Review Article

A Comparative Analysis of Fraudulent Recruitment Advertisement Detection Methods in the IoT Environment

Ruiqi Wang [ID],[1] **Ning Cao** [ID],[1] **Yajie Guo** [ID],[1] **Shujuan Ji** [ID],[1] **and Sachin Kumar** [ID][2]

[1]*Key Laboratory for Wisdom Mine Information Technology of Shandong Province, Shandong University of Science and Technology, Qingdao, China*
[2]*Department of Computer Science and Engineering, Ajay Kumar Garg Engineering College, Ghaziabad, India*

Correspondence should be addressed to Shujuan Ji; jane_ji2003@aliyun.com

The growth of the Internet of Things has changed the way of job hunting. Online recruitment has gradually replaced the traditional offline recruitment mode. Some unscrupulous people use online recruitment platforms to publish fraudulent recruitment advertisements, which not only bring financial and reputational losses to job seekers but also harm the sustainable development of society. However, previous studies have not used unified evaluation metrics and datasets, and detecting fraudulent recruitment advertisements lacks systematic research. To resolve this problem, this paper selects four representative traditional learning methods (i.e., random forest, support vector machine (SVM), logistic regression, and Naïve Bayes) and three deep learning methods (i.e., TextCNN, gate recurrent unit (GRU), and bidirectional long-short-term memory (Bi-LSTM)), which perform good in natural language processing (NLP) and use the same evaluation metrics and datasets conducting comparative experiments on balanced and unbalanced datasets, respectively. The experimental results show that the TextCNN method achieves the best detection performance with relatively low energy consumption on the balanced dataset. All the metrics values are more significant than 0.93. On unbalanced datasets, the TextCNN method still performs best with increasing imbalanced proportion.

1. Introduction

With the development of the Internet of Things (IoT), people can quickly get information from electronic devices. At the same time, the primary recruitment method in the labor market has rapidly shifted from offline to online, and getting recruitment information from the Internet has become a mainstream way. IoT has changed the inefficient way of finding a job. Online recruitment has the advantages of effectiveness, easiness, and efficiency [1]. However, some unscrupulous people use the network platforms' weakness to post fraudulent recruitment advertisements on the Internet to deceive money and exploit labor in the name of recruitment. Some fake recruitment has evolved from fraud to violent robbery, threats, restriction of personal freedom, and other serious violations [2].

Fraudulent recruitment advertisements have become a nationwide social nuisance. According to the *2017 China Internet Users' Consumer Protection Rights Report* (https://wenku.baidu.com/view/f50682067ed5360cba1aa8114431b90d6d85894e.html), among all the fraud cases reported by protection rights, fraudulent part-time work was the most frequently reported type of fraud (accounting for 22.1%), and most of the fraudulent recruitment occurred in well-known recruitment platforms. Data from *China's Justice Big Data Service Platform* (http://data.court.gov.cn/pages/research.html) shows that fraudulent recruitment cases increased yearly. Data from *Ai Media Consulting* through the *2019 China Internet Recruitment Industry Market Research* (https://www.iimedia.cn/c400/63879.html) shows that among the various bad experiences on the online platforms, the most situation that job seekers minded was that the

enterprise information was not true (accounting for 34.8%). The second was personal information leakage (accounting for 31.8%). It is thus clear that the detection of fraudulent recruitment advertisements is a critical problem to be solved urgently. Detecting fraudulent recruitment advertisements based on data generated by the IoT system not only helps safeguard the rights and interests of job seekers but also promotes economic growth and creates a green IoT environment. Figure 1 shows the detection process for fraudulent recruitment advertisements in an IoT environment. However, this area is still a relatively untapped field, not receiving much attention from the academic community. In addition, detecting fraudulent recruitment advertisements among legitimate ones is a technically challenging problem [3]. Most research on fraudulent recruitment advertisements is carried out from the theoretical aspect. For example, Rubin [4] analyzed the causes and countermeasures of commercial fraud through advertising from the perspective of information economics, stating that deception was manipulating information to gain advantages. From a legal perspective, Jiang [5] puts forward the slogan "*Taking the law as a guarantee, strengthening advertising supervision functions, increasing rectification efforts, and cracking down on false and fraudulent advertisements.*"

The technology for detecting fraudulent recruitment advertisements is limited and immature. According to the methods adopted, the limited existing studies on the detection of fraudulent recruitment advertisements generally can be divided into three parts: traditional learning-based detection methods, traditional learning + feature extraction-based detection methods, and deep learning-based detection methods. Traditional learning-based detection methods mainly use traditional learning algorithms to detect fake job advertisements. Traditional learning + feature extraction-based detection methods mainly use feature extraction methods to improve the performance of traditional learning algorithms. Deep learning-based detection methods use various deep learning algorithms to detect employment frauds without feature engineering. The similarity of these detection methods is to identify the implicit fraud patterns in data. In particular, existing research involved different detection methods, different feature extraction methods, different evaluation metrics, and different datasets. Therefore, it is necessary to study the detection of fake recruitment advertisements systematically. To conduct a systematic study on detecting fake recruitment advertisements, this paper selects seven algorithms with good performance in NLP. It conducts systematic and comprehensive experiments using the same evaluation metrics and datasets. The main contributions are summarized as follows.

(1) The existing detecting methods of fraudulent recruitment advertisements are described and analyzed in detail

(2) A comparative analysis of the existing work is carried out experimentally using the same dataset and evaluation metrics. Seven algorithms are used to conduct comparative experiments on the public employment fraud detection dataset, and the experimental results are analyzed in detail

(3) The experimental results show that the TextCNN of the deep learning methods outperforms all other compared methods in the accuracy, precision, recall, and $F1$-score. In terms of time performance, though the training time of TextCNN is much higher than the traditional learning methods and the traditional learning + feature extraction methods, its testing time is similar to the ones of SVM and SVM+TF-IDF, which is acceptable for IoT devices. Therefore, comprehensively considering the accuracy, precision, recall, $F1$-score, and testing time, the TextCNN method performs best among all these compared methods

(4) This paper has a specific reference value for further research of higher performance and lower energy consumption detection methods to achieve the goal of green IoT

The organization of this paper is as follows. Section 2 reviews the relevant research on fraudulent recruitment detection. Section 3 introduces the representative methods. Section 4 presents the setting of experiments and the analysis of experimental results. Section 5 is the summary of the work of this paper and the prospect for the future.

2. Related Work

This section reviews the related work from three categories (i.e., the traditional learning-based detection methods, the traditional learning + feature extraction-based detection methods, and the deep learning-based detection methods) that we classify in Section 1 in detail.

2.1. Traditional Learning-Based Detection Methods. Some researchers used traditional learning-based methods to learn some rules. For example, Vidros et al. [6] analyzed employment fraud for the first time. They explained the work process of online recruitment and the role of the applicant tracking system in this process in detail. A set of rules of thumb was summarized from analyzing feasible data in the real world. Meanwhile, spam [7], insurance fraud, and phishing are highly similar to recruitment fraud, and they also discussed these areas in detail.

Besides, a more comprehensive and extensive study [8] was conducted based on the previous work [6]. They contributed and evaluated a real dataset of 17880 recruitment advertisements—Employment Scam Aegean Dataset (EMS-CAD) (https://www.kaggle.com/datasets/shivamb/real-or-fake-fake-jobposting-prediction). Based on a subset of this dataset, they conducted word bag modeling and trained six WEKA (http://www.cs.waikato.ac.nz/ml/weka) classifiers (logistic regression, J48 decision trees, Naïve Bayes, random forest, etc.). As a result, an extension of the empirical ruleset was derived.
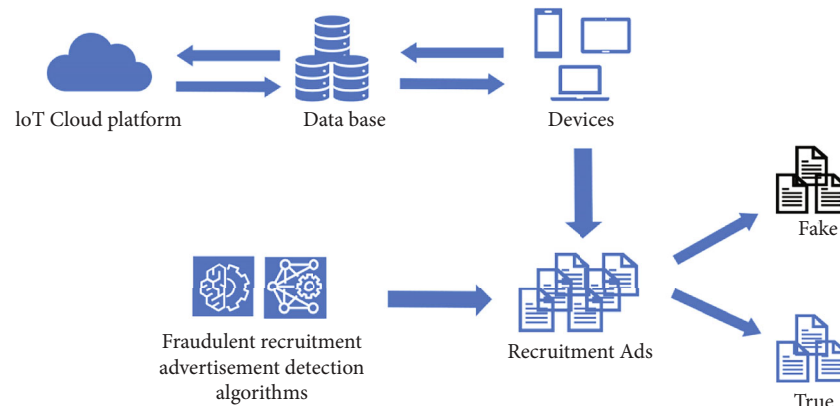
FIGURE 1: Detection process for fake job ads in an IoT environment.

Based on supervised learning methods, Dutta and Bandyopadhyay [9] proposed an automatic classification tool. They used single and ensemble classifiers to detect fraudulent recruitment advertisements, respectively. The single classifiers applied Naïve Bayes, multi-layer perceptron (MLP), K-nearest neighbor (KNN), and decision tree. And in the ensemble classifiers, random forest, and AdaBoost were applied. In addition, they compared the performance of these classifiers on the original highly unbalanced dataset.

Recruitment advertisements should be drawn from various sources to collect data in an all-around way. To solve this problem, Nindyati and Nugraha [10] extracted the Indonesian Employment Scam Dataset (IESD) from Indonesian recruitment data and manually labeled it based on empirical analysis and public reports of employment scam complaints. They considered the platforms where fraudulent recruitment advertisements were posted. In addition, they added behavioral features to previous studies [8] and used behavioral activities as contextual features for fraud detection. Naïve Bayes, logistic regression, KNN, decision tree, and SVM were applied as classifiers. The result indicated that adding behavioral characteristics can improve the detection effect of fake recruitment advertisements.

2.2. Traditional Learning + Feature Extraction-Based Detection Methods. According to the classification method proposed by Vidros et al. [8], the features were divided into three categories in the feature extraction stage, i.e., language-based, context-based, and metadata-based features [3]. They selected J48, logistic regression, and random forest methods as three baselines. Then, they combined voting techniques, including average vote, majority vote, and maximum vote, with these three baselines when constructing detection models. Moreover, they evaluated the performance of the models on unbalanced datasets, which made the experiment more comprehensive.

The lack of sufficient background information on recruitment websites makes the detection of fraudulent recruitment advertisements even more challenging. To address this problem, Mahbub and Pardede [11] focused on a novel feature space design that further extracted information about recruitment companies. They extracted contextual features manually and considered not only textual

and structural features but also contextual features. The experiments result on the Naïve Bayes, J48, and JRip classifiers suggested that adding contextual features improved the detection performance and further enriched the ruleset.

Alghamdi and Alharby [12] used ensemble method based on random forest classifier to detect fraudulent recruitment advertisements. The SVM algorithm extracted the main features, including company profile, company logo, and required experience. Moreover, the detection performance was improved based on the reliable model obtained in the preprocessing and feature selection stages.

Mehboob and Malik [13] focused on the influential features of the EMSCAD to detect fake recruitment advertisements. They used information gain to select significant features. Their experimental results showed that the company profile, salary range, organization type, and required education were the most influential. Therefore, they considered combining well-performing features and adding valuable features to help improve the model's performance. In addition, they established a robust detection model using gradient boosting techniques.

2.3. Deep Learning-Based Methods. Fraudsters may know the ruleset in advance, which makes detecting fraudulent recruitment advertisements increasingly tricky. Kim et al. [14] believed there was an internal correlation between frauds, so they proposed a deep neural network algorithm based on hierarchical clustering to detect implicit fraud in data. They took the anomaly characteristics as the initial weights of the deep neural network and trained them with an autoencoder. This method discarded the feature information, took the global and local data structure as the entry point, and used clustering and deep neural networks to reveal the internal relationships between frauds.

2.4. Summary. Existing studies did an excellent job of detecting fraudulent recruitment advertisements. However, using the ruleset to detect fake job advertisements has poor expansion and is challenging to apply to new datasets. At the same time, existing studies used different detection methods, different evaluation metrics, and different datasets. They lacked comparative analysis, which makes the detection of fake job advertisements lack systematic research.

To solve this problem, this paper selects four traditional algorithms, random forest, logistic regression, SVM, and Naïve Bayes, which are frequently used in the above literature and are proven to be good on various datasets under multifarious evaluation metrics. Moreover, three popular deep learning algorithms, including GRU, Bi-LSTM, and TextCNN, are adopted to detect fake recruitment advertisements, all performing well in the field of NLP. For machine learning methods, the Bag of Words (BoW) algorithm and the Term Frequency-Inverse Document Frequency (TF-IDF) algorithm [15] are adopted to realize feature extraction. At the same time, in deep learning methods, we also try to use pretraining models, including Word to Vector (Word2-Vec) [16] and Global Vectors (GloVe) [17] for word embedding. Moreover, we use the same evaluation metrics and datasets to carry out systematic and comprehensive comparative experiments.

3. Comparison and Analysis of Fraudulent Recruitment Advertisement Detection Methods

Since this paper is aimed at comparing and analyzing the detection methods used in the existing work experimentally, this section will analyze the four typical traditional learning methods frequently adopted in recruitment advertising detection literature [3, 8–10, 12, 13] and the three currently popular deep learning methods. In detail, the four traditional learning methods include random forest, logistic regression, SVM, and Naïve Bayes. The three deep learning methods include GRU, Bi-LSTM, and TextCNN.

3.1. Random Forest Method. As a classical ensemble learning method, random forest was first proposed by Breiman [18], who combined the Bagging ensemble learning theory [19] with the random subspace method [20]. Random forest is a classifier based on the decision tree, which can solve the performance bottleneck of the decision tree by outputting the result by voting. In addition, random forest has better tolerance to noise and outliers, has stronger scalability for high-dimensional data classification, and has a stronger generalization ability of the model. Moreover, random forest is a data-driven nonparametric classification method that only needs to learn classification rules from a given sample without prior knowledge.

3.2. SVM Method. Based on statistical learning theory [21], SVM is a data mining method that can successfully handle regression and pattern recognition problems. The target of SVM is to search for the optimal hyperplane in space to satisfy the classification requirements. SVM is one of the most commonly used and most effective classifiers, and it has good generalization performance based on the principle of structural risk minimization. Moreover, SVM has a solid theoretical basis and specific mathematical model and has been widely concerned by researchers since it was proposed.

3.3. Naïve Bayes Method. Based on the probability model, Naïve Bayes was proposed by Maron and Kuhns [22]. The "naive" of Naïve Bayes refers to the two primary hypotheses: conditional independence and positional independence. In detail, the conditional independence hypothesis assumes that the property values are independent of each other, namely, there is no dependency between terms. The positional independence hypothesis means that the term's location in the document does not affect the probability calculation.

3.4. Logistic Regression Method. The mechanism of logistic regression [23] uses a group of data to fit a logistic regression model and form multiple regression relationships to predict the occurrence probability of an event in any area. The advantage of logistic regression lies in that the independent variables in statistical analysis can be continuous or discrete, which does not need to meet the normal distribution.

Logistic regression is good at solving binary classification problems, and detecting fraudulent recruitment advertisements is a common textual binary classification problem. The logistic regression classifier is simple and easy to understand, and the model is highly interpretable. It does not need to assume data distribution in advance and directly models the possibility of classification, avoiding the problem of inaccurate hypothesis distribution. In addition, only the eigenvalues of each dimension are stored, and the memory resource consumption is small.

3.5. TextCNN Method. Convolutional neural network (CNN) [24] is a feedforward neural network that recognizes two-dimensional images with amplification, shrinkage, and displacement invariance. In recent years, CNN has been mostly used for image processing or classification recognition. Kim [25] first adopted CNN for text classification and proposed the TextCNN model. Figure 2 is the structure of TextCNN. TextCNN and CNN are very similar in design. The difference is that CNN uses convolution kernels of the same width and height when processing images. Still, TextCNN's convolution kernel width is consistent with the word vector dimension. When CNN processes images, it carries out a two-dimensional convolution operation at the convolution layer, while TextCNN carries out a one-dimensional convolution operation when it processes text.

3.6. LSTM Method. Long-short-term memory (LSTM) [26] is a special kind of recurrent neural network (RNN) [27]. Its network structure is the same as general RNN. The difference is that the memory module replaces the summation unit in the hidden layer. Figure 3 is the structure of the LSTM model. The information of cell state can be enhanced or weakened by the design of the "gate" of LSTM, so that long-term dependent information can be learned, effectively overcoming the defect of traditional RNN.

GRU [28] and Bi-LSTM [29] are the two most classical variants of LSTM. GRU was proposed to solve long-term memory problems and gradient in backpropagation. And Bi-LSTM overcomes the shortcoming that the LSTM model can only get one-way information from front to back but cannot from back to front. The forward and backward LSTM networks obtain the context information, and the model's performance is effectively improved.
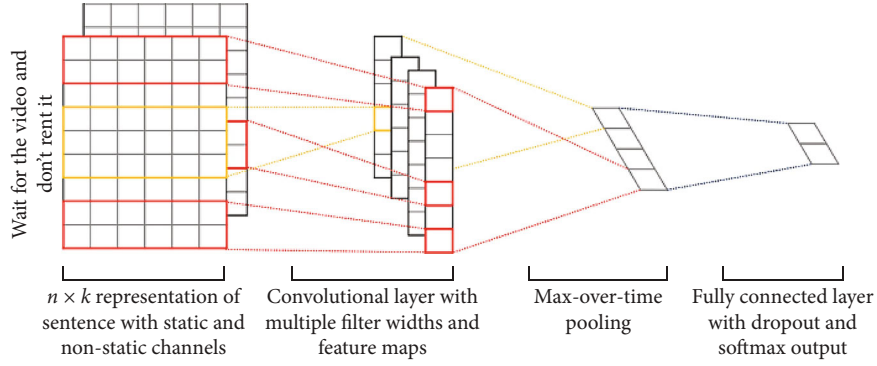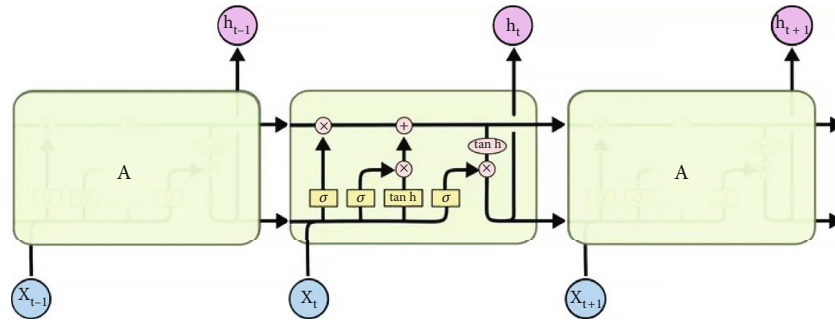
FIGURE 2: Structure of TextCNN model.

FIGURE 3: Structure of LSTM model.

TABLE 1: Dataset description.

Attribute number	Attribute name	Attribute type
1	Title	String
2	Location	String
3	Department	String
4	Salary_range	String
5	Company_profile	HTML fragment
6	Description	HTML fragment
7	Requirements	HTML fragment
8	Benefits	HTML fragment
9	Telecommuting	Binary
10	Has_company_logo	Binary
11	Has_questions	Binary
12	Employment_type	Nominal
13	Required_experience	Nominal
14	Required_education	Nominal
15	Industry	Nominal
16	Function	Nominal

4. Experiments and Results

In this paper, we have designed two sets of experiments. The first set of experiments is to verify the performance of each algorithm on the balanced dataset. The second set of experiments is conducted to verify the influence of the dataset imbalance ratio on the experimental performances. The following subsections illustrate the data and experimental setting details, the evaluation metrics, and the experimental results.

4.1. Data and Experimental Settings. In the experimental data section, the EMSCAD is used in this paper. The dataset contains 17880 real-life recruitment advertisements from 2012 to 2014, classified as legitimate and fraudulent, with 17014 legitimate and 866 fraudulent. The dataset description is shown in Table 1. We construct five datasets based on the original dataset using the downsampling method, including one balanced dataset and four unbalanced datasets. Table 2 is the detailed information on the datasets, and the last column is the datasets for each set of experiments we used. As many values are missed, we select company profile, description, requirements, and benefits, primarily selected in existing literature as text features. Before starting the experiment, we "clean" the experimental data, including removing punctuations, stopping words, and processing missing values.

In experiments, seven algorithms are selected for comparison, that is, the random forest method (abbr. RF), the logistic regression method (abbr. LR), the SVM method, and the Naïve Bayes method (abbr. NB), which are the four typical traditional learning algorithms adopted in recruitment advertisement detection literature [3, 8–12] that are described in the related work section. The TextCNN method, the GRU method, and the Bi-LSTM method are three promising deep learning algorithms. For feature

TABLE 2: Datasets statistics established based on the EMSCAD.

Datasets	True	Fraudulent	Fraudulent%	Total	Used experiments
Balanced dataset	866	866	50%	1732	1st
Unbalanced dataset-1	1300	866	40%	2166	2nd
Unbalanced dataset-2	2598	866	25%	2364	2nd
Unbalanced dataset-3	4908	866	15%	5774	2nd
Unbalanced dataset-4	17017	866	4.8%	17880	2nd

TABLE 3: Confusion matrix for binary classification.

Confusion matrix		Predict	
		Positive	Negative
True	Positive	TP	FN
	Negative	FP	TN

extraction, the BoW algorithm is used in the traditional learning methods, and the classical TF-IDF algorithm is adopted in traditional learning + feature extraction methods. Besides, deep learning methods use Word2Vec and GloVe for word embedding. All the experiments are validated by five-fold cross-validation.

4.2. Evaluation Metrics. Accuracy, precision, recall, and $F1$-score are adopted as evaluation metrics. A confusion matrix is introduced first to introduce the above four evaluation metrics. Table 3 is a binary confusion matrix. In this table, TP represents the quantity of truly positive samples and classified as positive, FP represents the quantity of actually negative samples but classified as positive, FN represents the quantity of actually positive samples but classified as negative, and TN represents the quantity of actually negative samples and classified as negative.

The details of accuracy, precision, recall, and $F1$-score are shown as follows.

(1) *Accuracy* is the proportion of the number of correctly classified samples to the total samples

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{TN} + \text{FN}} \tag{1}$$

(2) *Precision* is the proportion of truly positive samples in a group of predicted positive samples

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{2}$$

(3) *Recall* is the percentage of all truly positive samples that are successfully predicted

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{3}$$

(4) *F1 -score* refers to the weighted summed average of precision and recall

$$F1\text{-score} = 2 * \frac{\text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}} \tag{4}$$

4.3. Experimental Results and Analysis

4.3.1. Results and Analysis of the First Set of Experiments. Table 4 lists the results of experiments on the balanced dataset. Table 4 shows that all the traditional learning methods achieve good results (A, P, R, and F are all greater than 0.88), and random forest performs best on the balanced dataset. After using the TF-IDF algorithm for feature extraction, the results of all the traditional learning + feature extraction methods are improved (A, P, R, and F are all greater than 0.9), except for the Naive Bayes. In particular, the SVM+TF-IDF methods achieve the best performance. TF-IDF algorithm measures the importance of a word in terms of frequency. These results show that using TF-IDF for feature extraction plays a significant part in enhancing the effectiveness of methods. In the deep learning methods, TextCNN performs best (A, P, R, and F are all greater than 0.93). The reason is that TextCNN uses three different sizes of convolution kernels for word embedding. Compared with random forest and SVM +TF-IDF, the performance of TextCNN improves (A: 3.1%, P: 3.2%, R: 3%, F: 3%) and (A: 0.2%, P: 0.1%, R: 0.3%, F: 0.2%), respectively. After using Word2Vec and GloVe for word embedding, the performance of GRU, Bi-LSTM, and TextCNN are worse than before. In my opinion, these two pretraining models are trained on a specific dataset, which is different from the writing style of the EMSCAD dataset, so the experimental results are not ideal.

4.3.2. Results and Analysis of the Second Set of Experiments. Table 5 lists the results of experiments on the four unbalanced datasets. From Table 5, when the datasets are slightly unbalanced, for example, on the unbalanced dataset-1 and unbalanced dataset-2, the results are roughly the same as those on the balanced dataset. In detail, on the unbalanced dataset-1 and unbalanced dataset-2, we can see that random forest performs best in traditional learning methods, while SVM+TF-IDF performs best in traditional learning + feature extraction methods. Similarly, TextCNN in the deep learning category

TABLE 4: Experimental results on the balanced dataset.

Category	Model	Balanced dataset			
		A	P	R	F
Traditional learning	RF	0.899	0.899	0.900	0.900
	SVM	0.884	0.885	0.885	0.884
	LR	0.897	0.898	0.899	0.897
	NB	0.893	0.894	0.894	0.893
Traditional learning + feature extraction	RF+TF-IDF	0.923	0.923	0.923	0.923
	SVM+TF-IDF	0.928	0.930	0.927	0.928
	LR+TF-IDF	0.908	0.908	0.909	0.908
	NB+TF-IDF	0.876	0.879	0.879	0.876
Deep learning	GRU	0.835	0.836	0.835	0.834
	Bi-LSTM	0.884	0.886	0.884	0.884
	TextCNN	**0.930**	**0.931**	**0.930**	**0.930**
Deep learning + pretraining	GRU+Word2Vec	0.748	0.750	0.745	0.747
	GRU+GloVe	0.748	0.748	0.748	0.748
	Bi-LSTM + Word2Vec	0.875	0.875	0.876	0.875
	Bi-LSTM + GloVe	0.831	0.832	0.830	0.830
	TextCNN + Word2Vec	0.923	0.923	0.923	0.923
	TextCNN + GloVe	0.926	0.927	0.927	0.927

performs best compared to the above methods. On the unbalanced dataset-1, compared with random forest and SVM+TF-IDF, the performance of TextCNN improves (A: 3.1%, P: 2.3%, R: 3.8%, F: 3.3%) and (A: 3%, P: 2.7%, R: 3.3%, F: 3.2%), respectively. On the unbalanced dataset-2, compared with random forest and SVM+TF-IDF, the performance of TextCNN improves (A: 3.2%, P: 1.2%, R: 6.8%, F: 4.8%) and (A: 2.4%, P: 0.8%, R: 5%, F: 3.5%), respectively. In the deep learning + pretraining methods, after using the pretraining model Word2Vec and GloVe for word embedding, the performance of GRU and Bi-LSTM decreased significantly. And the result of the TextCNN methods decreases slightly. We guess these results also may lead by the fact that these two pretraining models are trained on a specific dataset, which is different from the writing style of the EMSCAD dataset.

On the unbalanced dataset-3 and unbalanced dataset-4, though the unbalanced ratio of these two datasets increases, the TextCNN method maintains its advantage. Still, it has the best detection effect compared with other methods. In particular, on the unbalanced dataset-3 and unbalanced dataset-4, TextCNN performs best compared to the other methods. As the datasets become more unbalanced, the influence of TF-IDF on accuracy, precision, recall, and $F1$-score gradually decreases, and the recall and $F1$-score of the traditional learning methods and traditional learning + feature extraction methods significantly reduce, which shows that the TextCNN method has good robustness even when the dataset is very unbalanced. We think the TextCNN method uses multiple convolution kernels of different sizes to embed documents that enrich the semantic representation. In conclusion, TextCNN has advantages in dealing with balanced and unbalanced datasets; therefore, it is more suitable for dealing with the data in real life.

4.4. Analysis of Energy Consumption. Since the goal of the green IoT is to achieve better results in an environmentally friendly manner (i.e., less computing consumption), for the computing consumption, two aspects should be considered, i.e., training time and testing time. Testing time is even more critical for IoT devices than training time for getting a good model. This is because IoT devices with less testing time are more sensitive and can contribute to better human-computer interaction. Therefore, to further compare the performance of the compared methods, this section analyzes the experimental results from the perspective of training time and testing time and takes the testing time as the primary consideration. Figures 4 and 5 are the results of training time and testing time on the balanced dataset. Two figures are drawn because the training time of traditional and deep learning methods is of different orders of magnitude.

As can be seen from Figure 4(a), among the four traditional learning methods (see left columnar plexus of Figure 4(a)), Naïve Bayes has the shortest training time and SVM has the longest one. After feature extraction (see right columnar plexus of Figure 4(a)), expecting for logistic regression, the training time of all the other methods, such as RF, SVM, and NB, increases. Compared with traditional learning methods, the training time of the detection methods based on deep learning is significantly increased to a larger order of magnitude, shown in Figure 5(a). That is because deep learning methods need much longer to train the deep neural network. In addition to the GRU method, the training time of Bi-LSTM and TextCNN increases after using pretraining models for word embedding (see right columnar plexus of Figure 5(a)), and the training time of the TextCNN method increases most significantly.

TABLE 5: Experimental results on the unbalanced datasets.

Category	Model	Unbalanced dataset-1				Unbalanced dataset-2			
		A	P	R	F	A	P	R	F
Traditional learning	RF	0.910	0.919	0.896	0.905	0.928	0.946	0.864	0.896
	SVM	0.896	0.898	0.885	0.890	0.924	0.932	0.865	0.892
	LR	0.877	0.872	0.874	0.873	0.906	0.874	0.880	0.877
	NB	0.830	0.824	0.830	0.826	0.832	0.781	0.826	0.796
Traditional learning + feature extraction	RF+TF-IDF	0.904	0.912	0.890	0.898	0.918	0.940	0.844	0.879
	SVM+TF-IDF	0.911	0.915	0.901	0.906	0.936	0.950	0.882	0.909
	LR+TF-IDF	0.882	0.885	0.870	0.877	0.904	0.912	0.830	0.860
	NB+TF-IDF	0.836	0.834	0.824	0.828	0.861	0.855	0.761	0.791
Deep learning	GRU	0.865	0.861	0.857	0.857	0.912	0.893	0.868	0.879
	Bi-LSTM	0.886	0.881	0.881	0.880	0.915	0.894	0.875	0.883
	TextCNN	**0.941**	**0.942**	**0.934**	**0.938**	**0.960**	0.958	**0.932**	**0.944**
Deep learning + pretraining	GRU+Word2Vec	0.802	0.796	0.808	0.798	0.855	0.820	0.783	0.800
	GRU+GloVe	0.808	0.804	0.797	0.799	0.859	0.814	0.787	0.799
	Bi-LSTM + Word2Vec	0.889	0.884	0.884	0.884	0.918	0.896	0.885	0.889
	Bi-LSTM + GloVe	0.854	0.848	0.855	0.851	0.901	0.895	0.825	0.853
	TextCNN + Word2Vec	0.938	0.939	0.932	0.936	0.955	**0.960**	0.920	0.938
	TextCNN + GloVe	0.932	0.937	0.922	0.928	0.949	0.958	0.906	0.928
Category	Model	Unbalanced dataset-3				Unbalanced dataset-4			
		A	P	R	F	A	P	R	F
Traditional learning	RF	0.944	**0.963**	0.824	0.875	0.977	0.985	0.768	0.842
	SVM	0.940	0.941	0.821	0.867	0.974	0.984	0.733	0.811
	LR	0.937	0.879	0.881	0.880	0.969	0.830	0.832	0.831
	NB	0.853	0.738	0.840	0.769	0.868	0.613	0.827	0.646
Traditional learning + feature extraction	RF+TF-IDF	0.939	0.962	0.806	0.861	0.976	**0.987**	0.755	0.831
	SVM+TF-IDF	0.954	0.960	0.861	0.901	0.980	0.984	0.799	0.867
	LR+TF-IDF	0.923	0.920	0.771	0.822	0.965	0.938	0.656	0.724
	NB+TF-IDF	0.890	0.891	0.657	0.705	0.955	0.961	0.541	0.564
Deep learning	GRU	0.947	0.918	0.861	0.885	0.973	0.875	0.833	0.850
	Bi-LSTM	0.946	0.920	0.862	0.888	0.979	0.922	0.837	0.874
	TextCNN	**0.969**	0.956	**0.923**	**0.938**	**0.986**	0.954	**0.896**	**0.922**
Deep learning + pretraining	GRU+Word2Vec	0.916	0.855	0.781	0.809	0.967	0.895	0.733	0.791
	GRU+GloVe	0.923	0.867	0.791	0.791	0.969	0.885	0.775	0.820
	Bi-LSTM+Word2Vec	0.947	0.908	0.869	0.887	0.973	0.905	0.802	0.845
	Bi-LSTM+GloVe	0.932	0.902	0.807	0.845	0.972	0.920	0.758	0.817
	TextCNN+Word2Vec	0.968	0.967	0.900	0.929	0.985	0.981	0.860	0.911
	TextCNN+GloVe	0.949	0.958	0.906	0.928	0.962	0.963	0.879	0.915

For the testing time, according to Figure 4(b), we can see that almost all the methods need less than 0.05 s in response to each test except the SVM and SVM+TF-IDF methods. I think SVM and SVM+TF-IDF spend more time searching for the optimal hyperplane. In particular, the logistic regression method has the shortest testing time, which means it is the most sensitive method for IoT devices. After using TF-IDF for feature extraction (see right columnar plexus of Figure 4(b)), the testing time of all the other two methods, such as NB and LR, is decreased except for the RF and SVM methods. From Figure 5(b), we can see that in the deep learning methods, the testing time of TextCNN is shorter than GRU and Bi-LSTM. With the adoption of Word2Vec and GloVe, the testing time of GRU, Bi-LSTM, and TextCNN all increased. It is worth noting that the response time of TextCNN is about 0.6 s per test, which is slightly higher than that of traditional learning methods. Compared with the results, we get by using the traditional learning methods and the traditional learning + feature extraction methods in Table 4 and Table 5, we can see that TextCNN achieves the best detection performance on accuracy, precision, recall, and $F1$-score. Moreover, comparing the results
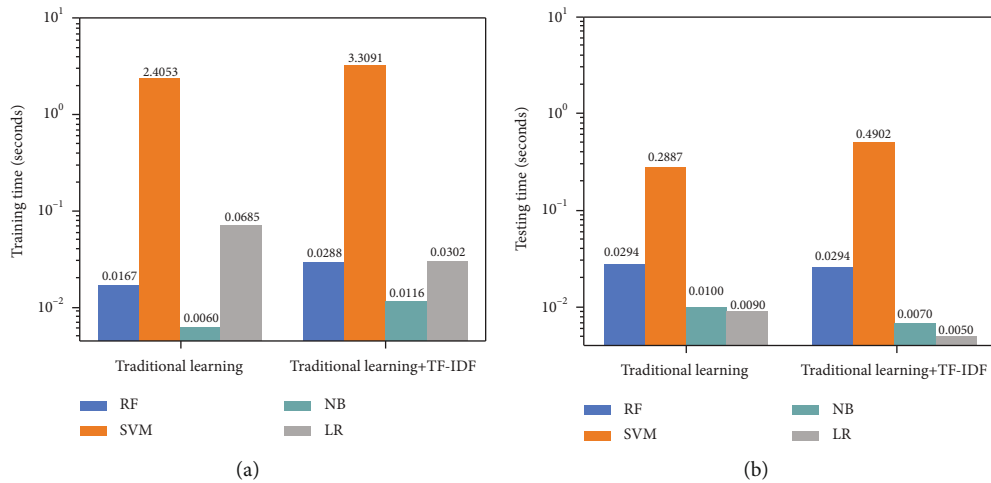
Figure 4: (a) Training time and (b) testing time on the four traditional learning methods.
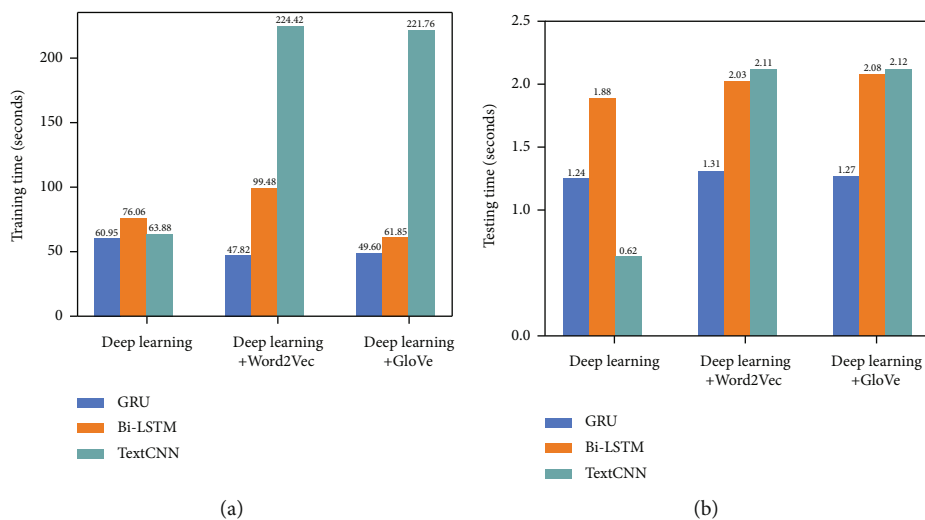

Figure 5: (a) Training time and (b) testing time on the two deep learning methods.

in Figures 4 and 5, we can see that TextCNN achieves acceptable energy consumption during testing time. Therefore, TextCNN is a worthwhile method for detecting fraudulent recruitment advertisements in the IoT Environment.

5. Conclusion

This paper analyzes and compares seventeen methods, four traditional learning methods (i.e., random forest (RF), SVM, logistic regression (LR), and Naïve Bayes (NB)). These four traditional learning methods combined with feature extracting method (i.e., RF+TF-IDF, SVM+ TF-IDF, LR +TF-IDF, and NB+ TF-IDF) and three deep learning methods (i.e., GRU, Bi-LSTM, and TextCNN); these three deep learning methods combined with pretraining model for word embedding (i.e., GRU+Word2Vec, GRU+GloVe, Bi-LSTM+Word2Vec, Bi-LSTM+GloVe, TextCNN+Word2-Vec, and TextCNN+GloVe). To further analyze the perfor-

mance of each method, comprehensive experiments are carried out based on the EMSCAD dataset.

The experimental results indicate that the deep learning methods are generally better than the traditional learning methods, the traditional learning + feature extraction methods, and even the deep learning + pretraining-based methods, regardless of the balanced or the unbalanced datasets. In particular, TextCNN outperforms other deep learning methods. In terms of time performance, though TextCNN needs considerably longer offline training time, the testing time (i.e., response time) is slightly higher than the traditional learning-based methods. Those results indicate that the TextCNN method can detect real-life fraudulent recruitment advertisements in the IoT environment.

In summary, using unified evaluation metrics and datasets and considering the impact of the imbalance rates make the comparison and analysis of fraudulent recruitment advertisement detection methods more systematic and comprehensive.

Therefore, this paper can help researchers systematically understand the detecting methods of fraudulent recruitment advertisements and provide directions for selecting and exploring suitable methods. The experimental results have a specific reference value for further research of higher performance recruitment advertising detection methods.

Based on this paper, in the future, we aim to collect our employment fraud detecting dataset, and we will study a higher performance and lower energy consumption fraudulent recruitment advertising detection method to help achieve the goal of green IoT. It is an exciting direction to ensemble high-performance methods such as TextCNN, LSTM, or other popular deep learning methods and technologies (e.g., attention mechanism, mask mechanism). Moreover, deeper pretraining models for word embeddings are also an interesting direction.

Data Availability

The data we used is available and can be obtained from the author (202082060057@sdust.edu.cn).

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

References

[1] P. Kaur, "E-recruitment: a conceptual study," *International Journal of Applied Research*, vol. 1, no. 8, pp. 78–82, 2015.

[2] C. Wang, "False information analysis of online recruitment," *Modern Marketing (Business Edition)*, vol. No. 335, no. 11, pp. 140-141, 2020.

[3] S. Lal, R. Jiaswal, N. Sardana, A. Verma, A. Kaur, and R. Mourya, "ORFDetector: ensemble learning based online recruitment fraud detection," in *2019 Twelfth International Conference on Contemporary Computing (IC3)*, pp. 1–5, Noida, India, 2019, August.

[4] P. Rubin, "Regulation of information and advertising," *CPI Journal*, vol. 4, 2008.

[5] Y. Jiang, "On the construction of advertising credit supervision system," *Industrial and Commercial Administration*, vol. 4, 2004.

[6] S. Vidros, C. Kolias, and G. Kambourakis, "Online recruitment services: another playground for fraudsters," *Computer Fraud & Security*, vol. 2016, no. 3, pp. 8–13, 2016.

[7] I. Androutsopoulos, J. Koutsias, K. V. Chandrinos, and C. D. Spyropoulos, "An experimental comparison of naive Bayesian and keyword-based anti-spam filtering with personal e-mail messages," in *Proceedings of the 23rd annual international ACM SIGIR conference on research and development in information retrieval*, pp. 160–167, Athens, Greece, 2000, July.

[8] S. Vidros, C. Kolias, G. Kambourakis, and L. Akoglu, "Automatic detection of online recruitment frauds: characteristics, methods, and a public dataset," *Future Internet*, vol. 9, no. 1, p. 6, 2017.

[9] S. Dutta and S. K. Bandyopadhyay, "Fake job recruitment detection using machine learning approach," *International Journal of Engineering Trends and Technology*, vol. 68, no. 4, pp. 48–53, 2020.

[10] O. Nindyati and I. G. B. B. Nugraha, "Detecting scam in online job vacancy using behavioral features extraction," in *2019 International Conference on ICT for Smart Society (ICISS)*, vol. 7, pp. 1–4, Bandung, Indonesia, 2019, November.

[11] S. Mahbub and E. Pardede, "Using contextual features for online recruitment fraud detection," in *the 27th International Conference on Information Systems Development*, Lund, Sweden, 2018.

[12] B. Alghamdi and F. Alharby, "An intelligent model for online recruitment fraud detection," *Journal of Information Security*, vol. 10, no. 3, pp. 155–176, 2019.

[13] A. Mehboob and M. S. I. Malik, "Smart fraud detection framework for job recruitments," *Arabian Journal for Science and Engineering*, vol. 46, no. 4, pp. 3067–3078, 2021.

[14] J. Kim, H. J. Kim, and H. Kim, "Fraud detection for job placement using hierarchical clusters-based deep neural networks," *Applied Intelligence*, vol. 49, no. 8, pp. 2842–2861, 2019.

[15] G. Salton and C. T. Yu, "On the construction of effective vocabularies for information retrieval," *ACM SIGPLAN Notices*, vol. 10, no. 1, pp. 48–60, 1975.

[16] T. Mikolov, G. Corrado, C. Kai, and J. Dean, "Efficient estimation of word representations in vector space," in *Proceedings of the international conference on learning representations (ICLR 2013)*, Scottsdale, AZ, USA, 2013.

[17] J. Pennington, R. Socher, and C. Manning, "Glove: global vectors for word representation," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Doha, Qatar, 2014.

[18] L. Breiman, "Random forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.

[19] S. W. Kwok and C. Carter, "Multiple decision trees," in *Machine Intelligence and Pattern Recognition*, vol. 9, pp. 327–335, North-Holland, 1990.

[20] T. K. Ho, "The random subspace method for constructing decision forests," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 8, pp. 832–844, 1998.

[21] V. Vapnik, *The Nature of Statistical Learning Theory*, Springer Science & Business Media, 2013.

[22] M. E. Maron and J. L. Kuhns, "On relevance, probabilistic indexing and information retrieval," *Journal of the ACM (JACM)*, vol. 7, no. 3, pp. 216–244, 1960.

[23] B. Efron, "The efficiency of logistic regression compared to normal discriminant analysis," *Journal of the American Statistical Association*, vol. 70, no. 352, pp. 892–898, 1975.

[24] Y. LeCun, B. Boser, J. S. Denker et al., "Backpropagation applied to handwritten zip code recognition," *Neural Computation*, vol. 1, no. 4, pp. 541–551, 1989.

[25] Y. Kim, "Convolutional neural networks for sentence classification," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Doha, Qatar, 2014.

[26] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

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

[28] K. Cho, B. Van Merriënboer, C. Gulcehre et al., "Learning phrase representations using RNN encoder-decoder for statistical machine translation," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Doha, Qatar, 2014.

[29] M. Schuster and K. K. Paliwal, "Bidirectional recurrent neural networks," *IEEE Transactions on Signal Processing*, vol. 45, no. 11, pp. 2673–2681, 1997.