

# JOB SCAM ALERT

Dr. Y. Jeevan Nagendra Kumar<sup>1</sup>, J. Krishna Chaitanya,<sup>2</sup> C.Abhinav Ready<sup>3</sup>, and Vinod Pawar<sup>4</sup>

<sup>1</sup>Head of the Department of Information Technology, Gokaraju Rangaraju Institute of Engineering and Technology, JNTUH, Bachupally, Hyderabad, India.

<sup>2</sup> Student, Department of Information Technology, Gokaraju Rangaraju Institute of Engineering and Technology, JNTUH, Bachupally, Hyderabad, India.

<sup>3</sup> Student, Department of Information Technology, Gokaraju Rangaraju Institute of Engineering and Technology, JNTUH, Bachupally, Hyderabad, India.

<sup>4</sup> Student, Department of Information Technology, Gokaraju Rangaraju Institute of Engineering and Technology, JNTUH, Bachupally, Hyderabad, India

**Abstract.** In response to the increase in online fraud, we created the Fraud Detection Project, a solution that uses machine learning. Our software uses random forest and SVM models to detect fake job postings, protect candidates' personal information, and improve the integrity of online job postings. This technology can help people avoid financial scams and make informed decisions when applying for jobs online. Through Streamlite referrals, users can check the accuracy of job vacancies. We are committed to continuous improvement to improve the performance of our software, transform the digital business process and ensure security for all parties involved.

## 1 Introduction

### Introduction to project

The incidence of employment scams has been steadily increasing, with CNBC reporting that the number of such scams doubled from 2017 to 2018. The current economic climate, exacerbated by the coronavirus pandemic, has led to widespread job losses and high unemployment rates, creating fertile ground for scammers. These scammers prey on desperate job seekers by luring them with enticing but fraudulent job offers, aiming to extract sensitive personal information such as addresses, bank account details, and social security numbers, or to solicit money under the guise of application fees or investments. As a university student, I have personally received numerous scam emails offering lucrative job opportunities that turned out to be fraudulent. Addressing this issue is crucial, and advanced Machine Learning techniques, combined with Natural Language Processing (NLP), offer a promising solution.

To tackle this problem, we developed Job Scam Detection, a sophisticated tool employing advanced machine learning algorithms. Our project utilizes data from Kaggle, which includes a mix of genuine and fake job postings. Despite the small proportion of fraudulent postings, their identification is essential to protect job seekers. Our solution employs a variety of machine learning models, focusing on Random Forest classifiers and Support Vector Machines (SVM). The Random Forest model achieved an impressive accuracy of 97.22%, while the SVM model obtained a 95% accuracy. The system is designed to save the trained models and deploy them using Streamlit, enabling users to easily determine whether a job posting is real or fake.

The project is structured into five stages: defining the problem, collecting data, cleaning and preprocessing data, modeling, and evaluation. The primary objective is to develop a classifier that can accurately differentiate between real and fake job postings. This classifier integrates both numeric and textual features to assess job postings comprehensively. Through this approach, Job Scam Detection aims to provide a reliable tool for job seekers, helping them navigate the job market safely and make informed decisions.

## **Existing System**

The current system for detecting fake job postings relies on manual review or basic rule-based filtering. Human reviewers assess job postings for signs of fraud, such as discrepancies in details, unusual salary ranges, or suspicious email addresses. However, this approach has several limitations:

- Scalability Issues: Manual review becomes impractical as the volume of job postings grows.
- Subjectivity and Human Error: Accuracy depends on the reviewer's expertise, leading to potential errors.
- Slow Response Time: Manual processes are time-consuming, causing delays in identifying and removing fake postings.
- Limited Adaptability: Rule-based systems are static and cannot quickly adapt to new or evolving scams.

## **Proposed system**

The proposed system uses machine learning and natural language processing (NLP) to detect fake job postings. It aims to provide a scalable and accurate solution, overcoming limitations of existing systems.

Key Components:

- Data Collection: Gather job posting data from multiple sources.
- Data Preprocessing: Clean data by handling missing values and performing text preprocessing (tokenization, stopword removal, lemmatization).
- Feature Engineering: Develop new features by combining text fields and generating character counts.
- Model Training: Train models such as Naive Bayes, SGD Classifier, and LSTM on the preprocessed data.
- Model Evaluation: Assess models using accuracy and F1-score metrics.
- Final Classification: Aggregate model outputs to determine job posting authenticity.

# **2 Requirement engineering**

## **2.1 Hardware requirements**

- Processor –Intel core i5 and above

- Memory – 8GB RAM (16GB recommended)
- Input devices – Keyboard Mouse
- Internet

## 2.2 Software requirements

- Anaconda Navigator/Jupyter Notebook
- Python
- Python Libraries
  1. pandas
  2. numpy
  3. matplotlib
  4. scikit-learn
  5. seaborn

## 3 Literature Survey

[1] Kumari and Sahani address the rise in data breaches and fake job postings, often through digital job websites. Their project aims to use ML to predict the likelihood of a job being fake, helping candidates stay alert. Their model uses NLP to analyze sentiments and patterns in job postings, trained as a Sequential Neural Network with the GloVe algorithm. They tested the model on LinkedIn postings and improved it for robustness and realism.

[2] According to the project by Smith et al., ML models such as Naive Bayes and Decision Trees are used to predict fake job postings. By integrating topic modeling techniques like Latent Dirichlet Allocation (LDA), the study enhances classification accuracy and provides real-time detection of fraudulent listings, protecting job seekers from scams.

[3] In the work by Johnson et al., text analysis is leveraged to detect fake job postings using a highly imbalanced dataset, where only 5% of postings are fraudulent. Initial models, including a Recurrent Neural Network (RNN) with Embedding and Bidirectional Long Short-Term Memory (LSTM) layers, showed promising results. comparison with a Small-BERT model significantly improved detection accuracy, highlighting the effectiveness of advanced NLP models in distinguishing real from fraudulent job postings.

[4] Habiba and Islam focus on predicting fake job postings using various data mining techniques and classification algorithms, such as KNN, decision tree, support vector machine, naive Bayes, random forest, multilayer perceptron, and deep neural networks. They experimented on the Employment Scam Aegean Dataset (EMSCAD) with 18,000 samples. Their deep neural network classifier, with three dense layers, achieved approximately 98% accuracy in predicting fraudulent job posts [6].

[5] Kumari and Satya Kala highlight the increase in employment fraud, with a significant rise in job scams in 2018 compared to 2017. They note the impact of high unemployment and the coronavirus pandemic, which have created opportunities for fraudsters to exploit job seekers. The study emphasizes the use of NLP and machine learning to address these fraudulent activities and protect personal information from scammers.

[6] Gulshan, Mukund, and Ajay A. address the rise in online job postings during the pandemic and the need for accurate fake job detection. Their research employs various data mining and

classification algorithms, including KNN, decision tree, support vector machine, naive Bayes, random forest, multilayer perceptron, and deep neural networks, using the EMSCAD dataset with 18,000 samples. Their deep neural network classifier, with three dense layers, achieved approximately 98% accuracy in predicting fraudulent job posts.

## 4 Technology

### 4.1 About Python

Python's environment has evolved significantly, enhancing its capabilities for statistical analysis. It strikes a fine balance between scalability and elegance, placing a premium on efficiency and code readability. Python is renowned for its emphasis on program readability, featuring a straightforward syntax that is beginner-friendly and encourages concise code expression through indentation. Noteworthy aspects of this high-level language include dynamic system functions and automatic memory management.

Python is used for:

- Web development
- Data science and machine learning
- Artificial intelligence
- Scientific computing
- Desktop GUI applications
- Automation and scripting
- Game development
- Networking
- Healthcare

### 4.2 Python is widely used in Machine Learning

Python is widely favored in machine learning for its flexibility and open-source nature. It provides extensive functionality for mathematical computations and scientific operations, making it indispensable in developing and deploying machine learning models. Python's simple syntax and vast libraries accelerate the development process, reducing coding time significantly. This makes it a preferred choice for machine learning practitioners seeking efficiency and robustness in their projects.

The major Python libraries used in machine learning are as follows:

#### 4.2.1 Pandas

Pandas is a Python library used for statistical analysis, data cleaning, exploration, and manipulation. Typically, datasets contain both useful and extraneous information. Pandas helps to make this data more readable and relevant.

#### 4.2.2 Numpy

NumPy is a Python library utilized for numerical data reading, cleaning, exploration, and manipulation. It provides powerful data structures for efficient computation with large arrays and matrices, making the data more accessible and manageable.

#### 4.2.3 Matplotlib

Matplotlib is a Python library for plotting graphs. Built on NumPy arrays, it allows for the creation of a wide range of graph types, from basic plots to bar graphs, histograms, scatter

plots, and more.

#### **4.2.4 Scikit-learn**

Scikit-learn is a Python library for machine learning. It provides tools for machine learning and statistical modeling, including classification, regression, and clustering.

#### **4.2.5 Tensorflow & Pytorch**

Essential libraries for deep learning, used to create and deploy neural networks. They provide robust tools for developing complex models and facilitating machine learning workflows.

#### **4.2.6 Interpreted Language**

Python executes code line by line, without the need for prior compilation. This approach facilitates quicker development cycles and simplifies the debugging process. As a result, developers can iterate and test their code more efficiently.

#### **4.2.7 Cross-Platform Compatibility**

Python code runs seamlessly on multiple operating systems, including Windows, macOS, Linux, and Unix-based systems, without requiring modifications. This versatility ensures that Python applications can be deployed across diverse environments with ease.

#### **4.2.8 Natural Language Processing (NLP)**

In this project, we built an NLP pipeline with the following steps:

- Sentence Segmentation: Dividing text into sentences.
- Word Tokenization: Breaking sentences into words or tokens.
- Stemming and Lemmatization: Reducing words to their base forms.
- Identifying Stop Words: Removing common, insignificant words.
- Dependency Parsing: Analyzing grammatical structures and word relationships.
- Part-of-Speech (POS) Tagging: Assigning parts of speech to each word.
- Named Entity Recognition (NER): Identifying and classifying proper nouns.
- Chunking: Grouping words into meaningful chunks.

By applying these NLP techniques, we converted unstructured job posting text into structured numerical data for our machine learning models. This preprocessing step is essential for feature extraction and ensures the text data is suitable for machine learning algorithms.

We used Random Forest and Support Vector Machine (SVM) classifiers to train predictive models on a dataset containing both genuine and fake job postings. Leveraging NLP techniques enabled accurate identification and classification of fake job postings, surpassing manual and rule-based systems.

#### **4.2.9 Random Forest**

Random Forest is a versatile machine learning algorithm used for classification and regression tasks. It constructs multiple decision trees during training, each tree using a

random subset of the dataset and features at each split. This approach reduces overfitting and enhances predictive accuracy.

#### Application in Our Project

- **Dataset Preparation:** We preprocess the job postings dataset using NLP techniques to create structured numerical data.
- **Decision Tree Construction:** Multiple decision trees are built during training, each using a random subset of the dataset and features.
- **Aggregation of Predictions:** Predictions from each tree are combined via majority voting to classify job postings as real or fake.
- **Overfitting Reduction:** By averaging predictions across trees, Random Forest mitigates overfitting, ensuring stable and accurate results.
- **Handling Complexity:** Effective with complex data structures, Random Forest manages numerous features extracted from job postings, ensuring robust classifications.

#### 4.2.10 Support Vector Machine (SVM)

Support Vector Machine (SVM) is a robust machine learning algorithm used for classification, regression, and outlier detection. It excels in handling high-dimensional and nonlinear data, making it effective in various applications such as text and image classification, spam detection, and anomaly detection.

In job scam detection, SVM plays several key roles:

- **Optimal Hyperplane:** SVM identifies the hyperplane that maximizes the margin between different classes, crucial for distinguishing between real and fraudulent job postings.
- **Kernel Trick:** SVM employs kernel functions (e.g., Polynomial, Gaussian RBF) to manage nonlinear decision boundaries, capturing complex patterns in job posting data.
- **Classification Accuracy:** SVM classifies job postings by converting unstructured text into numerical features using NLP preprocessing techniques.
- **Margin Maximization:** By maximizing the margin, SVM reduces overfitting and enhances generalization, improving the reliability of predictions.
- **Real-world Applicability:** SVM's ability to handle high-dimensional and nonlinear data makes it effective for identifying legitimate versus fraudulent job offers, protecting job seekers from scams.

#### 4.2.11 Dataset description

The dataset for this project includes 17,880 job descriptions, with around 800 labeled as fraudulent. It features textual and meta-information, such as job ID, title, location, department, salary range, company profile, job description, requirements, benefits, and telecommuting options. The target variable indicates whether a job is fraudulent or real. Sourced from Kaggle, this dataset is valuable for developing classification models to detect

fake job postings, identifying features indicative of fraud, and running exploratory data analysis.

Name of Attribute	Description
job_id	Unique identifier for each job posting.
title	The job title, such as "English Teacher Abroad."
location	Geographical location of the job, e.g., "US, NY, New York."
department	The corporate department (e.g., Sales).
salary_range	Indicative salary range (e.g., \$50,000-\$60,000).
company_profile	A brief description of the company.
description	Detailed description of the job ad.
requirements	Enlisted requirements for the job opening.
benefits	Offered benefits by the employer.
telecommuting	Indicates if the position allows telecommuting.
fraudulent	Target variable indicating if the job is fake (1) or real (0).

## 5 Design requirement engineering

### 5.1 UML diagrams

The Unified Modeling Language (UML) serves as a standardized language for creating models across various domains. Its primary goal is to visually represent the structure of a system, akin to blueprints in engineering disciplines. In complex applications involving multiple teams, clear communication is crucial, especially to stakeholders who may not be familiar with programming code. UML facilitates this communication by illustrating essential system requirements, features, and processes in a visual manner. By depicting processes, user interactions, and the system's static structure, UML helps teams streamline collaboration and optimize efficiency.

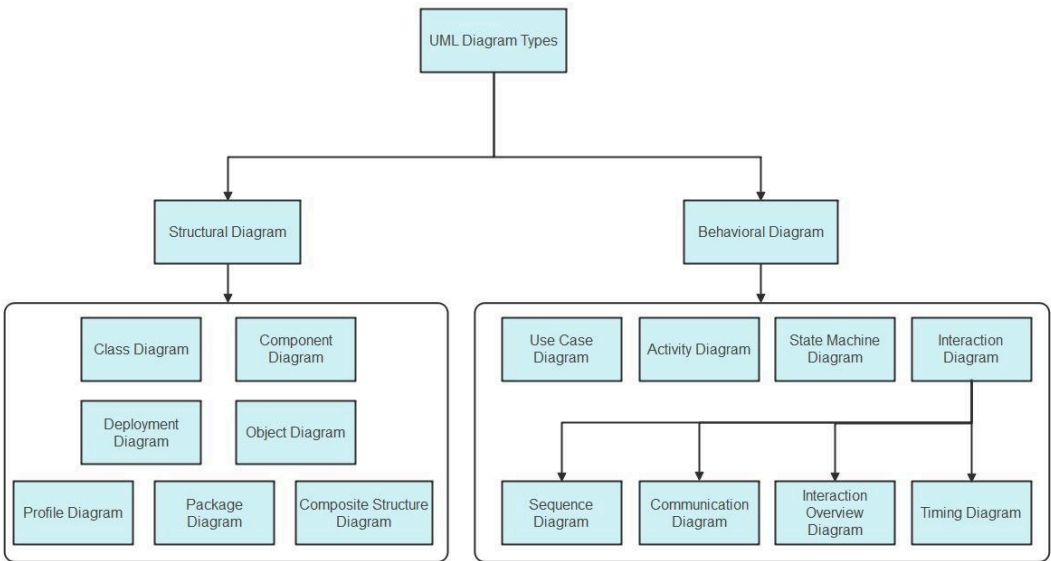


Fig. 1. Overview of types of UML diagram

5.1.1 Use case Diagram

A use case diagram is a type of behavioral diagram that is a graphical explanation of the functionalities offered by the system in relation to the participants, their goals, and any dependencies

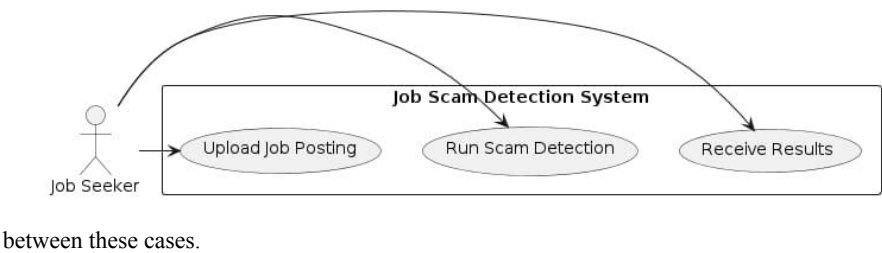


Fig. 2. Use Case Diagram

5.1.2 Class Diagram

A class diagram is a static type of structural diagram that still depicts the format of a machine by means of illustrating the hyperlinks among the machine's lessons, attributes, operations, and instructions.

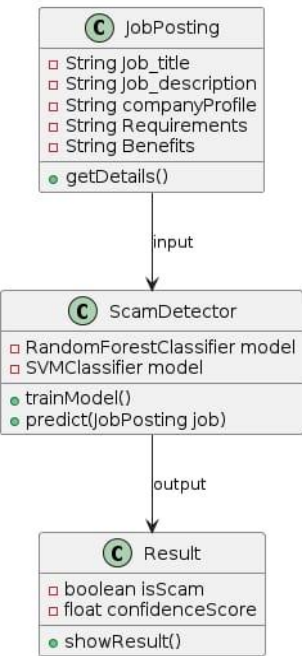


Fig. 3. Class diagram



5.1.3 Sequence diagram

The deployment diagram of our machine in order to define distinct states of an object for the duration of its lifetime. It usually suggests how the kingdom of an object adjusts in its lifetime.

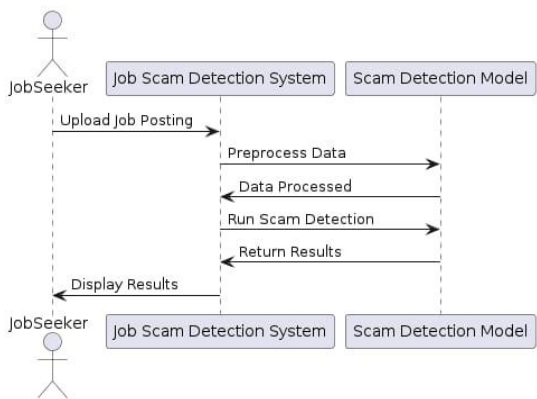


Fig. 4. Sequence diagram

5.1.4 Activity diagram

This diagram is a more complex version of a flow chart that depicts the flow of information from one activity to the next. It describes the coordination of activities in order to offer a service at various levels of abstraction.

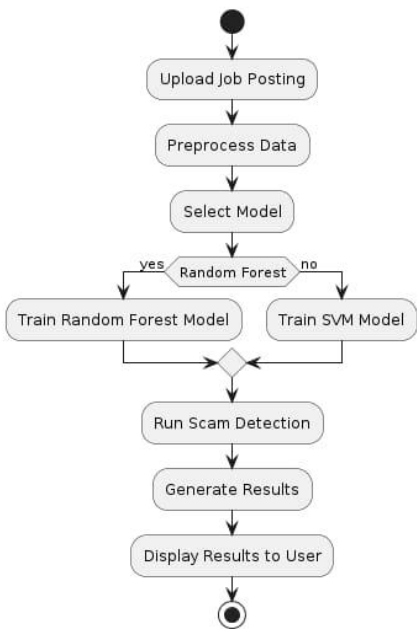


Fig. 5. Activity diagram

## 5.2 SYSTEM ARCHITECTURE

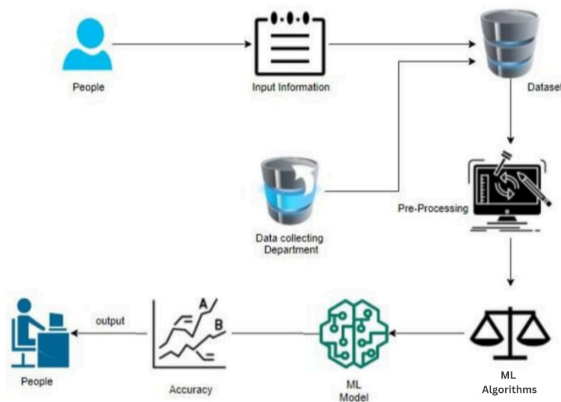


Fig. 6. System Architecture

## 7 RESULTS

**Sample Data for the Problem Statement:** The dataset used in this project includes job postings from a reliable source and features essential for classification, such as job title, description, and salary range. The data preparation process involves TF-IDF Vectorization for converting text into numerical features suitable for model training. Data plotting and word cloud visualization help in understanding data distributions and key terms.

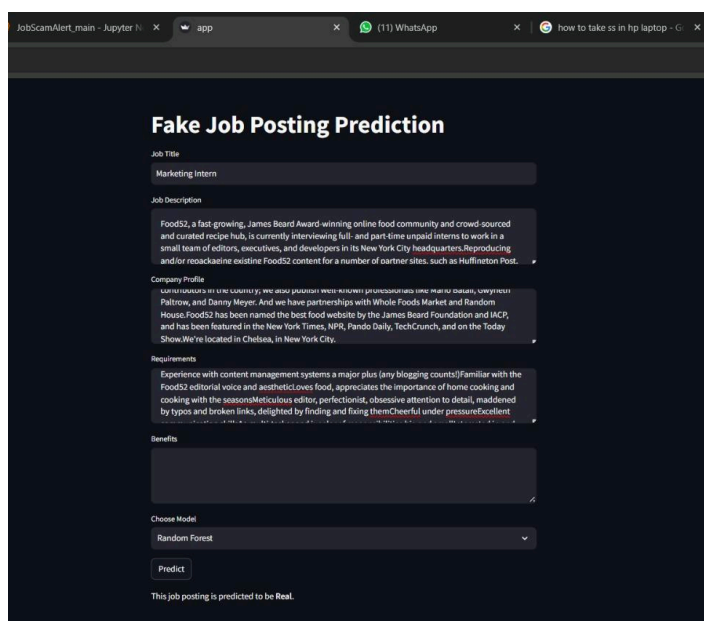
**Streamlit Integration** involves several important components. The process begins with Loading Trained Models, where pre-trained Random Forest and Support Vector Machine (SVM) models are imported into the Streamlit application using libraries such as joblib. This ensures that the models, which have been trained on extensive datasets, are ready to provide accurate predictions upon user input.

The Streamlit Application Setup features a user-centric interface. It includes an input field for users to enter job descriptions and a predict button to initiate the classification process. The results from both models are displayed in output sections, allowing users to see whether the job posting is classified as fake or real. This setup ensures that users receive immediate feedback based on the machine learning models' analysis.

Deployment and Usage of the Streamlit application offer several advantages. The application can be hosted on a web server or run locally, making it accessible from any device with internet connectivity. Users interact with the application by entering job details and receiving instant predictions, facilitating quick decision-making about job postings. The integration of Random Forest and SVM into the Streamlit application ensures that the predictions are both accurate and reliable, leveraging the strengths of each model.

Additionally, the Streamlit Integration allows for ongoing enhancements. The feedback mechanism enables users to provide input on the predictions, which can be used to fine-

tune the models and improve accuracy over time. This dynamic interaction with users ensures that the application remains relevant and effective in detecting fraudulent job postings.



The screenshot shows a web application interface for 'Fake Job Posting Prediction'. The interface is dark-themed and includes the following elements:

- Job Title:** A text input field containing 'Marketing Intern'.
- Job Description:** A text area containing a detailed description of a role at 'Food52', mentioning its growth, awards, and team structure.
- Company Profile:** A text area containing information about 'Food52', including its location in Chelsea, New York City, and its partnerships with Whole Foods Market and Random House.
- Requirements:** A text area containing a list of requirements for the job, such as experience with content management systems and familiarity with the Food52 editorial voice.
- Benefits:** A text area that is currently empty.
- Choose Model:** A dropdown menu set to 'Random Forest'.
- Predict:** A button to trigger the prediction.
- Result:** A message at the bottom stating 'This job posting is predicted to be Real.'

## 8 CONCLUSION AND FUTURE ENHANCEMENTS

In conclusion, this project successfully developed a machine learning-based system for detecting fraudulent job postings by utilizing a combination of text and numeric data. The project employed Random Forest and Support Vector Machine (SVM) models, showcasing the effectiveness of machine learning techniques in combating employment scams. The Random Forest model demonstrated superior performance compared to the SVM model, proving to be more effective in distinguishing between genuine and fake job postings. This success was further enhanced by the integration of Streamlit, which provided an interactive platform for users to receive real-time predictions. Overall, the project offers a practical solution for identifying fraudulent job postings and protecting job seekers, laying a solid foundation for future developments in this area.

Despite the project's success, there are several areas for future enhancement. One key improvement could involve incorporating more advanced natural language processing (NLP) techniques. Models such as transformers and BERT could enhance detection accuracy by better capturing nuanced patterns and understanding the context within job descriptions.

Expanding the dataset to include a broader and more diverse range of job postings is another important step. A larger dataset would improve the model's ability to generalize across various types and sources of job listings, making it more robust.

Additionally, implementing real-time detection capabilities would allow the system to continuously adapt to new and evolving scam patterns. This could be achieved through automated updates and model retraining based on new data and user feedback, ensuring the system remains effective over time.

Exploring hybrid models that combine multiple machine learning techniques or integrating ensemble methods could also offer improvements in classification accuracy. Expanding the system's reach to cover different languages and regions could further enhance its global applicability and utility.

Overall, these enhancements aim to refine the system's

## 9 Reference

[1] Journal of Engineering Sciences, K.V. Jhansi Rani, Priyanka Rompalli, Leela Nagababu Kathi, Abhiram Tholam, Ramakrishna Gudla, B.Tech, Department of CSE, Eluru College of Engineering and Technology, Duggirala, Andhra Pradesh-534004.

[2] w.ijcrt.org © 2024 IJCRT | Volume 12, Issue 4 April 2024 | ISSN: 2320-2882 Mr. B.J.M. Ravi Kumar, Mr. Melkamu Boka Eba, Mr. Ridwan Mohammed Department of Information Technology and Computer Application, Engineering College A, Andhra University, Visakhapatnam, India - 530003

[3] e-ISSN: 2582-5208 Aravind Sasidharan Pillai\*1 \*1The University of Illinois, Urbana-Champaign, IL, USA. DOI: <https://www.doi.org/10.56726/IRJMETS35202>

[4] 2021 2nd International Conference on Robotics, Electrical and Signal Processing Techniques Sultana Umme Habiba, Md. Khairul Islam, Farzana Tasnim Dept. of CSE, International Islamic University Chittagong, Dhaka, Bangladesh

[5] <https://economictimes.indiatimes.com/jobs/hr-policies-trends/job-scams-on-the-rise-as-fraudsters-target-desperate-jobseekers/increasing-job-scams/slideshow/101636401.cms?from=mdr>

[6] International Journal for Multidisciplinary Research (IJFMR) Maddi Sravya Reddy<sup>1</sup>, Maddikera Hemanth Lal<sup>2</sup>, Lingam Sainad<sup>3</sup>, Sandeep Agarwalla<sup>4</sup> Student, Department of Computer Science and Engineering, Malla Reddy College of Engineering and Technology, Hyderabad