JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

Fake Job Recruitment Detection Using Supervised Machine Learning Approaches

Dr. Radhamani V
Assistant Professor
Coimbatore Institute of Technology
Coimbatore, India

Dhinesh Kumar S
MSc Decision and Computing Sciences
Coimbatore Institute of Technology
Coimbatore, India

Raghul Manickam V S
MSc Decision and Computing Sciences
Coimbatore Institute of Technology
Coimbatore, India

Abstract:

The proliferation of online job listings and the increasing reliance on digital recruitment platforms have given rise to a pressing issue of fake job postings. In this era of digital job searching, job seekers are vulnerable to deceptive job offers, which can lead to financial losses and emotional distress. This research paper explores the development and application of machine learning algorithms for the detection of fake job postings. This study compiles a comprehensive dataset of job listings, encompassing various attributes such as job descriptions, company details, and application processes. By leveraging state-of-the-art machine learning techniques, including natural language processing and feature engineering, this research aims to identify patterns and characteristics that distinguish genuine job opportunities from fraudulent ones. The outcomes of this research are not only expected to enhance the job-seeking experience for individuals but also to assist job platforms, employers, and regulatory authorities in preventing the dissemination of deceptive job postings.

Keywords: Machine Learning, Fake job, Classifier, Supervised learning, Random Forest Classifier

I. INTRODUCTION

In today's digital age, the internet has revolutionized the job search process, making it more convenient than ever to seek employment opportunities. However, this convenience has also given rise to a concerning issue – the proliferation of fake job postings. As job seekers increasingly turn to online platforms to find their next career move, they are confronted with an ever-growing number of deceptive job listings. These listings, often crafted with the intent to defraud, pose a serious threat to job seekers, leading them into scams, identity theft, and financial loss. The rise of fake job postings represents a critical problem that warrants immediate attention. Job platforms, recruitment agencies, and job seekers alike are facing the challenge of distinguishing between legitimate job opportunities and deceptive offers. To combat this issue effectively, a multidisciplinary approach that leverages data science, machine learning, and natural language processing is essential. This research paper aims to address the pressing concern of fake job postings by investigating techniques for predicting and identifying deceptive employment opportunities. By analysing a vast dataset of job listings, we intend to develop a predictive model capable of discerning fraudulent job postings from authentic ones. Our study encompasses a broad spectrum of data sources, including textual analysis, user-generated reviews, and historical job posting patterns, to create a comprehensive framework for detecting fraudulent activity. The significance of this research extends beyond the job market itself. Identifying and combatting fake job postings can help protect job seekers from financial exploitation and preserve their personal information. Moreover, it can enhance the integrity of job platforms and recruitment agencies, fostering trust within the employment ecosystem.

In the following sections of this paper, we will delve into the methodology employed, the data sources harnessed, and the machine learning algorithms used to predict fake job postings. Our ultimate goal is to contribute valuable insights and tools that will assist job seekers, job platforms, and regulatory bodies in safeguarding the integrity of the job market and promoting a safe and reliable space for job seekers.

II. LITERATURE SURVEY

In the paper titled *"Fake Job Detection and Analysis Using Machine Learning and Deep Learning Algorithms"* by C.S. Anita et al. explores this problem statement. In order to identify fraudulent jobs and distinguish them from actual employment, machine learning and deep learning techniques are employed in this work. In order to ensure that the classification method used is extremely precise and accurate, the data analysis and cleaning parts are also proposed in this study. It should be emphasized that the data cleaning phase is crucial to any machine learning project since it directly affects how accurate both machine learning and deep learning

algorithms will be. As a result, this study places a lot of emphasis on the data cleaning and preprocessing step. High accuracy and high precision can be used in the classification and detection of bogus jobs.

The paper written by C. Jagadeesh et al. titled *"Artificial intelligence based Fake Job Recruitment Detection Using Machine Learning Approach"* implemented the Artificial Intelligence based approach to serve this problem. The paper proposes an automated tool that utilizes machine learning classification techniques to prevent fraudulent job postings on the internet. It explores two primary categories of classifiers: single classifiers and ensemble classifiers, with results favoring the latter for the detection of fraudulent job posts. Within the single classifier category, the paper employs various algorithms, including Naive Bayes, Multi-Layer Perceptron Classifier, K-nearest Neighbor, and Decision Tree Classifier. In contrast, the ensemble approach incorporates Random Forest Classifiers and boosting algorithms such as AdaBoost and Gradient Boosting. The primary goal of this research is to determine the authenticity of a job posting. The data preprocessing steps involve handling missing values, eliminating stop words, removing irrelevant attributes, and eliminating extra spaces, ultimately preparing the dataset for categorical encoding to create feature vectors. These feature vectors are then applied to multiple classifiers, and the study evaluates their performance using metrics like Accuracy, F-measure, and Cohen's Kappa score. Detecting fraudulent job descriptions can significantly benefit job seekers by reducing the risks associated with job hunting. However, the challenge of identifying fake job postings is compounded by class imbalance, where the number of genuine job postings far exceeds that of fraudulent ones. To address this, the paper introduces an effective framework known as FJD-OT (Fake Job Description Detection Using Oversampling Techniques), which leverages an oversampling technique to enhance the accuracy of detecting fake job descriptions.

III. DATASET DESCRIPTION

We've conducted training and testing using a dataset acquired from Kaggle to identify counterfeit job postings. Our dataset comprises approximately 18,000 rows and 17 columns, encompassing both text and numerical data. These columns pertain to the job post details found on online job platforms like Internshala and Naukri, offering a comprehensive overview of how jobs are advertised online.

job_id	int64
title	object
location	object
department	object
salary_range	object
company_profile	object
description	object
requirements	object
benefits	object
telecommuting	int64
has_company_logo	int64
has_questions	int64
employment_type	object
required_experience	object
required_education	object
industry	object
function	object
fraudulent	int64

Fig. 1. Schema of Dataset

This dataset comprises 17,880 job postings and serves as a testbed for evaluating a proposed analytical approach. To establish a reliable baseline, a series of steps are taken to balance the dataset. Prior to deploying any classification models, the dataset undergoes essential preprocessing steps, including the removal of missing values, the elimination of stop words, the removal of irrelevant attributes, and the cleaning of extra spaces. These preprocessing steps are necessary to prepare the dataset for subsequent categorical encoding, enabling the creation of feature vectors. These feature vectors are then used as input for various classifiers to assess their performance.

A. *Exploratory Data Analysis*

Conduct exploratory data analysis to understand the distribution of features, identify trends, and potential correlations. Visualize data to gain insights into the characteristics of fake and legitimate job listings.

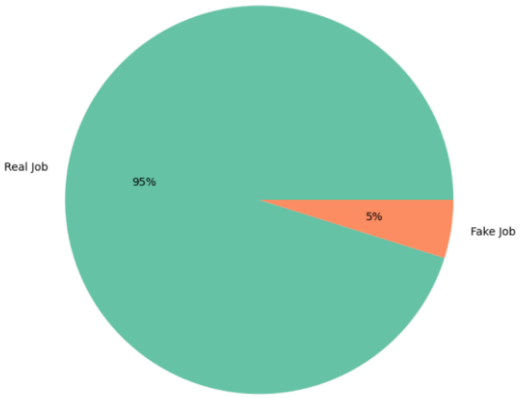

Fig. 2. Split of Real and Fake Jobs

Fig. 2. explains the dataset split of real and fraudulent job listings that is available in the dataset.

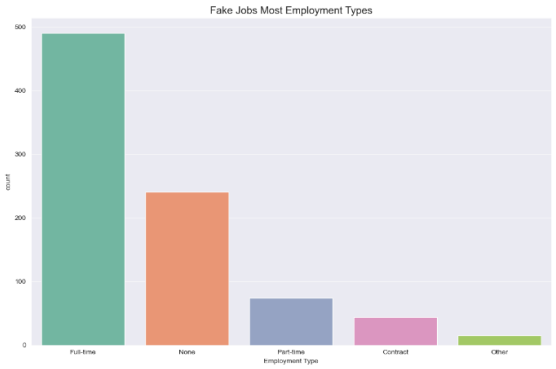

Fig. 3. Fake Jobs – Employment Types

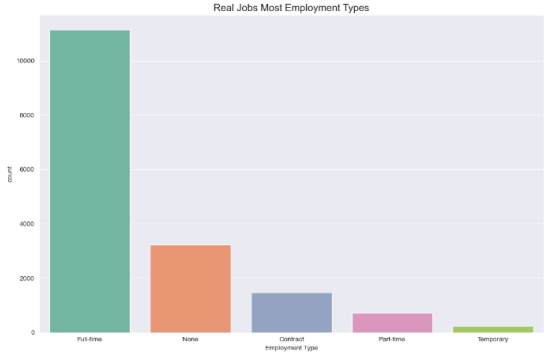

Fig. 3. Real Jobs – Employment Types

Fig. 3. and Fig. 4. explains the variations of count in the employment types in Fake and Real jobs that is provided in the dataset which provides insights on the data that the paper has employed in this project.

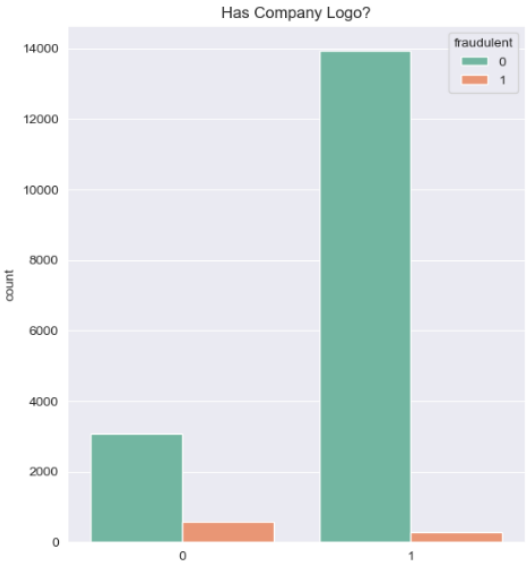

Fig. 5. Number of Fake and Real Jobs containing Company Logo

Fig. 5. briefly interprets the amount of fraudulent job listings and real job listings that does contain company logo and the ones which does not contain company logos.

B. Data Pre-processing

Data preprocessing refers to the set of procedures and techniques used to clean, transform, and organize raw data into a format suitable for analysis or machine learning. It is a crucial step in the data analysis pipeline, as the quality of the data greatly influences the results and effectiveness of data-driven models. Here are some common data preprocessing tasks that the project has employed on the dataset.

Lowercase conversion:

Also known as text normalization, is a common data preprocessing technique where all the characters in a text are converted to lowercase. This process helps to ensure uniformity in the text and can be beneficial in various natural language processing (NLP) and text analysis tasks.

Punctuation removal:

In Data pre-processing, it is often important to clean and preprocess text data, which can involve removing punctuation. Punctuation removal can be useful for tasks such as text analysis, sentiment analysis, and natural language processing.

Removing Stopwords:

Removing stopwords is a common text preprocessing step in natural language processing. Stopwords are words that are frequently used in language but often don't carry significant meaning in text analysis tasks. Examples of stopwords in English include "the," "is," "and," "in," and "of."

Remove extra white spaces:

Removing extra white spaces involves reducing multiple consecutive white spaces in text to a single space. This process helps standardize and clean text data. You can achieve this using regular expressions in programming languages like Python. The regular expression \s+ matches one or more consecutive white spaces and replaces them with a single space, effectively eliminating the extra white spaces. This can be particularly useful when processing and analysing text data.

Removing HTML tags:

Removing HTML tags involves eliminating the markup language used for formatting web content from a text string. This is often done during text preprocessing for data analysis. You can use regular expressions or specialized HTML parsing libraries to achieve this. By removing HTML tags, you extract the plain text content, making it suitable for various text analysis tasks, such as natural language processing or sentiment analysis.

IV. PROPOSED METHODOLOGY

The paper uses multiple classification algorithms approaches which come under supervised machine learning techniques.

Naive Bayes Classifier

The Naive Bayes classifier is a supervised machine learning algorithm, which is used for classification tasks, like text classification. It is also part of a family of generative learning algorithms, meaning that it seeks to model the distribution of inputs of a given class or category.

$$P(A|B) = \frac{P(B|A)\ P(A)}{P(B)}$$

Logistic Regression

This type of statistical model (also known as logit model) is often used for classification and predictive analytics. Logistic regression estimates the probability of an event occurring, such as voted or didn't vote, based on a given dataset of independent variables.

$$P = \frac{e^{a+bX}}{1+e^{a+bX}}$$

Decision Tree Classifier

Decision Tree Classifier is a class capable of performing multi-class classification on a dataset. In case that there are multiple classes with the same and highest probability, the classifier will predict the class with the lowest index amongst those classes.

Gini Index	Entropy
$I_G = 1 - \sum_{j=1}^{c} p_j^2$	$I_H = -\sum_{j=1}^{c} p_j log_2(p_j)$

Random Forest Classifier

Random forests or random decision forests is an ensemble learning method for classification, regression and other tasks that operates by constructing a multitude of decision trees at training time. For classification tasks, the output of the random forest is the class selected by most trees.

$$K_k^{cc}(\mathbf{x},\mathbf{z}) = \sum_{k_1,\ldots,k_d,\sum_{j=1}^{d} k_j=k} \frac{k!}{k_1!\ldots k_d!}\left(\frac{1}{d}\right)^k \prod_{j=1}^{d} 1_{\lceil 2^{k_j} x_j \rceil = \lceil 2^{k_j} z_j \rceil},$$

$$\text{for all } \mathbf{x},\mathbf{z} \in [0,1]^d.$$

V. EXPERIMENTAL RESULTS

The previously mentioned classifiers have been trained and evaluated to identify fraudulent job listings within a dataset comprising both counterfeit and genuine job posts. This assessment is carried out to detect fake job postings effectively.

Classification reports are a helpful way to evaluate the performance of a classification model, such as a machine learning classifier. They provide detailed information about the model's ability to correctly classify data points. A common tool for generating classification reports in Python is the *classification_report* function from the *sklearn.metrics* module in *scikit-learn*.

Testing Classification Report for Naive Bayes

	precision	recall	f1-score	support
0	0.99	0.94	0.96	3403
1	0.40	0.87	0.55	173
accuracy			0.93	3576
macro avg	0.70	0.90	0.76	3576
weighted avg	0.96	0.93	0.94	3576

Testing Classification Report for Decision Tree Classifier

	precision	recall	f1-score	support
0	0.99	0.99	0.99	3403
1	0.80	0.80	0.80	173
accuracy			0.98	3576
macro avg	0.90	0.90	0.90	3576
weighted avg	0.98	0.98	0.98	3576

Testing Classification Report for Random Tree Classifier

	precision	recall	f1-score	support
0	0.98	1.00	0.99	3403
1	1.00	0.58	0.73	173
accuracy			0.98	3576
macro avg	0.99	0.79	0.86	3576
weighted avg	0.98	0.98	0.98	3576

Fig. 6. Classification report for different classifiers

VI. CONCLUSION

The primary goal of employment scam detection is to assist job-seekers in securing legitimate employment opportunities from trustworthy companies. In addressing the challenge of identifying employment scams, the paper introduces various machine learning algorithms as proactive measures. The paper adopts a supervised approach, employing multiple classifiers to illustrate their effectiveness in detecting fraudulent job offers.

The experimental outcomes underscore the superiority of the Random Forest classifier compared to its counterparts in the field of job scam identification. Notably, the proposed methodology demonstrates an impressive accuracy rate of 98.27%, a substantial improvement when compared to currently established methods in this domain. This innovation in employment scam detection promises to enhance the job-seeking experience and significantly reduce the risk of falling victim to fraudulent employment schemes.

REFERENCES

[1] B. Alghamdi and F. Alharby, ―An Intelligent Model for Online Recruitment Fraud Detection,‖ J. Inf. Secur., vol. 10, no. 03, pp. 155–176, 2019, doi: 10.4236/jis.2019.103009.

[2] I. Rish, ―An Empirical Study of the Naïve Bayes Classifier An empirical study of the naive Bayes classifier,‖ no. January 2001, pp. 41–46, 2014.

[3] D. E. Walters, ―Bayes's Theorem and the Analysis of Binomial Random Variables,‖ Biometrical J., vol. 30, no. 7, pp. 817–825, 1988, doi: 10.1002/bimj.4710300710.

[4] F. Murtagh, ―Multilayer perceptrons for classification and regression,‖ Neurocomputing, vol. 2, no. 5–6, pp. 183–197, 1991, doi: 10.1016/0925-2312(91)90023-5.

[5] P. Cunningham and S. J. Delany, ―K -Nearest Neighbour Classifiers,‖ Mult. Classif. Syst., no. May, pp. 1–17, 2007, doi: 10.1016/S0031-3203(00)00099-6.

[6] H. Sharma and S. Kumar, ―A Survey on Decision Tree Algorithms of Classification in Data Mining,‖ Int. J. Sci. Res., vol. 5, no. 4, pp. 2094–2097, 2016, doi: 10.21275/v5i4.nov162954.

[7] E. G. Dada, J. S. Bassi, H. Chiroma, S. M. Abdulhamid, A. O. Adetunmbi, and O. E. Ajibuwa, "Machine learning for email spam filtering: review, approaches and open research problems,‖ Heliyon, vol. 5, no. 6, 2019, doi: 10.1016/j.heliyon.2019.e01802.

[8] L. Breiman, ―ST4_Method_Random_Forest,‖ Mach. Learn., vol. 45, no. 1, pp. 5–32, 2001, doi: 10.1017/CBO9781107415324.004.

[9] B. Biggio, I. Corona, G. Fumera, G. Giacinto, and F. Roli, ―Bagging classifiers for fighting poisoning attacks in adversarial classification tasks," Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), vol. 6713 LNCS, pp. 350–359, 2011, doi: 10.1007/978-3-642-21557-5_37.