mathbb{R}^m$ is the feature vector for a. Define the adjacency matrix *B* of the heterogeneous resume graph *G* and its degree matrix *D* as $D_{pp} = \sum_p B_{pq}$. The adjacency matrix's diagonal elements *B* are assigned to 1 due to the self-loops. Also, the edges among nodes are built on also-viewed attribute between resumes (resume-resume edges).

4.2. IHGCN model

In IHGCN model, we consider a three-layer heterogeneous graph convolutional network model for semi-supervised node classification. GCNs are to aggregate feature information from a node's first-order

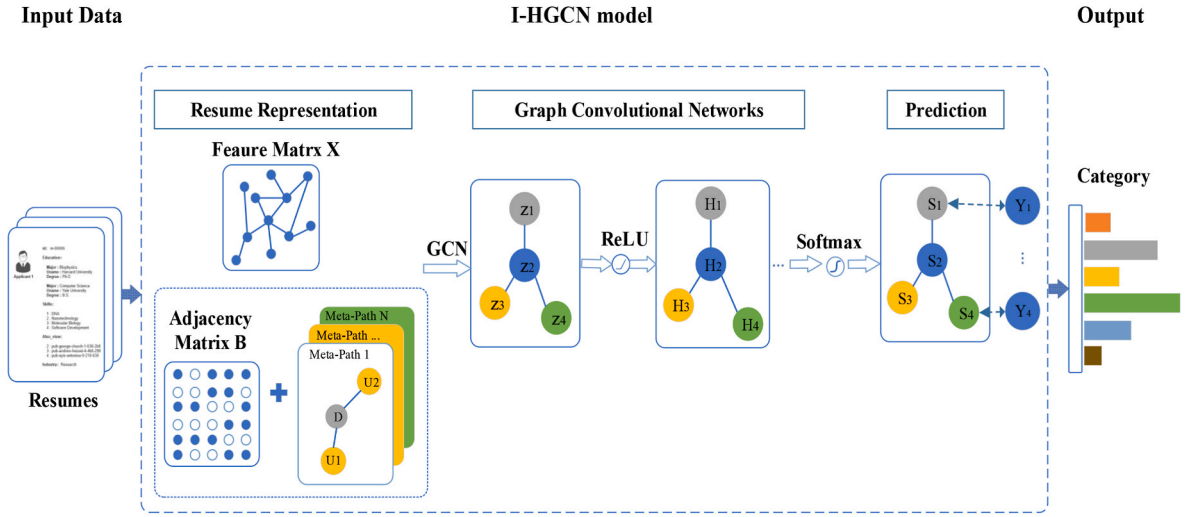


Fig. 4. The overall framework of our proposed IHGCN. The variables of Z, H, S and Y in GCNs denote hidden representations via GCN layers, hidden representations via ReLU layer, hidden representations via SoftMax layer, and labels respectively.

neighbors, as shown in Fig. 5. Therefore, IHGCN model can capture the information about near neighbors with two layers of convolution. IHGCN model operates directly on the graph and induces the node embedding vectors based on their neighborhood properties. When stacking multiple GCN layers, information about larger neighborhoods is aggregated.

The first layer of our IHGCN model is the input layer. Suppose there are N nodes in the resume graph, and each node represents an early jobseeker's resume. Each node has its own features, and these node features form an $N \times D$-dimensional feature matrix X, where N is the number of nodes and D is the number of input features. In addition, a representative description of the graph structure in matrix form is $N \times N$-dimensional adjacency matrix B, which denotes the relationship between each node. X and B are the inputs of our IHGCN model.

The second and third layer of our IHGCN model is the hidden layer. Every layer of the neural network can then be expressed as a non-linear function:

$$f\left(H^{(l)}, B\right) = RELU\left(BH^{(l)}W^{(l)}\right) \tag{1}$$

where $H^{(0)} = X$ and $H^{(L)} = Z$, where l is the number of layers, and Z is the graph-level outputs. Also, $W^{(l)}$ is a weight matrix for the l_{th} neural network layer and $RELU(\cdot)$ is a non-linear activation function ReLU, and $RELU(x) = \max(0, x)$.

To sum up all the feature vectors of the node itself and all neighboring nodes, we enforce the self-loops in the resume graph by adding the identity matrix to B: $\widetilde{B} = B + I$, where I is the identity matrix. Also, in order to take the average of neighboring node features, we normalizing B by summing all rows to one: $D^{-1}B$, where D is the diagonal node degree matrix. And then, the normalized symmetric adjacency matrix $\widetilde{B}$ is calculated as: $\widetilde{B} = \widetilde{D}^{-\frac{1}{2}}B\widetilde{D}^{-\frac{1}{2}}$, where D is a diagonal degree matrix of $\widehat{B}$, $\widetilde{D}_{ii} = \sum_j \widetilde{B}_{ij}$.

In our IHGCN model, every HGCN layer can be updated by:

$$H^{(l+1)} = \sigma\left(\widetilde{D}^{-\frac{1}{2}}\widetilde{B}\widetilde{D}^{-\frac{1}{2}}H^{(l)}W^{(l)}\right) + H^{(l)} \tag{2}$$

where $\widetilde{B}$ is the adjacency matrix and $\widetilde{D}$ is the diagonal node degree matrix of $\widetilde{B}$.

4.3. Prediction and training

After building the resume graph, we feed the graph into a three-layer HGCN model. The first layer is the input layer, and the second and the third layer node (resume) embeddings have the same size as the labels set. And then, we fed the node embeddings into a *softmax* classifier:

$$Z = f(X) = \sigma(ReLU(XW_0)W_1) \tag{3}$$

Where X is the feature matrix, $W_0 \in \mathbb{R}^{C\times H}$ and $W_1 \in \mathbb{R}^{H\times F}$ is the weight matrix for a hidden layer with H feature maps, that from input to hidden layer and from hidden to output layer respectively. σ represent the *Softmax* layer: $soft\max\left(x_i\right) = \frac{\exp(x_i)}{\sum_{j=1}^{C}\exp(x_j)}$. In addition, the function mentioned above generates an output Z at the node level, which is represented as a feature matrix with dimensions $N \times F$, where F corresponds to the number of features produced for each node.

Since our problem is a multi-classification problem, we define the loss function as the Negative Log Likelihood loss:

$$\log P(\mathscr{D}|\theta) = \sum_{i=1}^{n}\log\left(\widehat{y}_{\theta,i}^{(y_i)}\right) \tag{4}$$

where y_i represent only calculate the logarithm of the probability value corresponding to the true category for C categories. For example, if $C = 2$, then $\widehat{y}_{\theta,i}^{(1)} = \widehat{y}_{\theta,i}$ and $\widehat{y}_{\theta,i}^{(1)} = 1 - \widehat{y}_{\theta,i}$, where $y \in (0, C-1)$ for the C categories labels, $\widehat{y}_i \in (0, 1)$ follow the probability distribution and sum to 1.

5. Experiments

This section provides the details of experiments carried out on an actual dataset for verify the effectiveness of our IHGCN model.

5.1. Baselines

For multi-class classification problems, the category is judged by existing feature attributes, and the result is multiple categories. To study the effectiveness and feasibility of IHGCN, we select 8 comparison methods and transform the resume data into the form models needed. For GNN-based algorithms, we chose **GCN** (Kipf and Welling, 2016), **GAT** (Velickovic et al., 2017) and **NodeFormer** (Wu et al., 2022) as baselines for their powerful classification ability. Also, we select some typical graph embedding algorithms, such as **DeepWalk** (Perozzi et al., 2014), **Node2vec** (Grover and Leskovec, 2016), **LINE** (Tang et al., 2015), **SDNE** (Wang et al., 2016), **Struc2vec** (Ribeiro et al., 2017).Some classical methods are chosen as baselines, including Support **Vector Machine (SVM)** (Cortes and Vapnik, 1995), **K-Nearest Neighbor (KNN)** (Tibshirani), **Adaboost (AB)** (Freund and Schapire, 1996), **Random Forest (RF)** (Breiman, 2001), **Logistic Regression (LR)** (Bailly et al., 2022) and **Naive Bayes Classifier (NBC)** (Rrmoku et al., 2022).

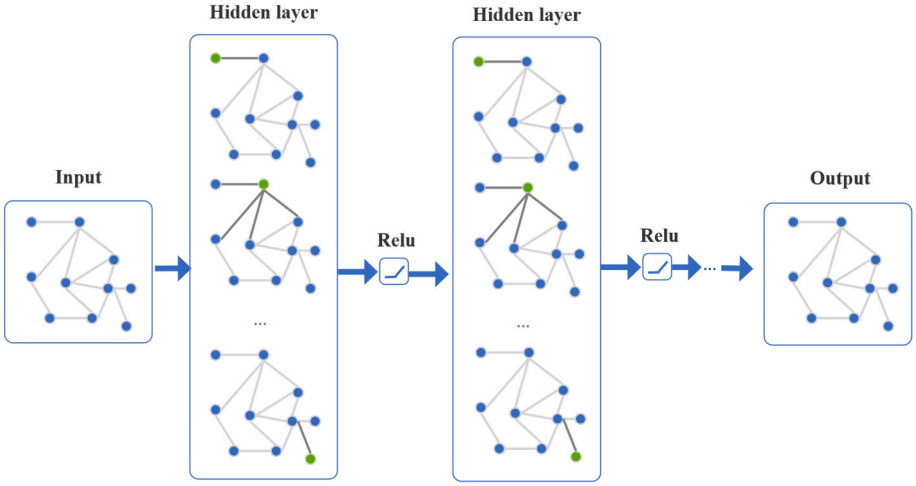


Fig. 5. Schematic diagram of multi-layer graph convolutional neural network structure.

- **GCN** is utilized for semi-supervised learning on data structured in graphs, and it serves as a more efficient version of convolutional neural networks. It is ingeniously designed for the extraction of features from graph data, enabling the use of these features for node classification purposes.
- **GAT** employs an attention mechanism for more effective neighbor aggregation, in contrast to GCN. GAT does not require complex matrix operations or prior knowledge of graph structures. GAT assigns different importance to different nodes in the neighborhood during the convolution process by overlaying self-attention layers.
- **NodeFormer** is a novel Transformer-style network for node classification on large graphs, addressing deficiencies of GNNs. It introduces efficient all-pair message passing scheme and kernelized Gumbel-Softmax operator for learning latent graph structures, demonstrating promising efficacy in various tasks.
- **DeepWalk** is an unsupervised graph embedding model which generates embeddings by simulating random walks on a graph and using skip-gram to learn node representations. It preserves the local structural information of the graph.
- **Node2vec** is a graph embedding method which builds on DeepWalk by introducing biased random walks that can explore both breadth-first and depth-first search strategies, which can capture both the local and global structures of the graph.
- **LINE** generates embeddings by preserving both the first-order (directly connected nodes) and second-order (co-occurring nodes in the neighborhood) proximity of the graph, but not performing well as the other models on highly sparse graphs.
- **SDNE** is a deep learning-based method that reconstructs the adjacency matrix of the graph using a stacked denoising autoencoder. It can preserve both the global and local structures of the graph.
- **Struc2vec** is a clustering-based graph embedding model which introduces a hierarchical clustering technique to group nodes into different levels of granularity, and performs random walks on the hierarchical clusters to generate embeddings.
- **SVM** approaches classification tasks by identifying a separation boundary, known as a classification plane, that maximizes the distance between the boundary and the nearest data points on each side, with the aim of effectively dividing the data points into separate classes.
- **KNN** is relatively mature in theory and simple in idea, and can be used for both classification and regression. The idea of KNN is that if the majority of the K nearest (i.e., the closest neighbor in the feature space) samples near a sample belong to a certain category in the feature space, the sample also belongs to that category.
- **AB** is an approach involves training multiple simple classifiers using the training data, and subsequently merging them into a powerful classifier
- **RF** is a classifier that contains multiple decision trees and can handle multi-classification problems very well. The output class of RF is judged by each decision tree in the forest separately, and finally the predicted class is decided by the class selected the most.
- **LR** is capable of addressing multiple classification tasks and is employed to express the probability of an event occurring. LR is straightforward to deploy, requiring minimal computational power, delivering fast results, and using minimal storage resources during the classification process.
- **NBC** is one of the most widely used classification algorithms based on Bayesian theorem. The essence of the Bayesian classification algorithm is calculating the conditional probability.

5.2. Experimental settings and evaluation metrics

For our proposed IHGCN model, we implemented the experiment in Pytorch Geometric 1.11.0. The experimental environment involved an Intel(R) Core (TM) i7-1065G7 CPU @ 1.30 GHz (8 CPUs). To evaluate the robustness of our proposed IHGCN model, we divided the dataset into three sets: 60% for training, 20% for validation to tune the parameters, and 20% for testing the performance. We apply grid search for tunning the hyper-parameters of the model: the learning rate is tuned amongst {0.00, 0.01, …, 0.05} and the dropout ratio in {0.1, 0.2, …, 0.5}. The obtained optimized value for learning rate is 0.01 and dropout ratio is 0.2. We trained the model over 900 epochs and set the decay weight as 0.0005. In order to generate the same learnable parameters of the model initialization each time, the random number seed we set is 42. In this case, we set the number of hidden IHGCN layers with 16 units each. We adopt 2-step meta-paths for IHGCN model.

To assess the classification performance of all methods, we selected five representative evaluation metrics in the literature (Grandini et al., 2022): accuracy, precision, macro F1_score, and AUC.

- **Accuracy** measures the proportion of predictions that a model is correct. It is calculated by the labels are correctly predicted over the total number of the instances. $Accuracy = \frac{TP+TN}{TP+FP+TN+FN}$.
- **Precision** refers to the proportion of items identified as Positive by the model that are truly Positive. It can be calculated by dividing the number of accurately predicted positive labels by the total number of positive predictions. A higher precision indicates a lower rate of false positive identifications.

$$Precison = \frac{TP}{TP + FP}$$

- **Macro-F1 score** combines both precision and recall under the concept of harmonic mean and finds the best trade-off between the two quantities. Recall is the ratio of correctly predicted positive items divided by the number of positively classified items. $Recall = \frac{TP}{TP+FN}$, $F1\ score = 2 \cdot \left(\frac{Precision \cdot Recall}{Precision + Recall}\right)$ and $macro - F1\ score = \frac{F1-score_1+F1-score_2+F1-score_3+F1-score_4+F1-score_5+F1-score_6}{6}$.

where TP, TN, FP, FN denotes true positives, true negatives, false positives and false negatives, respectively.

- **AUC** represents a probability and is indicative of how well a classification algorithm can distinguish between positive and negative samples. A higher AUC value suggests that the algorithm is more effective at placing positive samples ahead of negative ones, indicating better classification performance. $AUC = \frac{\sum_{i \in PositiveClass} rank_i - \frac{P(1+P)}{2}}{P \times N \times M}$ Where P, N, M represent the number of positive samples, negative samples and classification categories correspondingly.

5.3. Overall performance

To evaluate the effectiveness of IHGCN, we analyze its performance against various machine learning models that exhibit differences in their classifier stages. Also, we evaluate the performance of GAT and GCN as its outstanding performance on other existing studies.

Table 4 summarizes the performance comparison between IHGCN and various baseline models. IHGCN outperforms all the baselines significantly, demonstrating its superior performance. In the following, we discuss these results to gain some important insights. When comparing IHGCN with GNN without meta-paths, it is clear that adding meta-paths to the resume graph in HGNN improves the performance. This suggests that meta-paths capture syntactic and semantic relations among resumes, providing valuable additional information from resume data. The superior performance of our proposed IHGCN over the baselines highlights its ability to effectively combine node features and information from neighboring nodes for analyzing resume data. Overall, deep learning-based GNN algorithms achieve better performance, while GAT performs worse than IHGCN, indicating that unsupervised embeddings from GAT lack discriminative power in job classification. In

Table 4
Overall performance.

	Accuracy	Precision	Macro-F1 score	AUC
IHGCN	**0.8591**	**0.9011**	**0.8218**	**0.8475**
NodeFormer	0.6237	0.2435	0.6322	0.5706
GCN	0.7350	0.4905	0.5173	0.6911
GAT	0.6644	0.6532	0.6189	0.6299
DeepWalk	0.5426	0.1843	0.1929	0.5335
Node2vec	0.6064	0.7669	0.6914	0.6696
Line	0.3830	0.0645	0.0923	0.4960
SDNE	0.3936	0.1407	0.1209	0.4949
Struc2vec	0.4681	0.1964	0.1621	0.5288
SVM	0.5745	0.5209	0.5401	0.5495
KNN	0.5532	0.1829	0.1879	0.5345
AB	0.5390	0.4683	0.5003	0.5382
RF	0.5887	0.1951	0.2105	0.5553
LR	0.6312	0.3206	0.2714	0.5829
NBC	0.0922	0.0768	0.0974	0.5643

the task of using resume data for job recommendation, the attention mechanism in GAT performs slightly worse compared to the IHGCN. Machine learning algorithms generally perform better than graph embedding algorithms, except for Node2vec and NBC. Node2vec excels in capturing local and global graph information, generating rich semantic node embeddings. However, NBC performs poorly in this node classification task, possibly due to its feature independence assumption may not hold in this complex resume graph structures. These insights provide valuable understanding of the performance and strengths of the different models.

The overall performance results are presented in Table 4, with superior outcomes highlighted in bold. The following observations can be made based on the findings in Table 4: (a) GNN-based models consistently outperform classical machine learning baselines, indicating that the incorporation of node features and the utilization of deep neural networks can generate desirable node embeddings; (b) IHGCN surpasses all baselines across different scenarios involving early jobseekers' resume data. The relative improvements (%) compared to the best baseline were 12.4%, 24.8%, 20.3%, and 15.6% for accuracy, precision, Macro-F1, and AUC, respectively. Furthermore, it becomes evident that the proposed enhanced GCN framework is highly effective and yields superior node embedding outcomes compared to the baseline methods. Also, the confusion matrix of the classification results is shown in Fig. 6.

Due to severe imbalance of realistic data, we draw the confusion matrixes for marketing (Mktg) and technical (Tech) classes with relatively large amount of data. Notably, the IHGCN model demonstrates versatility in addressing multi-classification problems and various types of recommender systems.

The main reasons why IHGCN works well are two folds:

1) IHGCN is an improved GCN model, where the node's new feature is computed as a weighted average of the node itself and its second-order neighbors, using a special form of Laplacian smoothing (Li et al., 2018). By incorporating the characteristics of both the node and its neighbors in the heterogeneous graph, IHGCN effectively analyzes the nodes and performs well in node classification tasks.
2) In IHGCN, meta-paths are introduced as features, allowing the heterogeneous resume graph to capture both the relations between nodes and the global node-metapath relations. This addition of meta-paths enhances the interpretability of the model and leads to improved performance in the experiments. By considering both local node-node relations and global node-metapath relations, IHGCN gains a better understanding of the resume graph, resulting in more effective job recommendations.

5.4. Robust and sensitivity

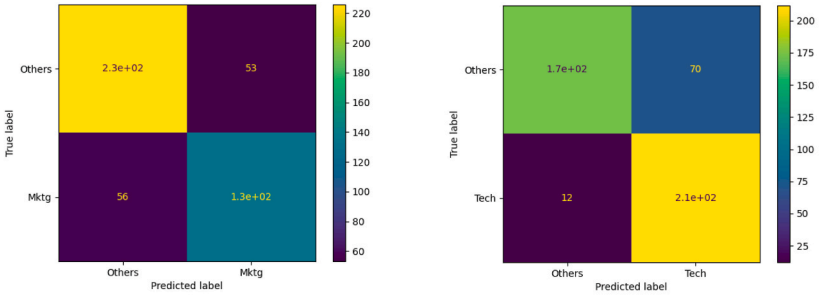
Fig. 7 illustrates the variation of test accuracy under different parameter settings. In.

Fig. 7-(a), we observed that accuracy peaks when dropout reaches 0.2. Then, accuracy gradually fluctuates and decreases with increasing dropout. In Fig. 7-(b), we plotted the classification performance of IHGCN with different learning rates. We found that the test accuracy reaches its optimal value and remains relatively robust at a learning rate of 0.01. Moreover, excessively high learning rates do not improve the classification performance and may result in oscillations.

Fig. 8 illustrates the performance of the IHGCN model with different adjacency matrix configurations. And, the explanations of variables representing different adjacency matrix configurations are show in Table 5. The best performance, labeled as "AVDMUS" is achieved when the adjacency matrix includes the combination of four meta-paths related to the "also-viewed" attribute. In Fig. 8, "AV" represents the adjacency matrix constructed by the original GCN model using the "also-viewed" attribute. "AVD", "AVM", "AVU" and "AVS" correspond to the experimental results of introducing the meta-paths RDR, RMR, RUR, and RSR in Table 3 into the adjacency matrix respectively. The performance of the "AVD" is slightly reduced compared to the "AV" when the meta-path RDR is introduced. However, the performance of the "AVDMSU" is significantly improved compared to the "AVMSU" by incorporating the meta-path RDR. These experimental results indicate that job recommendations should not solely rely on the degree-related information of early job seekers. Instead, a comprehensive assessment of their educational background and skill-related information is crucial for improving the accuracy of job recommendations. This figure allows us to observe the accuracy variations when different meta-paths are included in the adjacency matrix. It provides insights into the impact of incorporating various meta-paths on the performance of the IHGCN model in the job recommendation task.

6. Conclusions

In this paper, we addressed the problem of job recommendations for early jobseekers and proposed the IHGCN model as our solution. To



(a) The confusion matrix for Marketing class and others. (b) The confusion matrix for Technical class and others.

Fig. 6. The confusion matrix for classification results.

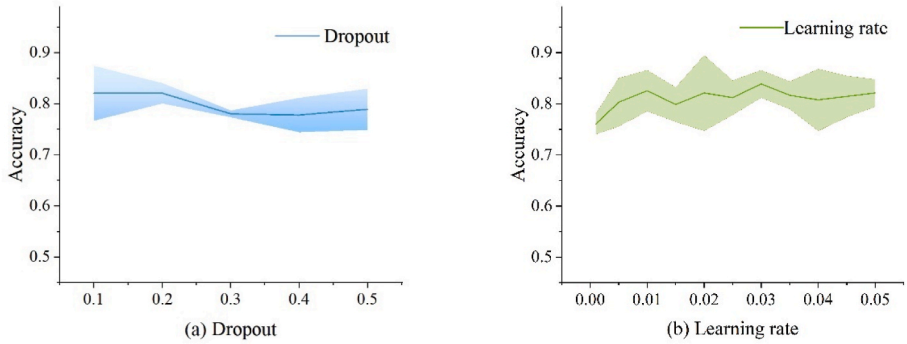


Fig. 7. Test accuracy of the IHGCN model with different hyperparameter settings. Figure (a) and figure (b) show the variation of accuracy with dropout and learning rate respectively.

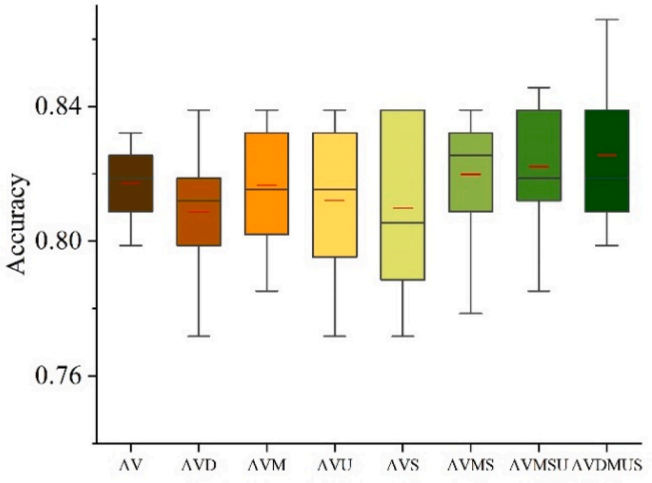


Fig. 8. The accuracy performance achieved by different adjacency matrix configurations. The variable "AV" on the x-axis represent only use the "also-viewed" attribute as the basic adjacency matrix. The variables "AVD", "AVM", "AVU" and "AVS" on the x-axis represent adding the meta-paths RDR, RMR, RUR and RSR to the basic adjacency matrix. The variable "AVMS" represent two meta-paths RMR and RSR are introduced to the basic adjacency matrix. The variable "AVMSU" represent two meta-paths RMR, RSR and RUR are introduced to the basic adjacency matrix. The variable "AVDMSU" denote four meta-paths RDR, RMR, RUR and RSR are introduced to the basic adjacency matrix.

Table 5
The adjacency matrix configurations.

Variable	Explanation
AV	Only use "also-viewed" attribute as the basic adjacency matrix.
AVD	Add one meta-path RDR to the basic adjacency matrix AV.
AVM	Add one meta-path RMR to the basic adjacency matrix AV.
AVU	Add one meta-path RUR to the basic adjacency matrix AV.
AVS	Add one meta-path RSR to the basic adjacency matrix AV.
AVMS	Add two meta-paths RMR and RSR to the basic adjacency matrix AV.
AVMSU	Add three meta-paths RMR, RSR and RUR to the basic adjacency matrix AV.
AVDMUS	Add all four meta-paths RDR, RMR, RUR and RSR to the basic adjacency matrix AV

facilitate this research, we developed a specialized labeling method tailored to early jobseekers' resumes and invested over 500 h in manual labeling. After that, we constructed a heterogeneous resume graph where each node represents an early jobseeker's resume. The connections between early jobseekers were established through an ensemble of the also-viewed attribute and several meta-paths. In our approach, we framed the job recommendation task as a multi-class node classification problem, leveraging the IHGCN model to capture the crucial aspects of education background and skill information. Notably, our proposed model effectively handles limited labeled documents, demonstrating its adaptability to practical scenarios. The experimental results confirm the superiority of our IHGCN model across all evaluation metrics when compared to the baseline models. In the future, we intend to explore the temporal nature of relevant attributes and incorporate additional data to further validate and enhance our model's performance.

CRediT author statement

Hao Wang: Conceptualization, Methodology, Software, Writing-Original draft preparation; **Wenchuan Yang**: Validation, Methodology; **Jichao Li:** Funding acquisition, Project administration; **Junwei Ou**: Validation; **Yanjie Song**: Supervision; **Yingwu Chen:** Resources.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

The authors do not have permission to share data.

Acknowledgments

This study was supported by General Program of National Natural Science Foundation of China (72371244) and Excellent Youth Scholars of Hu'nan Provincial Natural Science Foundation (2022JJ20047).

References

Alhijawi, B., Kilani, Y., 2020. A collaborative filtering recommender system using genetic algorithm. Inf. Process. Manag. 57 (6), 102310.
Bailly, A., Blanc, C., Francis, É., et al., 2022. Effects of dataset size and interactions on the prediction performance of logistic regression and deep learning models. Comput. Methods Progr. Biomed. 213, 106504.
Basilico, J., Hofmann, T., 2004. Unifying collaborative and content-based filtering. In: Proceedings of the Twenty-First International Conference on Machine Learning, p. 9.
Börner, K., Scrivner, O., Gallant, M., et al., 2018. Skill discrepancies between research, education, and jobs reveal the critical need to supply soft skills for the data economy. Proc. Natl. Acad. Sci. USA 115 (50), 12630–12637.
Boselli, R., Cesarini, M., Mercorio, F., et al., 2017. In: Using Machine Learning for Labour Market intelligence[C]//Machine Learning and Knowledge Discovery in Databases: European Conference, ECML PKDD 2017, Skopje, Macedonia, September 18–22, 2017, Proceedings, Part III 10. Springer International Publishing, pp. 330–342.
Breiman, L., 2001. Random forests. Mach. Learn. 45 (1), 5–32.
Chen, Y., Zheng, Y., Xu, Z., et al., 2022. Cross-domain few-shot classification based on lightweight Res2Net and flexible GNN. Knowl. Base Syst. 247, 108623.
Cortes, C., Vapnik, V., 1995. Support-vector networks. Mach. Learn. 20, 273–297.
Dave, V.S., Zhang, B., Al Hasan, M., et al., 2018. A combined representation learning approach for better job and skill recommendation[C]. In: Proceedings of the 27th

ACM International Conference on Information and Knowledge Management, pp. 1997–2005.
Flambeau, J.K.F., Norbert, T., 2021. Skills prediction based on multi-label resume classification using CNN with model predictions explanation. Neural Comput. Appl. 33 (10), 5069–5087.
Freund, Y., Schapire, R.E., 1996. Experiments with a new boosting algorithm. In: International Conference on Machine Learning. Morgan Kaufmann Publishers, Bari, Italy, pp. 148–156.
Ghorvei, M., Kavianpour, M., Beheshti, M.T.H., et al., 2023. Spatial graph convolutional neural network via structured subdomain adaptation and domain adversarial learning for bearing fault diagnosis. Neurocomputing 517, 44–61.
Grandini, M., Bagli, E., Visani, G., 2022. Metrics for Multi-Class Classification: an Overview, pp. 1–17 arXiv preprint. 2020.
Grover, A., Leskovec, J., 2016. node2vec: scalable feature learning for networks[C]. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 855–864.
Guo, D., Wang, H., Wang, M., 2021. Context-aware graph inference with knowledge distillation for visual dialog. IEEE Trans. Pattern Anal. Mach. Intell. 44 (10), 6056–6073.
Hamilton, W., Ying, Z., Leskovec, J., 2017. Inductive representation learning on large graphs. Adv. Neural Inf. Process. Syst. 30.
Hao, X., Li, J., Guo, Y., et al., 2021. Hypergraph neural network for skeleton-based action recognition. IEEE Trans. Image Process. 30, 2263–2275.
He, M., Shen, D., Zhu, Y., et al., 2019. Career Trajectory Prediction Based on cnn[C]// 2019 IEEE International Conference on Service Operations and Logistics, and Informatics (SOLI). IEEE, pp. 22–26.
He, Y., Ren, T., Tang, J., et al., 2022. Heterogeneous learning for scene graph generation [C]. In: Proceedings of the 30th ACM International Conference on Multimedia, pp. 4704–4713.
Kavianpour, P., Kavianpour, M., Jahani, E., et al., 2021a. A Cnn-Bilstm Model with Attention Mechanism for Earthquake Prediction, 13444 arXiv preprint arXiv:2112.
Kavianpour, P., Kavianpour, M., Jahani, E., et al., 2021b. Earthquake magnitude prediction using spatia-temporal features learning based on hybrid cnn-bilstm model. In: 2021 7th International Conference on Signal Processing and Intelligent Systems (ICSPIS). IEEE, pp. 1–6.
Kavianpour, M., Ghorvei, M., Kavianpour, P., et al., 2022a. An intelligent gearbox fault diagnosis under different operating conditions using adversarial domain adaptation. In: 2022 8th International Conference on Control, Instrumentation and Automation (ICCIA). IEEE, pp. 1–6.
Kavianpour, M., Ramezani, A., Beheshti, M.T.H., 2022b. A class alignment method based on graph convolution neural network for bearing fault diagnosis in presence of missing data and changing working conditions. Measurement 199, 111536.
Kenthapadi, K., Le, B., Venkataraman, G., 2017. Personalized job recommendation system at linkedin: practical challenges and lessons learned[C]. In: Proceedings of the Eleventh ACM Conference on Recommender Systems, pp. 346–347.
Kipf, T.N., Welling, M., 2016. Semi-supervised Classification with Graph Convolutional Networks arXiv preprint arXiv:1609.02907.
Koren, Y., Rendle, S., Bell, R., 2021. Advances in collaborative filtering. Recom. Sys. Handbook 91–142.
Li, Q., Han, Z., Wu, X.M., 2018. Deeper insights into graph convolutional networks for semi-supervised learning[C]. Proc. AAAI Conf. Artif. Intell. 32 (1).
Li, X., Zhou, T., Li, J., et al., 2021. Group-wise semantic mining for weakly supervised semantic segmentation[C]. Proc. AAAI Conf. Artif. Intell. 35 (3), 1984–1992.
Liu, J., Ng, Y.C., Wood, K.L., et al., 2019. Ipod: an Industrial and Professional Occupations Dataset and its Applications to Occupational Data Mining and Analysis arXiv preprint arXiv:1910.10495.
Lops, P., De Gemmis, M., Semeraro, G., 2011. Content-based Recommender Systems: State of the Art and Trends. Recommender systems handbook, pp. 73–105.
Narasimhan, M., Lazebnik, S., Schwing, A., 2018. Out of the box: reasoning with graph convolution nets for factual visual question answering. Adv. Neural Inf. Process. Syst. 31.
Nguyen, T., Grishman, R., 2018. Graph convolutional networks with argument-aware pooling for event detection[C]. Proc. AAAI Conf. Artif. Intell. 32 (1).
Papadakis, H., Papagrigoriou, A., Panagiotakis, C., et al., 2022. Collaborative filtering recommender systems taxonomy. Knowl. Inf. Syst. 64 (1), 35–74.
Paparrizos, I., Cambazoglu, B.B., Gionis, A., 2011. Machine learned job recommendation [C]. In: Proceedings of the Fifth ACM Conference on Recommender Systems, pp. 325–328.
Pazzani, M.J., 1999. A framework for collaborative, content-based and demographic filtering. Artif. Intell. Rev. 13, 393–408.
Perozzi, B., Al-Rfou, R., Skiena, S., 2014. Deepwalk: online learning of social representations[C]. In: Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 701–710.
Pradhyumna, P., Shreya, G.P., 2021. Graph neural network (GNN) in image and video understanding using deep learning for computer vision applications[C]//2021 Second International Conference on Electronics and Sustainable Communication Systems (ICESC). IEEE 1183–1189.
Qin, C., Zhu, H., Xu, T., et al., 2018. Enhancing person-job fit for talent recruitment: an ability-aware neural network approach[C]. In: The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval, pp. 25–34.
Qin, C., Zhu, H., Xu, T., et al., 2020. An enhanced neural network approach to person-job fit in talent recruitment. ACM Trans. Inf. Syst. 38 (2), 1–33.
Qiu, J., Tang, J., Ma, H., et al., 2018. Deepinf: social influence prediction with deep learning[C]. In: Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, pp. 2110–2119.
Ribeiro, L.F.R., Saverese, P.H.P., Figueiredo, D.R., 2017. struc2vec: learning node representations from structural identity[C]. In: Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 385–394.
Rrmoku, K., Selimi, B., Ahmedi, L., 2022. Application of trust in recommender systems—utilizing naive Bayes classifier. Computation 10 (1), 6.
Seveso, A., Giabelli, A., Malandri, L., et al., 2021. Skills2Job: a recommender system that encodes job offer embeddings on graph databases (student abstract)[C]. Proc. AAAI Conf. Artif. Intell. 35 (18), 15885–15886.
Si, X., Dai, H., Raghothaman, M., et al., 2018. Learning loop invariants for program verification. Adv. Neural Inf. Process. Syst. 31.
Su, Y., Bayoumi, M., Joachims, T., 2022. Optimizing rankings for recommendation in matching markets[C]. In: Proceedings of the ACM Web Conference 2022, pp. 328–338.
Sun, Y., Zhuang, F., Zhu, H., et al., 2021. Cost-effective and interpretable job skill recommendation with deep reinforcement learning[C]. In: Proceedings of the Web Conference, 2021, pp. 3827–3838.
Tang, J., Qu, M., Wang, M., et al., 2015. Line: large-scale information network embedding. In: Proceedings of the 24th International Conference on World Wide Web, pp. 1067–1077.
Tarus, J.K., Niu, Z., Mustafa, G., 2018. Knowledge-based recommendation: a review of ontology-based recommender systems for e-learning. Artif. Intell. Rev. 50, 21–48.
Tibshirani R. Nonparametric Regression: Nearest Neighbors and Kernels[J].
Velickovic, P., Cucurull, G., Casanova, A., et al., 2017. Graph attention networks. stat 1050 (20), 10.48550.
Wang, D., Cui, P., Zhu, W., 2016. Structural deep network embedding. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 1225–1234.
Wu, X., Xu, T., Zhu, H., et al., 2019. Trend-Aware Tensor Factorization for Job Skill Demand Analysis. IJCAI, pp. 3891–3897.
Wu, Q., Zhao, W., Li, Z., et al., 2022. Nodeformer: a scalable graph structure learning transformer for node classification. Adv. Neural Inf. Process. Syst. 35, 27387–27401.
Xu, T., Zhu, H., Zhu, C., et al., 2018. Measuring the popularity of job skills in recruitment market: a multi-criteria approach[C]. Proc. AAAI Conf. Artif. Intell. 32 (1).
Yang, C., Hou, Y., Song, Y., et al., 2022a. Modeling two-way selection preference for person-job fit[C]. In: Proceedings of the 16th ACM Conference on Recommender Systems, pp. 102–112.
Yang, W., Chen, G., Zhao, Z., et al., 2022b. iCGPN: interaction-centric graph parsing network for human-object interaction detection. Neurocomputing 502, 98–109.
Yao, H., Wu, F., Ke, J., et al., 2018. Deep multi-view spatial-temporal network for taxi demand prediction[C]. Proc. AAAI Conf. Artif. Intell. 32 (1).
Yao, L., Mao, C., Luo, Y., 2019a. Graph convolutional networks for text classification[C]. Proc. AAAI Conf. Artif. Intell. 33 (1), 7370–7377.
Yao, L., Mao, C., Luo, Y., 2019b. Graph convolutional networks for text classification[C]. Proc. AAAI Conf. Artif. Intell. 33 (1), 7370–7377.
Yao, K., Zhang, J., Qin, C., et al., 2022. Knowledge enhanced person-job fit for talent recruitment. In: 2022 IEEE 38th International Conference on Data Engineering (ICDE). IEEE, pp. 3467–3480.
Yasunaga, M., Ren, H., Bosselut, A., et al., 2021. QA-GNN: Reasoning with Language Models and Knowledge Graphs for Question Answering arXiv preprint arXiv: 2104.06378.
Zhang, Y., Tang, J., Yang, Z., et al., 2015. Cosnet: connecting heterogeneous social networks with local and global consistency[C]. In: Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 1485–1494.
Zhang, X.X., Zhou, Y., Ma, Y., et al., 2016. Glmix: generalized linear mixed models for large-scale response prediction[C]. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 363–372.
Zhu, C., Zhu, H., Xiong, H., et al., 2016. Recruitment market trend analysis with sequential latent variable models. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 383–392.
Zihayat, M., Ayanso, A., Zhao, X., et al., 2019. A utility-based news recommendation system. Decis. Support Syst. 117, 14–27.