IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received 4 November 2024, accepted 27 November 2024, date of publication 3 December 2024, date of current version 11 December 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3510581

RESEARCH ARTICLE

On Block Classification for Automatic Content Extraction From Chinese Resumes

LI-HUI ZHAO[1], JI ZHANG[1], (Senior Member, IEEE), QIQIANG XU[2], LINLIN HOU[3], AND XIN WANG[4], (Member, IEEE)
[1]School of Software, North University of China, Taiyuan 030051, China
[2]ByteDance, Shenzhen 518048, China
[3]Zhejiang Lab, Hangzhou 311121, China
[4]School of Computer Science and Software Engineering, Southwest Petroleum University, Chengdu 610500, China

Corresponding authors: Ji Zhang (ji.zhang.nuc@gmail.com) and Linlin Hou (houlinlin@zhejianglab.com)

This work was supported in part by the Natural Science Foundation of Shanxi Province under Grant 202203021212114.

ABSTRACT Resume information extraction technologies are crucial for automating the shortlisting and evaluation of resumes, benefiting both enterprises and job seekers. Resume block classification plays a pivotal role in the process of resume information extraction, as it significantly impacts the accuracy of the subsequent intra-block information extraction. However, as reported in the existing research, the classification of resume blocks often uses traditional and general-purpose text classification models, failing to consider the intrinsic contextual order relationships amongst resume blocks. Moreover, there are few studies that consider the transferability of classification models. Therefore, to address these limitations, we propose a series of methods to enhance the performance of resume block classification. Our approach focuses on three key aspects. Firstly, we introduce a novel sequence encoder that effectively extracts the sequence features among blocks of the same resume, leveraging the intrinsic contextual order relationships to improve classification accuracy. Secondly, considering the large classification variance of the existing models in different scenarios, we enhance the sequence encoder with a feature fusion strategy, which combines multiple features to improve the model's robustness and transferability. Thirdly, from the perspective of ensemble learning, we propose a dynamic weighted hybrid model that dynamically generates weighting for each participating sub-model, enabling adaptive integration of different classification models. Finally, to alleviate the huge workload of cross-domain relabeling, we develop a transfer learning model specifically designed for the resume block classification task, facilitating the application of our approach across different domains. To evaluate the effectiveness of our proposed methods, we release three Chinese datasets that include 4,500 Chinese resumes on https://github.com/xqqhelloword/resume-block-classification. Experimental results show that our hybrid model achieves 97.6% accuracy, outperforming existing methods and establishing a new state-of-the-art in this field.

INDEX TERMS Resume block classification, recurrent neural network, Transformer, feature fusion, hybrid model, transfer learning.

I. INTRODUCTION

Automatic processing and intelligent information extraction from large volumes of resumes have become increasingly important tasks for job-seeking websites and human resource departments across various organizations. These tasks enable efficient management and utilization of the information contained in resumes [1], [2], [3], [4]. However, the vastly differing writing styles employed in different resumes make accurate information extraction a non-trivial task. As described in Figure 1, this intelligent extraction process usually involves three key steps: resume segmentation, resume block classification, and intra-block information extraction [5], [6]. Among these steps, resume block classification plays a crucial role, as it essentially tackles a text classification problem [7], [8], [9]. However, unlike general

The associate editor coordinating the review of this manuscript and approving it for publication was Biju Issac.

classification tasks, resume block classification targets special scenarios where the samples are orderly organized in a semi-structured document, such as a resume. This unique characteristic of resumes presents additional challenges and opportunities for developing effective classification methods.

Resume block classification is a significant step in resume information extraction [10], [11], [12], as it plays a crucial role in guiding the subsequent step of intra-block information extraction. The intra-block extraction algorithms vary depending on the resume block type, making accurate block classification essential for the overall success of the information extraction process. However, the existing research works only treat the resume block classification as a general classification problem, and they do not sufficiently consider the contextual sequential relationship among different resume blocks in one resume, such as [13] and [14]. This oversight can lead to suboptimal block classification results and may require a relatively large amount of training data to compensate for the lack of contextual information. Moreover, previous works rarely consider the transferability of models, which can result in a significant workload when applying the classification methods to new domains. The need for extensive relabeling of data in each new domain hinders the practical applicability and scalability of these approaches.

In the past research, most of the algorithms applied to resume block classification are the traditional text classification algorithms, such as Naive Bayes (NB), Support Vector Machine (SVM), and Random Forest (RF) [15]. While these algorithms have proven effective for general text classification tasks, they often fail to consider the unique intrinsic structural features of resumes, which can limit their performance in this specific domain. Resumes exhibit a distinct structure, typically consisting of a few blocks arranged in a certain order. This order is influenced by several factors, such as: 1) most job seekers fill in the resume template on recruitment websites, which implement pre-formatted templates, and 2) the general writing style of job seekers is usually similar. These factors contribute to the semi-structured nature of resumes, where the contextual order of blocks within the document carries important information. However, existing algorithms, such as [16], [17], and [18], often treat each block as largely independent from each other, failing to consider the contextual order of each block within the resume. This oversight can lead to suboptimal classification performance, as the algorithms do not fully leverage the intrinsic structural features of resumes.

Therefore, in this paper, to improve the performance of resume block classification, we propose a block sequence encoder that learns the intrinsic contextual order relationship between different resume blocks in a resume. Our method involves building a block-level bi-directional recurrent neural network (BRNN) encoder to capture the contextual order features among these resume blocks. By leveraging the sequential information inherent in the resume structure, the proposed encoder can better understand the relationships between different blocks and improve classification accuracy.

However, when the training data is not comprehensive enough to reflect the special scenarios of block sequence feature distribution, there might still be some anomalous resumes whose block sequence features do not follow the normal features observed in the majority of resumes. These anomalies can arise due to various reasons, such as applicants using casual formats or changes in the original order of resume blocks during the segmentation process. To reduce the impact of the sequential order of the abnormal resumes on the performance of the model, we further propose several strategies for resume block classifications. These strategies aim to reduce the weight of order features in the model's classification decisions while correspondingly increasing the weight of text content features.

Firstly, building on the idea of the aforementioned block sequence encoder BRNN, we additionally employ the feature fusion strategy [19] to enhance the stability of block features. This enhanced encoder, which we refer to as the BRNNF encoder, combines the contextual order features learned by the BRNN with additional features, such as textual content features and layout features. By integrating multiple types of features, the BRNNF encoder can provide a more comprehensive and robust representation of resume blocks, reducing the impact of anomalous resumes on the classification performance.

Secondly, to further mitigate the influence of sequential order in abnormal resumes, we design an additional recurrent neural network (RNN) structure that classifies each resume block individually. This individual block classifier (IBC) focuses solely on the textual content of each block, disregarding the sequential characteristics of different resume blocks. By treating each block independently, the IBC can provide a complementary perspective to the BRNNF encoder, which considers the contextual order information.

Finally, to leverage the strengths of both the BRNNF encoder and the IBC, we propose a novel hybrid model strategy based on ensemble learning. Our hybrid model dynamically adjusts the weight of each base model (BRNNF and IBC) for each input sample through a label sequence probability distribution (LSPD) method. LSPD assesses the orderliness level of resume blocks, allowing the hybrid model to adaptively balance the contributions of the BRNNF encoder and the IBC based on the characteristics of each resume.

In addition, based on the BRNN and BRNNF encoders mentioned above, we make some changes on the domain adversarial neural network (DANN) transfer model [20], introducing two new variants called DA-BRNN and DA-BRNNF. These variants are specifically designed to address the challenge of cross-domain adaptation in resume block classification and significantly reduce the workload of relabeling when transforming to a new domain. Overall, the advantage of the method proposed in this paper over other methods lies in its ability to capture the contextual

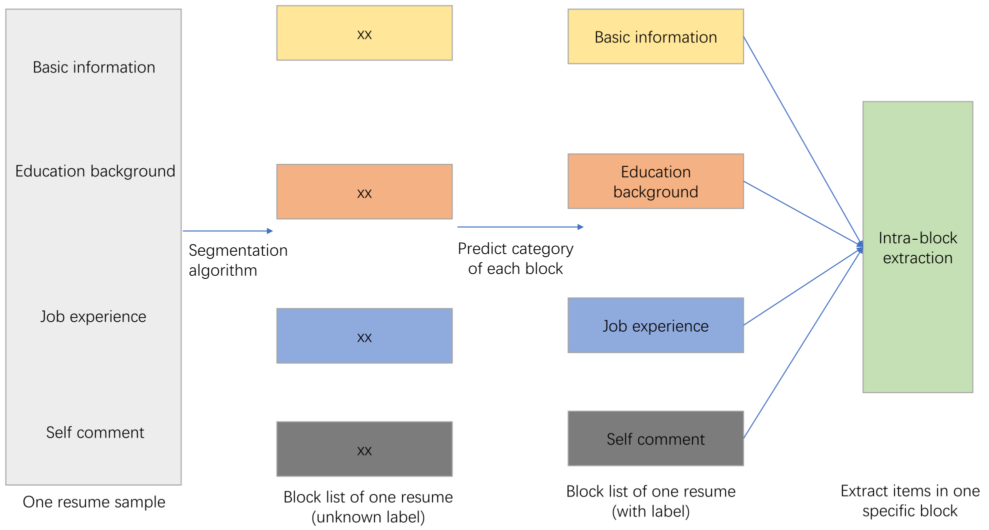


FIGURE 1. The general three steps resume information extraction framework.

sequential relationships between resume blocks through a sequence encoder. Additionally, it enhances the robustness and transferability of the sequence encoder through feature fusion, effectively improving recognition accuracy by leveraging transfer learning techniques.

The main contributions of this paper are summarized as follows:

1) We propose the block sequence model which consists of a block feature extractor and a BRNN encoder which can fully consider the sequential feature of resume blocks, thereby improving the classification accuracy of resume blocks. This model leverages the intrinsic contextual order relationship between different resume blocks to enhance the classification performance.
2) To specifically deal with the abnormal resumes, we further propose an enhanced BRNN encoder with feature fusion, called BRNNF, and a hybrid model to further improve the model's adaptability. The BRNNF encoder combines the contextual order features learned by the BRNN with additional features, such as textual content features and layout features, to provide a more robust representation of resume blocks. The hybrid model dynamically adjusts the weights of the BRNNF encoder and an IBC based on the characteristics of each resume, maximizing the overall block classification performance.
3) For the hybrid model, we design an innovative dynamic weighted hybrid strategy which combines a RNN classifier and a LSPD structure to increase the classification accuracy of the hybrid model. The LSPD assesses the orderliness level of resume blocks, allowing the hybrid model to adaptively balance the contributions of the BRNNF encoder and the IBC based on the characteristics of each resume. This dynamic weighting scheme distinguishes our approach from existing hybrid model integration methods and enhances the classification accuracy.
4) We also consider the transferability of the model for resume block classification and to propose an effective transfer method to significantly reduce the workload of relabeling when transforming to a new domain. We introduce two variants of the DANN transfer model, called DA-BRNN and DA-BRNNF, which incorporate the BRNN and BRNNF encoders, respectively. These variants enable the efficient and effective adaptation of the classification model to new domains by learning domain-invariant features through adversarial training, greatly reducing the need for extensive relabeling in the target domain.
5) The extensive experiments demonstrate that our model can not only achieve a high accuracy (97%) on normal resumes but a high accuracy (95.5%) on abnormal resumes, which is the state-of-the-art performance on the resume block classification task. These results showcase the effectiveness of our proposed methods in handling both normal and abnormal resumes, outperforming existing approaches and setting a new benchmark for resume block classification.

The remainder of this paper is organized as follows: Section II discusses the related work. Section III introduces our proposed model for resume block classification. Section IV describes the experimental design for evaluating our model and analyzes the experimental results. The final section concludes the paper and highlights some future research directions.

II. RELATED WORK

A. RESUME BLOCK CLASSIFICATION

A method of extracting resume information was first proposed by using the Hidden Markov Model (HMM) to divide the resume into blocks, and then by selecting the corresponding extracting algorithm according to different categories of resume blocks [5]. This approach regards the resume text as a token sequence, and each token is labeled with the "BISO" mechanism, where "B" represents the beginning of a block, "I" represents the intro of a block, "S" represents an independent block, and "O" represents useless information. This method achieves both resume segmentation and resume block classification. Experiments show that this method outperforms full-text information extraction methods that do not take the step of resume segmentation [21], [22]. However, in this scheme, how to accurately identify the category of the resume block becomes a technical bottleneck, which directly affects the selection of the extraction algorithm in the subsequent steps. Using only the labeling result of HMM may lead to low segmentation accuracy, since HMM model adopts the Homogeneous Markov Hypothesis and fails to catch a long distance dependence between two tokens. Therefore, in this paper, we separate two steps: firstly, use a segmentation algorithm to perform the resume segmentation, and secondly, conduct the resume block classification.

Currently, there are two main types of resume block classification methods [23], [24], i.e., classifying by detecting titles and determining categories based on content through traditional MI-based algorithms. For the first type, there are two ways to detect block titles, one is to match the title according to keywords, such as [25], [26], [27] and [28], while the other way uses a certain multi-class classification model, such as [14]. Keywords matching is easy to implement, but the drawbacks are also obvious. It requires building a large keyword library, and the potential block titles are quite diverse. Furthermore, the text length of block titles is generally short, which leads to low accuracy, and the real category of corresponding content may not be consistent with the title. On the other hand, MI-based algorithms also have two different ways to predict the categories of resume blocks. The first way is to predict the category after resume segmentation, as illustrated in [13], and the second way is to perform the resume segmentation and block classification simultaneously, as with the aforementioned HMM model [5]. MI-based methods do not depend on the keyword library and are generally much more accurate than methods of detecting titles. However, the disadvantage of MI-based methods is that they need a large amount of labeled data for supervised training, which is a time-consuming task. In the complex real-world situation such as resume block classification, due to the noise data, the classification performance of MI-based methods still has significant room for improvement. There is little research that explores the use of deep learning to leverage the contextual order of resume blocks for resume block classification [29], [30], [31]. Our model proposed in this paper takes this into account.

B. ENSEMBLE LEARNING

The existing hybrid model strategies in ensemble learning include Bagging such as Random Forest [32], Boosting such as [33], Blending such as [34], and Stacking such as [35]. But in essence, the importance of each basic sub-model in these hybrid models has been fixed in the training stage, and in the prediction stage, for any test sample, the hybrid model will assign the same static weight of each base model learned in the training stage to produce the final classification result, while in this paper, our novelty lies in dynamic decision. For a given specific test sample, our hybrid model judges the importance of each base model for the current classification task and dynamically assign a new weight for each sub-model to produce the final decision result.

C. TRANSFER LEARNING

Transfer learning refers to a model trained in task A and having the ability to solve problems in task B. In recent years, transfer learning has been widely used in NLP, such as sentiment analysis [36], sequence translation [20], word vector pre-training model [37], etc. According to its internal principles, transfer learning can be divided into four types: instance-based learning [38], feature-based learning [39], shared parameter [40], and the combination of deep learning and transfer learning [41]. Among them, the instance-based method aims to find out the samples in the source domain that are useful for the classification of the target domain through analysis, and increase the weight of these samples. The feature-based scheme aims to find the common feature representation method of the samples in the two domains, and the parameter-based scheme aims to find the common parameters of the source domain and the target domain model. The transfer learning scheme combined with deep learning focuses on the idea of pre-training and fine-tuning of the deep learning model.

DANN is a variant of the Generative Adversarial Network (GAN) [42], which is mainly used to solve the adaptation problem between the source domain dataset and the target domain dataset. Some researchers use DANN for transfer learning, such as [41] and [42]. The source domain dataset has classification labels, while the target domain dataset has not been manually labeled. DANN assumes that the feature space distribution of source domain data and target domain data is different, but the task is the same, that is, the classification categories are the same. Through training, DANN finds a common feature space distribution between source domain and target domain, so that the data characteristics in this feature space can represent the sample data characteristics of both source domain and target domain. In this paper, we make some changes based on the DANN transfer model [41], which is not only a feature-based method but also a combination of deep learning and transfer learning.

III. OUR MODEL

In order to make full use of the sequence feature among the resume blocks to improve the performance of block classification and enhance the model's adaptability to abnormal resumes, a dynamic weighted hybrid model that combines the block sequence model with a jointly trained

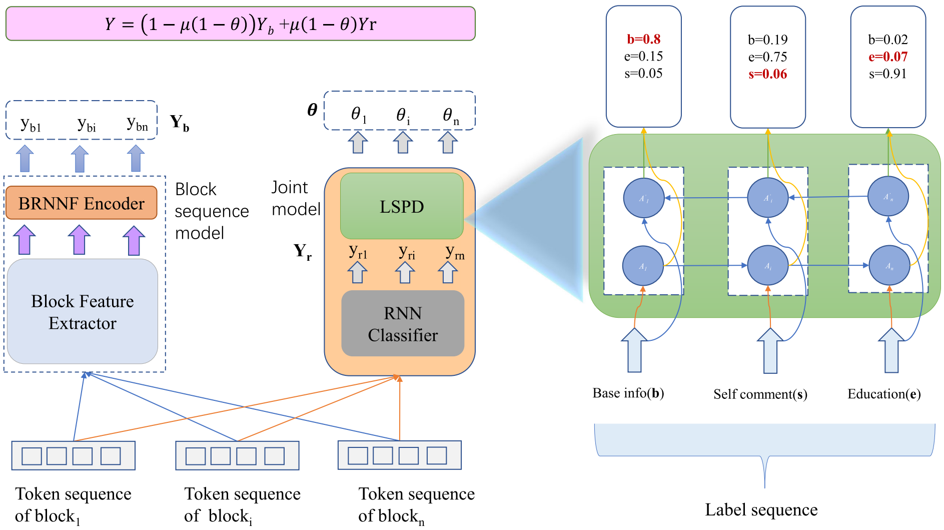


FIGURE 2. This figure is the dynamic weight hybrid model overview. It includes a block sequence model which consists of a block feature extractor and a BRNNF encoder, and a RNN + LSPD jointly trained model where RNN produces the predicted label sequence, and LSPD produces the confidence level of the block sequence model for current block classification. The output Y is calculated by the formula at the top of this figure.

RNN + LSPD model is proposed in this paper. Figure 2 provides an overview of our proposed model. The input is a block sequence obtained by the resume segmentation step. In the following, we will elaborate on the method of resume segmentation. Each block consists of several word tokens, each of which is embedded by the pre-trained word2vec model [43]. The input shape is $b \times n \times m \times d$, where b is the batch size, n is the maximum block number in a resume, m is the maximum word number in a block, and d is the length of the embedding vector. The output are the labels of all the blocks in a resume. Note that our model predicts all the block categories in a resume at once. So, the output shape is $b \times n \times k$, where k is the class number. In this paper, we set $k = 10$.

A. RESUME SEGMENTATION

In the resume segmentation step, we consider the fusion of text features and format features, and use these features to train a binary classification model to judge whether each line in the resume is a block title. When we detect that a line is a block title, we truncate it to obtain a new block. The binary classification model we use in this paper is a shallow neural network described as Figure 3. For instance, if the content of one line is "education background:", firstly, we use word2vec model to transform each word into embedding vector. Assume that the length of line is m, the transformation result is a matrix with the shape $m \times d$. This matrix is then input into the RNN Encoder and a max-pooling layer to obtain the text feature vector h_t. At the same time, we use the one-hot technique to represent the format feature, which is then input into a fully connected layer to obtain the final format feature h_f. Finally, the output is given by: $sigmoid(W[h_t; h_f] + b)$.

Here, the reasons for choosing the RNN encoder in our work are as follows: 1. Sequence Data Processing Capability: The text in resumes exhibits strong sequential characteristics, and RNNs have a significant advantage in processing sequential data. RNNs can effectively capture contextual information through hidden states, which is crucial for identifying key information in resumes, such as educational background and work experience. 2. Information Transmission and Memory: RNNs can transmit information between different time steps in a sequence, making them suitable for handling the contextual dependencies that may exist in resumes. The reasons for choosing max pooling in our work are: 1. Feature Selection: Max pooling helps select the most significant features during the feature extraction process, reducing redundant information and emphasizing important content. For different sections of a resume (such as skills and project experience), max pooling can highlight the most critical information. 2. Computational Efficiency: Max pooling reduces the complexity and computational requirements of the model, allowing us to process data more efficiently during both the training and inference stages.

B. BRNN ENCODER

Figure 4 (left) describes the BRNN structure. The input is the block sequence $\{block_0, \ldots, block_i, \ldots, block_{n-1}\}$, where n is the maximum number of blocks, and $block_i$ is composed of $\{word_0, \ldots, word_i, \ldots, word_{m-1}\}$, where m is the maximum size of words in one block, each word is embedded by word2vec. Then, we put each $block_i$ into a block feature extractor (the feature extractor can be a CNN, RNN, Transformer, BERT). Finally, the corresponding x_i is obtained, which represents the feature vector of $block_i$. The input of BRNN encoder is $X = \{x_0, \ldots, x_i, \ldots, x_{n-1}\}$. After passing

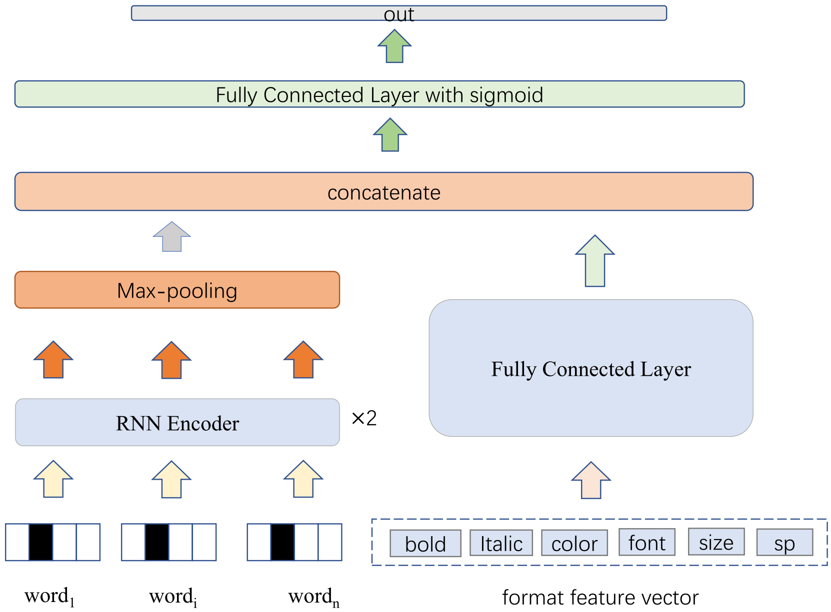


FIGURE 3. Resume segmentation model in this paper, the format feature vector is a vector of length 6, and the "sp" means the number of special char in current line, such as ":", "-", etc.

through the bidirectional gated recurrent unit (Bi-GRU) [44], the hidden output $H = \{h_0, \ldots, h_i, \ldots, h_{n-1}\}$ is obtained, as the input of a dense layer with softmax active function, finally obtains the output $Y = \{y_0, \cdots, y_i, \cdots, y_{n-1}\}$. Then, we choose cross entropy as the loss function, the total loss of one batch is described by equation 2:

$$Loss = \frac{1}{N}\sum_{s=1}^{N}\sum_{i=1}^{n} L_s(y_i, y_i') \tag{1}$$

$$= -\frac{1}{N}\sum_{s=1}^{N}\sum_{i=1}^{n}\sum_{j=0}^{k-1} p_{i(s)}'^{(j)} \log p_{i(s)}^{(j)} \tag{2}$$

where N is the batch size, $L(y_i, y_i')$ is the cross entropy of one time step in BRNN, and the total loss is the sum of the loss in each time step.

C. BRNNF ENCODER

As mentioned above, due to the disadvantages of BRNN, we make some improvements, as described in Figure 4 (right), which is called BRNNF encoder. We employ a feature fusion strategy to concatenate the input X and the hidden output of Bi-GRU, H. The purpose of this is to allow the model not only pay attention to the sequence features among resume blocks but also to some weights to the text content features of the resume block itself, thereby reducing the sensitivity of BRNN to abnormal resume data with different context sequence characteristics.

D. DYNAMIC WEIGHTED HYBRID MODEL

We leverage the idea of ensemble learning to combine our BRNNF model with a general classification model which is not affected by the sequence features among different resume blocks. Specifically, we design a dynamic weighted hybrid model which combines BRNNF with a jointly trained model, as presented in Figure 2. The joint model consists of a RNN structure and the aforementioned LSPD. Note that the RNN only predict one block individually. Therefore, the result of the joint model won't be affected by the sequence feature of resume blocks. The reason we choose the RNN structure is that RNN can achieve a high accuracy when facing the abnormal samples based on the experiment statistic results, and has a very good stability, although it has a lower accuracy than our block sequence model when facing normal samples. In our model, when abnormal data appears, the RNN classifier in the joint model plays a major role, and when facing normal resume data, the weight of BRNNF is increased to improve the generalization ability of the model and reduce the variance of the model.

The specific way to implement "dynamic weight" is described as Figure 2. Specifically, the LSPD is a Bi-GRU structure, used to judge the order rationality of the resume blocks' arrangement in current resume sample, since the performance degradation of BRNNF is caused by the disorderliness of the arrangement among different blocks. Our LSPD is based on the assumption that when the RNN classifier in joint Model predicts a resume block sequence, we regard the result of RNN classifier as the ground truth. Then, the LSPD scores the order rationality of the "ground truth" label sequence, and takes it as the weight assigned to the block sequence model and the RNN classifier. For example, in Figure 2, RNN classifier generates a prediction label sequence: "base info (b)", "self comment (s)", "Education (e)". After LSPD scoring, the

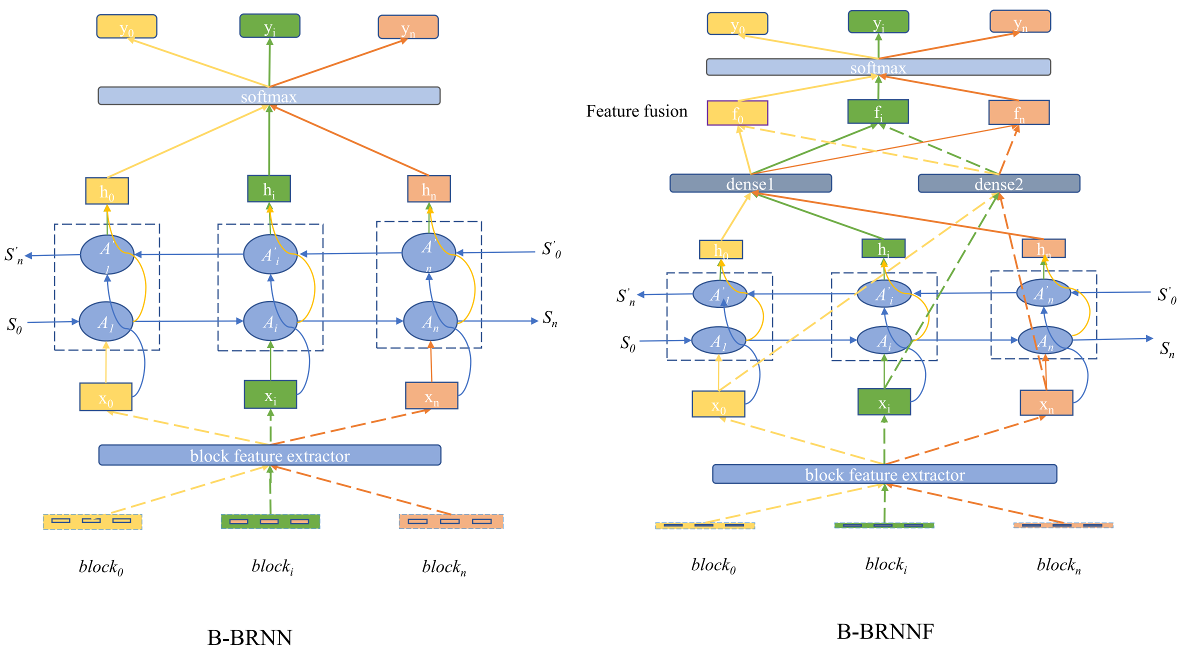


FIGURE 4. BRNN and BRNNF structure proposed in this paper.

score of the first prediction label "base info (b)" is obtained: $score_1 = (0.8, 0.15, 0.05)$. Then, according to $b = 0.8$, the confidence weight of the block sequence model is 0.8. For the second prediction label "self comment (s)", $score_2 = (0.19, 0.75, 0.06)$. According to $s = 0.06$, the confidence weight of the block sequence model is 0.06, which indicates that the probability of the second sequence labeled "self comment" is very low under normal circumstances, i.e., the arrangement of resume blocks is irrational. Therefore, the weight of the block sequence model for classification decision should be reduced.

The RNN classifier and LSPD are trained through joint training. When given a block sequence $input = \{block_1, \cdots, block_i, \cdots, block_n\}$, we input it into the RNN classifier, and obtain $Y_r = \{y_{r1}, \cdots, y_{ri}, \cdots, y_{rn}\}$. In the training stage, we define the input of LSPD as the mixture of ground truth label sequence and RNN classifier prediction label sequence with a parameter T:

$$input_{LSPD} = (1 - T) \cdot Y' + T \cdot Y_r \tag{3}$$
$$= \{L_1, \cdots, L_i, \cdots, L_n\}, \tag{4}$$

where T is a time-varying coefficient that increases with the number of iterations, $Y' = \{y'_1, \cdots, y'_i, \cdots, y'_n\}$ is the ground truth label sequence. L_i is a k-dimension one-hot vector, which is computed by the follow equation:

$$L_i = argmax[(1 - T) \cdot softmax(y'_i) + T \cdot y_{ri}]. \tag{5}$$

The output of LSPD is the probability distribution of categories at each time step. For instance, given a specific label sequence input $input_{LSPD}$, we design an embedding layer to learn the distributed representation of each class label. After the embedding layer, we get the embedding vector sequence $Embedding(input_{LSPD}) = \{E_1, \cdots, E_i, \cdots, E_n\}$. Then, the context feature of the i-th time step calculated by the Bi-GRU structure is: $c(E_1, \cdots, E_{i-1}, E_{i+1}, \cdots, E_n)$. Note that the embedding vector of current time step E_i is not included here. The output of the i-th time step in LSPD is a probability distribution calculated by a dense layer with the softmax function, which represents the probability of each resume block category at the current time step given the context feature $c(E_1, \cdots, E_{i-1}, E_{i+1}, \cdots, E_n)$. The formula is described as:

$$output_i = P(K|E_1, \cdots, E_{i-1}, E_{i+1}, \cdots, E_n) \tag{6}$$
$$= (p_i^1, \cdots, p_i^j, \cdots, p_i^k)^T, \tag{7}$$

where $K = \{1, 2, \cdots, k\}$ represents the all possible categories of the i-th resume block and k is the count of all categories.

Then, we use the label smoothing technique [45] to avoid the problem that the probability distribution of different labels generated in each time step of LSPD is too concentrated in one category, while the probability difference of other categories is too small. Specifically, we set a parameter λ to control the smoothing degree:

$$L_i(\lambda) = (1 - \lambda) \cdot L_i + \frac{\lambda}{k}. \tag{8}$$

where L_i is the label in the i-th time step. So, the loss of LSPD can be described as:

$$Loss = \sum_{i=1}^{n} H(L_i(\lambda), output_i), \tag{9}$$

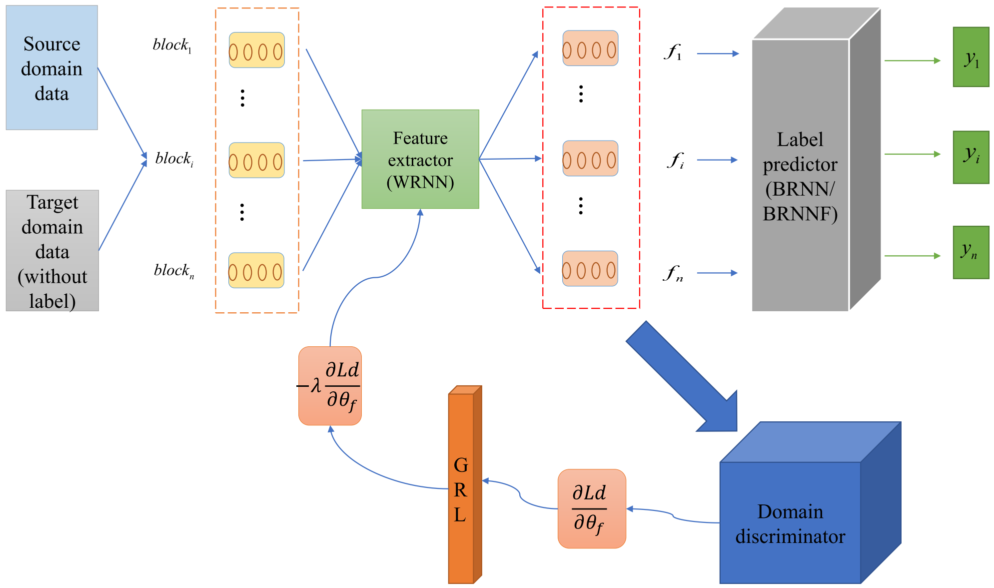


FIGURE 5. DA-BRNN and DA-BRNNF transfer model proposed in this paper.

where $H(\cdot)$ is the cross entropy function. Then the weight θ is produced by the follows:

$$\theta 7 = \{\theta_1, \cdots, \theta_i, \cdots, \theta_n\} \tag{10}$$
$$= \{p_1^{\arg\max(L_1)}, \cdots, p_i^{\arg\max(L_i)}, \cdots, p_n^{\arg\max(L_n)}\}. \tag{11}$$

The final output of this hybrid model Y is calculated by:

$$Y = (1 - \mu(1-\theta)) \cdot Y_b + \mu(1-\theta) \cdot Y_r, \tag{12}$$

where Y_b is the output of the block sequence model, μ is the penalty factor. We consider that the RNN classifier in the joint model also has a prediction error rate, and the θ generated may not be completely correct. Therefore, a penalty factor is used in here.

E. TRANSFER MODEL FOR RESUME BLOCK CLASSIFICATION

We apply the DANN structure to our resume block classification model, as shown in Figure 5. Different from the general DANN, the input of DANN in this paper is a sequence of data, representing a resume sample, and each item in the sequence is a text vector, representing a resume block. The label classifier is BRNN or BRNNF proposed in this paper. Inside the feature extractor is a bidirectional GRU, which we call Word-level RNN (WRNN). Each item in the input sequence is a word. N inputs share the parameters in the feature extractor, and each gets the corresponding feature vector after passing through the feature extractor.

After the DA-BRNN model training is completed, we can separate the parameters of the feature extractor and the label classifier and package them into an independent model. In this way, the independent model acts as a shared model with an effective classification capability for both the source and target domains. Then, we use a small number of labeled samples from the target domain to fine-tune the parameters of the shared model to obtain the final transfer classification model.

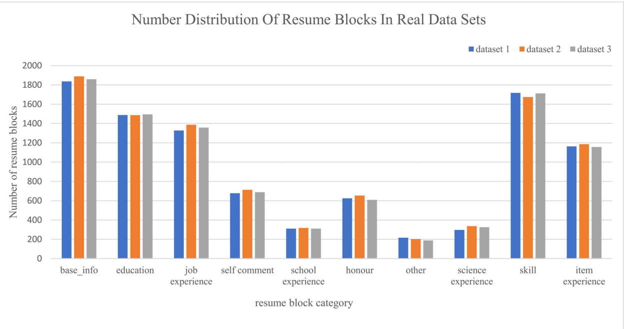


FIGURE 6. The distribution of the number of categories in our dataset.

IV. EXPERIMENTAL EVALUATION

A. DATASETS

Due to the complexity of data collection and the scarcity of resume resources, there are no open source datasets available. Therefore, we open source three datasets on Github[1] from real-world scenarios, which are related to the art, engine, and IT fields, respectively. Each dataset contains 1,500 resumes, and a total of 4,500 resumes are collected. Then, we segment each resume into a block sequence arranged by the original order and provide the label of each block as the input of the proposed hybrid model. We divide all resume blocks into 10 categories: base information, education background, honour, project experience, academic experience, work

[1]https://github.com/xqqhelloword/resume-block-classification

experience, school experience, self comment, skill, other. The distribution of the number of categories in our dataset is shown in Figure 6.

B. EXPERIMENT SETTINGS

In our experiments, each dataset is divided randomly into a training set, a verification set, and a test set with a proportion of 7: 2: 1. And the dimension of word embedding is set to 300. The input shape is $8 \times 12 \times 400 \times 300$, i.e., $b = 8$, $n = 12$, $m = 400$, $d = 300$. Here, the batch size b or N is 8, the maximum block number n is 12, the maximum word number m is 400. length of the embedding vector d is 300. The output shape is $8 \times 12 \times 10$, where the class number k is 10 in this paper. In RNN, the dimension of the feature vector h is 400×300. The learning ratio and the dropout rate are set to 0.001 and 0.5, respectively. Parameter θ is the dynamic output weight vector of LSPD during inference. Each element in the theta vector has a value range of (0, 1). Parameter u is the attenuation factor, which weakens the weight of Y, and its value is 0.9. And, we set λ to 0.2. We use the Adam optimizer to minimize the loss function. For the package-library, our model is implemented using Pytorch and Python 3.7.

In this paper, the hyperparameters are selected through a fixed strategy. Specifically, we empirically chose a set of fixed hyperparameter values (e.g., learning rate, batch size, etc.) based on a literature review of prior relevant work and preliminary experimental results. We considered the model's generalization ability in our experiments, demonstrating its performance across our three domain datasets using cross-validation methods. We continuously adjusted the combinations of different values for the fixed hyperparameters and then selected the one with the optimal performance. We ran each experiment five times and used the mean as the result. The experimental results indicate that the fixed hyperparameter settings can ensure good generalization performance in most cases.

In this section, four main experiments are designed. The detailed experimental settings are as follows:

(1) Compare the classification performance of each model on the normal resume test set. The size of each training set is 700 resumes (about 4,500 blocks), the test set is 400 resumes (about 3,500 blocks), and the validation set is 300 resumes. We choose TextCNN [46], RNN (Bi-GRU) [44], MLP [47], [48], Transformer [49], BERT [50] as the block feature extractors to obtain the feature vectors of blocks. Based on this, we test the accuracy change after adding the BRNNF encoder, i.e, TextCNN + BRNNF, RNN + BRNNF, MLP + BRNNF, Transformer + BRNNF, BERT + BRNNF.

(2) Evaluate the adaptability of our proposed model to the test containing abnormal resumes. We choose RNN, RNN + BRNN (ours), RNN + BRNNF (ours), the static weight hybrid model, the dynamic weight hybrid model (ours) to train on the same 700 normal resumes. Note that the static weight hybrid model only averages the results of the block sequence model and the RNN classifier of the joint model with a constant weight. We select RNN because it achieves the best result among general models (i.e., without our BRNNF encoder) except the BERT. BERT is a large pre-trained model which needs substantial memory and computational power. Then, we observe the accuracy change on the test set composed of 700 resumes with the proportion of abnormal samples ranging from 0% to 100%.

(3) In order to evaluate the transfer learning ability of the proposed DA-BRNN model for resume block classification in different domains, we design a cross-domain transfer learning experiment. We reorganize the three previously collected datasets, divide them into electronic, non electronic engineering, and social sciences, according to their respective industry domains, and mark them as IT, NoIT, and ART, with each domain containing 1,500 samples. Then, we record the performance of the DA-BRNN model in the target domain, at the same time, we also record the performance of model which only uses source domain dataset as the comparison result.

(4) In addition, we conduct some additional detailed experiments to further explain how our model works, such as: a) The weight value θ distribution generated by the LSPD structure in Figure 2 on normal and abnormal datasets. b) From the data perspective, the reason for the decrease of BRNN classification performance under abnormal resume samples is the lack of learning of abnormal samples. Therefore, we design an experiment to add a certain proportion of abnormal data into the normal training set and observe its performance changes under different proportions of abnormal data in the test set, to verify the effectiveness of simply mixing abnormal data.

C. EXPERIMENTAL RESULTS

1) COMPARATIVE STUDY

Table 1 shows that when using block feature extractors based on CNN, RNN, MLP, Transformer, and BERT, respectively, with the addition of the BRNN or BRNNF encoder, the accuracy rate is improved compared with the situation that only using block feature extractor. As results show, our hybrid model achieves 97.6% accuracy on the average dataset, outperforming all comparison methods. Specifically, the hybrid model outperformed the following models: SVM, TextCNN, RNN, MLP, Transformer, and BERT, with improvement rates of 0.2%, 1.4%, 4.2%, 1.2%, 3.3%, and 6.0%, respectively. According to previous research in resumes block classification studies, an increase of 1%-3% on the accuracy metric is considered a significant effect. Resumes-A, Resumes-B, and Resumes-C denote the three datasets we collected. Table 1 proves that the BRNN encoder proposed in this paper is very effective. Furthermore, the accuracy of BRNNF has increased on the basis of BRNN, which validates the proposed feature fusion method. Among the general baseline models, including SVM, TextCNN, RNN, MLP, Transformer, and BERT, the RNN model performs better than other, excluding BERT.

TABLE 1. The proposed methods with baseline models for classification performance on our Chinese datasets. The best accuracy results are in bold.

Compare models	Resumes-A	Resumes-B	Resumes-C	Average
SVM	91.0 ± 0.01	92.3 ± 0.03	91.4 ± 0.02	91.6 ± 0.01
TextCNN	94.0 ± 0.03	94.5 ± 0.05	94.3 ± 0.01	94.3 ± 0.01
TextCNN + BRNN (our)	95.2 ± 0.03	96.1 ± 0.02	95.8 ± 0.02	95.7± 0.03
TextCNN + BRNNF (our)	**96.4 ± 0.02**	**97.1 ± 0.01**	**96.8 ± 0.02**	**96.8± 0.03**
RNN (Bi-GRU)	96.1 ± 0.01	96.8 ± 0.02	96.3 ± 0.02	96.4 ± 0.02
RNN + BRNN (our)	**97.5 ± 0.03**	97.8 ± 0.02	97.5 ± 0.02	97.6± 0.04
RNN + BRNNF (our)	**97.5 ± 0.03**	**98.1 ± 0.02**	**98.2 ± 0.02**	**97.9± 0.03**
MLP	93.1 ± 0.01	93.9 ± 0.01	93.1 ± 0.02	93.4 ± 0.01
MLP + BRNN (our)	95.5 ± 0.03	95.2 ± 0.04	**95.7 ± 0.02**	95.4± 0.03
MLP + BRNNF (our)	**95.7 ± 0.03**	**95.6 ± 0.02**	**95.7 ± 0.02**	**95.6± 0.03**
Transformer	95.4 ± 0.01	96.8 ± 0.01	96.3 ± 0.02	96.2 ± 0.03
Transformer + BRNN (our)	**97.1 ± 0.3**	98.0 ± 0.02	97.8 ± 0.02	97.6± 0.03
Transformer + BRNNF (our)	97.0 ± 0.03	**98.1 ± 0.02**	**97.9 ± 0.02**	**97.7± 0.02**
BERT (base-Chinese)	**97.1 ± 0.02**	98.0 ± 0.04	**97.2 ± 0.03**	97.4 ± 0.02
BERT + BRNN (our)	97.0 ± 0.02	98.3 ± 0.01	96.9 ± 0.02	97.4± 0.01
BERT + BRNNF (our)	**97.1 ± 0.03**	**98.6 ± 0.02**	97.1 ± 0.02	**97.6± 0.01**
Hybrid model (our dynamic weight)	97.2 ± 0.01	97.9 ± 0.01	97.7 ± 0.02	**97.6± 0.03**

TABLE 2. The performance comparison between BERT-based models and our models.

Compare models	Params(M)	Memory(G)	GPU(G)	Inference time(ms)
BERT-Large	340	≥ 16	≥ 16	$\geq$1026
BERT-Base	110	≥ 12	≥ 12	$\geq$873
ALBERT-Base	12	≥ 1.2	≥ 2.0	≥ 101
ALBERT-Large	48	≥ 1.6	≥ 2.5	≥ 156
Dynamic Weight Hybrid Model(ours)	2	≤ 0.5	≤ 1.0	≤ 35

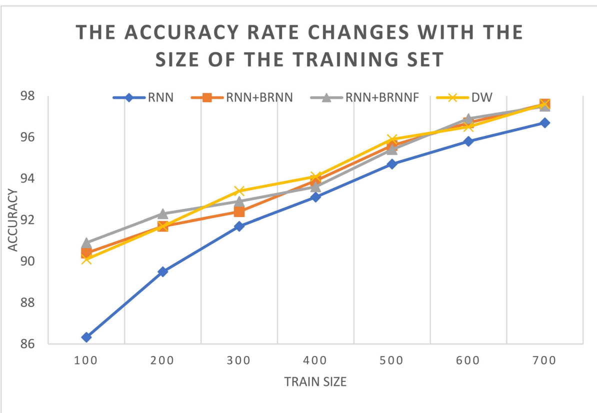


FIGURE 7. Accuracy under varying sizes of the training set, where the "DW" means "Dynamic Weight" hybrid model proposed in this paper.

However, RNN has a significantly lower computational cost and time consumption than BERT, which is the reason we use it to fuse our block sequence model. By utilizing the sequential information among different resume blocks, our proposed models can further improve the classification performance compared with the existing baseline models.

To further illustrate that our model is sufficiently lightweight compared to the BERT-based model, we list the parameter information and inference time of BERT and ALBERT [51], as well as the space and time requirements for inference, as shown in Table 2. Obviously, compared with these huge pre-trained models, our model is lightweight enough. Because our model does not need any pre-training on a vast corpus followed by fine-tuning. Our dynamic weight hybrid model has the least number of parameters and the least inference time than other four BERT-based model. Due to the reasonable combination of different modules we designed, meanwhile, our model can nearly achieve the same performance with the BERT model.

Moreover, Figure 7 shows that, under different small sizes of the training set (varies from 100 to 700 resumes with an increase step of 100), using BRNN or BRNNF encoder can achieve a better accuracy than the existing models. This suggests that BRNNF can effectively mitigate the disadvantage of requiring a large number of training samples for deep-learning-based models, making it capable of dealing with the practical scenarios where only a small training set is available.

2) ADAPTABILITY ON ABNORMAL RESUMES

The experiment's results about the adaptability on abnormal resumes are shown from Figure 8 to 12. We can see that regardless of the structure used as the block feature extractor, the accuracy does not decrease as the noise ratio increases when BRNN/BRNNF is not utilized. After adding the BRNN or BRNNF encoder, as the proportion of noise increases, the accuracy rate gradually decreases. This indicates that our BRNN or BRNNF encoder is susceptible to the influence of abnormal sequence features, leading to a bad anti-interference performance. Conversely, when the dynamic weight hybrid model is used, the accuracy rate drops the slowest, even when the noise ratio reaches 100%, it is

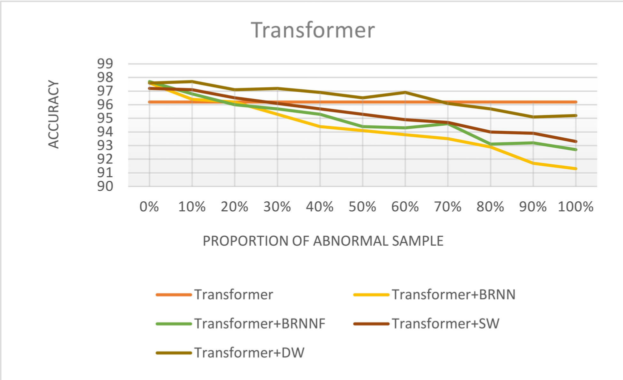


FIGURE 8. The accuracy of Transformer-based methods on different proportion of abnormal samples.

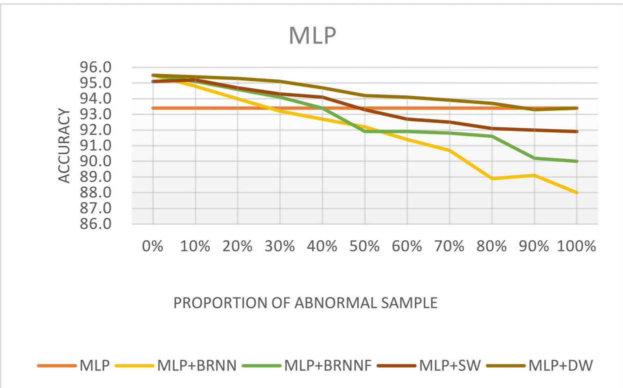


FIGURE 9. The accuracy of MLP-based methods on different proportion of abnormal samples.

only 1% lower than the baseline model. This fully verifies the effectiveness of the dynamic weight hybrid model we proposed, and significantly improves the accuracy of BRNNF under abnormal resume samples. At the same time, under the normal resume test set, i.e., where the corresponding noise ratio is 0, the model maintains near-original high accuracy levels of BRNNF. It is worth mentioning that in Figure 8 to 12, we also count the accuracy of the model under another hybrid strategy—the static weight hybrid model, called "sw". In the prediction stage, the weights of the two sub-models are no longer changed, and the results are a weighted average according to the pre-set weights w_1 and w_2. The results show that our dynamic weight strategy significantly outperforms this static weight strategy.

3) TRANSFER LEARNING ABILITY

Table 3-4 respectively show the accuracy of transfer model based on BRNN and BRNNF structures on the target domain. It can be concluded from the tables that for any combination of source domain and target domain, the transfer model proposed in this paper achieves a certain performance improvement compared with the model which only uses source domain dataset.

TABLE 3. Accuracy of DANN-BRNN model on target domain.

Source $\rightarrow$ Target	Source only(%)	DANN-BRNN(%)	Diff(%)
NoIT $\rightarrow$ ART	88.9	85.6	3.3 $\uparrow$
ART $\rightarrow$ NoIT	94.8	95.3	3.5 $\uparrow$
IT $\rightarrow$ ART	87.2	83.6	3.6 $\uparrow$
ART $\rightarrow$ IT	94.5	90.5	2.5 $\uparrow$
IT $\rightarrow$ NoIT	93.4	92.3	1.1 $\uparrow$
NoIT $\rightarrow$ IT	95.3	92.2	3.1 $\uparrow$

TABLE 4. Accuracy of DANN-BRNNF model on target domain.

Source $\rightarrow$ Target	Source only(%)	DANN-BRNNF(%)	Diff(%)
NoIT $\rightarrow$ ART	86.7	84.6	2.1 $\uparrow$
ART $\rightarrow$ NoIT	94.0	90.5	3.5 $\uparrow$
IT $\rightarrow$ ART	86.9	85.2	1.7 $\uparrow$
ART $\rightarrow$ IT	94.7	90.1	4.6 $\uparrow$
IT $\rightarrow$ NoIT	93.7	92.5	1.2 $\uparrow$
NoIT $\rightarrow$ IT	95.3	91.3	4.0 $\uparrow$

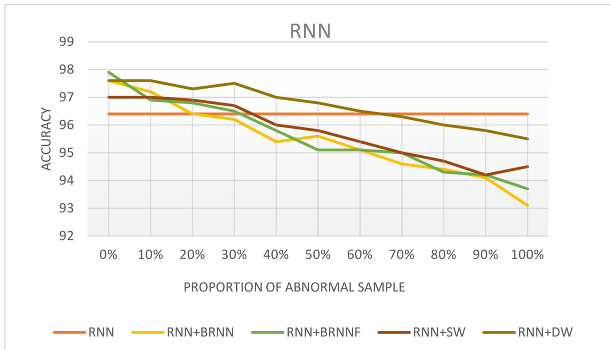


FIGURE 10. The accuracy of RNN-based methods on different proportion of abnormal samples.

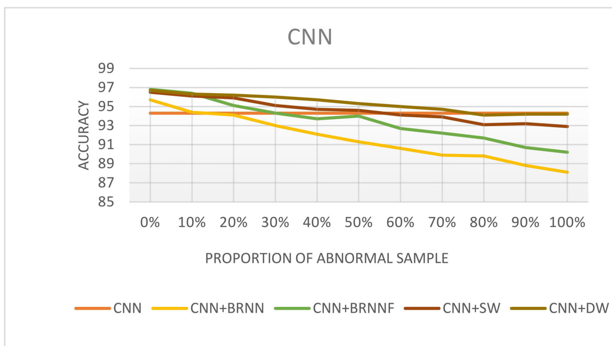


FIGURE 11. The accuracy of CNN-based methods on different proportion of abnormal samples.

4) ADDITIONAL EXPERIMENTS

LSPD structure dynamically produces a small weight for the block sequence model when facing abnormal data and a large weight when facing normal data. Therefore, in order to further validate the effectiveness of our LSPD structure, we observe the average value distribution difference of the weight θ in Figure 2 when test on abnormal and normal data. Results are shown as Figure 14. From this figure, we can see that when test on the abnormal test set (red points), the

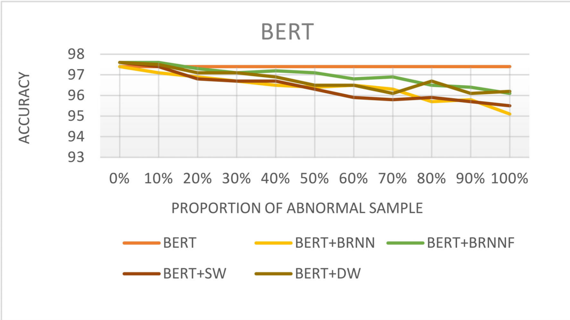


FIGURE 12. **The accuracy of BERT-based methods on different proportion of abnormal samples.**

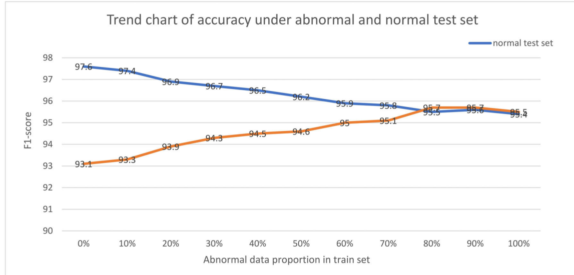


FIGURE 13. **Add 0-100% abnormal resume samples to the 700 resume size training set, and the change of accuracy on normal resume test set and abnormal resume test set (100% random shuffling).**

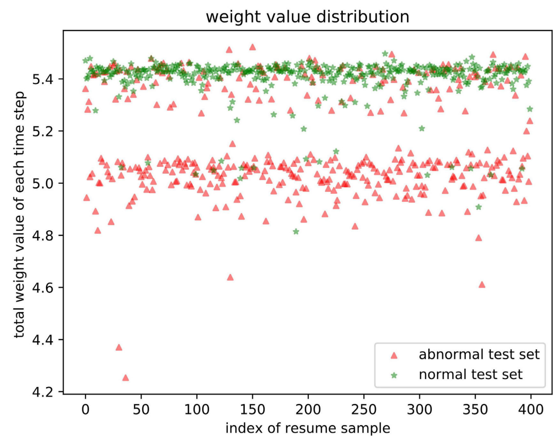


FIGURE 14. **θ distribution when test on abnormal and normal test sets.**

distribution of the θ value is obviously lower than when test on the normal test set (green points). This suggests that our hybrid model can dynamically adjust the weight of the block sequence model and the joint model according to the weight θ generated by LSPD. When the resume blocks of the input sample are arranged in a confusing order, LSPD will produce a smaller weight, whereas when the resume blocks of the input sample are arranged in a sufficient order, a larger weight will be assigned to the block sequence model.

We also design an experiment to verify whether the naive method of simply adding a certain proportion of abnormal resume samples to the training set rather than adjusting the structure of model is effective. We choose the RNN + BRNN as the structure of model. The results are shown in Figure 13. It can be seen from the figure that when we increase the proportion of disordered abnormal resume samples in the training set (700 resumes), the accuracy on the abnormal resume test set (100% random shuffling) gradually increases (from 93.1% to 95.5%), but at the same time, the accuracy on the normal resume test set gradually decreases (from 97.6% to 95.4%). In essence, this naive idea only reduces the variance of BRNN, making the prediction of BRNN closer to the mean level. After adding disordered abnormal resumes to the training set, the sequence features learned by BRNN are seriously damaged, as a result, the classification performance on normal resume samples is greatly reduced, compared with this naive method, our proposed dynamic weight hybrid model can not only maintain the high accuracy when facing normal resume set (97.6%), but also a high accuracy when facing abnormal resume set (95.5%). Therefore, it is of significance for us to improve the classification performance from the model structure aspect rather than the data aspect, which sufficiently proves the necessity and research significance of our work.

D. DISCUSSIONS

The following conclusions can be drawn from the above experimental results.

(1) In specific classification scenarios like resume block classification, the algorithm based on deep learning that considers the sequence feature among resume blocks outperforms the general text classification algorithms which only take one block as the input and ignore the sequence feature among resume blocks. In fact, the distribution of resume block categories in real datasets is unbalanced. Therefore, when the general classification algorithm trains on a resume block dataset formed by real resume blocks, it will face the problem of unbalanced categories. However, our proposed models can better fit the real block distribution.

(2) Using the BRNN or BRNNF encoder can achieve ideal classification results with a small training set, and overcome the disadvantage that deep learning-based algorithms require large-scale data training to perform well. The reason may be that our model takes into account the important feature of contextual order between resume blocks. For the specific scenario of resume block classification, the model can still achieve good performance even if the dataset used for training is small.

(3) Through the feature fusion and dynamic weight hybrid strategy, we successfully solve the problem that BRNN encoder performs poorly when facing the resume sample that differ significantly from the sequence features BRNN has learned, improving the adaptability of BRNN and reducing the model variance. Experiments show that the hybrid strategy is effective, probably because we take full advantage

of the advantages and disadvantages of each sub-model, since the output of the RNN classifier in the joint model (Figure 2) is not affected by the sequence characteristics between resume blocks impact, and BRNNF can achieve better results when facing normal resume data.

(4) Experiments show that our DA-BRNN and DA-BRNNF models improve the domain transfer ability. The DA-BRNN model has better performance than the non-transfer model using only the source domain dataset, which fully proves the transfer effectiveness of DA-BRNN model. Our models only need an unlabeled corpus in the target domain and a labeled corpus in the source domain to realize domain adaptation. Hence, the features extracted by DA-BRNN model are domain independent, thereby reducing the differences caused by domain changes and realizing the transfer learning from source domain to target domain.

V. CONCLUSION AND FUTURE WORK

In this paper, we introduce a sequence encoder named BRNN, designed to extract sequence features from blocks within the same resume. We further enhance the BRNN with a feature fusion strategy, termed BRNNF. In addition, we propose a dynamic weighted hybrid model based on ensemble learning principles, which dynamically generates weights for each participating sub-model. We also introduce a transfer model specifically tailored for the resume block classification task. To support our research, we have released three Chinese datasets comprising a total of 4,500 resumes. Experimental results demonstrate that our model achieves an impressive accuracy of 97% on normal resumes and 95.5% on abnormal resumes, setting a new state-of-the-art performance for the resume block classification task compared to other baseline models.

The research issue addressed in this paper is a step within the resume information extraction framework. There is a lack of a comprehensive and unified end-to-end framework for resume information extraction, which fails to adequately consider the intrinsic contextual order relationships between resume blocks. This oversight may lead to decreased accuracy in information extraction. In the future, we will investigate an automatic extraction system that integrates the three sub-tasks: resume segmentation, resume block classification, and intra-block information extraction. Currently, the selection of hyperparameters is based on human experience, which lacks automation and increases manual effort. Therefore, we will also explore automatic hyperparameter learning and tuning in future research. In the future, we will consider adding more datasets, including resume data in other languages, such as English. We believe that our method will perform well on other datasets or languages. In addition, our model is not only limited to the classification of resume blocks but also applicable to a more general application scenario, for example, a text classification scenario in which each object entity of the classification task has a certain sequence relationship. The techniques and strategies introduced in this paper, such as the BRNN and BRNNF encoders, the dynamic weighted hybrid model, and the transfer learning approach, can be adapted to other text classification tasks where the input data exhibits sequential dependencies. By leveraging the contextual order information and dynamic weighting schemes, our model has the potential to improve the performance of various text classification applications beyond resume block classification.

TABLE 5. A table of acronyms gives all abbreviated symbols and their meanings.

Symbols	Meanings
NB	Naive Bayes
SVM	Support Vector Machine
RF	Random Forest
BRNN	block-level bi-directional recurrent neural network encoder
BRNNF	a feature fusion block-level bi-directional recurrent neural network encoder
RNN	recurrent neural network
IBC	individual block classifier
LSPD	a label sequence probability distribution method
DANN	the domain adversarial neural network transfer model
DA-BRNN	the domain adversarial block-level bi-directional recurrent neural network
DA-BRNNF	the domain adversarial block-level bi-directional recurrent neural network with feature fusion
HMM	the Hidden Markov Model
GAN	the Generative Adversarial Network
Bi-GRU	the bidirectional gated recurrent unit
CNN	the Convolutional Neural Network
WRNN	Word-level RNN
TextCNN	Convolutional neural networks for text
MLP	Multilayer Perceptron
BERT	the Bidirectional Encoder Representation from Transformers
ALBERT	a lite BERT
IT	electronic domain
NoIT	non electronic engineering domain
ART	social sciences domain according to their respective industry domains

APPENDIX

For clarity and reference, we compile Table 5, which includes all the abbreviated symbols used throughout this paper, encompassing those found in the text, figures, and tables. This notation table clearly lists each abbreviation along with its corresponding meaning.

ACKNOWLEDGMENT

(Li-Hui Zhao and Ji Zhang are co-first authors.)

REFERENCES

[1] B. Gaur, G. S. Saluja, H. B. Sivakumar, and S. Singh, "Semi-supervised deep learning based named entity recognition model to parse education section of resumes," *Neural Comput. Appl.*, vol. 33, no. 11, pp. 5705–5718, Jun. 2021.

[2] M. Alamelu, D. S. Kumar, R. Sanjana, J. S. Sree, A. S. Devi, and D. Kavitha, "Resume validation and filtration using natural language processing," in *Proc. 10th Int. Conf. Internet Everything, Microw. Eng., Commun. Netw. (IEMECON)*, Dec. 2021, pp. 1–5.

[3] S. Luo and J. Yu, "ESGNet: A multimodal network model incorporating entity semantic graphs for information extraction from Chinese resumes," *Inf. Process. Manage.*, vol. 61, no. 1, Jan. 2024, Art. no. 103524. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0306457323002613

[4] A. Heakl, Y. Mohamed, N. Mohamed, A. Elsharkawy, and A. Zaky, "ResuméAtlas: Revisiting resume classification with large-scale datasets and large language models," *Pro. Comput. Sci.*, vol. 244, pp. 158–165, Jan. 2024. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1877050924029909

[5] K. Yu, G. Guan, and M. Zhou, "Resume information extraction with cascaded hybrid model," in *Proc. 43rd Annu. Meeting Assoc. Comput. Linguistics (ACL)*, 2005, pp. 499–506.

[6] G. R. Y. Naik, B. Shashidhara, and G. K. Amith, "A review on text extraction techniques for degraded historical document images," in *Proc. 2nd Int. Conf. Adv. Inf. Technol. (ICAIT)*, vol. 1, Jul. 2024, pp. 1–8.

[7] T. Pham, X. Tao, J. Zhanag, J. Yong, W. Zhang, and Y. Cai, "Mining heterogeneous information graph for health status classification," in *Proc. 5th Int. Conf. Behav., Econ., Socio-Cultural Comput. (BESC)*, Nov. 2018, pp. 73–78.

[8] T. Pham, X. Tao, J. Zhang, and J. Yong, "Constructing a knowledge-based heterogeneous information graph for medical health status classification," *Health Inf. Sci. Syst.*, vol. 8, no. 1, p. 10, Dec. 2020.

[9] K. Gawhankar, A. Deorukhkar, A. Miniyar, H. Kapure, and B. Ivin, "NLP-driven ML for resume information extraction," in *Proc. IEEE 9th Int. Conf. Converg. Technol. (I2CT)*, Apr. 2024, pp. 1–6.

[10] A. Espinal, Y. Haralambous, D. Bedart, and J. Puentes, "A format-sensitive BERT-based approach to resume segmentation," in *Proc. 33rd Conf. Open Innov. Assoc. (FRUCT)*, May 2023, pp. 30–37.

[11] K. Yao, J. Zhang, C. Qin, X. Song, P. Wang, H. Zhu, and H. Xiong, "ResuFormer: Semantic structure understanding for resumes via multi-modal pre-training," in *Proc. IEEE 39th Int. Conf. Data Eng. (ICDE)*, Apr. 2023, pp. 3154–3167.

[12] V. V. S. Tallapragada, V. S. Raj, U. Deepak, P. D. Sai, and T. Mallikarjuna, "Improved resume parsing based on contextual meaning extraction using BERT," in *Proc. 7th Int. Conf. Intell. Comput. Control Syst. (ICICCS)*, May 2023, pp. 1702–1708.

[13] Z. Jiang, C. Zhang, B. Xiao, and Z. Lin, "Research and implementation of intelligent Chinese resume parsing," in *Proc. WRI Int. Conf. Commun. Mobile Comput.*, vol. 3, Jan. 2009, pp. 588–593.

[14] J. Chen, C. Zhang, and Z. Niu, "A two-step resume information extraction algorithm," *Math. Problems Eng.*, vol. 2018, pp. 1–8, May 2018.

[15] Q. Li, H. Peng, J. Li, C. Xia, R. Yang, L. Sun, P. S. Yu, and L. He, "A survey on text classification: From shallow to deep learning," 2020, *arXiv:2008.00364*.

[16] A. Barducci, S. Iannaccone, V. La Gatta, V. Moscato, G. Sperlì, and S. Zavota, "An end-to-end framework for information extraction from Italian resumes," *Expert Syst. Appl.*, vol. 210, Dec. 2022, Art. no. 118487.

[17] P. Swami and V. Pratap, "Resume classifier and summarizer," in *Proc. Int. Conf. Mach. Learn., Big Data, Cloud Parallel Comput. (COM-IT-CON)*, vol. 1, May 2022, pp. 220–224.

[18] H. Sajid, J. Kanwal, S. U. R. Bhatti, S. A. Qureshi, A. Basharat, S. Hussain, and K. U. Khan, "Resume parsing framework for E-recruitment," in *Proc. 16th Int. Conf. Ubiquitous Inf. Manage. Commun. (IMCOM)*, Jan. 2022, pp. 1–8.

[19] S. Chaib, H. Liu, Y. Gu, and H. Yao, "Deep feature fusion for VHR remote sensing scene classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 8, pp. 4775–4784, Aug. 2017.

[20] B. Zoph, D. Yuret, J. May, and K. Knight, "Transfer learning for low-resource neural machine translation," 2016, *arXiv:1604.02201*.

[21] D. Freitag and A. McCallum, "Information extraction with HMMs and shrinkage," in *Proc. AAAI- workshop Mach. Learn. Inf. Extraction*, Orlando, Florida, 1999, pp. 31–36.

[22] N. Kushmerick, E. Johnston, and S. McGuinness, "Information extraction by text classification," in *Proc. Workshop Adapt. Text Extraction Mining*, 2001, pp. 1–7.

[23] N. Roopesh and C. N. Babu, "Robotic process automation for resume processing system," in *Proc. Int. Conf. Recent Trends Electron., Inf., Commun. Technol. (RTEICT)*, Aug. 2021, pp. 180–184.

[24] K. Tejaswini, V. Umadevi, S. M. Kadiwal, and S. Revanna, "Design and development of machine learning based resume ranking system," *Global Transitions Proc.*, vol. 3, no. 2, pp. 371–375, Nov. 2022.

[25] E. Karamatli and S. Akyokus, "Resume information extraction with named entity clustering based on relationships," in *Proc. Int. Symp. Innov. Intell. Syst. Appl.*, 2010.

[26] J. Chen, L. Gao, and Z. Tang, "Information extraction from resume documents in PDF format," *Electron. Imag.*, vol. 28, no. 17, pp. 1–8, Feb. 2016.

[27] Y. Wentan and Q. Yupeng, "Chinese resume information extraction based on semi-structured text," in *Proc. 36th Chin. Control Conf. (CCC)*, Jul. 2017, pp. 11177–11182.

[28] B. C. L. Tobing, I. R. Suhendra, and C. Halim, "Catapa resume parser: End to end Indonesian resume extraction," in *Proc. 3rd Int. Conf. Natural Lang. Process. Inf. Retr.*, Jun. 2019, pp. 68–74.

[29] D. S. Chavare and A. B. Patil, "Resume parsing using natural language processing," *Grenze Int. J. Eng. Technol.*, vol. 9, no. 1, pp. 1–6, 2023.

[30] M. Sharma, G. Choudhary, and S. Susan, "Resume classification using elite bag-of-words approach," in *Proc. 5th Int. Conf. Smart Syst. Inventive Technol. (ICSSIT)*, Jan. 2023, pp. 1409–1413.

[31] C. Gan and T. Mori, "A few-shot approach to resume information extraction via prompts," in *Proc. Int. Conf. Appl. Natural Lang. Inf. Syst.* Cham, Switzerland: Springer, 2023, pp. 445–455.

[32] A. Liaw and M. Wiener, "Classification and regression by randomForest," *R News*, vol. 2, no. 3, pp. 18–22, 2002.

[33] N. Chakrabarty, T. Kundu, S. Dandapat, A. Sarkar, and D. K. Kole, "Flight arrival delay prediction using gradient boosting classifier," in *Advances in Intelligent Systems and Computing*. Cham, Switzerland: Springer, 2019, pp. 651–659.

[34] P.-L. Chen et al., "A linear ensemble of individual and blended models for music rating prediction," in *Proc. KDD Cup*, 2012, pp. 21–60.

[35] W. Sun and B. Trevor, "A stacking ensemble learning framework for annual river ice breakup dates," *J. Hydrol.*, vol. 561, pp. 636–650, Jun. 2018.

[36] H. Zhao, B. Wu, H. Wang, and C. Shi, "Sentiment analysis based on transfer learning for Chinese ancient literature," in *Proc. Int. Conf. Behav., Econ., Socio-Cultural Comput. (BESC)*, Oct. 2014, pp. 1–7.

[37] M. Artetxe, G. Labaka, and E. Agirre, "Learning bilingual word embeddings with (almost) no bilingual data," in *Proc. 55th Annu. Meeting Assoc. Comput. Linguistics*, 2017, pp. 451–462.

[38] S. Ben-David, J. Blitzer, K. Crammer, A. Kulesza, F. Pereira, and J. W. Vaughan, "A theory of learning from different domains," *Mach. Learn.*, vol. 79, nos. 1–2, pp. 151–175, May 2010.

[39] S. J. Pan, I. W. Tsang, J. T. Kwok, and Q. Yang, "Domain adaptation via transfer component analysis," *IEEE Trans. Neural Netw.*, vol. 22, no. 2, pp. 199–210, Feb. 2011.

[40] A. Bordes, N. Usunier, A. Garcia-Duran, J. Weston, and O. Yakhnenko, "Translating embeddings for modeling multi-relational data," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 26, 2013, pp. 1–9.

[41] Y. Ganin, E. Ustinova, H. Ajakan, P. Germain, H. Larochelle, F. Laviolette, M. Marchand, and V. Lempitsky, "Domain-adversarial training of neural networks," *J. Mach. Learn. Res.*, vol. 17, no. 1, pp. 2030–2096, May 2016.

[42] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 27, 2014, pp. 1–9.

[43] Y. Goldberg and O. Levy, "word2vec explained: Deriving Mikolov et al.'s negative-sampling word-embedding method," 2014, *arXiv:1402.3722*.

[44] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," 2014, *arXiv:1412.3555*.

[45] R. Müller, S. Kornblith, and G. E. Hinton, "When does label smoothing help?" in *Proc. Adv. Neural Inf. Process. Syst.*, 2019, pp. 4694–4703.

[46] Y. Kim, "Convolutional neural networks for sentence classification," 2014, *arXiv:1408.5882*.

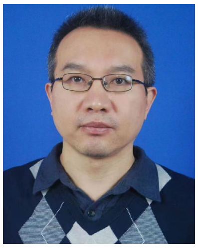
[47] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," *Nature*, vol. 323, no. 6088, pp. 533–536, Oct. 1986.

[48] H. Taud and J. F. Mas, "Multilayer perceptron (MLP)," in *Lecture Notes in Geoinformation and Cartography*. Cham, Switzerland: Springer, 2018, pp. 451–455.

[49] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017, pp. 5998–6008.

[50] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," 2018, *arXiv:1810.04805*.

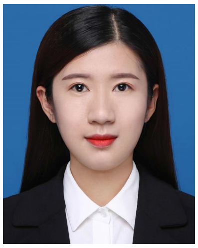
[51] Z. Lan, M. Chen, S. Goodman, K. Gimpel, P. Sharma, and R. Soricut, "ALBERT: A lite BERT for self-supervised learning of language representations," 2019, *arXiv:1909.11942*.



LI-HUI ZHAO is currently an Associate Professor of computer science with the North University of China. His research interests include data science, data mining, intelligent technologies.

JI ZHANG (Senior Member, IEEE) is currently a Full Professor of computer science. He has published over 300 papers in top peer-reviewed international journals and conferences. His research interests include intelligent computing, big data analytics, and data privacy.

QIQIANG XU received the master's degree from Nanjing University of Aeronautics and Astronautics. He is currently a Research and Development Engineer with ByteDance, Shenzhen, China. His research interests include natural language processing and text data mining.



LINLIN HOU received the bachelor's degree from Jilin University, China, and the Ph.D. degree in applied mathematics from Nankai University, China. She is currently a Postdoctoral Fellow with Zhejiang Laboratory. Her research interests include natural language processing and data mining.

XIN WANG (Member, IEEE) is currently a Full Professor with the School of Computer Science and Software Engineering, Southwest Petroleum University. He has filed over 50 domestic and international invention patents. His research interests include knowledge discovery, data mining, and artificial intelligence. He has published over 70 papers in top international conferences and journals, authored two monographs, and received runner-up awards for the Best Paper at ICDE and WISE as well as the Outstanding Paper Award at DEXA.