A Novel Employment Recommendation for Chinese College Students: Combining RNN and Topic Model

Jin Xie, *Member, IAENG,* Mohamad Fadli Zolkipli, and Mohamad Farhan Mohamad Mohsin

***Abstract*—To enhance job matching accuracy and optimize human resource utilization for Chinese college students, developing effective job recommendation algorithms is essential. However, existing methods often fail to capture the personalized needs of students, resulting in suboptimal recommendations. This paper presents a novel job recommendation algorithm based on Recurrent Neural Networks (RNN) and Biterm Topic Modeling (BTM). Detailed user profiles are constructed using university library data, encompassing academic records, social activities, and skill certifications. These profiles are then clustered using BTM to identify latent topics. A bipartite graph framework is employed to establish an initial recommendation structure, which is further refined using a weighted random forest classifier that incorporates user preferences. Experimental results demonstrate that the proposed model significantly outperforms baseline algorithms in both hit rate and recommendation accuracy.**

***Index Terms*—Employment recommendation; User profiling; Recurrent Neural Network; Topic Modeling; Machine Learning**

I. Introduction

IN recent years, the employment landscape for Chinese college students has become increasingly competitive, raising concerns among policymakers and educators[1]. Despite the growing number of job listings, many students struggle to secure positions that align with their individual skills and aspirations. A key contributor to this mismatch is the lack of personalization in traditional job recommendation systems, which often rely on static resumes and subjective assessments.[2], [3]. With the growing number of graduates each year, improving the employment matching rate and enhancing the efficiency of human resource utilization have become urgent problems. Traditional employment guidance methods often rely on subjective judgment, which is not only inefficient but also unable to address the personalized employment needs of college students [4], [5]. Therefore, providing users with accurate and personalized recommendations has become a critical challenge for current student employment platforms. To address these issues, some platforms have started incorporating artificial intelligence technologies to classify, filter, and recommend jobs using natural language processing and machine learning algorithms. Consequently, employment recommendation systems have emerged. These technologies not only enhance the user experience by offering a wider range of job options but also facilitate broader communication channels for recruitment companies, fostering information exchange. An effective employment recommendation system requires accurate user profiles, advanced technologies for personalization, and continuous adaptation to user needs.

Currently, most of the existing employment recommendation methods have the following shortcomings:

1) Limited Data Diversity: Traditional recommendation algorithms often depend on a single data source, such as resumes or job search website records. This reliance significantly limits the system's ability to capture the complex, multi-dimensional nature of users. As a result, the generated recommendations tend to overlook important personal factors. For example, systems based solely on resume content fail to account for students' interests, social networks, or extracurricular involvement—all of which are essential for making informed and personalized career decisions.
2) Inaccurate User Classification: Many existing systems adopt overly simplistic classification strategies, assigning users to broad categories and generating recommendations accordingly. This method neglects the heterogeneity and individuality inherent in user profiles, leading to generalized results that lack nuance and relevance. For instance, grouping all computer science majors together ignores their specific skill sets, project experience, and career aspirations—factors that are critical for delivering precise and meaningful job matches.

Therefore, developing an automated and accurate job recommendation method is of substantial practical relevance and real-world value. In response to this need, this study proposes a novel recommendation framework tailored for college students, which integrates Recurrent Neural Networks (RNN) and topic modeling techniques. Specifically, the system utilizes a Conditional Random Field (CRF) model combined with a Long Short-Term Memory (LSTM) neural network to extract structured user data from university library archives and construct comprehensive user profiles. These profiles are subsequently clustered using the Biterm Topic Model (BTM) to identify latent semantic patterns. Finally, a graph-based network is employed to establish a foundational recommendation framework that effectively matches students with suitable employment opportunities.

Manuscript received October 13, 2024; revised June 20, 2025.

This work was supported by Universiti Utara Malaysia and in part by the Local Science and Technology Development Project Guided by the Central Government of China (Grant No. 2018ZYYD007).

Jin Xie is a lecturer at Hubei University of Automotive Technology (HUAT), Shiyan 442001, China (email: rudyxie@huat.edu.cn).

Mohamad Fadli Zolkipli is an associate professor of Universiti Utara Malaysia (UUM), Sintok 06010, Malaysia (Corresponding author, e-mail: m.fadli.zolkipli@uum.edu.my).

Mohamad Farhan Mohamad Mohsin is a senior lecturer at Universiti Utara Malaysia (UUM), Sintok 06010, Malaysia (email: farhan@uum.edu.my).

II. RELATED WORK

In the field of recommendation algorithms, the continuous development of technology and the rapid growth of data have led to the proposal of many innovative methods. These methods can be broadly categorized into traditional recommendation algorithms and domain-specific recommendation algorithms, such as job recommendation systems.

Traditional recommendation algorithms primarily include collaborative filtering, content-based methods, and hybrid approaches. Collaborative filtering recommends items based on behavioral similarities between users or associations between items, suggesting items that similar users have liked or items resembling those a user has previously engaged with. Content-based recommendation analyzes a user's historical behaviors and preferences, as well as the content characteristics of items, to recommend items that align with the user's interests. Hybrid recommendation combines the strengths of both collaborative filtering and content-based recommendation to enhance the accuracy and diversity of recommendations. For example, Mao et al.[6] proposed a user coverage model based on product rating differences to address the issue that traditional recommendation algorithms often prioritize accuracy over diversity. This model combines a rating difference matrix with user interest domain calculations to define user interest ranges more accurately. By mapping the user and the recommended list of interest domains to vector space, the model optimizes the recommendation objective function. Despite these improvements, the model's recommendation results can still be uncertain due to mismatches between covered products and actual user needs. Shokrzadeh et al.[7] tackled this issue by creating behavior graphs and behavior paths based on user behavior sequences. They used vectorization techniques to convert text-type paths into vectors and calculated the similarity between multi-dimensional behavior path vectors. This approach allowed for multi-dimensional path collaborative filtering recommendations based on similarity. However, the sparsity of the scoring matrix limits the path vector's effectiveness in calculating similarity, affecting recommendation accuracy. Teng et al.[8] developed a job recommendation model using multi-task learning, incorporating reciprocity constraints, attention mechanisms, and fuzzy gate mechanisms. This model aggregates the needs of college students and employers to provide appropriate job recommendations. While the inclusion of attention and fuzzy gate mechanisms enhances the model's capabilities, it also increases computational complexity, resulting in longer processing times during job recommendation.

As a significant branch of recommendation systems, job recommendation algorithms have garnered extensive attention in recent years. These algorithms aim to address the inefficiencies of information matching in traditional recruitment processes by providing more accurate matches for job seekers and recruiters through automated and intelligent methods.

Yu et al.[9] proposed a cross-domain collaborative filtering algorithm that utilizes latent factor space in the auxiliary domain for recommendations by extending the features of users and items. This method effectively leverages shared information across different domains to enhance recommendation performance. Liu et al.[10] introduced a person-job matching algorithm based on joint representation learning, which achieves more accurate job recommendations by learning a shared representation space for job seekers and jobs, as well as the matching function between them. Alatrash et al.[11] developed a hybrid recommendation algorithm for developers that combines explicit features (such as skills and experience) with implicit features (such as behavioral data and social networks). This approach captures the needs and preferences of developers more comprehensively, thereby improving the accuracy of job recommendations.

In summary, job recommendation algorithms have continuously improved in accuracy and efficiency through the introduction of new techniques and methods. However, existing algorithms still face challenges such as data sparsity, cold start problems, and limited ability to meet users' personalized needs. Therefore, this paper proposes a job recommendation algorithm for college students based on recurrent neural networks (RNN) and topic modeling to address these issues more effectively. By constructing user profiles and performing topic clustering analysis, the algorithm delivers personalized job recommendations tailored to the unique characteristics and needs of college students. The method is tested and validated using the employment service platform of a university in Hubei, China. The performance analysis from verification experiments demonstrates the superior effectiveness of the proposed method.

III. PROPOSED METHODOLOGY

This section outlines the core components of the proposed employment recommendation framework, which consists of three main stages: (1) user profile construction using sequential and textual data, (2) topic modeling via the Biterm Topic Model (BTM) to uncover latent semantic structures, and (3) a personalized recommendation mechanism based on graph modeling and a weighted random forest classifier. Together, these components form a cohesive system that generates accurate and individualized job recommendations for college students.

A. User Profile Construction

Prior to job recommendation, a conditional random field model integrated with a long short-term memory (LSTM) neural network, a variant of the recurrent neural network (RNN), is applied. Leveraging the information field definition, comprehensive student data is extracted to construct detailed user profiles [12]. Using Long Short-Term Memory (LSTM) networks is advantageous for building user profiles due to their ability to effectively capture and learn from sequential data. LSTM networks, a type of recurrent neural network (RNN), are specifically designed to address the limitations of traditional RNNs, such as the vanishing gradient problem, which hampers the learning of long-term dependencies. By maintaining long-term memory through gated mechanisms, LSTM can retain and utilize information over extended periods. This capability is particularly beneficial for user profiling, as it allows the model to incorporate a user's historical behaviors and interactions, thereby capturing temporal patterns and trends that are essential for accurate profile construction. Furthermore, LSTM can manage varying lengths of input sequences, making them well-suited for

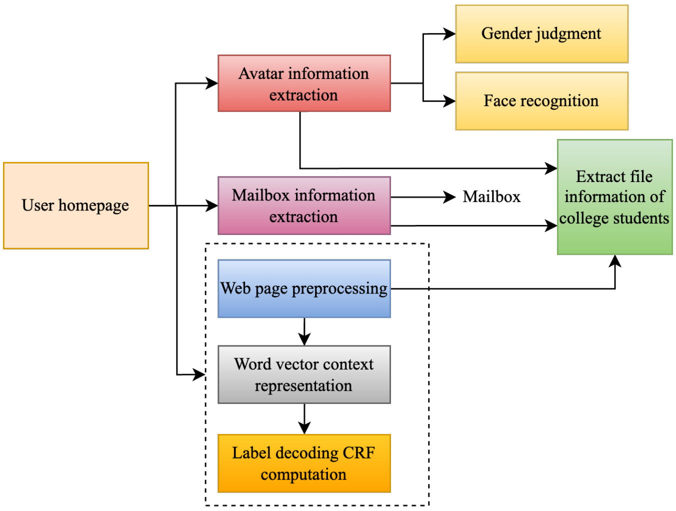

Fig. 1. The architecture of the user portrait information extraction model

handling diverse user activity data. Consequently, LSTM-based models can generate more comprehensive and dynamic user profiles, leading to improved personalization and recommendation outcomes. In the actual operation process, the structure of the user profile information extraction model is shown in Figure 1.

The user information extraction model, illustrated in Figure 1, is applied to the university library's homepage. This model extracts college students' profile information through several steps: mailbox information extraction, avatar information extraction, and web page preprocessing. Due to the structural and textual similarities across different college students' homepages, this information is input into a long short-term memory (LSTM) neural network model to automatically extract webpage information features. The text is processed using word segmentation to accurately filter and retrieve the relevant information.

During webpage preprocessing, redundant information in user profiles can interfere with job recommendations and increase the algorithm's computational complexity. Therefore, the initial step in extracting college students' user information involves cleaning the webpages. Invalid tags are used to identify redundant information, which is then removed through text filtering [13]. This process yields simplified user webpage text information, enhancing the efficiency and accuracy of the job recommendation algorithm.

Valid labels are assigned to the types of information to be extracted, and these labels are compiled into a thesaurus table. This table is then input into the long short-term memory (LSTM) neural network to extract word features and character word vectors. The processes of word feature extraction and character word vector extraction are represented by Formula (1) and Formula (2), respectively.

$$h(t) = LSTM(x(t), h(t-1)) \quad (1)$$

$$v(t) = LSTM(c(t), v(t-1)) \quad (2)$$

Where $x(t)$ represents the tth input word, $h(t-1)$ represents the hidden state at the previous time step, and $h(t)$ represents the hidden state at the current time step. By inputting words one by one and continuously updating the hidden state, the LSTM model captures the dependencies between words and extracts their feature representations. $c(t)$ denotes the tth character, $v(t-1)$ denotes the vector representation of the previous time step, and $v(t)$ denotes the vector representation of the current time step. By inputting characters one by one and constantly updating the vector representation, the LSTM model can learn the character-level context information and generate the corresponding word vector representation.

All text nodes are represented as two-dimensional word vectors. A sequence is then generated based on the user information word vector, which is combined with the text node sequence. This word vector is decoded through the fully connected layer of the long short-term memory (LSTM) neural network model [14] to classify all user profile text label information. The classified user profile information is stored in the data layer. Upon receiving a profile query command, the corresponding classified user profile information is retrieved and displayed as a visual user profile using modern programming techniques.

B. Topic Modeling with BTM

To better uncover the hidden information within different user profiles, user topic modeling is applied to the established college student data through processes such as numerical analysis and discretization [15].

Biterm Topic Modeling (BTM) is particularly useful for establishing user profiles due to its ability to effectively capture the latent topics within short texts, such as social media posts, reviews, or user comments. Unlike traditional topic models like Latent Dirichlet Allocation (LDA), which are better suited for longer documents, BTM is designed to identify co-occurring word pairs (biterms) within the entire corpus, making it more adept at handling the sparsity and brevity of user-generated content. This approach allows for the extraction of more coherent and meaningful topics from fragmented and short pieces of text, which are common in user profiles. By leveraging BTM, researchers and practitioners can gain a more nuanced understanding of users' interests, preferences, and behaviors, leading to the creation of more accurate and comprehensive user profiles. These enriched profiles can subsequently enhance the personalization and effectiveness of recommendation systems and other user-centered applications [16]. Hence, BTM is employed to assess the divergence among the traits delineated in college students' user profiles. Subsequently, students awaiting job recommendations are segmented into distinct topic clusters. Through this process, user profiles exhibiting high congruity are identified within the pool of employed students, facilitating the tailored allocation of suitable job opportunities in accordance with each student's employment status.

D represents the training set corpus, $W = w_1, w_2 ... w_N$ represents the biterm set obtained from the training set corpus, and w_1 represents a biterm unit $(q_{i,1}, q_{i,2})$. Assume that α and β are Dirichlet prior parameters. The whole process of model building using the BTM model is shown in Figure 2.
In the figure above, K is the number of artificial topics, θ is the topic distribution over the text corpus, φ is the term distribution over the topic, $Mult()$ is the multi-term distribution, and $Dir()$ is the Dirichlet distribution. Z is the Dirichlet distribution of the topic θ. BTM building steps are as follows:

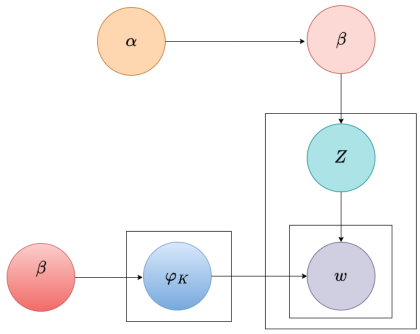

Fig. 2. BTM model training process

- Step 1: The word distribution $\varphi K \sim Dir(\beta)$ under a topic is sampled from the parameter β by Dirichlet distribution;
- Step 2: Sampling the global topic distribution $\theta \sim Dir(\alpha)$ from the Dirichlet distribution of α;
- Step 3: Select the topic Z from the common parameter θ of the short text corpus, and serve $Z \sim Mult(\theta)$;
- Step 4: Suppose that a word pair in the corpus is b, $b = (w_i, w_j)$. Take the words w_i, w_j from the topic Z extracted above, and make them obey $w_i, w_j \sim Mult(\varphi Z)$;
- Step 5: Select the topic Z from the common parameter θ of the short text corpus, and serve $Z \sim Mult(\theta)$.

Through the aforementioned steps, the BTM topic model is established on the training set, yielding the document-topic distribution matrix and the topic-word distribution matrix for the large-scale corpus. The probability distribution of each topic feature word is arranged in descending order, and the top w topic words in the probability ranking are extracted as the expansion of the feature word. The formula for this process is as follows:

$$D_q = \{v_{q1}, (u_{11}, u_{12}, ..., u_{1w}), ...v_{qn}, (u_{n1}, u_{n2}, ..., u_{nw})\} \tag{3}$$

Where D_q is the expanded text generated by BTM topic model training of the qth text in corpus D, v_{qn} refers to the nth word in the qth text, and $(u_{n1}, u_{n2}, ..., u_{nw})$ is the w expanded words obtained by v_{qn} using BTM topic model.

C. Bipartite Graph-Based Recommendation

Upon completing the topic modeling of college student user profiles, a set of suitable job opportunities corresponding to a particular user group is identified [17]. Considering each user and project as input vectors, each college student user would need to traverse and evaluate all job opportunities to determine the most suitable matches. To streamline this selection process, we construct a bipartite graph-based job recommendation framework for college students, wherein each edge in the network structure represents a user's potential selection of an item.

Prior to employing the bipartite graph-based job recommendation framework for college students, corresponding energy values are allocated to the users. Following the principle of equal distribution, users distribute their energy across different selected job opportunities [18], with each job receiving a certain amount of energy. This allocated energy can be transferred back to the users, enabling the update of user energy values [19]. In the bipartite graph job recommendation structure for college students, initial energy needs to be assigned to each user at the outset. Adhering to the equal distribution principle, assuming a total of n users, the initial energy λ is divided equally into n parts, with each user receiving $1/n$ initial energy. The specific process of energy allocation is illustrated in Figure 3.

In the process of energy allocation, when a user selects a job, a portion of their energy is assigned to that job. In Figure 3, a, b, and c represent users. User a has chosen job 1 and thus allocates $1/3$ of their energy to job 1. When user b subsequently selects job 1, which was also chosen by user a, job 1 receives $1/3$ of the energy from user b. Consequently, job 1 accumulates a total energy of $1/3\lambda_a + 1/2\lambda_b$ from users a and b. This process is analogous for other jobs. When multiple users select the same job opportunity, the energy value associated with that position accumulates. Ultimately, each job acquires a certain energy value, which represents the degree of preference or selection by users.

After the energy allocation of the bipartite graph network structure is completed, the results of energy distribution are counted and arranged from high to low, and the jobs with the most energy distribution are found and recommended to college students.

IV. Accurate Employment Recommendation

As the bipartite graph job recommendation framework does not account for individual user interests, the authors incorporate a weighted random forest model into the recommendation structure. This model leverages user profile information to vote and classify the recommended job opportunities [20], thereby yielding more accurate job recommendation results tailored to each user's preferences.

The utilization of a weighted random forest model in recommendation algorithms offers several advantages that contribute to improved performance and efficacy. Firstly, weighted random forest models enhance prediction accuracy by incorporating feedback mechanisms that iteratively refine the model based on past recommendations and user interactions [21]. This iterative learning process enables the model to adapt and evolve over time, thereby accommodating changes in user preferences and environmental dynamics. Additionally, weighted random forest models enhance robustness against overfitting by aggregating predictions from multiple decision trees and introducing randomness during the training process [18]. This ensemble approach helps mitigate biases and variance, leading to more stable and reliable recommendations. Furthermore, weighted random forest models are inherently interpretable, allowing researchers and practitioners to gain insights into the decision-making process and understand the factors driving recommendations [22]. The specific architecture of the weighted random forest model is depicted in Figure 4. Each decision tree is assigned a corresponding weight value, which should be factored into the classification process to yield the optimal classification outcome.

As illustrated in Figure 4, random sampling with replacement is initially performed on the original dataset to create several distinct training sets. An independent decision tree is then constructed for each training set, with computed weights

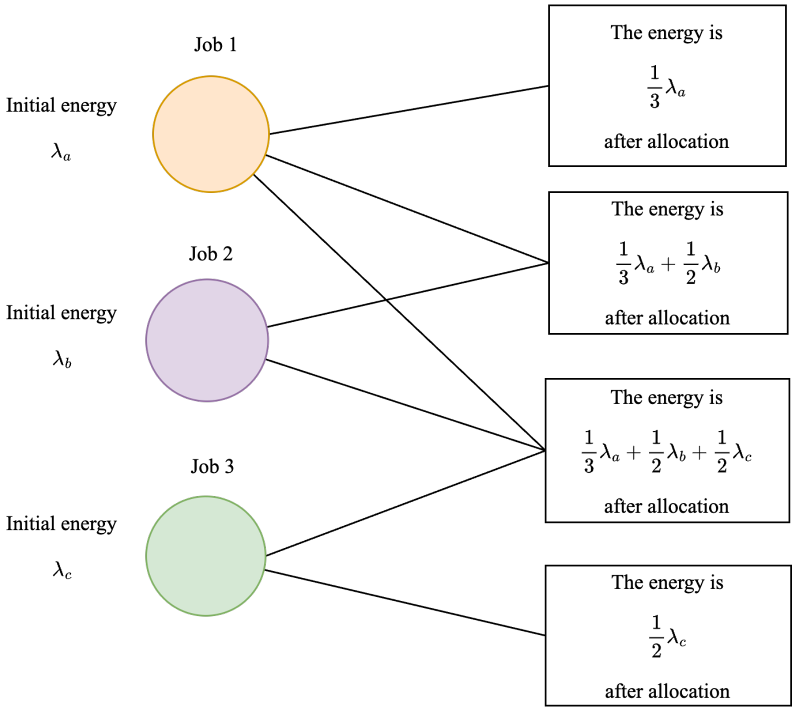

Fig. 3. Schematic diagram of energy allocation based on bipartite graph network structure

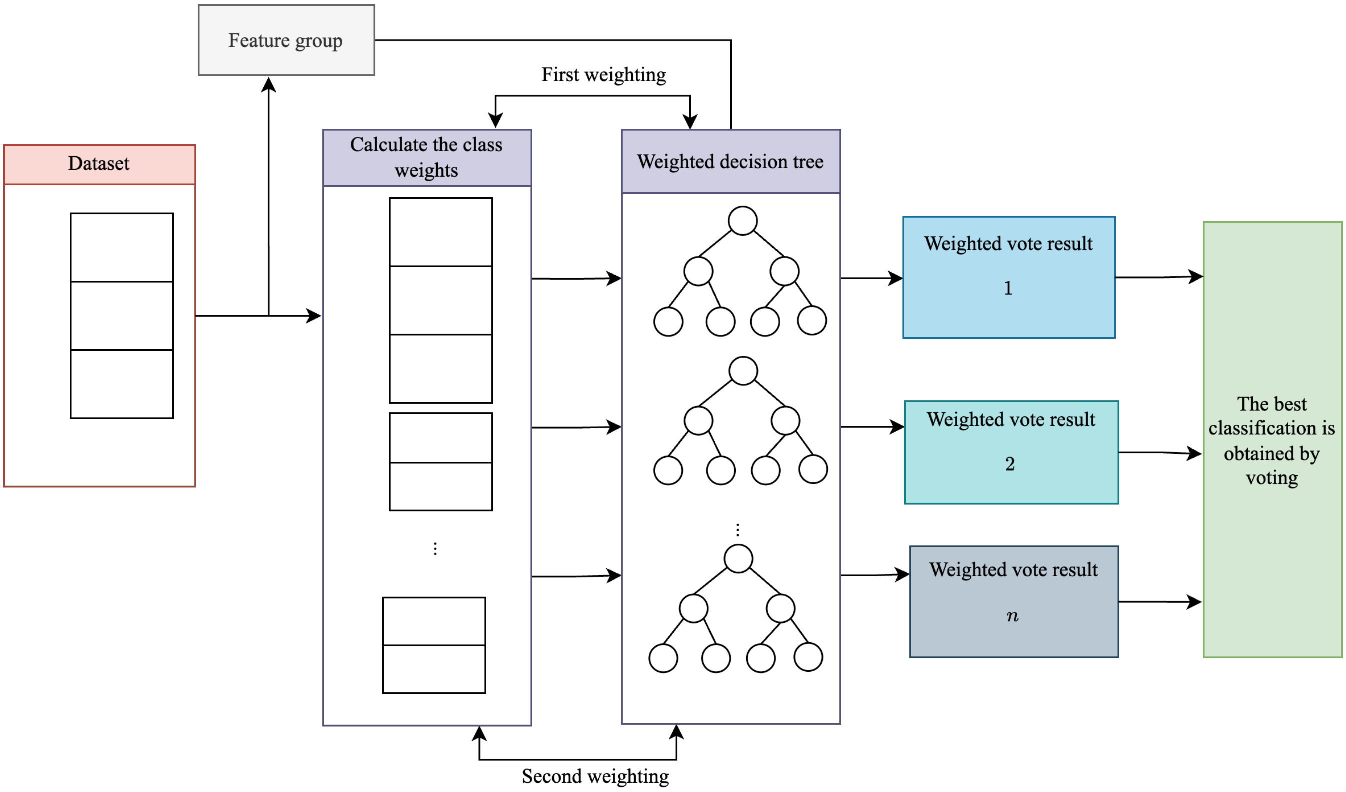

Fig. 4. Weighted random forest model

assigned to each tree. When classifying a new sample, each decision tree provides a classification result. These results are then weighted according to the corresponding tree's weight, and a weighted voting process is used to determine the final classification. Decision trees with higher weights have a greater influence on the final result. The category with the highest number of votes is selected as the final classification. In the case of a tie, methods such as majority voting or random selection are employed. In the weighted random forest model applied to job recommendations for college students, parameters such as the number of decision trees and value thresholds are set based on experience. In this study, the feature vector sets of college student user profiles and employment positions are fed into the weighted random forest model [23], and ensemble classification is performed through multiple decision trees.

Based on the number of input vectors, a corresponding number of classification tree models are established to calculate the probability of a particular user profile group at a given decision tree node [24]. Considering job characteristics, the expected value required for the division of recommended jobs is calculated [25], as shown in Equations (4) and (5).

$$\sigma(B) = -E\sum_{g=1}^{n}\frac{\theta_g}{B}\lg(\frac{\theta_g}{B}) \tag{4}$$

$$\psi_A(B) = \sum_{g=1}^{n}\frac{|\theta_g|}{|B|}\times\sigma(B_g) \tag{5}$$

Where B represents the decision tree node, σ represents the probability that certain user profile group appears in the node, g represents the user profile group category, n represents the total number of user clusters, θ_g represents the number of users in the gth group, A represents the feature, and ψ represents the expected information.

The information gain is calculated based on Equations (4) and (5), and the attribute corresponding to the maximum gain rate is screened out, which is regarded as the classification attribute. The specific calculation process is as follows:

$$G(A) = \sigma(B) - \psi_A(B) \tag{6}$$

Where G represents the information gain.

According to the classification attributes, the weighted random forest model is classified, and several multiple decision tree classification models are generated. The input information is further analyzed to obtain the voting integration results of decision trees:

$$H(\eta) = \underset{Y}{argmin}\sum_{l=1}^{L}\tau v_l(\eta, G(A)) \tag{7}$$

Where H represents the decision tree voting ensemble result obtained after analyzing the input information, Y represents the predicted label value of the decision tree voting ensemble result, η represents the employment of college students, l represents the number of branches of the decision tree, L represents the maximum number of branches of the decision tree, τ is the decision function, and v_l represents the classification model.

Using the weighted random forest model, job recommendations for college students provided by the bipartite graph are further refined through scoring [26]. Based on these classification results, a new scoring label is defined and applied to the bipartite graph recommendation structure to adjust the original project scores. The jobs with the highest revised scores are then identified and compiled into a recommendation list, which is presented to the college students.

TABLE I
DATASET FEATURE STATISTICS TABLE

Feature class	Feature type	Specific Features
Data of students	Discrete feature	Gender, political status, nationality, province of origin, major
Data of students	Continuous feature	Average grades for each semester
Data of Employment units	Discrete feature	The name of the unit, the industry of the unit, the province and the city of the unit

V. VERIFICATION EXPERIMENTS AND ANALYSIS

A. Dataset Selection

To evaluate the effectiveness of the proposed job recommendation algorithm integrating recurrent neural networks and topic modeling, a series of experiments were conducted using real-world data. The dataset was collected from a university in Shiyan, Hubei Province, China, and included nearly three years of graduate records. The corresponding employment units were treated as potential job recommendations, thereby forming the experimental dataset. To ensure a balance between accuracy and computational efficiency, the random forest classifier was configured with 50 decision trees. A latent dimension of 10 was selected to represent key student and job attributes, and the learning rate was set to 0.05 to facilitate stable and efficient model convergence.

To facilitate the employment recommendation evaluation for college students, the dataset encompasses numerous student characteristics and job attributes. The experiment utilizes a dataset comprising 2,000 students and 100 job opportunities, which is divided as follows: 80% (1,600 students) is used as the training set, while the remaining 20% (400 students) serves as the test set. Table 1 presents the statistical summary of the dataset's characteristics.

Using the constructed experimental dataset, the cross-domain collaborative filtering recommendation algorithm [9] and the joint representation learning recommendation algorithm [8] were applied to perform job recommendations for college students. The recommendation results of these three methods were compared to more intuitively demonstrate the superiority of the algorithm proposed in this paper.

B. Experimental Setup

The experimental environment utilized in this study was the employment service platform of a university located in Shiyan, Hubei Province, China. This platform maintains a comprehensive database encompassing students' job search activities, resume submissions, academic records, and employment unit information. It provides structured access to both personal and institutional data, supporting a range of functions including data entry for students and employers,

Fig. 5. User profile word cloud

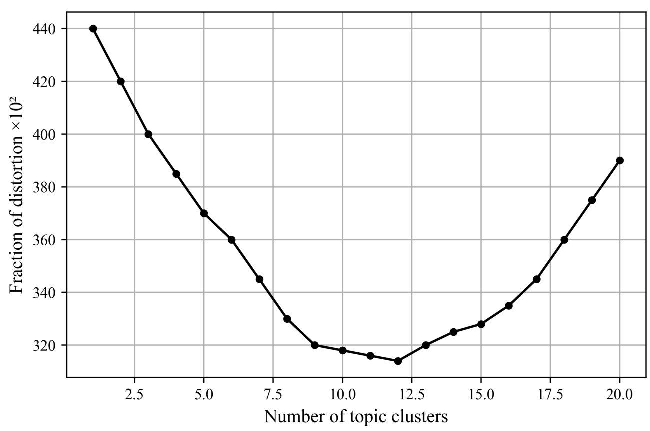

Fig. 6. Distortion scores under different topic numbers

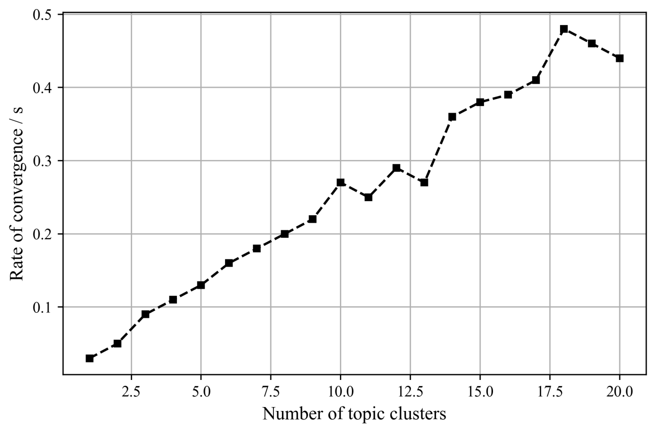

Fig. 7. Convergence time under different topic numbers

job matching, and visualization of recommendation outcomes. The platform's dataset was used to implement and evaluate the performance of the proposed recommendation model in a real-world setting.

To generate the user profile word cloud described in the previous section, student attributes—including personal details and academic performance—were preprocessed and analyzed. The word cloud was created using the Python WordCloud library, which visualizes the relative importance of different features based on their frequency and assigned weights.

C. Employment Recommendation Results

Fundamental student data were extracted from archival records in the university library and used to construct user profiles through the proposed methodology. A visual representation of these profiles in the form of a word cloud is illustrated in Figure 5. Following the construction of user profiles, topic modeling was performed to identify latent semantic structures. Figures 6 and 7 present the statistical outcomes of this process, specifically evaluating clustering error and convergence time under different topic numbers.

As illustrated in Figure 6, the distortion score is used to evaluate the quality of topic modeling, with lower values indicating better clustering performance. The results reveal that as the number of topic clusters increases, the distortion score initially declines, reaches a minimum, and then begins to rise. This trend suggests the existence of an optimal cluster count that balances granularity and coherence in topic representation. Conversely, convergence time, depicted in Figure 7, reflects the computational efficiency of the modeling process. While shorter convergence times are generally desirable, extremely low values may indicate premature convergence, leading to underfitting. The observed pattern shows that convergence time initially increases with the number of clusters, then decreases after a certain point—implying that adequate optimization time is required before achieving the most effective clustering configuration. By jointly analyzing distortion scores and convergence times, the optimal number of topic clusters was determined to be nine. At this setting, the model achieves a minimal distortion score while maintaining a moderate convergence time, thereby balancing modeling effectiveness with computational efficiency. Figure 8 illustrates the centroid of each identified topic cluster, and Figure 9 provides a visual depiction of the overall clustering structure based on these centroids.

Upon completion of topic modeling using the identified cluster centroids, the bipartite graph recommendation framework and weighted random forest model were applied to generate personalized job recommendations for graduating students. The final set of recommended positions is visualized in Figure 10. The analysis shows that the 13 suggested job opportunities are not only appropriate in quantity but also exhibit a strong alignment with the characteristics of the student user profiles. This alignment is reflected in the feature words extracted from each profile, further confirming the effectiveness of the proposed algorithm.

Moreover, the consistency of these results across different student samples and job datasets underscores their statistical robustness. Even under varying input conditions, similar recommendation patterns were observed, owing to the use of random data partitioning, which ensured the representativeness of both training and testing sets. These findings reinforce the generalizability and reliability of the proposed method. Taken together, the results depicted in Figure 10 validate the statistical significance and practical feasibility of the recommendation framework for real-world deployment in college employment platforms.

```
array([
[ 9.25156960e-01,  9.34459458e-02,  4.63426047e-02], [-5.87550127e-01,  9.85503917e-01,  2.85608967e-02],
[ 2.82917504e-03, -9.87612334e-01,  3.57240377e-02], [ 9.25156960e-01,  1.24423862e-01,  8.47278105e-03],
[ 1.24487587e+00, -8.76793800e-02,  4.92217487e-02], [ 1.12213993e-02, -9.87612334e-01, -5.03381975e-02],
[-1.08089767e+00,  5.92884447e-02,  8.44598874e-03], [-2.44486236e-01,  1.10410630e-01,  7.70288700e-01],
[ 1.33935222e-01,  1.01254305e+00, -1.03756352e-01], [-1.08089767e+00, -4.37897688e-02, -3.73389598e-02],
[ 4.23835409e-02,  9.85503917e-01, -6.00969642e-02], [ 5.36374140e-03, -9.87612334e-01,  1.98680687e-02],
[-1.08089767e+00, -3.04247681e-01, -4.39955334e-02], [ 2.03848353e-01,  6.27353672e-04, -8.98413207e-01],
[-2.10852916e-01,  1.01254305e+00,  2.89494260e-02], [ 9.25156960e-01,  8.80361598e-02, -9.06311476e-03],
[-6.16988873e-01, -1.01470931e+00, -8.15426225e-04], [ 7.06183797e-03, -9.87612334e-01, -9.59478610e-03],
[ 9.25156960e-01,  5.29867978e-02,  4.37607093e-04], [ 2.16469718e-02,  9.85503917e-01,  1.16204440e-02],
[ 4.31034431e-04,  1.01254305e+00,  6.49856430e-02], [ 9.25156960e-01,  3.71758406e-02,  8.85997311e-03],
[ 3.49939041e-02, -1.01470931e+00,  4.05465383e-03], [-3.19940972e-02,  1.01254305e+00, -1.23536568e-03],
[-1.08089767e+00, -8.02128012e-02,  2.38697549e-02], [ 1.81554470e-03, -1.01470931e+00,  6.33331515e-03],
[ 6.71769480e-02, -9.87612334e-01,  2.03860376e-03]])
```

Fig. 8. Center points of each cluster and visual results

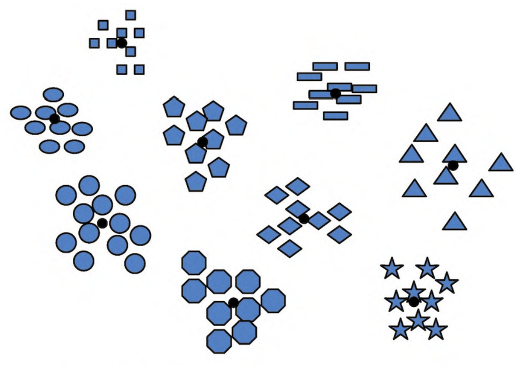

Fig. 9. Center points of each cluster and visual results

Digital Signal Processing engineer User match: 0.3145676054	Mechanical engineer User match: 0.32456/9913
Game 2D art designer User match: 0.4397860040	Data Analyst (Industrial) User match: 0.3477091430
Data Governance Engineer User match: 0.2356832350	Water supply and drainage engineer User match: 0.4605211360
Data Governance Engineer (2) User match: 0.4216932170	Data Analyst (Finance) User match: 0.3652689160

Fig. 10. Results of job recommendation for college students

D. Performance Evaluation and Metrics

To assess the effectiveness of the three job recommendation algorithms, user interactions with the recommended job lists were analyzed, and the hit rate for each method was calculated. The hit rate, denoted by Φ, quantifies the proportion of recommended positions that were actually accessed by users. It is computed using the following formula:

$$\Phi = \frac{1}{Q}\sum_{f=1}^{Q} r(f) \tag{8}$$

In this equation, Q denotes the total number of student users, f represents an individual user, and $r(f)$ is a binary indicator function that equals 1 if the user accessed a job contained in the recommendation list, and 0 otherwise.

Based on the computation in Equation (8), the hit rate comparison results for the three recommendation algorithms are illustrated in Figure 11. As shown, the hit rate increases as the length of the recommendation list grows, reflecting a higher probability of users engaging with at least one suggested position. For example, when the list length reaches 120 jobs, the hit rate of the proposed method climbs to 0.94, significantly outperforming the other two baseline algorithms, which remain below 0.8.

This marked improvement highlights the effectiveness of the proposed approach, which combines a bipartite graph recommendation framework with a weighted random forest model. The graph structure captures user-job associations efficiently, while the random forest refines the recommendation list by incorporating feature-based compatibility between user profiles and job attributes. Together, these components enable more precise and personalized job matching, resulting in a substantial increase in hit rate performance. The experimental results are statistically significant because they are based on a representative dataset, which proves the significant superiority of the algorithm proposed in this study by comparing it with other algorithms, and this advantage is verified across different portions of the dataset. Therefore, it can be concluded that the job recommendation algorithm presented in this paper holds high practical value for real-world applications.

To further evaluate the effectiveness of the proposed method, average cosine similarity was employed as an additional performance metric. Cosine similarity quantifies the similarity between two vectors by measuring the cosine of the angle between them—where higher values indicate closer alignment. In the context of job recommendation, a larger cosine similarity score reflects a stronger match between the attributes of recommended jobs and the capabilities of student users.

Thus, an elevated average cosine similarity suggests that the algorithm is better able to align candidate profiles with suitable job postings, resulting in more accurate and meaningful recommendations. The mathematical formulation for computing this metric is shown in Equation (9).

$$Q_{ACS} = exp\frac{\sum_{i=1}^{N-1}\sum_{j=i+1}^{N}(Z_i, Z_j)}{N(N-1)/2} \tag{9}$$

Where N is the number of samples; Z_i, Z_j denote the user sample.

To empirically assess the robustness of the proposed model, a total of 100,000 data samples were randomly selected from the university employment platform and partitioned into ten equal subsets of 10,000 records each. Multiple recommendation algorithms were applied to these subsets, and their performance was evaluated using average cosine

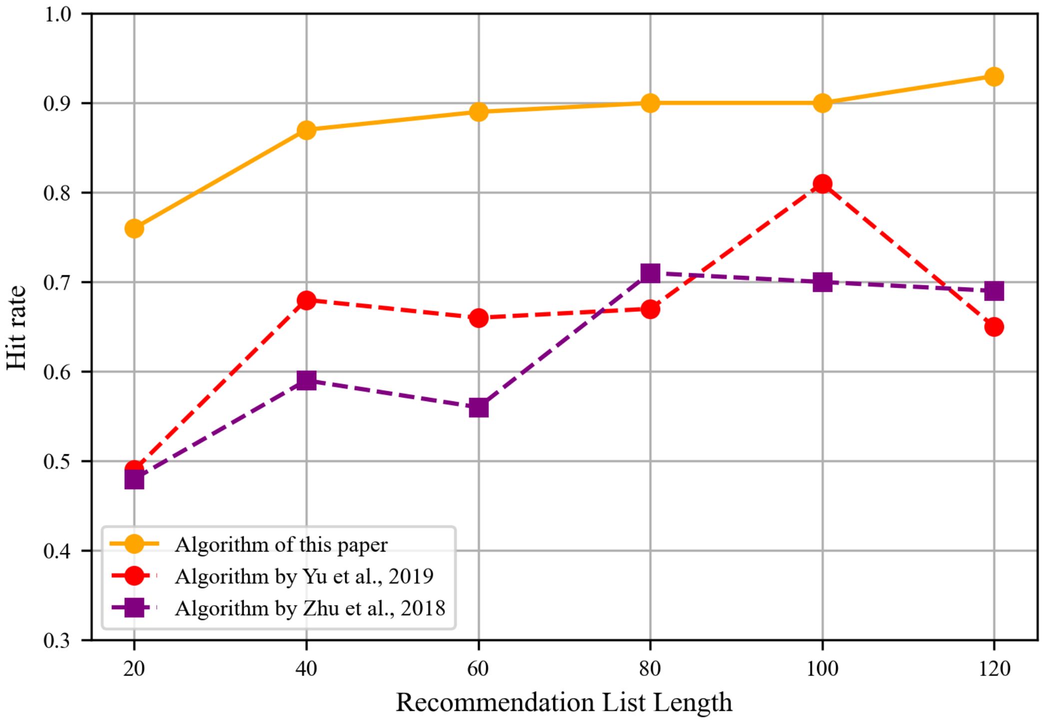

Fig. 11. Comparison of hit rat recommendation results of different methods

similarity, as depicted in 12.

The results clearly show that the proposed method consistently outperforms the benchmark approaches in terms of vector similarity, indicating its superior capability in aligning student competencies with job requirements. This performance advantage highlights the model's effectiveness in addressing challenges such as data sparsity and evolving user preferences. By maintaining high matching precision across diverse datasets, the proposed algorithm demonstrates strong adaptability and real-world applicability for personalized job recommendations.

To evaluate the contribution of the topic modeling component, an ablation study was conducted to compare recommendation accuracy before and after integrating the Biterm Topic Model (BTM). The results, presented in Figure 13, show a substantial improvement in accuracy following the inclusion of BTM.

The topic model enhances performance by clustering a large corpus of job descriptions into distinct semantic groups, each capturing shared requirements and characteristics. This grouping enables the system to recognize patterns and similarities across jobs more effectively, thereby improving the alignment between student profiles and job demands. By leveraging topic-based categorization, the model is better equipped to match students' skills and interests with appropriate positions.

In summary, the integration of BTM significantly improves the precision of job recommendations by semantically organizing extensive job data, ultimately facilitating more targeted and personalized employment guidance for college students.

To further validate the effectiveness of the proposed job recommendation algorithm, a comparative experiment was conducted using precision and recall as evaluation metrics.

TABLE II
THE TESTING RESULTS OF ACCURACY AND RECALL WITH DIFFERENT APPROACH

Methods	Accuracy	Recall
Method of this paper	0.96	0.89
Method by Yu et al., 2019	0.81	0.77
Method Zhu et al., 2018	0.74	0.67

The performance of the proposed method was benchmarked against two existing approaches from the literature [9][8], and the results are summarized in Table 2.

As shown in the table, the proposed method outperforms both alternatives, achieving higher scores in both precision and recall. This indicates superior accuracy in matching student profiles with job characteristics, while also covering a broader set of relevant opportunities. The performance gain can be attributed to key innovations in the model architecture—particularly the integration of recurrent neural networks and topic modeling.

By capturing sequential behavioral patterns through RNNs, and organizing job data semantically via the Biterm Topic Model, the system creates richer user representations. Additionally, the optimization of node relationships and weight distribution in the bipartite graph further enhances the system's ability to detect nuanced similarities between students and job postings, resulting in more effective recommendations.

E. Ablation Study

To evaluate the individual contributions of each component in the proposed employment recommendation model, we conducted an ablation study by progressively removing or

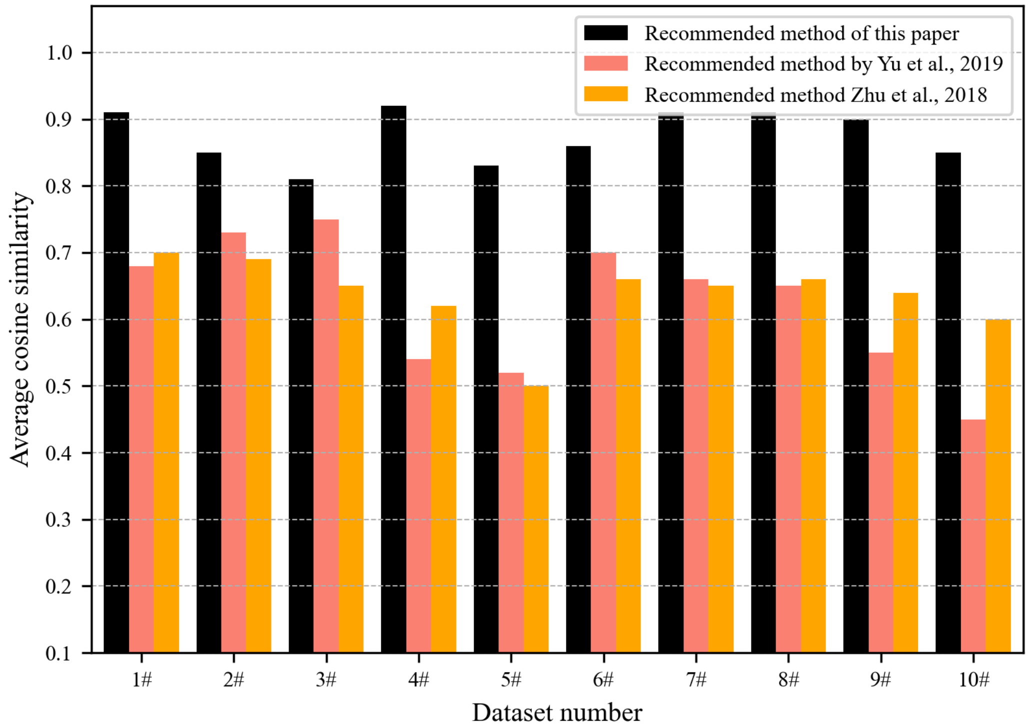

Fig. 12. Average cosine similarity comparison of the three methods

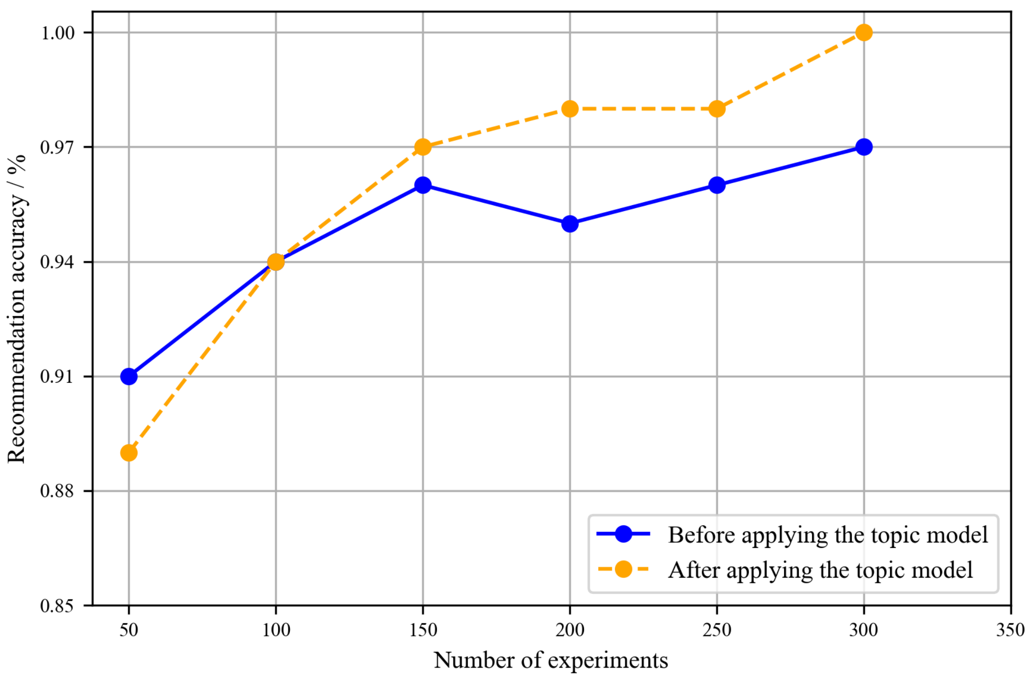

Fig. 13. Comparison of recommendation accuracy before and after the application of topic algorithm

replacing specific modules. The components under examination include the Biterm Topic Model (BTM), the bipartite graph recommendation structure, and the weighted random forest classifier. The performance of each ablated variant was assessed using accuracy and recall metrics on the same dataset. We designed the following model variants:

- Model A (Full model): Includes all modules—RNN-based user portrait, BTM clustering, bipartite graph recommendation, and weighted random forest.
- Model B (w/o Topic Model): Replaces BTM with simple keyword frequency-based clustering.
- Model C (w/o Bipartite Graph): Removes the bipartite graph structure and uses only RNN + BTM + RF.
- Model D (w/o Random Forest): Removes the final RF classifier, and uses graph-based ranking as the final recommendation list.

Each variant was trained and tested on the same dataset of 2,000 students and 100 job positions (with an 80/20 training-test split), using identical parameters to ensure fair comparison. The results in Table III and Figure clearly demonstrate that removing any component leads to a noticeable performance drop. In particular, the removal of the topic modeling module (BTM) resulted in the most significant decrease, indicating that semantic clustering of user profiles plays a key role in enhancing recommendation precision. Likewise, removing the bipartite graph structure diminished the system's ability to capture relational patterns between users and jobs, leading to less personalized recommendations. The weighted random forest module, while not as impactful as BTM, still contributed meaningfully by refining final recommendation rankings based on multi-dimensional features.

TABLE III
PERFORMANCE COMPARISON IN ABLATION STUDY

Model Variant	Accuracy	Recall
Model A (Full model)	0.96	0.89
Model B (No BTM)	0.84	0.76
Model C (No Graph)	0.87	0.79
Model D (No Random Forest)	0.89	0.81

These findings validate the synergistic effect of integrating all modules in the proposed model and highlight the necessity of each for delivering high-quality job recommendations.

VI. CONCLUSION

This study presents a novel job recommendation algorithm tailored for Chinese college students, leveraging recurrent neural networks (RNN) and Biterm Topic Modeling (BTM) to enhance the personalization and accuracy of employment recommendations. By constructing detailed user profiles and integrating semantic clustering with graph-based matching and ensemble learning, the proposed model effectively improves job-student alignment.

Experimental evaluations demonstrate that the approach significantly outperforms baseline methods across multiple performance metrics, including hit rate, cosine similarity, precision, and recall. Furthermore, the ablation study confirms the importance of each model component in achieving optimal results.

Despite these contributions, the study has limitations. The current user data were extracted solely from the university library's archive system, which may not fully capture students' diverse experiences and career preferences. Future research will aim to incorporate richer, multi-source datasets—including academic records, extracurricular engagement, online behavior, and internship history—to construct more holistic user profiles. This expansion will further enhance the system's adaptability, robustness, and applicability in real-world employment platforms..

ACKNOWLEDGMENT

The authors would like to thank all School of Computing members who invented this study. This study was conducted for the requirement of Post-Graduate Study.

REFERENCES

[1] B. Parida, P. KumarPatra, and S. Mohanty, "Prediction of recommendations for employment utilizing machine learning procedures and geo-area based recommender framework," *Sustainable Operations and Computers*, vol. 3, pp. 83–92, Jan. 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2666412721000489

[2] J. Betáková, J. Wu, I. Rudnák, and R. Magda, "Employment of foreign students after graduation in Hungary in the context of entrepreneurship and sustainability," *Entrepreneurship and Sustainability Issues*, vol. 8, pp. 553–570, Jun. 2021.

[3] L. Lu, X. Gao, and Y. Xiao, "Impact of college students' perceived teacher support on career exploration: A Self-Determination Theory approach," *Innovations in Education and Teaching International*, vol. 61, no. 3, pp. 528–540, May 2024, publisher: SRHE Website _eprint: https://doi.org/10.1080/14703297.2022.2147093. [Online]. Available: https://doi.org/10.1080/14703297.2022.2147093

[4] X. Xia, J. Li, X. Wang, Z. Zhang, and Y. Chen, "Study on Teaching Reform based on the Investigation of College Students' Entrepreneurial Environment," *Asian Agricultural Research*, vol. 14, no. 6, pp. 60–63, 2022. [Online]. Available: https://ageconsearch.umn.edu/record/333623/

[5] Z. Huang and G. Liu, "Prediction model of college students entrepreneurship ability based on artificial intelligence and fuzzy logic model," *Journal of Intelligent & Fuzzy Systems*, vol. 40, no. 2, pp. 2541–2552, 2021, publisher: IOS Press. [Online]. Available: https://content.iospress.com/articles/journal-of-intelligent-and-fuzzy-systems/ifs189247

[6] F. Mao and X. Yang, "Literary therapy based on positive psychology impact on college students' happiness," *Journal of Poetry Therapy*, vol. 37, no. 1, pp. 16–34, Jan. 2024, publisher: Routledge _eprint: https://doi.org/10.1080/08893675.2023.2188126. [Online]. Available: https://doi.org/10.1080/08893675.2023.2188126

[7] Z. Shokrzadeh, M.-R. Feizi-Derakhshi, M.-A. Balafar, and J. Bagherzadeh Mohasefi, "Knowledge graph-based recommendation system enhanced by neural collaborative filtering and knowledge graph embedding," *Ain Shams Engineering Journal*, vol. 15, no. 1, p. 102263, Jan. 2024. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2090447923001521

[8] Y. Teng and K. Yang, "Research on Social Recommendation Algorithm Based on PSO_kfcm Clustering and CBAM Attention Mechanism of Graph Neural Networks," *IAENG International Journal of Computer Science*, vol. 51, no. 8, pp. 936–948, 2024.

[9] X. Yu, F. Jiang, J. Du, and D. Gong, "A cross-domain collaborative filtering algorithm with expanding user and item features via the latent factor space of auxiliary domains," *Pattern Recognition*, vol. 94, pp. 96–109, 2019, publisher: Elsevier. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0031320319302122

[10] J. Liu, Q. Liu, and G. Sun, "Collaborative Graph Filtering Recommendation Algorithm Based on Public Interaction Ratio and Structural Neighborhood Contrastive Learning," *IAENG International Journal of Computer Science*, vol. 51, no. 8, pp. 1011–1019, 2024.

[11] R. Alatrash, R. Priyadarshini, and H. Ezaldeen, "Collaborative filtering integrated fine-grained sentiment for hybrid recommender system," *The Journal of Supercomputing*, vol. 80, no. 4, pp. 4760–4807, Mar. 2024. [Online]. Available: https://doi.org/10.1007/s11227-023-05600-w

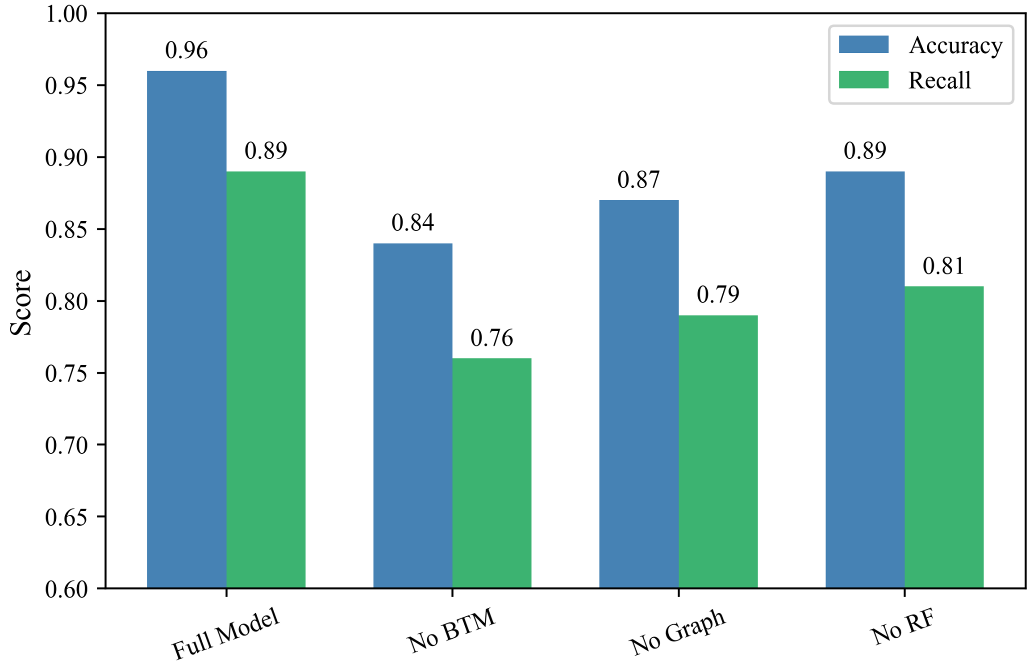

Fig. 14. Ablation Study Results by Model Variant

[12] Y. Mashayekhi, N. Li, B. Kang, J. Lijffijt, and T. De Bie, "A Challenge-based Survey of E-recruitment Recommendation Systems," *ACM Comput. Surv.*, vol. 56, no. 10, pp. 252:1–252:33, Jun. 2024. [Online]. Available: https://dl.acm.org/doi/10.1145/3659942

[13] X. Wang, Z. Dai, H. Li, and J. Yang, "Research on Hybrid Collaborative Filtering Recommendation Algorithm Based on the Time Effect and Sentiment Analysis," *Complexity*, vol. 2021, no. 1, p. 6635202, Jan. 2021. [Online]. Available: https://onlinelibrary.wiley.com/doi/10.1155/2021/6635202

[14] Y. Gao, H. Liang, and B. Sun, "Dynamic network intelligent hybrid recommendation algorithm and its application in online shopping platform," *Journal of Intelligent & Fuzzy Systems*, vol. 40, no. 5, pp. 9173–9185, 2021, publisher: IOS Press. [Online]. Available: https://content.iospress.com/articles/journal-of-intelligent-and-fuzzy-systems/ifs201579

[15] R. Vinodha and R. Parvathi, "Content and Location Based Point-of-Interest Recommendation System Using HITS Algorithm," *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, vol. 31, no. Supp01, pp. 31–45, May 2023. [Online]. Available: https://www.worldscientific.com/doi/10.1142/S0218488523400032

[16] X. Yan, J. Guo, Y. Lan, and X. Cheng, "A biterm topic model for short texts," in *Proceedings of the 22nd international conference on World Wide Web*. Rio de Janeiro Brazil: ACM, May 2013, pp. 1445–1456. [Online]. Available: https://dl.acm.org/doi/10.1145/2488388.2488514

[17] E. Ephzibah, R. Sujatha, and J. M. Chatterjee, "An adaptive neuro-fuzzy inference for blockchain-based smart job recommendation system," *International Journal of Information and Decision Sciences*, vol. 14, no. 1, p. 1, 2022.

[18] A. Brek and Z. Boufaida, "Enhancing information extraction process in job recommendation using semantic technology," *International Journal of Performability Engineering*, vol. 18, no. 5, p. 369, 2022. [Online]. Available: https://www.ijpe-online.com/EN/abstract/abstract4686.shtml

[19] T. T. Wells, C. A. Schofield, E. M. Clerkin, and E. S. Sheets, "The academic job market: Advice from the front lines," *The Behavior Therapist*, vol. 36, no. 2, pp. 39–45, 2013.

[20] D. Triantini and N. Yuniarti, "The innovation of technology utilisation in distributing employee by vocational secondary school career centre," in *Journal of Physics: Conference Series*, vol. 1833, no. 1. IOP Publishing, 2021, p. 012025.

[21] M. M. Afsar, T. Crump, and B. Far, "Reinforcement learning based recommender systems: A survey," *ACM Computing Surveys*, vol. 55, no. 7, pp. 1–38, 2022.

[22] C. Strobl, A.-L. Boulesteix, T. Kneib, T. Augustin, and A. Zeileis, "Conditional variable importance for random forests," *BMC Bioinformatics*, vol. 9, no. 1, p. 307, Dec. 2008. [Online]. Available: https://bmcbioinformatics.biomedcentral.com/articles/10.1186/1471-2105-9-307

[23] G. Egede Hansen, G. Bel, and O. Helby Petersen, "The Unequal Distribution of Consequences of Contracting Out: Female, Low-skilled, and Young Workers Pay the Highest Price," *Journal of Public Administration Research and Theory*, vol. 33, no. 3, pp. 434–452, 2023, publisher: Oxford University Press US. [Online]. Available: https://academic.oup.com/jpart/article-abstract/33/3/434/6730982

[24] W. U. Yun, L. Jian, and M. A. Yanlong, "A hybrid music recommendation model based on personalized measurement and game theory," *chinese Journal of electronics*, vol. 32, no. 6, pp. 1319–1328, 2023, publisher: CIE. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/10292733/

[25] O. Al-Sabana and S. O. Abdellatif, "Optoelectronic devices informatics: optimizing DSSC performance using random-forest machine learning algorithm," *Optoelectronics Letters*, vol. 18, no. 3, pp. 148–151, Mar. 2022. [Online]. Available: https://link.springer.com/10.1007/s11801-022-1115-9

[26] T. M. Akhriza and I. D. Mumpuni, "Quantitative class association rule-based approach to lecturer career promotion recommendation," *International Journal of Information and Decision Sciences*, vol. 13, no. 2, p. 192, 2021. [Online]. Available: http://www.inderscience.com/link.php?id=116507