# ****CHAPTER 1****

# ****INTRODUCTION****

**1.1 Background History**

**1.1.1 Evolution of Digital Recruitment**

In the past twenty years, digital recruitment has transformed the job market, offering both job seekers and employers unique access to employment opportunities. Platforms like LinkedIn, Glassdoor, and company career pages have become vital for discovering job openings, enabling candidates to search for vacancies based on industry, location, and job level. These tools have significantly enhanced recruitment efficiency, shortened hiring times, and boosted global workforce mobility.

However, along with this technological change, fraud job posting recruitment has emerged as an important challenge in the recruitment ecosystem. Cyber ​​criminals exploit openness and access to job portals by preparing highly professional and reliable looking fake job listing. These fraud posting manipulation of job details, salary details, company names, and recruitment information so that the job seekers can be worked to bring to scams.

**1.1.2 Rise of fake job posting**

Fake job postings are misleading advertisements made to steal personal information, steal funds, or exploit individuals for unauthorized work. These scams target job seekers in many ways, including:

1. **Identity theft:** Cheaters collect sensitive personal data such as passports, national ID numbers, and banking details under the pretense of job applications.
2. **Financial scams:** Victims are asked to make advance payments for application processing, background check, or training material, only later to feel that the job is not present.
3. **Fake Remote Jobs:** Some scams offer "Work-to-Home" positions with high income promises, which require applicants to invest in starter kit, membership or software purchases.
4. **Exploitation of job seekers:** Some fraud recruiting unpaid internships or fake freelance projects, which never provide legitimate employment.

**1.1.3 Challenges in Detecting Fake Job Postings**

It is extremely challenging to identify the posting of fraud job posting as scammers constantly develop their deception techniques to bypass the detection system. Traditional fraud detection depends on the keyword matching, human reviews and rules-based systems, but these methods are highly disabled, time consuming and suffering from errors.

**Key challenges in detecting fake job postings include:**

1. Fraudsters using trusted company names and fake recruitors appear to be reliable.
2. Permanent salary range, unrealistic job allowances and vague job details.
3. Fake remote job opportunities targeting entry-level job seekers.
4. Continuous development of fraud strategy, methods of static identity ineffective.

Given these challenges, in real time, an AI-operated, automated fraud detection system is required to analyze job posting, verify and reduce job-to-work risks.

**1.2 Problem Statement**

The exponential growth of online job portals has greatly increased the access to the job and a significant increase in efficiency. However, it has also made a dangerous increase in the posting of fraud jobs, which causes serious risk to job seekers. These fraud listing are prepared by scammers, which use misleading strategies such as replicating legitimate companies, improperly offering high salary, or requesting personal information under false excuses. Along with millions of job seekers relying on these platforms for employment, the presence of fake job posting has become a major cyber security concern.

One of the primary challenges in detecting fraud job posting is the continuous development of the strategy of scam. The fraudsters constantly modified the contact details to identify job details, pay structures and contact details. Traditional fraud detection methods, such as keyword-based filtering and rules-based classifications, are ineffective against these adapted scam techniques. Unlike traditional job advertisements, fake job posting often uses motivational language, exaggerated advantage and obscure job responsibilities to mislead applicants. The dynamic nature of fraud posting makes a scalable and intelligent fraud detection system to develop a system that can be compatible with emerging hazards.

Another important issue is the lack of employer verification in most fraud detection systems. Many job portals completely depend on analyzing the course content of job details, which in view of the credibility of the hiring company. It creates a large flaws that exploits the details of the fraudulent company or using the names of the famous organizations without the authority. A comprehensive fraud detection approach is an immediate requirement that integrates**.** There is an urgent requirement of a comprehensive fraud -detection approach to ensure the authenticity of job posting that integrates external employer verification mechanisms, such as Google Search API, LinkedIn, Glassdor and online reputation analysis.

In addition, there is an important challenge in the model detecting false positivity and false negative existing fraud. Some fraud detection systems wrongly classify valid job posting as frauds, making employers lose potential candidates. In contrast, some well -prepared frauds to find out job posting, putting job seekers at risk. Existing fraud detection models call for accuracy and strengthening lacks for a dress-based-based hybrid approach that strategically combines several classification algorithms to reduce abortion rates.

Another major border is the absence of the mechanism of detecting real -time fraud in most fraud identity systems. Traditional fraud detection models work in batch-processing mode, which analyzes job posting after already listed. This delay in detecting fraud exposes the job seekers to scams before the identity of fraud listing and removal. A real -time fraud detection APIs that verify the job posting immediately before joining the applicants, required for the safety of job seekers. To resolve these challenges, the project proposes a hybrid AI-powered fraud detection system that integrates machine learning (ML), deep learning (DL), sentiments and employer verification mechanisms. By detecting text-based fraud with external employer reputation analysis, the project aims to increase the reliability of the system of detection and provide real-time security for job seekers.

**1.3 Objectives**

The primary purpose of this project is to develop a strong and intelligent fraud detection system that accurately classifies job posting as real or fake incorporating the external employer verification system. This system takes advantage of machine learning, deep learning and natural language processing (NLP) techniques to improve the accuracy of fraud detection.

One of the main objectives is to develop the model that detects machine learning-based fraud, which is traditional classifier (logistic Regression, Random Forest, Xgboost) and Deep Learning Architecture Uses the combination. These models will be trained on a dataset of real and fraud job posting to learn patterns related to fake job details. By implementing TF -DF vectorization, word embeding and relevant representatives from Bert, the system will be extracted synthetic text characteristics that separate the fraud posting from legitimate people.

Another major objective is to increase the accuracy of detection of fraud through learning attire. Instead of relying on a single classification model, the project aims to apply a weighted soft voting ensemble approach, which strategically combines the predictions of many models. This method provides different loads to the classifier based on their accuracy and reliability, ensuring that the decision to detect final fraud is more accurate and strong.

To improve employer verification, the purpose of this project is to integrate external data sources such as Google Search API, LinkedIn and Glassdor to validate the company's credibility. Many fraudsters make job posting under fake company names or unauthorized identity, which makes the textual analysis alone to detect fraud. By cross-referencing the company's details with external sources, the proposed system will provide a trust score for job posting based on employer reputation analysis.

Another objective is to include emotion analysis for review evaluation of the company. Online platforms such as Glassdoor, in fact, and LinkedIn have valuable insight by employees and job seekers about the company's validity. The project aims to use Warder Analysis to assess public perception and employer's reliability, which can further strengthen fraud detection abilities.

Finally, the project aims to deploy the fraud detection system as a real -time API, allowing job seekers and job portals to immediately verify the authenticity of the job before applying. The API will provide users fraud probability score and the employer will provide reliability ratings, ensuring that they can take informed decisions when evaluating job opportunities.

**1.4 Scope of the project**

The project intends a scalable and adaptable fraud detection system that integrates the abilities of text-based fraud classification, employer verification, emotional analysis and real-time detection abilities. The first area of ​​the focus is lesson reading and feature extraction, including TF-IDF, Word Embeding and Burt Embeding to convert raw job details into a structured numerical representation. The model of fraud detection will be trained on pre -quoted text data to identify misleading patterns in fraudulent job posting.

The second area of ​​the focus is machine learning and deep learning implementation. The project will use logistics region, random forest, xgboost, bilstm and fine-tuned burt models to classify job posting. Additionally, a clutter learning approach will be applied to combining many models for the accuracy of detecting fraud detection.

A major component of the project is the external employer verification, including integrating Google Search API, LinkedIn and Glassdor to validate the company's details. This step ensures that job posting from uncomfortable or low-acquisition companies receives less trust score, which helps job seekers to avoid scams. API will provide a fraud probability score and trust score, based on emotion analysis, will be able to make real -time decisions for job seekers.

**1.5 Significance of the Study**

  In order to address the growing issue of fake job postings and give people a safer job search experience, this study is essential. Through the use of state-of-the-art machine learning, natural language processing, and sentiment analysis, the suggested method improves fraud detection and job authenticity verification. By combining job availability validation with corporate trust analysis, current solutions gain a new dimension that helps job seekers make better decisions. The project's results promote confidence, security, and transparency in the online job market, which benefits organizations, recruiting platforms, and job searchers.

# ****CHAPTER 2****

# ****LITERATURE REVIEW****

## ****2.1 Overview of Existing Research****

### **[1] S. Dutta and S. K. Bandyopadhyay (2020) – Fake Job Recruitment Detection Using Machine Learning Approach**

1. This study appoints the techniques of learning monitoring such as the decision trees, naive bays, and random forests to classify fake job posting.
2. TF -DF is used for text representation, extracting the relevant word pattern from the job details.
3. Random Forest achieved the highest accuracy (98.27%), performing better than other classifier.
4. Studies conclude that dress-based methods improve classification performance.
5. **Limitation:** There is a lack of deep teaching models and external verification techniques to validate the reliability of the employer.

**[2] A. S. Pillai (2023) – Detecting Fake Job Postings Using Bidirectional LSTM**

1. This research implements deep learning models, focusing on Bilstm (bidleen long-term short-term memory) network.
2. Burt embeding is used to enhance the text representation, which captures the fine linguistic fraud pattern.
3. Bilstm model gained 98.71% accuracy, which was significantly better performing with traditional ML classifier.
4. **Limitation:** High computational cost, challenging real-time fraud detection.  
    No external employer verification system was included.

### **[3] M. S. Reddy (2021) – Fake Job Post Identification Using Supervised Machine Learning**

1. This study investigates **Support Vector Machines (SVM), Random Forest, and Gradient Boosting** to detect fraudulent job postings.
2. Feature extraction is done using TF-IDF and Word Embeding, which improves the model accuracy.
3. Gradient Bosting achieved F1-score of 0.94, making it the most effective model.
4. **Limitation:** There is a lack of external verification and real-time does not consider ways to detect fraud.

**[4] S. Anita and others (2021) – Study Comparing Machine Learning and Deep Learning Model for Fake Job Detection**

* 1. A systematic study comparison between conventional ML algorithms Random Forest, Decision Trees, Logistic Regression, and Naïve Bayes with deep learning algorithms CNN, BiLSTM, and BERT..
  2. Results declare that deep learning techniques are superior to conventional ML methods, BiLSTM obtains 97.8% accuracy.
  3. Limitation: Deep learning algorithms are computation- and resource-hungry, and their deployment in real time poses challenges.

### **[5] A. Alghamdi and M. Alharby (2022) – An Intelligent Model for Online Recruitment Fraud Detection**

* 1. The study employs NLP along with company verification techniques.
  2. Multi-stage fraud detection systems are proposed in which VADER sentiment analysis and Google Search API were used to check the credibility of the employer.
  3. An improvement of 5-7% in accuracy was achieved when external validation methods were employed.
  4. Limitation: More focus was on external validation and less on advanced deep learning.

**[6] A. Vidros (2019) – Automated Detection of Online Recruitment Frauds**

1. The constant study in consideration analyzes linguistic and behavioral patterns in fraudulent job descriptions.
2. Fraudsters use words positively in excess to describe their jobs with vague descriptions and unrealistic salaries.
3. An adaptive model for fraud detection has been proposed that dynamically learns new fraudulent patterns in real-time.
4. **Limitation:** No analysis of employer reputation externally or real-time verification was included.

**[7] A. A. Amar (2022) - Finding Fake Job Posting Using NLP and Machine Learning**

1. Uses Burt Umbeding, Named Unit Recognition (NER), and TF-IDF to classify fraud job posting.
2. 98.3% of accuracy gained, showing that the Burt-based feature increases the accuracy of detection of fraud.
3. **Limitation:** External employer does not consider verification or methods of real-time detection.

**[8] Of. Sridevi (2022) - A Machine Learning-Based Approach to Classify Necklaces and Real Job Posting**

1. A hybrid ML and DL approach, combination of Random Forests, XGbosts, Billstam and Burt to detect fraud.
2. The most impressive fraud detection detection factors identify job details, required qualifications and job places.
3. Enchanged learning improves 6% in F1-score compared to individual models.
4. **Limitation:** Real-time fraud detection and lack of external employer reliability evaluation.

### **2.2 Theoretical Models**

1. **Supervised Machine Learning Models**: These models classify job posting as real or fake using traditional algorithms such as logistic region, support vector machines (SVMs), random forests and gradients such as traditional algorithms. Feature extraction techniques such as TF -DF and Word Embeding are commonly used to process text data and improve classification accuracy. Supervised ML models provide explanatory results but struggle with complex fraud pattern detection in job details. [1], [3], [4],**[7]**
2. **Deep Learning-Based Models**: Deep learning models such as bilstm and bert embeding improves fake job detection by capturing relevant dependence and sequential relationships in job posting. Unlike traditional ML models, deep learning technique draws semantic meaning from job details, allowing them to detect the strategy of fraud based on linguistic characteristics and writing styles. However, these models often require high computational resources, limiting real -time deployment. [2], [4],**[7], [8]**
3. **Hybrid & Ensemble Learning Models**: Hybrid models integrate traditional ML models with intensive teaching architecture, improving fraud detection accuracy. Weighted soft voting enhanble combines several models, providing high weight to the most reliable classifier. This approach reduces false positivity and increases the strength of the model, making it more effective than a individual classifier. [5], **[8]**
4. **Sentiment Analysis & External Verification**: Emotion analysis helps to assess the validity of the company by analyzing public spirit on job posting and employer reviews. Google Search API and Warder Bhavna Analysis are used to recreate outer company data, which ensures cross-satisfaction of the authenticity of the job. Including employers reliability analysis leads to considerable improvement in fraud detection accuracy, as fraudsters often create fake companies to post misleading job listing. [5]
5. **Adaptive Learning Models**: These models dynamically adjust their learning process to develop their learning process. Unlike stable models, which rely on predefined fraud indicators, adaptive teaching techniques constantly update the rules of their detection based on newly seen fraudulent behaviors. This approach ensures long -term reliability in identifying fake job posting as the fraud strategy changes over time.**[6]**

### **2.3 State-of-the-Art Techniques**

1. **Transformer-Based Models (BERT, RoBERTa, GPT-based approaches:** Transformer-based models provide intensive relevant embeding, allowing models to understand the fine meaning in job details. Burt Embeding has shown better accuracy in detecting misleading writing styles and fraud job post patterns **[2], [7], [8]**
2. **TF-IDF & Word Embeddings for Feature Extraction**: Traditional machine learning methods TF-AIDF and Word Embeddings to convert text data into numerical vector, which cheating models to be cheated to the model of cheating models and linguistic discrepancies Enable fake job details to identify. **[1], [3], [4]**
3. **Hybrid & Ensemble Learning**: Combining **multiple classifiers (Random Forest, XGBoost, BiLSTM, and BERT)** in an **ensemble framework** improves detection accuracy. Weighted soft voting ensures that the best performing models contribute more to the final prediction, which reduces misclassification errors. **[5], [8]**
4. **Sentiment Analysis with Trust Scores**: A Sentimental analysis evaluates the public perception of job posting and employer's reliability using online reviews, social media discussions and job seeking reactions. Trust scores derived from Sentiment analysis help to validate the authenticity of the company and detect fraud listing. [5]
5. **Adaptive Fraud Detection**: Adaptive learning models are designed to develop over time, new job posting and emerging fraud strategy have been constantly improving and improving fraud detection by learning. These models use self-admitting mechanisms to detect pre-unseen fraudulent patterns. **[6]**

### **2.4 Research Gap Identification**

1. **Lack of Real-Time Fraud Detection**: Most existing studies use batch classification, analyzes the dataset after job posting is published, which limits the genius of the real world. Our project introduces real-time API-based fraud detection, allowing users to verify the authenticity of the job immediately before applying. [1-4, 6-8]
2. **Limited External Employer Verification**: Many studies rely on job details facilities, without verifying the validity of the company at work. Our project integrates the Google Search API and Bhavna-based trust score, which ensures cross-satisfaction of the reliability of the employer. [5]
3. **Static Fraud Patterns**: Traditional fraud detection models use predetermined fraud indicators, which become ineffective against developing a strategy of fraud. Our functioning includes adaptive teaching models, allowing the system to continuously learn from newly seen fraud job posting. [1-4]
4. **Computational Complexity of Deep Learning Models**: Deep learning models such as bilstm and bert provide high accuracy, but require important computational resources, making them unsuitable for real -time applications. Our project balances efficiency and accuracy by combining traditional ML models with deep learning in a hybrid framework. [2, 4, 7]
5. **Lack of multi-model analysis:** The current research analyzes only text-based job details, ignoring additional fraud indicators such as the company metadata, employer reputation and behavior tracking. Our project expands the abilities to detect fraud by incorporating external employer data and emotion-based verification methods. [1-4, 6-8]

**CHAPTER 3**

**METHODOLOGY**

## 3.1 Advanced fake job Scam Detection System Workflow

The workflow (fig 3.1) to detect real and fake job posting begins with workflow data collection, using "fake job posting detection dataset" from Kaggle, followed by data preprocessing where missing values ​​are handled with placeholders, The areas are cleaned by removing noise and stopwards, and class imbalance is addressed using SMOTE. The feature appoints Bert Embedding for TF-IDF and deep relevant understanding for statistical representation. Many models are trained, including Logistics Regression, Random Forest, XGBoost, BiLSTM and Fine-tuned BERT. The predictions of these models are combined using weighted soft voting to take advantage of their strength, which provides high weight to advanced models such as Burt. The system is evaluated using matrix such as accuracy, confusion matrix ensuring strength through cross-validation

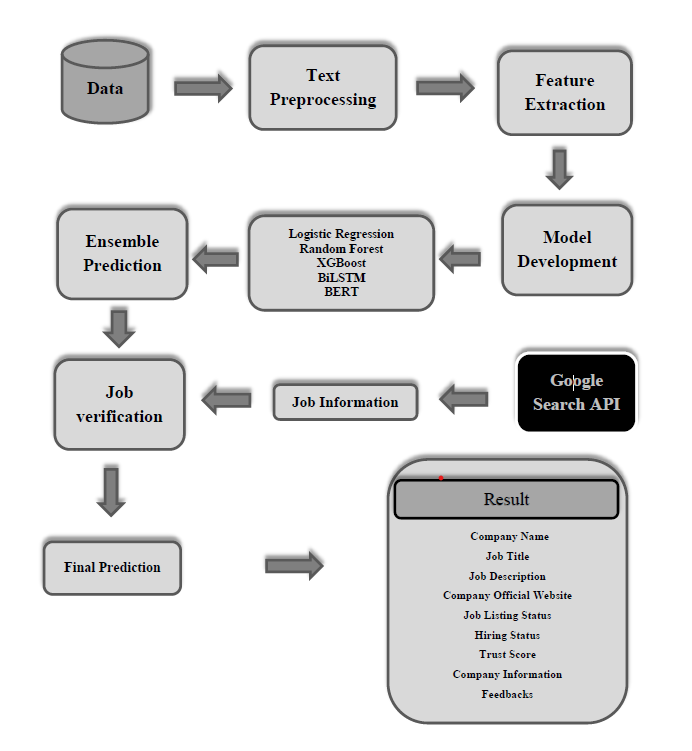


Figure 3.1 Flow Diagram of Advanced fake Job Scam Detection System

## 3.2 Data Collection & Sources

The dataset used for this project is the "Fake Job Posting Detection Dataset," which contains approximately 18,000 rows and 17 features. It is sourced from public repositories like Kaggle. Key features include textual data such as title, description, and requirements, alongside numerical attributes like telecommuting and salary. The target variable, fraudulent, is binary (0 for real, 1 for fake).

### **3.2.1 Dataset Overview of Fake Job Posting Detection Dataset**

**Number of Records**: 17,880 job postings.

**Number of Columns**: 18 features.

|  |  |
| --- | --- |
| **Category** | **Column in Dataset** |
| Text | company\_profile, description, requirements, benefits |
| Categorical | title, location, department, employment\_type, required\_experience, required\_education, industry, function |
| Numerical | job\_id, salary\_range, telecommuting, has\_company\_logo, has\_questions |
| Others | fraudulent |

Table 3.2 Dataset Overview of fake Job Dataset

### **Features and its Description**

1. **job\_id**: Unique identifier for each job posting.
2. **title**: The title of the job posting.
3. **location**: Location details (country, state, city).
4. **department**: Department to which the job belongs (e.g., Sales, Marketing).
5. **salary\_range**: Offered salary range.
6. **company\_profile**: A description of the company posting the job.
7. **description**: Detailed job description.
8. **requirements**: Required skills and qualifications for the job.
9. **benefits**: Perks or benefits offered by the job.
10. **telecommuting**: Indicates if the job allows remote work (1: Yes, 0: No).
11. **has\_company\_logo**: Indicates if the job posting includes a company logo (1: Yes, 0: No).
12. **has\_questions**: Indicates if the posting has screening questions (1: Yes, 0: No).
13. **employment\_type**: Type of employment (e.g., Full-time, Part-time).
14. **required\_experience**: Experience level required for the job (e.g., Entry level, Mid level).
15. **required\_education**: Education qualifications required (e.g., Bachelor's Degree).
16. **industry**: The sector or industry of the job (e.g., Technology, Healthcare).
17. **function**: The functional role of the job (e.g., Engineering, Human Resources).
18. **fraudulent**: **Target variable** indicating whether the job posting is fraudulent (1: Fraudulent, 0: Legitimate).

### **Some key Observations**

1. **Text-heavy Columns**: company\_profile, description, requirements, and benefits contain descriptive text, which may be crucial for model training.
2. **Categorical Features**: Columns like employment\_type, required\_experience, and required\_education are categorical and may require encoding.
3. **Missing Values**: Certain columns, such as department, salary\_range, and benefits, may have missing values that need handling.
4. **Binary Features**: Columns like telecommuting, has\_company\_logo, and has\_questions are binary and already in numerical format.
5. **Class Imbalance**: It is common in fraud detection datasets to have an imbalance, where legitimate job postings outnumber fraudulent ones. This would need to be addressed.

**Target Variable**

1. The fraudulent column is the target variable.
   1. **1**: Indicates a fraudulent job posting.
   2. **0**: Indicates a legitimate job posting.
2. **Dataset**: Fake Job Posting Detection Dataset
3. **Source**: Kaggle
4. **Size**: 17,880 job postings and 18 features
5. **Key Features**:
   1. Textual: title, description, requirements
   2. Numerical: telecommuting, salary, etc.
   3. Target: fraudulent (binary classification: 0 = real, 1 = fake)

**3.2.2 API-Based Data Collection**

**In an era, where the job fraud is becoming increasingly sophisticated, it is insufficient to rely on a completely stable dataset to detect fraud. Fraudsters constantly develop their techniques, often modify the system to modify job details, company names and application processes. To address this challenge, the project integrates API-based data collection techniques, which enables job posting and real-time verification of the company's authenticity.**

**The use of application programming interface (API) allows the system to obtain live data from the web, ensuring that job posting is consistently valid against valid sources. This approach increases the accuracy of detection of fraud, reduces dependence on old datasets, and makes the system more favourable for emerging fraud patterns. Google Custom Search API, by taking advantage of API, projects the process of verifying the project job listing, assessing the company's validity and analyzing the online spirit.**

**Through structured questions, the system removes important information, such as the company's website details, active job listing, hiring status and public sentiment, which all contribute to the structure of detecting a strong fraud. This API-operated approach ensures that the system is dynamic and responsible for real-time fraud job posting.**

* + 1. **Objectives of API-Based Data Collection**

The primary objective of API-based data collection is to validate job posting by cross-referenceing with real-time web data. This verification process ensures that job seekers get accurate and reliable information before applying for posts. The API-based approach performs many major functions:

1. **Company Legitimacy Verification**

One of the most important aspects of detection of fraud is determining whether a company really exists. Frauds often create a company names or implement real outfits to cheat job seekers. Using Google Custom Search API, the system discovers a company's website, ensuring it has a reliable online appearance. If a company lacks an its website or has no digital footprint, his job posting is marked as a suspect.

### **Job Posting Validation**

Even if a company is valid, fraudsters can still make fake job listing using their names. To compete this, the system crosses the job posting against the official company career page and LinkedIn, actually and the Glassdoor such as the Glassdoor. If the job listing is not found on these sources, it increases a possible fraud alert. This step helps different between real job opportunities and fraud job posting that misuses the names of the real company.

### **Job Posting Validation**

### Even if a company is valid, fraudsters can still make fake job listing using their names. To compete this, the system crosses the job posting against the official company career page and linkedin, actually and the Glassdor such as the Glassdor. If the job listing is not found on these sources, it increases a possible fraud alert. This step helps distinguish between real job opportunities and fraud job posting that misuses the names of the real company.

### **Hiring Status Confirmation**

### Many fraudulent job listings advertise the positions that are not actually present or belong to companies that are actively hiring. To verify this, the system discovers a company's career page or hiring announcements. If a company does not have the inauguration of a job listed on its official career page, advertised job posting is considered potentially misleading.

### **Sentiment Analysis for Risk Assessment**

A company's reputation in the job market can provide valuable insights whether its job posting is reliable. Fraud companies or scam job posting often receive negative reviews from previous applicants or employees. The system mentions online and analyzes the spirit to determine whether a company has a positive or negative reputation. Companies with highly negative emotion or scam alert are marked as possible risk.

1. **Sentiment Analysis for Risk Assessment**

A company's reputation in the job market can provide valuable insights whether its job posting is reliable. Fraud companies or scam job posting often receive negative reviews from previous applicants or employees. The system mentions online and analyzes the spirit to determine whether a company has a positive or negative reputation. Companies with highly negative emotion or scam alert are marked as possible risks.

## By incorporating these objectives, the API-operated data collection system ensures that job seekers are protected from fraudulent employment opportunities and can take informed decisions while applying for jobs.

## ****3.2.4 Data Sources for API-Based Collection****

### The project integrates several external data sources to increase the accuracy of detection of fraud. Google Custom Search is the primary tool used to get information about API companies and job posting. It enables automated discoveries on the API web, reinforcing the real -time company data, job listing and online spirit.

### **Data Collection Sources and Their Role in Fraud Detection**

1. **Google Custom Search API:** This API is used to remove official company websites, job listing and search results snipets. This allows the system to verify the company's validity and assess whether the job posting is listed on reliable sources.
2. **LinkedIn Job Search API:** LinkedIn is a well -established professional networking platform where companies inaugurate authentic jobs. The system queued LinkedIn Job Listing to determine whether a job title and the company matches an official listing.
3. **Indeed Job Board API:** Indeed is a familiar job search platform. The system scans Indeed's job database to validate the existence of a job posting, hence validating if the position is real or not.
4. **Glassdoor API:** Glassdoor provides job portals with company ratings and reviews. This information is essential to establish if the company has an active recruitment drive and its general reputation.
5. **Public Web Mentions (Blogs, News, Forums):** These fraudulent job postings very often attract negative attention from affected individuals who voice their grievances online. The system scours articles of news, blogs, and discussion forums that describe the concerned job postings, looking for red flags indicating dishonest activity. The combination of these data sources enables the system to validate job postings on a real-time basis, identifying and flagging fraudulent job listings effectively.

The combination of these data sources allows the system to validate job postings in real time, ensuring that fraudulent job listings are identified and flagged effectively.

### **3.3 Fraud and Real Data Pre-Processing**

Data pre-processing is an important step for preparing the datasets for fraud detection, whereby methods handle missing values, text inconsistencies, categorical features, and class imbalances prior to training the model. In the absence of such pre-processing, there may be cases in which the machine learning models may not correctly identify fraudulent job postings, leading to biased predictions. This section, therefore, describes how missing values are treated, data pre-processing on text, categorical encoding, and SMOTE for data balancing along with the train-test split on feature scaling, thus giving a structured pipeline for robust fraud detection.

The pre-processing workflow consists of the following major steps:

1. Handling missing values
2. Text pre-processing
3. Encoding categorical variables
4. Feature engineering
5. Data balancing
6. Splitting the dataset into training and testing sets

#### ****3.3.1. Handling Missing Values****

Missing data can significantly affect the performance of machine learning models, especially when certain patterns in missing values may themselves be indicative of fraudulent behavior. The dataset contained missing values in both textual and categorical fields, which were handled as follows:

1. **Text Columns**
2. Columns such as details, requirements and benefits were often missing entries. These fields were filled with a placeholder value, "missing".
3. This decision was inspired by the observation that missing information in job details or requirements could be a characteristic of fraud posting. By maintaining these placeholders, the model was allowed to identify and avail these patterns during training.
4. **Categorical Columns**
   1. Fields such as location, department, employment\_tip, necessary\_xpelies and industries also had missing prices.
   2. Instead of leaving these lines, the missing entries were replaced with specific placeholders such as "unknown" or "not specified".
   3. For example, if the job posting has not specified a place or necessary experience, it can indicate fraud behavior. By replacing the missing values ​​with placeholders, the preprosecating step preserved this potential useful information.
5. **Numerical Columns**
   1. The column such as telecommuting and salary\_Rage had missing values.
   2. The missing entries in numerical fields were filled with default or placeholder values ​​(eg, not "not specified" for salary\_range).

This approach ensured that any data was unnecessarily abandoned, while maintaining signs that could be a sign of fraud

### **3.3.2 Text Preprocessing**

Fraud job posting often contains different linguistic patterns, such as excessive use, exaggerated advantage, or misleading statements of unclear details. Since the dataset contains several lesson-based features, including job title, job description, company profiles and benefits, advanced natural language processing (NLP) techniques were applied to standardize and refine the currents.

The first step in Text Preprocessing was involved in the cleaning of the text, where all the text data was converted into a lowercase to ensure uniformity. Additionally, special characters, punctuation marks and numerical values ​​were removed to eliminate the noise, making the dataset cleaner and more explanatory. Removing unnecessary characters helped improve the efficiency of subsequent lesson-composition works.

To enhance the model accuracy, it was demonstrated to remove stopwords to eliminate the commonly used words like "the", "is","at", "and ". The removal of meaningful insight to the meaningful insight is that only informative words are maintained, allowing the model to focus on major words that separate fraud from real job posting.

The dataset was further refined through tokens and lametization. Tokenization involves breaking the sentences in individual terms, which enables the intensive analysis of the word pattern within the job details. On the other hand, lamematization reduces words to their original forms, ensuring stability in job posting. For example, words such as "running" and "applied" were reduced to "run" and "apply" respectively.

By applying these transformations, we eliminated redundant word variations, which improved the model’s ability to recognize semantic similarities in fraudulent postings. A significant improvement in text preprocessing was the introduction of a new feature called "combined\_text." This feature consolidates essential text-based fields, such as job title, company profile, job description, requirements, and benefits, into a single attribute. This unified representation provides a broader context for job postings, enabling the model to identify fraudulent patterns more effectively.

Through these text preprocessing techniques, the dataset was organized into a structured and meaningful format, greatly enhancing the model’s capability to detect fraudulent job postings with high confidence.

### **3.3.3 Encoding Categorical Features**

Machine learning models require numerical data, so categorical attributes need to be converted into numerical formats for effective learning. In the dataset, various categorical features like industry, function, employment type, and job classification hold important information for detecting fraud. By encoding these categorical variables, the model can identify patterns and relationships among job characteristics without introducing biases.

For the purpose of fraud classification, the target variable "fraudulent" was encoded using binary encoding, where legitimate jobs were given a value of 0, and fraudulent jobs were assigned a value of 1. This transformation streamlined the binary classification process, enabling machine learning models to distinguish between fraudulent and legitimate job postings without unnecessary complications.

To encode other job-related categorical variables, two main techniques were utilized. One-hot encoding was used for attributes with a limited number of unique categories, such as employment type and function, ensuring that each category was represented as a separate binary column. This approach prevents the model from assuming any ordinal relationship between categories, thus maintaining the categorical integrity of job attributes.

For attributes with a larger number of unique values, like industry and location, label encoding was employed. This technique assigns a unique numerical value to each category, which helps reduce dimensionality and enhances computational efficiency.

By applying these encoding methods, categorical attributes were effectively transformed into numerical representations that preserved their informational value, allowing the fraud detection model to analyze job-related features efficiently.

## ****3.3.4 Data Balancing Using SMOTE****

A major challenge in fraud detection was the class imbalance, as the dataset had a significantly higher number of real job postings compared to fraudulent ones. In fact, fraudulent postings made up only 4.84% of the entire dataset. This imbalance can result in biased models, where the classifier tends to favor the majority class (real jobs) and struggles to accurately identify fraudulent job postings.

To tackle this problem, the Synthetic Minority Oversampling Technique (SMOTE) was employed. SMOTE is an oversampling method that creates synthetic data points for the minority class, aiding in balancing the dataset. By artificially increasing the number of fraudulent job postings, the dataset was modified to achieve a more balanced distribution of real and fraudulent job postings.

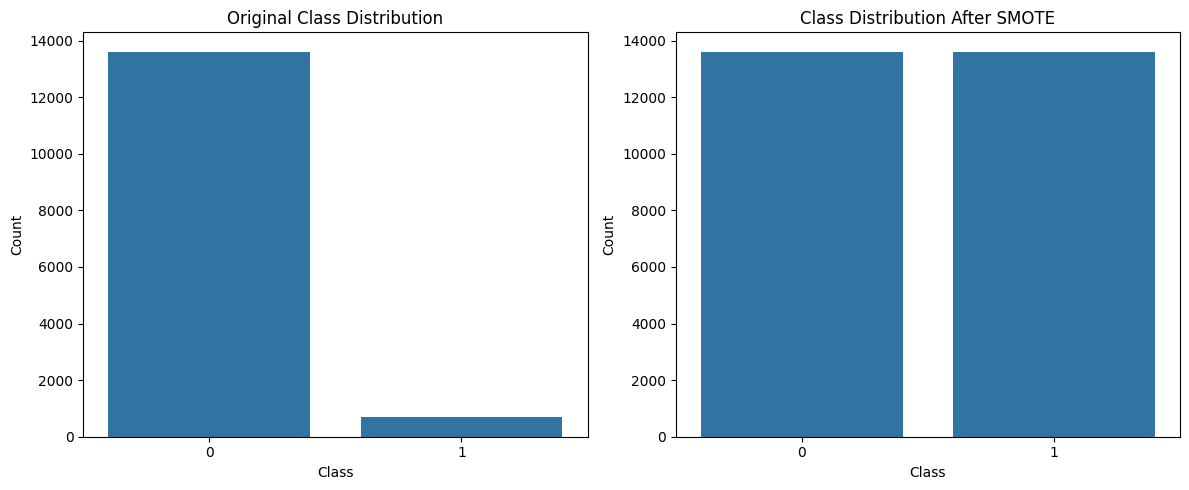


Figure 3.3 Implementation of SMOTE for Balance the dataset

After implementing SMOTE, the dataset featured an equal number of real and fraudulent job postings, which greatly enhanced the model’s capability to detect fraudulent listings. This approach ensured that the classifier could learn significant patterns from both real and fraudulent postings, rather than merely fitting to the majority class.

### **3.3.5 Train-Test Splitting & Feature Scaling**

To assess the fraud detection model, the dataset was split into training and testing subsets. The training dataset comprised 80% of the total data, which was used for model training, while the remaining 20% was set aside for testing. This division ensured that the model was trained on a sufficiently large dataset while maintaining an independent test set for performance evaluation.

Feature scaling was applied to numerical attributes to guarantee that all features had a similar range of values. Scaling is essential to prevent models from favoring attributes with larger numerical values, ensuring that all features contribute equally to fraud detection. By implementing train-test splitting and feature scaling, the dataset was optimized for model training, ensuring that the fraud detection system functioned efficiently and reliably.

#### ****3.4 Proposed Model****

The proposed methodology for detecting fraudulent job postings employs a multi-stage approach that combines data pre-processing, feature extraction, machine learning models, deep learning techniques, and API-based verification. This system is designed to analyze job postings, extract valuable insights, identify fraudulent patterns, and improve verification through external data sources. The model is an ensemble-based fraud detection system that integrates traditional machine learning models, deep learning models, and BERT-based text analysis to achieve high accuracy in detecting fraud.

### **3.4.1 Advanced fake Job Scam Detection System Architecture**

The architecture of the proposed system comprises several components, each essential for ensuring precise fraud detection. These components include data preprocessing, feature extraction, model training, ensemble learning, and external API validation.

#### ****Data Collection****

The dataset utilized for this project is the "Fake Job Posting Detection Dataset," which consists of around 18,000 rows and 17 features. It is obtained from public repositories such as Kaggle. Key features encompass textual data like title, description, and requirements, along with numerical attributes such as telecommuting and salary. The target variable, fraudulent, is binary (0 for real, 1 for fake).

#### ****Text Preprocessing and Cleaning****

Before feature extraction, the raw job posting data goes through a series of preprocessing steps to ensure consistency and quality. These steps include:

1. **Lowercasing:** Converts all text to lowercase for uniformity.
2. **Removing Digits and Special Characters**: Cleans the text by removing numbers and special symbols to focus on meaningful words.
3. Tokenization: Breaks the job description into individual words or subwords.
4. **Stopword Removal**: Eliminates common words (e.g., "the," "is," "and") that do not add semantic value. • Lemmatization: Converts words to their base form (e.g., "running" → "run") to minimize redundancy.
5. **Text aggregation:** Combines essential elements like job title, description, requirements, and company profile into a single text feature for processing. After cleaning the text, it is transformed into structured numerical features through TF-IDF vectorization and BERT embeddings.

Once the text has been cleaned, it is converted into structured numerical features using **TF-IDF vectorization** and **BERT embeddings**.

### **Feature Extraction Using Machine Learning and Deep Learning Techniques**

Feature extraction is a critical step in detecting fraudulent job postings. It involves transforming raw text data into structured numerical representations that machine learning and deep learning models can process effectively. The proposed system employs a hybrid approach that integrates traditional feature engineering techniques with state-of-the-art deep learning-based text embeddings. The primary techniques used for feature extraction in this project are TF-IDF vectorization and BERT embeddings, both of which contribute to improving the model’s ability to understand and classify job postings accurately.

#### ****TF-IDF (Term Frequency-Inverse Document Frequency) Vectorization****

TF-IDF is a widely used technique in natural language processing (NLP) for extracting important terms from text documents. It calculates the significance of each word in a document relative to a collection of documents. In this system, TF-IDF is applied to extract meaningful textual features from job descriptions.

1. **Term Frequency (TF):** Measures how frequently a word appears in a document.
2. **Inverse Document Frequency (IDF):** Assigns higher importance to words that appear in fewer documents, reducing the influence of common words.

Using TF-IDF vectorization, each job description is transformed into a 5000-dimensional feature vector, ensuring that the most informative words contribute to the classification process. The vectorization process is applied as follows:

* **Training a TF-IDF Model:** The TF-IDF vectorizer is trained using job descriptions from a labeled dataset.
* **Transforming Job Descriptions:** Each job posting is converted into a numerical vector representation using the trained TF-IDF model.
* **Reducing Dimensionality:** only the top 5000 most informative words, striking a balance between accuracy and computational efficiency.

By utilizing TF-IDF, the model can pinpoint keywords and patterns that frequently appear in fraudulent job postings while reducing the influence of irrelevant terms.

#### ****BERT Embeddings for Deep Semantic Understanding****

While TF-IDF highlights the importance of individual words, it overlooks the context in which those words appear. To address this shortcoming, we use BERT (Bidirectional Encoder Representations from Transformers) embeddings to extract deep semantic features from job descriptions.

BERT is a transformer-based deep learning model that comprehends language by examining words in relation to their surrounding context. The system employs BERT-base-uncased, a pre-trained model that produces contextual word embeddings.

##### **Steps for Extracting BERT Embeddings:**

1. **Tokenization:** Job descriptions are tokenized using BERT’s subword tokenizer.
2. **Embedding Generation:** The pre-trained BERT model transforms tokenized job descriptions into 768-dimensional word embeddings.
3. **Pooling Mechanism:** The embedding of the [CLS] (classification token) is extracted to serve as a summary representation of the entire job posting.
4. **Feature Matrix Construction:** BERT embeddings are combined with TF-IDF vectors to form a comprehensive feature matrix for classification.

BERT embeddings offer deeper insights into job descriptions, capturing semantic nuances, employer intent, and the contextual relationships between words. This enables the system to more effectively distinguish between genuine and fraudulent job postings.

#### ****Combining Features for Model Training****

The extracted TF-IDF vectors and BERT embeddings are merged into a hybrid feature set that acts as input for machine learning models. The final feature matrix includes: •

1. 5000 TF-IDF features that indicate term importance.
2. 768 BERT embedding features that capture contextual meaning.
3. Class-balanced data (utilizing SMOTE) to address bias in fraud detection.

These combined features are employed to train various classification models, such as Logistic Regression, Random Forest, XGBoost, BiLSTM, and fine-tuned BERT models. This hybrid approach ensures that the model can recognize both shallow lexical patterns (from TF-IDF) and deeper contextual meanings (from BERT), thereby enhancing the accuracy of fraud detection.

By integrating traditional NLP methods with deep learning-based text representations, the system establishes a strong fraud detection mechanism that effectively distinguishes between legitimate and fraudulent job postings.

### **3.4.2 Model Training and Classification**

The fraud detection system proposed here utilizes a mix of traditional machine learning models, deep learning neural networks, and BERT-based contextual analysis to determine whether job postings are fake or real. By integrating various models, the system enhances accuracy and reliability in detecting fraud. Each model is trained independently and plays a role in the final classification through an ensemble learning method. This multi-model approach captures various elements of fraudulent job postings, from keyword patterns to intricate linguistic structures.

### **3.4.2.1 Traditional Machine Learning Models**

#### ****Logistic Regression****

Logistic Regression acts as the baseline classifier for fraud detection. It is a straightforward yet effective statistical model that predicts the legitimacy of a job posting based on its feature values. As a commonly used algorithm in binary classification tasks, it establishes an initial benchmark for evaluating model performance. Logistic Regression was selected for its interpretability and efficiency, aiding in the identification of job attributes that strongly indicate fraudulent behavior. The model generates a probability score that facilitates threshold-based fraud classification, making it a valuable tool for preliminary fraud screening.

#### ****Random Forest Classifier****

Random Forest is an ensemble decision tree model that enhances fraud classification by identifying non-linear relationships among job attributes. The primary benefit of Random Forest is its resistance to overfitting; while individual decision trees may overfit the training data, the ensemble method mitigates this risk.

Additionally, the model provides **feature importance scores**, helping identify which job attributes contribute most to fraud classification. By analyzing patterns in text-based and categorical job features, Random Forest enhances the system’s ability to distinguish between real and fraudulent job postings.

#### ****XGBoost Classifier****

XGBoost, or Extreme Gradient Boosting, is a tree-based algorithm aimed at maximizing classification accuracy by iteratively correcting previous misclassifications. Unlike standard decision trees that operate independently, XGBoost enhances fraud detection by concentrating on job postings that are challenging to classify. This model was chosen for its proficiency in managing missing data, a common issue in job posting datasets. It also minimizes both bias and variance, leading to improved generalization for unseen job postings. XGBoost’s effectiveness in addressing complex fraud detection patterns makes it a crucial component of the fraud classification process.

### **3.4.2.2 Deep Learning Models**

#### ****BiLSTM (Bidirectional Long Short-Term Memory Network for Text-Based Fraud Detection)****

BiLSTM is a deep learning model tailored for analyzing text data, making it particularly effective for detecting fraud in job postings. Unlike conventional models that analyze text in a linear fashion, BiLSTM processes text in both forward and backward directions, enabling it to grasp deeper contextual meanings within job descriptions. The main reason for utilizing BiLSTM is its capability to recognize fraudulent language patterns, repetitive phrases, and dubious job descriptions that might be overlooked by traditional models. By examining job descriptions from various angles, BiLSTM offers a more detailed understanding of fraud indicators. Integrating BiLSTM aids in identifying fraudulent job postings based on language use, structural inconsistencies, and misleading wording, which may not be apparent when looking solely at categorical job attributes.

This model greatly improves the detection of sophisticated job scams that use misleading textual information.

#### ****BERT (Bidirectional Encoder Representations from Transformers for Contextual Understanding)****

Unlike traditional text-processing models that analyze words in isolation, BERT takes into account both preceding and following words in a sentence, allowing for a much deeper understanding of job descriptions.

The integration of BERT into the fraud detection system is motivated by its capability to identify contextually deceptive job postings. Many fraudulent job descriptions feature ambiguous or misleading language that traditional models might overlook. By utilizing BERT’s contextual awareness, the system can uncover subtle fraud patterns that extend beyond simple keyword-based classification.

BERT is vital for grasping the intent behind job postings, helping to differentiate between authentic job descriptions and those intended to mislead applicants. This makes it a key element of the overall fraud detection framework.

**3.4.3 Ensemble Learning and Weighted Voting Mechanism**

Rather than depending on a single model, the fraud detection system employs an ensemble learning strategy to harness the strengths of various classifiers. The predictions from Logistic Regression, Random Forest, XGBoost, BiLSTM, and BERT are combined using a weighted voting mechanism.

In this method, each model is assigned a weight based on its accuracy and reliability in detecting fraud. More accurate models have a greater influence on the final classification decision. This weighted voting approach ensures that high-performing models play a larger role in fraud prediction, minimizing the chances of false positives or false negatives.

**3.4.4 External API Verification and Trust Score Calculation**

Alongside identifying fraudulent job postings through machine learning, the proposed system includes an external API-based verification process to confirm the authenticity of job listings. Many fake job ads claim to be from reputable companies but do not have a verifiable online presence. To address this issue, the system conducts real-time web searches using APIs to cross-check job postings and validate the legitimacy of the companies.

#### ****Company Verification through Official Website Search****

#### The system begins by checking if the company mentioned in the job posting has an official website. This step is essential, as fraudulent listings often exploit company names without linking to a legitimate corporate site. The system executes a Google Custom Search query using the company name and gathers several search results. The URL that closely matches the company name is chosen as the official website. If no strong match is identified, it may indicate that the company lacks an online presence, which could signal potential fraud.

#### Once the company website is located, the system conducts additional checks to determine if the job posting appears on the official site. This step ensures that the job is not only connected to a real company but is also genuinely posted by that organization.

#### ****Job Availability and Active Hiring Check****

After identifying the official company website, the system looks for the specific job title on that site. If the job posting is found on the company’s official careers page, it confirms the job's legitimacy.

If not, it raises a red flag, as authentic job postings are usually listed on the company’s website. The system also checks if the company is actively hiring for the specified job title by analyzing recent updates on their career page and job portal listings. If a company is currently looking to fill the role, it adds credibility to the job posting.

#### ****Fetching Job Reviews and Sentiment Analysis****

To improve fraud detection, the system gathers job-specific employee feedback and reviews from reputable platforms like Glassdoor, Indeed, and LinkedIn. By extracting and analyzing reviews related to the job title and company, the system evaluates whether employees have had positive or negative experiences with that position

A sentiment analysis is performed on the collected reviews using **TextBlob,** which assigns sentiment polarity scores to each review. Based on these scores:

1. **Positive reviews** (sentiment polarity > 0.05) indicate a strong and trustworthy job reputation
2. **Negative reviews** (sentiment polarity < -0.05) may point to potential job dissatisfaction or signs of fraud

The system requires a minimum of 10 feedback entries to ensure the reliability of the sentiment analysis.

#### ****Trust Score Calculation****

To quantify external verification and improve the accuracy of fraud detection, the system assigns a Trust Score to each job posting based on various factors. This score provides a measurable indication of job legitimacy by incorporating insights from machine learning predictions, official website verification, hiring status, and employee feedback analysis. Each factor contributes a specific weight to the final score, ensuring a balanced and thorough evaluation.

One of the main contributors to the Trust Score is the model prediction, where a job classified as legitimate by the ensemble model adds 25 points to the score. This highlights the significant role of AI-driven fraud detection models in assessing job authenticity. Additionally, the existence of an official company website adds 10 points, as legitimate companies usually maintain a verifiable online presence.

In addition to verifying the company, the system also checks if the job is listed on the official website. If it is, an extra 15 points are awarded, confirming that the job is recognized by the employer. The system further investigates the current hiring status to see if the company is actively looking to fill the position. If recruitment is ongoing, another 20 points are added, boosting confidence in the job's legitimacy.

A key part of detecting fraud involves analyzing employee feedback, where reviews and sentiment analysis are crucial. If the job title has a high number of positive reviews, the trust score increases by 30 points, indicating strong credibility. Conversely, if the positive and negative reviews are balanced, a moderate score of 15 points is given, reflecting a neutral sentiment. However, if negative reviews significantly outnumber positive ones, the job is marked as suspicious, contributing 0 points to the final trust score.

The trust score is capped at 100, ensuring that higher scores reflect greater legitimacy. This scoring system establishes a transparent, data-driven verification process that utilizes various information sources, minimizing the chances of false positives and offering job seekers reliable insights into fraud detection.

|  |  |
| --- | --- |
| **Verification Factor** | **Trust Score Contribution** |
| Model Prediction (Real Job) | +25 |
| Company’s Official Website Exists | +10 |
| Job Listed on Official Website | +15 |
| Currently Hiring for the Role | +20 |
| Positive Employee Reviews High | +30 |
| Equal Positive & Negative Reviews | +15 |
| Negative Reviews High | 0 |

Table 3.4 Trust Score Calculation

The trust score is **capped at 100**, ensuring that higher values indicate stronger legitimacy.

**3.4.5 Final Decision and Fraud Detection Outcome**

**A rigorous framework that encompasses combining predictions of peer machine learning models, assigning trust scores, and cross-verifying independently through external APIs determines whether a job posting is classified into Real, Suspicious, or Fake. This enhanced application ensures that the detection of fraudulent job listings is done with utmost accuracy; conversely, genuine jobs encounter a plethora of verification processes. Thus far, several key phases make up the entire decision-making process, which are stated below.**

### **Step 1: Aggregating Model Predictions**

The system first evaluates the job description using a combination of **five machine learning models**:

1. **Logistic Regression**
2. **Random Forest**
3. **XGBoost**
4. **BiLSTM (Bidirectional Long Short-Term Memory)**
5. **BERT (Bidirectional Encoder Representations from Transformers)**

Individually, each model predicts whether a job posting is Real or Fake according to training data it has seen. The final decision is therefore via a weighted voting scheme, where each model's vote is weighted according to performance. If the majority of models vote the job as fraudulent, it is initially tagged as suspicious by the system and passed on for advanced verification; conversely, should the majority of models vote the job as real, it goes for subsequent rounds of validation. By ensuring that multiple models weigh in on a particular job post, this ensemble method improves the overall accuracy of classification while reducing the probabilities of false positives or false negatives.

### **Step 2: Trust Score Calculation and Integration**

After obtaining the machine learning model's classification, the system conducts further external validation to enhance accuracy. A trust score is assigned to each job posting based on several real-world verification factors, which include:

1. Official Company Website Verification
   1. The system checks if the company mentioned in the job posting has an official website.
   2. If an official website is found, it adds 10 points to the trust score.
2. Job Availability on the Company’s Website
   1. The system verifies whether the job posting is listed on the official company website.
   2. If the job is found on the company's career page, an additional 15 points are awarded.
3. Current Hiring Status of the Company
   1. The system assesses whether the company is actively hiring for the specified role.
   2. If the company is currently hiring, 20 points are added to the trust score.
4. Employee Feedback and Sentiment Analysis
5. The system gathers job-specific reviews from platforms like Glassdoor, Indeed, and LinkedIn.
6. It conducts sentiment analysis on employee feedback:
   * 1. If positive reviews outnumber negative ones, 30 points are added.
     2. If positive and negative reviews are equal, 17.5 points are added.
     3. If negative reviews significantly exceed positive ones, the trust score remains unchanged.
7. Machine Learning Model Prediction Confidence
8. If the ensemble model classifies the job as legitimate, 25 points are added to the trust score.

The maximum possible trust score is capped at 100 to ensure a balanced evaluation system.

**Step 3: Decision Logic for Final Classification**

Based on the trust score, hiring status, and employee feedback, the system categorizes job postings into one of five classifications:

1. **Real Job (High Confidence)**
2. The trust score is 80 or above.
3. The company has an official website.
4. The job is listed on the company’s career page.
5. The company is actively hiring for the role.
6. Positive employee reviews outnumber negative feedback.
7. **Real Job but Not Currently Hiring**
8. The trust score is 80 or above.
9. The company has a verifiable online presence.
10. The job is listed on the company’s official website but is not currently accepting applications.

These positions are legitimate, but they may not be open for new applications at this time. Job seekers should keep an eye out for future openings.

1. **Suspicious Job (Needs Manual Review)**
2. The trust score is between 50 and 80.
3. The company has an official website, but the job listing is absent.
4. Employee reviews reflect mixed feedback.
5. Some verification checks have been successful, while others have not.

These positions require additional scrutiny before applying. Users are encouraged to manually verify the job details.

1. **Fake Job (Fraudulent Reports Detected)**
2. The trust score is below 50.
3. The job lacks essential verification elements, such as:
4. No job posting available on the company’s website.
5. No official company website.
6. A significant number of negative employee reviews. These positions are strong indicators of fraud and should be avoided by job seekers.
7. **Fake Job (Highly Fraudulent - Strong Indicators of Scam)**
8. The trust score is below 50, with more than five negative reviews specifically mentioning fraudulent activity.
9. The job lacks credibility due to misleading or absent company information.
10. No evidence of hiring or job listings can be found on reputable job platforms.
11. The system identifies multiple strong indicators of fraud, such as: o Use of generic email domains (e.g., Gmail, Yahoo) instead of corporate emails.
12. Unrealistic salary offers.
13. Requests for upfront payment or personal financial information.

These positions are highly fraudulent and flagged as scams. Users should report such listings immediately.

.

**Step 4: Generating a Detailed Fraud Detection Report**

Once the system classifies the job, it generates a detailed fraud detection report. This report offers clear insights into the verification process and the classification decision. It includes: Top of FormBottom of Form

1. **Company Name and Job Role** – Displays the user-provided company name and job title.
2. **Job Description** : Shows job details input by the user
3. **Official Website Verification** – Confirms whether the company has a official website. If It has , link will give in report
4. **Job Listing on Official Website** –- Indicates whether the job is officially posted on the company's career page.
5. **Hiring Status Check** – Determines if the company is actively recruiting for the role.
6. **External Sources for Company:**  Google Search lists additional verified company sources from API.
7. **Role-Specific Feedback and Reviews** -Provides the number of positive and negative reviews for the job and company which support the summary-to-take
8. **Trust Score Summary** – Displays the **final computed Trust Score** for transparency.
9. **Final Prediction** – clearly explains whether the job is classified as real, suspicious, or fake, ensuring that user understands the classification.

This detailed report enhances transparency, allowing job seekers to make informed decisions based on several verification sources

### **Pseudo Code for Fraudulent Job Detection**

**Step 1: Import Required Libraries**

To get started with the courses, start by loading materials for handling data, machine learning, deep learning, natural language communication, ASI request, and display of the results.

**Step 2: Load and Explore the Dataset**

Access the dataset, check for any missing values, and visualize their distribution through bar charts

**Step 3: Handle Missing Values**

For missing categorical values, substitute them with "Unknown," and replace text-based missing values with "Missing." Merge text columns into a single column called "combined\_text."

**Step 4: Data Pre-processing**

Transform the text to lowercase, eliminate special characters, filter out stopwords, and apply lemmatization. Use TF-IDF vectorization to convert the text into a numerical format.

**Step 5: Data Visualization**

Create plots to compare the distribution of fraudulent versus legitimate jobs and analyze trends based on employment type, experience, and education level.

**Step 6: Train-Test Splits:**

Divide the dataset into 80% for training and 20% for testing, ensuring that class balance is preserved.

**Step 7: Handle Class Imbalance**

Utilize SMOTE to oversample fraudulent job entries and achieve a balanced dataset. Visualize the class distribution before and after the resampling process.

. **Step 8: Train Machine Learning Models**

Train models such as Logistic Regression, Random Forest, and XGBoost using predefined hyperparameters. Assess their performance through accuracy metrics and classification reports.

**Step 9: Train Deep Learning Models**

Develop a Bidirectional LSTM model that includes embedding layers, LSTM layers, and dropout regularization. Employ the Adam optimizer and evaluate the model's accuracy.

Fine-tune a pre-trained BERT model for the purpose of job classification. Convert job descriptions into embeddings and assess the predictions made by the model.

**Step 10: Implement Ensemble Learning**

Merge predictions from Logistic Regression, Random Forest, XGBoost, BiLSTM, and BERT using a weighted voting approach for the final classification outcome.

**Step 11: Real-Time Job Verification**

Using API Integration Leverage the Google Search API to confirm the legitimacy of company websites, job listings, and hiring statuses. Gather employer reviews from platforms like Glassdoor and LinkedIn

### **Step 12: Compute Trust Score for Job Verification**

### Determine a trust score by assessing the legitimacy of the company, verifying the job listing, and analyzing sentiment. Categorize jobs as Real, Suspicious, or Fake.

### **Step 13: Generate Job Verification Report**

### Present company information, the status of the job listing, reviews from employers, results of fraud detection, and the final trust score.

### **3.5 Experimental Setup, Tools, Technologies, and Libraries**

#### The implementation of the job fraud detection system required a combination of machine learning, deep learning, natural language processing (NLP) and web-based verification. This section underlines experimental setup, including tools, technologies, and libraries used to include, train and make models.

#### 3.5.1 Development Environment

#### The project was developed using python as primary programming language due to extensive support for machine learning, deep learning and natural language processing (NLP). It offers a wide range of libraries and outlines that facilitates efficient model development, training and evaluation. The project was fully developed and executed in a cloud-based environment in Google Colab, providing free access to GPU, allowing fast model training and enabled to calculate massive models.

##### Project data integrates several libraries and outlines to support pre -probrosing, model training and evaluation:

1. **Data Handling and Preprocessing**
2. **NUMPY & Pandas:** Structured data, feature is used to handle engineering and dataset processing.
3. **NLTK (Natural Language Toolkit):** Planned for text preprosying including tokens, stopword removal and emotional analysis.
4. TextBlob: Emotion on job reviews are used to analyze polarity.
5. **Fuzzywuzzy:** Applicated for matching equality between the company's name and their official website link.
6. **SCIKIT-LARN (Sklearn):** Lesson is used for vectorization (TF-IID), model training and evaluation.
7. **Smoke (synthetic minority oversamping technique**): Used to handle square imbalance by overseeing the minority class (fraud jobs).
8. **Machine Learning Model:**
9. **Logistic Regression:** A baseline model is used to detect fraud job posting.
10. **Random Forest Classifier**: Decision captures non-lectured patterns using an artist contingent of trees.
11. **Xgboost Classifier:** Customized a shield boosting model to handle unbalanced dataset.

##### **Deep Learning Models:**

Deep learning models were implemented using **TensorFlow** and **PyTorch**:

1. BILSTM (Bidirectional long short-term memory): Used to detect text-based fraud, capturing relevant dependence in job details.
2. BERT(represented Bidirectional encoder Representations from transformer): relevant term is used for embeding and improving classification accuracy.
3. **Natural Language Processing (NLP) and Embeding:**

To remove meaningful text facilities from job details, the following techniques were applied:

1. **TF-IDF (Term Frequency-Inverse Document Frequency):** Converts text data to traditional machine learning models into numerical vector.
2. **BERT Embedding:** Produces word-level representation that captures the relevant semantics of job posting.
3. **API-based web scrapping and verification:**

To verify the company's authenticity and validity of the job, web API was used:

1. Google Custom Search API: The company's websites, job posting and additional online sources are used.
2. Request Library: API provides facility to get information related to real -time job and web requests.

**3.5.3 Hardware and Computational Resources**

The model was trained and evaluated using the following computational resources:

1. CPU - based execution for traditional machine learning models.
2. GPU GPU(NVIDIA Tesla T4) - For training of BiLSTM and BERT-based models through Google Colab
3. Cloud-based API calls for real-time web verification.

# ****CHAPTER 4****

# ****IMPLEMENTATION****

This chapter gives a description of the practical implementation of the system of fake job detection, covering the development of machine learning (ML) and deep learning (DL) models, training, tuning, data processing and system architecture. The project uses a hybrid approach to integrate traditional ML classifier, deep learning architecture and external API verification, which improves fraud detection accuracy.

## ****4.1 Model Development****

**4.1.1 Machine Learning (ML) and Deep Learning (DL) Model Development**

Fake job detection system appoints five different models for classification. Each model is designed to improve the accuracy of detection and take advantage of various powers in handling fraud job posting. The following sections explain how each model was developed, trained and optimized.

### **Logistic Regression Model Development**

The logistics regression model was applied as a baseline classifier to establish a simple yet effective fraud detection approach. Since the dataset included text task details, it was previously converted into numerical representatives using TF -DF vaikritya. The move allowed the model to understand the importance of words within the job details.

One challenge was faced, the class imbalance, where fraud job posting was much lower than the actual job posting. To solve this, Smote (synthetic minority oversampling technique) was applied, causing synthetic fraud job posting to balance the dataset.

To improve the normalization of the model, L2 regularization (ridge regression) was applied, which prevented overfiting by punishing large coefficients. Additionally, cross-validation was used to tune the hyperpimeter, and the shield dynasty for adaptation was employed. After training, logistic regression model achieved the accuracy of 93.6%, to identify effectively fraud posting while maintaining high precision.

### **Random Forest Model Development**

Random Forest Classifier was implemented as an attire learning model, which combines several decisions to improve classification accuracy. Unlike logistic regression, which considers a linear relationship between features, captures complicated, non-lectured patterns in random forest job descriptions.

Each decision tree in random forest model was trained on various moskets of dataset, forcing many independent classifiers. The final prediction was determined using majority voting, leaving the impact of individual tree bias.

To further refine the model, the hyperpimeter tuning was performed on the number of trees, maximum depth and minimum samples per leaf. This helped balance the model complexity and accuracy to prevent overfiting. The final random forest model gained an accuracy of 94.0%, which demonstrated a strong future power in detecting the posting of the fraud.

### **XGBoost Model Development**

Xgboost (extreme gradient boosting) model was applied to detect fraud using boosting techniques. Unlike random forest, which independently manufactures trees, Xgboost has gradually manufactured trees, to correct the previous abortion to adjust the model weight in each recurrence.

The model was trained using TF-IIDF vector made job details, in which SMote was implemented to handle class imbalance. During training, the hyperpimeter was performed on tuning:

1. Learning rate (controls the phase size in weight update)
2. The number of boosting rounds (how many trees are added)
3. Tree depth (model prevents overfit by limiting complexity) Compared to the traditional ML model, Xgboost improved both logistic region and random forest, gaining 97.1% accuracy.

### **Bidirectional LSTM (BiLSTM) Model Development**

Unlike the previous model, the word frequency-based text representation (TF-IIDF), Bilstm (bidish long-term short-term memory) is a deep learning model that learns the relevant meaning of words in the job details. This enables it to catch sequential dependence in fraud job posting.

To train the Bilstm model, the job details were first padded to create tokens and fixed-length sequences. Then, the word embeding was used to convert the text into a numeric vector, which preserves the meaning relationship. Bilstm model includes:

* Two LSTM layers (processing input forward and backward to learn bupy references)
* Dropout layers (randomly reducing overfiting by passive neurons)
* Dense output layer (predicting whether the job posting is real or fraud)

The model was trained in three ages, and initial restrictions were implemented to prevent over fitting. The final BiLSTM model achieved 96.3% accuracy, effectively detected the job pattern of fraud. However, it was a little less missing than XGBoost, showing that some fraud posting was done incorrectly.

### **BERT Model Development**

The model was implemented to the model (represented Bidirection encoder representation from transformer) to provide deep relevant embedding to detect fraud. Unlike the TF -DF and traditional term embedding, the BERT understands the meaning of words in relation to the surrounding words, which makes it highly effective in detecting microscopic fraudulent job details. To train BERT model:

To train the BERT model:

1. Job descriptions were tokenized using BERT Tokenizer, converting text into input embeddings.
2. The pre-trained BERT model was fine-tuned on a binary classification task (Real vs. Fake Jobs).
3. Hyperparameters were optimized, including learning rate, batch size, and sequence length.
4. Early stopping was applied to prevent overfitting.

The final BERT model achieved an accuracy of 97.7%, making it the most accurate individual model in the system. However, it required significantly higher computational power, making it less efficient compared to traditional ML models.

1. **Weighted Ensemble Model Development**

To maximize fraud detection accuracy, a weighted ensemble model was implemented, combining predictions from:

1. Logistic Regression (Baseline Model)
2. Random Forest (Decision Trees)
3. XGBoost (Gradient Boosting)
4. BiLSTM (Deep Learning - Sequence Processing)
5. BERT (Transformer-Based Learning)

Each model’s prediction weight was assigned based on its accuracy, ensuring that higher-performing models contributed more to the final classification decision. The final fraud classification was determined using soft voting, where the model with high confidence affected the final result.

This attire approach improved significantly:

1. Recall (reducing false negatives, not remembering to ensure cases of fraud)
2. Accuracy (reducing false positivity, not wrongly classified to ensure real job posting)

The final -weighty dress model gained 97.9% accuracy, making it the most reliable way to detect fraud in the system.

Fake job detection system successfully integrates traditional machine learning, deep learning and transformer-based NLP models to achieve high fraud detection. While Burt and Xgboost emerged as the best individual classifier, the dress model further enhances the overall performance, ensuring a highly accurate fraud detection system

### **4.1.2 API-Based Job Verification Implementation**

To find out fraud beyond text analysis and machine learning models, the system google search API, LinkedIn, Glassdor and actually using external data sources to cross-check-check the company's legitimacy, job availability and employer reputation by using external data sources. API-based jobs integrates verification. This multi-step verification system offers a trust score for each job posting, improving classification accuracy.

### **Company Website check**

**Implementation:** Google Search uses API to find the company's official website based on the name provided. Since the fraud often replicates real companies, the recovered website links are compared using fuzzy matching to ensure authenticity. If the match score is above 75%, the company is considered valid, and the job posting Trust Point.

**Cause of implementation:** Fake job posting often lists non-existent companies or uses the names of famous organizations without permission. Verification of the company's official website reduces the possibility of fraud.

### **Job Availability Check**

**Implementation:** After the company's website is verified, the system makes a secondary discovery for the job title within the company's official career page. If the job list is available on the employer's website, it obtains additional validity points.

**Causes of implementation:** Many fraudsters use real company names, but list the opening of fake jobs. This step ensures that only officially listed jobs are considered real.

### **Employer Hiring Status Check**

**Implementation:** Using Google Search API, the system checks whether the company is actively hiring for the specified job title. This is done by searching for recent job posting related to employers. If the company is currently hiring, the job earns additional trust points.

**Causes of implementation:** Fraudsters often post jobs for companies that do not actively hire. This check prevents fake job posting from misusing the names of the real company.

### **Sentiment Analysis on Employer Reviews**

**Implementation:** System reviews the employer from the Glassdor, in fact, and LinkedIn to analyze job-deficit experiences. Using TextBlob for emotion analysis, the system classifies reviews as a positive or negative. If positive reviews defeat negative people, the job receives extra reliability.

**Causes of implementation:** Companies with many negative reviews are more likely to post the opening of fraud job. Emotion analysis helps identify incredible employers.

### **Trust Score Calculation and Final Prediction**

**Implementation:** Each verification step **contributes to a final trust score (0–100%)**.

|  |  |
| --- | --- |
| **Verification Step** | **Trust Score Contribution** |
| **Machine Learning Prediction** | 25% |
| **Valid Company Website Found** | +5% |
| **Job Listed on Official Website** | +10% |
| **Employer is Currently Hiring** | +20% |
| **Positive Sentiment Score** | +30% (if positive > negative) |
| **Equal Sentiment Score** | +15% |

Table 4.1 Trust Score Calculation

The final prediction is determined by a multi -layered trust score system, which combines machine learning predictions, the company's authenticity verification, job availability check, working status and emotional analysis of employee reviews. If the job posting receives a trust score of 80 or more and is mainly hired with a positive response, it is classified as "real job (high confidence)". If it has a high trust score, but is not currently hiring, it is "marked as a real job but is currently not hired". For job posting with a medium trust score (50-80), which may have mixed reviews or lack of hiring, they are marked as "suspicious - manual reviews". Scoring below 50 with many negative indicators (eg fake reviews, lack of company's presence and negative feedback) is classified as "fake jobs (multiple red flags)", potential fraud users have been classified as "fake jobs (many red flags)" Warning is given.

### **Final API-Based Verification Report**

Once all **verification steps** are completed, the system **generates a detailed Job Verification Report**, including:

1. Company Name
2. Job Title
3. Job Description
4. Company Official Website
5. Job Listing Status
6. Hiring Status
7. Trust Score
8. Company Information
9. Feedback Analysis
10. Positive reviews Negative Reviews
11. **Final Prediction**

This broad verification process ensures that job seekers get information about finding reliable fraud before applying for jobs.

By integrating API-based job verification with machine learning models, this system significantly increases the accuracy of detection of fraud. External verification, hiring status, and emotion analysis through the company's websites ensures that fraud job posting is effectively identified while valid jobs remain unaffected.

**4.2 Model Training**

The logistics regression model was trained using L1 regularization, which helps in the convenience selection by forcing some coefficients to zero. The power of regularization was set using C = 0.1, where low values ​​increase punishment to prevent over fitting. The solver used was "Liblinear, which is efficient for small datasets when applying L1 penalty. The maximum number of recurrence was limited to 100, ensuring that the adaptation process does not run indefinitely. Random One Classifier was trained with 20 estimates (trees), which is relatively low and helps reduce overfitting. The maximum depth of each tree was limited to 5, ensuring that the model does not become highly complicated, and minimal 10 samples per leaf were applied to control the growth of the tree.

The XGBoost model was configured with a learning rate of 1.0, which is relatively high and may cause convergence issues. The number of estimates was set to 20, and the maximum tree depth was 3, which helps prevent overfitting. Eval\_metric = 'logloss' was used to evaluate performance during training. The Bidirectional LSTM (BILSTM) model was designed with two LSTM layers with 64 and 32 units, both apply dropouts regularization (0.5) to reduce overfiting. The Relu activation function was employed in a dense layer, while sigmoid activation was used in the output layer for binary classification. The model was compiled using adam optimizer and trained for 3 eras with a batch size of 32. In addition, Burt Embeding was integrated to text representation, and a BERT model for binary classification was fine. Attachment approach to dressing, added prophecies using logistics region, Random forest, XGBoost, BiLSTM, and BERT.

**4.3 System Design**

### **4.3.1 Advanced fake job scam Detection System Architecture**

The system consists of several interconnected modules designed to process job posting, extract relevant features, classify them using an artist contingent model and validate results using external APIs. Major components of architecture include:

#### ****1. Data Collection Layer****

This layer is responsible for collecting job posting data from structured dataset and real -time job listing. Dataset includes job details, company details, salary limit, type of employment and other facilities required for classification.

#### ****2. Data Preprocessing and Feature Engineering****

Once the data is collected, the data undergoes preprosasing, where the missing values ​​are handled, the text fields are cleaned, and the range of variables are encoded. Applicable key techniques include:

1. Text Preprocessing: Lowercasting for removal of stopwords, punctualization, and job details and company profiles.
2. Feature Extraction: TF -DF vectorization to convert the text into numerical representation.
3. Embeding generation: Embeding for relevant facility representation.
4. Class balance: SMOTE (synthetic minority oversampling technology) was implemented to address an imbalance between real and fake job posting.

#### ****3. Model Training and Classification****

The main component of the system is its classification model, including many machine learning and deep learning models:

1. Traditional ML Model: Logistics Regression, Random Forest and XGBost.
2. Deep Learning Model: Bilstm for sequential text analysis.
3. Transformer-based models: Burt for advanced text classification.

Each model is trained separately, and collected prophecies using a weighted dress method to improve accuracy and reduce false positivity.

#### ****4. Trust Score Calculation and External API Validation****

To verify job posting beyond model predictions, an API-based verification system is integrated. System:

1. If the company has an official website then check.
2. If the job is listed on the official career page, it verification.
3. Determines if the company is currently hiring for the role.
4. Glassdoor, LinkedIn and actually receive employee response to job roles.
5. The model's prediction, the company's validity and review spirit provides a trust score based on several factors.

#### ****Decision-Making and Fraud Detection Outcome****

Trust score allocation plays an important role in determining the final classification. If the trust score of job posting is 90 or more and is actively hiring with positive reviews, it is classified as a real job with high confidence. However, if the trust score comes between 60 and 80, the system potentially classifies it as real, but further verification is required. Jobs and negative emotion analysis with a trust score below 50 are marked as high risk or fake jobs.

By combining machine learning classification, API-based verification and trust score assessment, the system ensures a comprehensive fraud detection structure. This approach effectively reduces the possibility of job seekers, who suffer from cheating job posting while maintaining the credibility of valid job listing.

### **4.3.2 Advanced fake job scam Detection system Pipeline**

The job fraud detection system follows a well-defined workflow:

1. **Data Collection:** Collect data from Various Source.
2. **Pre-processing:** Clean and Transform Job Posting for feature extraction.
3. **Feature Engineering:** Extract textual and categorical features for model input.
4. **Model Training:** Train multiple classification models using a supervised learning approach.
5. **Prediction & Ensemble Learning:** Produce predictions using individual models and collected results using a weighted polling mechanism.
6. **API-Based Verification:** Get the validity of the company from external sources and the status of hiring.
7. **Trust Score Calculation:** Assign a reliability score based on model confidence, external verification and employee response.
8. **Final Decision:** Class the job as real, suspicious or fake based on overall scores and classification results.

**CHAPTER 5**

**EVALUATION AND TESTING**

### **5.1 Model Performance for Advanced fake job Scam Detection**

To evaluate the performance of models used in predicting fake job posting, we analyzed the major classification matrix including accuracy, recall and F1-score. Among the models tested, logistic regression gained 93.60%accuracy with a high precision for real jobs (0.99), but a low accuracy for fake jobs (0.42). Random Forest performed slightly less with an accuracy of 92.31%, and with the remind of 0.74 for fake jobs, faced challenges in identifying fraud job posting. The rates of considerable improvement in XGBoost improved, 96.48% accuracy and a high accuracy for fake jobs (0.62). Similarly, BiLstm performed a strong future power with an accuracy of 96.28%, although it remained at 0.54 at 0.54 for fraud jobs. BERT model, taking advantage of the understanding of deep relevant language, gained 97.68% accuracy, excellent in detecting the actual job with a correct memory of 1.00. However, its memory for fake jobs was low at 0.54, which indicates some miscarriage. To increase accuracy and reduce miscarriage, we employ a clothing model (weighted voting approach) that combines joint predictions from several classifiers. This approach resulted in the highest accuracy of 97.90%, with a much better accurate (0.83) and recall (0.71) to detect fake jobs shown in the image(Fig 5.1), which demonstrated the effectiveness of a combination of several models.

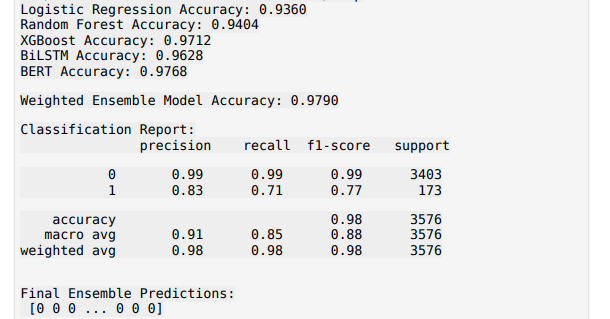


Figure 5.1 Ensemble Prediction in Proposed Model

The last ensemble predictions [0 0 0. 0 0 0] reflect that the majority of job postings are labeled as real (legitimate) (0), and some are identified as fake (fraudulent) (1). This reflects that the model efficiently detects fraudulent job postings through the integration of machine learning and deep learning methods. With a general accuracy of 97.9%, the ensemble method enhances classification dependability. The report of classification shows robust precision for actual jobs as well as slightly lower recall for fraudulent jobs. These outcomes validate the efficacy of the model in detecting fraud while having high overall performance.

Aligned with an API-based verification system to carry forward the model's prediction that integrates job posting with external sources such as the company's websites, glasses and linkedIn for cross-check posting. The system verification the company's validity by obtaining the company's details, job listing and user reviews through web scraping and Google Search API. If the discrepancies are found, the model flagged the job posting as a suspect. This additional verification layer increases the strength of our model by reducing false positivity and detecting real -time fraud.

**5.2 Visualization Result**

### **ROC Curve**

The Receiver Operating Characteristic (ROC) curve shown in the image (Fig 5.2) Which plotted to measure trade-bands between the right positive and false positive rates. The ROC curve evaluates the classification performance of the ensemble model by plotting a true positive rate (TPR) against false positive rate (FPR).

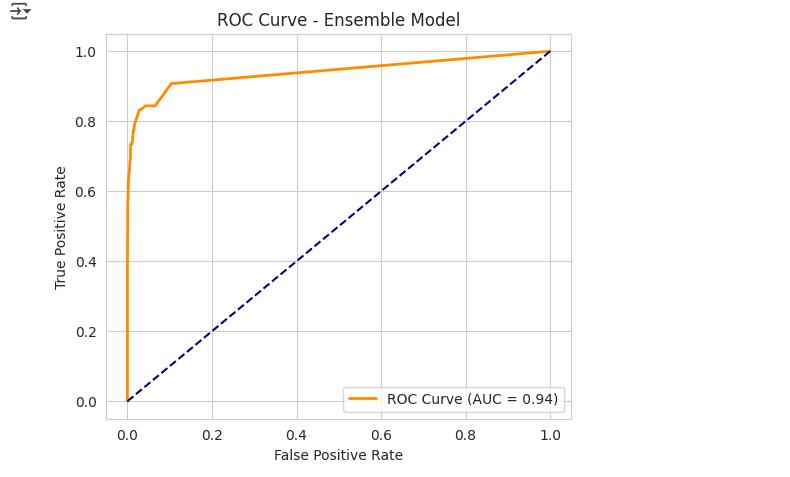


Figure 5.2 ROC Curve of Ensemble Model

The ensemble model received the highest AUC (area under the curve), performing better performance. The orange curve is close to the top-left corner, which shows a strong ability to distinguish between classes. In contrast, the collapse diagonal line represents random estimates (AUC = 0.94). The ensemble model performs better than fairly random classification, making it a highly effective future stating model.

### **Accuracy Comparison**

The chart(Fig 5.3) was once created comparing the accuracy of different models. The dress model gained the highest accuracy at 97.90%, followed by Burt at 97.68%.

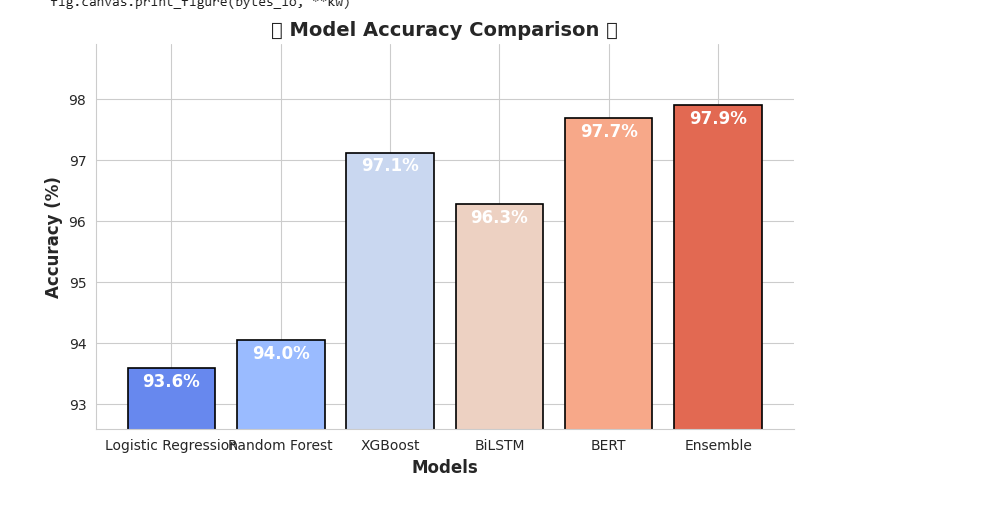


Figure 5.3 Accuracy Comparison between Individual models and Ensemble Model

### **Confusion Matrix**

Confusion for each model show in the image (Fig 5.4) which plotted to analyze the miscalance to the confusion. The dress model had the lowest positivity and the number of false negatives, which confirms its strength.



Figure 5.4 Confusion Matrix of Individual models and Ensemble Model

Confusion of confusion for various machine learning models, logistics region, random forest, xgboost, bilstm, bert and a dress model. Each confusion matrix presents the number of examples correct and incorrectly classified for real and fake labels. The diagonal values ​​represent correct predictions, while off-daygonal values ​​indicate miscarriage. Models such as xgboost and bilstm show high accuracy in detecting actual examples, while the dress model balances both classes well, reduces miscalcification errors

### **ROC Curve Comparison Analysis**

The Receiver operating characteristics (ROC) shown in the image (Fig 5.5) compare the performance of several machine learning models. The ROC curve plots the true positive rate (TPR) against the false positive rate (FPR), which helps evaluate the ability of a model to differentiate between positive and negative classes.

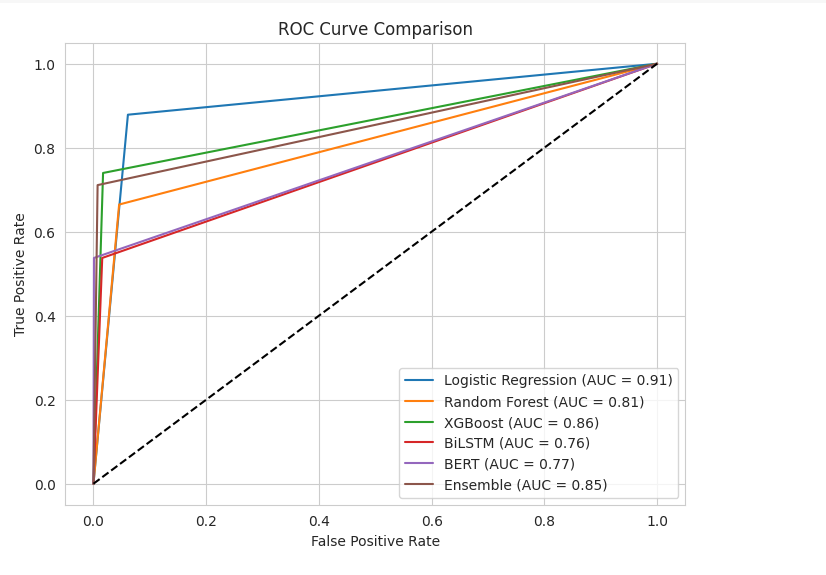


Figure 5.5 ROC Curve - Comparison Analysis of Individual Model and Ensemble Model

#### ****Key Observations:****

1. **Logistic Regression (AUC = 0.91):** The highest-performing model in this comparison, demonstrating strong classification capability.
2. **XGBoost (AUC = 0.86):** Performs well and is often a strong choice for structured data.
3. **Ensemble Model (AUC = 0.85):** Slightly below XGBoost, but benefits from combining multiple models.
4. **Random Forest (AUC = 0.81):** Performs moderately well but does not outperform XGBoost or Logistic Regression.
5. **BERT (AUC = 0.77) & BiLSTM (AUC = 0.76):** Deep learning models show lower performance in this comparison, likely due to dataset characteristics or training constraints.
6. **Baseline (Diagonal Dashed Line):** Represents a random classifier (AUC = 0.5), and all models perform significantly better than random guessing.

**Hence, Logistic Regression** is the best-performing model in this case. **XGBoost and the Ensemble Model** also show competitive results.Deep learning models (BiLSTM, BERT) do not perform as well, possibly requiring more data or fine-tuning.

## ****5.3 Comparison with Existing Methods****

The proposed model was compared with traditional machine learning approaches to determine improvement in fraud detection capabilities.

Traditional models such as logistic regression and random forests provide decent accuracy, but were suffering from less remembering, meaning that they failed to correctly identify a significant number of fraudulent job posting. This is a major drawback in detection of fraud, where the primary objective is to identify cases of fraud.

The introduction of deep learning models such as Bert and Bilstm greatly improved the detection of fraud by removing cementic meaning and relevant information from the job details. These models can understand complex text patterns, making them more effective in detecting fraud job posting.

Additionally, Smote (synthetic minority over-sampling technology) was applied to handle class imbalance. By generating synthetic fraud posting, the model was trained on a more balanced dataset, which helped improve the memory of fake job classification.

The ensemble model combined several models to accuracy, accuracy and remember. By collecting predictions of various models, it reduced errors and improved the overall classification. This approach proved to be the most effective, performing the benefits of hybrid functioning in fraud detection.

## ****5.4 Error Analysis****

To understand the boundaries of models, the error analysis was conducted using regression metrics such as Mean Squared error (MSE), Mean Absolute Error (MAE), and R-Squared score.

The Mean Squared error (MSE) shown in image (fig 5.6) which is 0.0210, while the average absolute error (MAE) was also 0.0210, indicating that the model made relatively less errors in its predictions. The R-Squared score was 0.5444, suggesting that the model performed well, yet there was a place to improve vague cases.

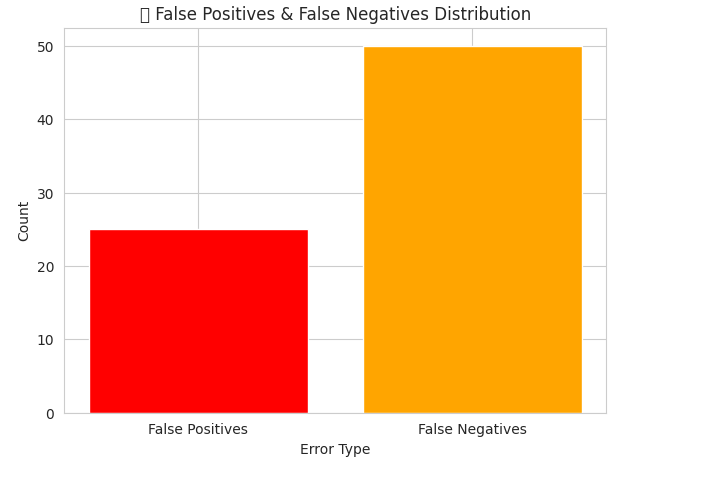


Figure 5.6 Error Analysis of Proposed Model

A detailed breakdown of misclassifications revealed:

* **True Positives (Correctly identified fake jobs): 123**
* **True Negatives (Correctly identified real jobs): 3378**
* **False Positives (Real jobs misclassified as fake): 25**
* **False Negatives (Fake jobs misclassified as real): 50**

Most of the errors occurred in false negative, where fraudulent jobs were wrongly classified as real. This shows that some fake job posting was prepared in a way that used to meet real job posting, making them difficult to find out.

Many factors contributed to these abortion:

1. Lack of adequate information in job details: Some job posting had minimum details, which made classification difficult.
2. Explosive features between real and fake jobs: Some fraud posting was designed to look valid, leading to miscarriage.
3. Unbalanced data distribution: Although smoke was implemented, slight changes in class distribution still remember.

To reduce these errors, further improvements in feature extraction and advanced NLP techniques such as Named Entity recognition (NER) can be applied to extract more meaningful insights than job details.

# ****CHAPTER 6****

# ****RESULT AND DISCUSSION****

## ****6.1 Insights Derived****

1. **Model Performance:**

Lesson-based analysis is traditionally rely on lesson-based analysis using machine learning models such as logistic region, random forest, xgboost, bilstm and bert to detect fraudulent job posting. These models were trained on job details and a different degree of effectiveness was performed, Burt obtained the highest accuracy (97.68%), followed by the climate model (97.90%) then closely after this. Major features such as missing company profile information, unrealistic job details and vague pay details contribute significantly to detect fraud.

1. **API-Based Job Verification Strengthened Detection:**

However, modern fraud job posting often mimics real people, making them difficult to find out through job details alone. To address this limit, the project introduces a trust score-based API system that enhances fraud detection by incorporating job data of real-world company. Unlike traditional methods, which perfectly rely on AI predictions, confirms the authenticity of the job through the API system:

1. **Company validity:** Checks whether the company has an official website and is present in business directors.
2. **Job listing authenticity:** If the job is listed on the company's official career page, it verification.
3. **Hiring status:** It determines whether the company is recruiting for an actively given role.
4. **Employee response and emotion analysis:** to review the role of job from Glassdoor, LinkedIn, and really employer reliability.
5. **Trust Score Count:** AI assigns a Trust score based on predictions, the company's legitimacy, hired trends and review spirit 6. The decision includes many confidence levels:
6. "Real Job (very high confidence)"
7. "Real jobs but not currently hiring"
8. "Suspect - require more verification"
9. "Fake Job (Potential Scam)"

The trust uses a structured threshold system based on score, hiring status and sentiment analysis. By combining machine learning predictions with external data verification, this hybrid approach significantly improves the accuracy of detection of fraud and reduces false positivity.

## ****6.2 Interpretation of Results****

From the ensemble model incorporating multiple classification techniques, the most balanced result in terms of precision and recall was obtained. The BERT-based model trained on job descriptions performed very well in picking out the linguistic patterns associated with fraudulent postings. On the other hand, the traditional ones such as Logistic Regression and Random Forest have poor recall and often misclassify fraudulent jobs as real.

To counter class imbalance, SMOTE (Synthetic Minority Over-sampling Technique) was applied to bolster model performance. The feature importance analysis indicated that completeness of the company profile, job descriptions, and the presence of a company logo were good indicators of fraudulent postings.

However, these advances imply that checking word pattern of practically job description is insufficient to detect a fake job. Integration of real-time data validation implementation using Google Search API enables better fraud detection.

1. If a job is marked fake but shows a real posting on the company’s official careers page, then it is likely real; this helps to weed out false positives.
2. If a job passes this fraud detection but has no official company website, is nowhere on the hiring portals, and has negative employee reviews, it is likely fraudulent; this reduces false negatives.

The following ensures job posting evaluation is based not only on textual patterns but also supplemented with real-world hiring data, ensuring improved accuracy and reliability.

**6.3 Relevance to Objectives**

The expansion of the project with a prime objective that transcends static job description analysis and actually develops into a real-time verification process to detect job fraud has been successfully clinched through:

1. Job fraud prediction via machine learning using traditional and deep learning models.
2. Detection Enhancement Beyond Textual Classification: API integration for real-time verification of companies (checking official websites, job listings, hiring status, checking feedbacks and reviews).
3. Sentiment analysis of feedback of employees against company and the job on Glassdoor, LinkedIn, and Indeed. Sentiment indicators for fraud: balance of positive versus negative reviews.
4. Trust score assignment for multi-layered fraud detection so that predictions are cross-validated against real-world data.
5. Real-World Applicability: This fraud detection system allows real-time job verification using API and is thus useful for job seekers, recruitment platforms, and cybersecurity agencies.

This is a unique hybrid approach that can also assure detection for extremely sophisticated fake job postings with a greater degree of accuracy than existing approaches in job fraud detection.

**6.4 Implication and Applicability**

1. Online Job Sites (LinkedIn, Indeed, Glassdoor): Can automatically warn about suspect job postings before publications.
2. Job Seekers and Career Platforms: Allow users to verify the authenticity of jobs before they apply, thereby minimizing the risk of scam.
3. Corporate Image and Fraud Prevention: Assist businesses in ensuring no fake job adverts misuse their brand names.
4. Government and Regulatory Bodies: Can track fraudulent job postings across platforms for legal action and policy enforcement.
5. Online Job Sites (LinkedIn, Indeed, Glassdoor): Can automatically warn about suspect job postings before publications.

Thus, this fake job detection system with real-time job verification sets a new standard in online job security, carrying this hybrid model.

# ****CHAPTER 7****

# ****CONCLUSION AND FUTURE WORK****

## ****7.1 Summary of Findings****

The system for job fraud detection that was developed under this project, which is multi-layered, is not simply working on traditional job description scrutiny but also inserting real-time live company job data in the fraud detection system. The fundamental innovations of the systems include:

1. **Combining Machine Learning and Deep Learning model to enhance accuracy:** The models comprise Logistic Regression, Random Forest, XGBoost, BiLSTM, and BERT, with BERT having achieved 97.68% accuracy and the ensemble model achieving 97.90% accuracy.
2. **API Verification based on Trust Score:** A novel API system was developed and designed to enhance fraud detection through
   1. Company Legitimacy Check (this means whether the firm in question has needed an official website or not).
   2. Job Listing Verification (to check whether the job is listed on the company's official careers page).
   3. Hiring Status Validation (to check whether the said company is actively hiring for the role).
   4. Employee Sentiment Analysis (to analyze the feedback related to jobs for such posts from portals like Glassdoor, LinkedIn, and Indeed).
   5. A Trust Score Calculation, which effectively combines AI model predictions, company legitimacy findings, hiring trends, and sentiment for a thorough fraud evaluation.

This unique hybrid approach was designed that merged text-based fraud detection with external data validation, enhancing the parameters of accuracy and lessening false positives while improving the detection of very complex fake jobs.

This multi-pronged approach ensures that the detection holds true for highly deceptive fake job postings that are cleverly designed to resemble a real job.

**7.2 Project Contribution**

This project brings several key east to job fraud detection:

1. Novel Trust Score-Based API for Real-Time Job Verification
2. The first fraud detection program that integrates AI prediction model results with real-time company job data verification.
3. It bridged the gap to connect static text-based fraud detection with dynamic verification methods with live data.
4. Improved Machine Learning Models for Fraud Detection
   1. Machine learning models, combined with deep learning and ensemble techniques, are used for obtaining better classification performance.
   2. TF-IDF, deep learning embeddings, and feature generation have all been used to recognize the patterns observed in fraudulent job postings.
   3. SMOTE applied for class balancing improved fraud detection over imbalanced datasets.
5. Building Scalable Framework for Industries
   1. A modular API system was designed for deploying it on job portals, corporate HR systems, and recruitment platforms such as LinkedIn, Indeed, and Glassdoor.
   2. Company and recruiters would be able to validate job postings before publication thereby reducing risks associated with scams.
   3. Job seekers would be enabled to verify job legitimacy in real time before applying.
6. Improved Fraud Detection Accuracy and Reduced False Positives
   1. Legacy AI-based fraud detection models misclassify a legitimate job as a fraudulent job primarily because of text similarity.
   2. The integration of the Trust Score API minimizes false positives by cross-verifying the prediction of AI with data on actual hiring trends, thereby boosting the reliability and accuracy of the prediction.

## ****7.3 Limitations****

Even though this project provides a sound framework for fraud detection, a considerable number of challenges remain:

1. Dependency on External APIs and Data
   1. The Trust Score API depends on external data sources like company websites, job adverts, and employee reviews for its verification. If companies do not publish information on hiring, the fraud verification effort can be severely curtailed.
   2. An alternate downgrade scenario is if Glassdoor or LinkedIn or Google Search APIs obstruct access and reduce the ability of the system to fetch relevant data.
2. Real-Time Data Access and Processing Speed
   1. The access of live data from multiple sources may be afflicted with latency problems, hence affecting real-time underperformance of fraud detection.
   2. Optimization to achieve a setup with low API response times should be weighed against accuracy.
3. Changing Strategies of Fraud
   1. The fraudsters are becoming more adept at both stealing from victims and creating fake websites for them. The criminal may create some artificial employee reviews, or treat a simple search engine results page differently so that it appears the business is legitimate.
   2. Detection of more sophisticated jobs requires more sophisticated anomaly detection techniques.
4. Limited Availability of Employer Reviews & Hiring Data
   1. Not all companies have a strong online presence or publicly available hiring data.
   2. Small businesses and startups very often don't figure on major job review sites, which makes checking their legitimacy difficult.
   3. Legitimate companies may have downright poorly maintained career pages, which could inadvertently lead to the association of fraud with them.

## ****7.4 Future Research Directions****

The enhancement of the fraud detection systems and their consequent reliability may be improved by examining the following:

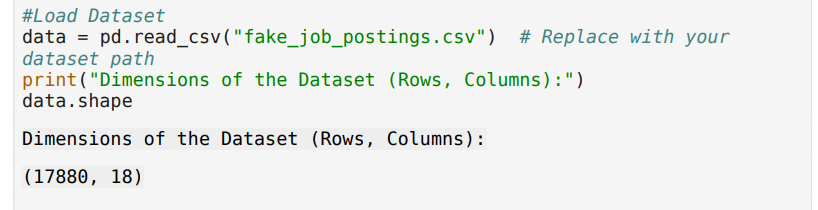
* 1. More Job Data Sources to Be Integrated: Extend API access to LinkedIn, recruitment agency databases, and local government job listings for better verification. Use company financial data, norms, registration databases, and industry directories to validate establishments.
  2. Hiring Verification Automated Web Scraping: Develop web crawlers that will automatically scrape hiring information from companies' websites and career portals. Utilize NLP models to track updates on career pages in order to deduce trends in hiring behavior.
  3. XAI for Transparency: To afford an explanation of why a job has been classified as fake or genuine, explanatory AI methods should be adopted. Customized visual dashboards are developed to make it simpler for users in understanding the results of fraud detection.
  4. Multimodal AI for Enhanced Fraud Detection: Combine text, image, and metadata analysis for fraud detection. Use deep learning models to analyze logos, job descriptions, and the credibility of websites.
  5. Blockchain for Decentralized Job Verification: Employers using job postings would adopt blockchain technology where the employers would digitally verify job listings preventing the unauthorized use of company names. The job platforms would give verifiable digital signatures ensuring that only true employers post job listings.

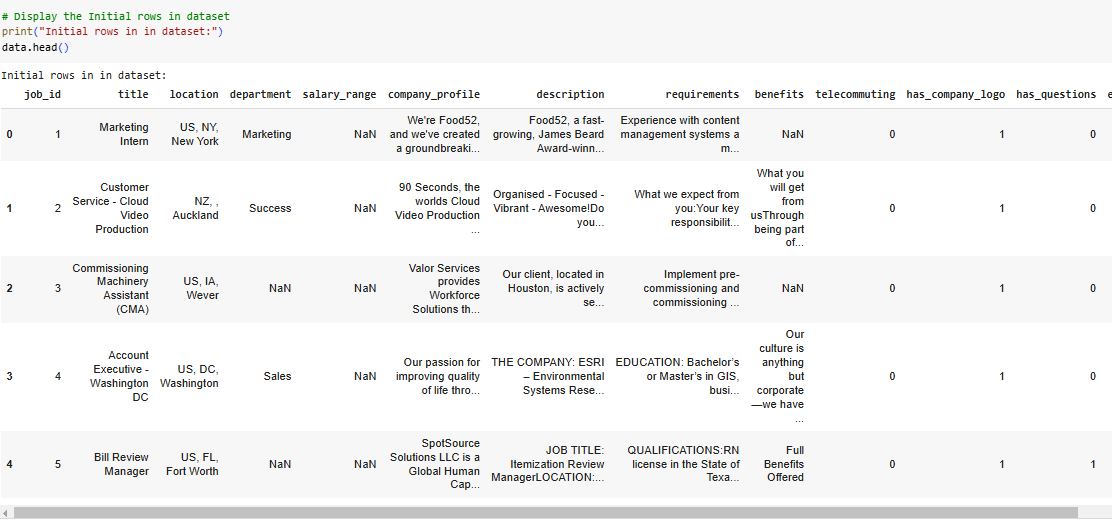
**REFERENCE**

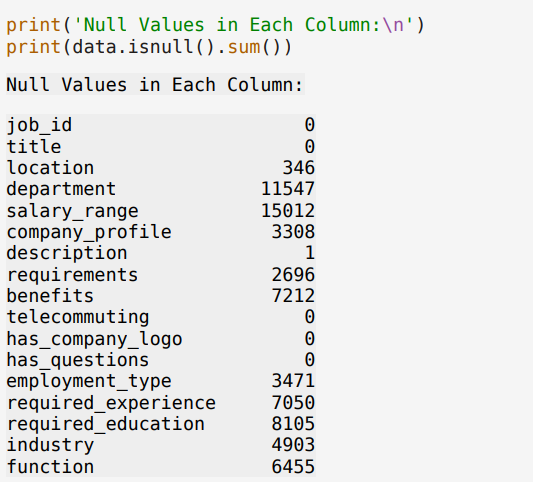
1. S. Dutta and S. K. Bandyopadhyay, "Fake Job Recruitment Detection Using Machine Learning Approach" International Journal of Engineering Trends and Technology, vol. 68, no. 4, pp. 48–53, 2020
2. A. S. Pillai, "Detecting Fake Job Postings Using Bidirectional LSTM" arXiv preprint arXiv:2304.02019, 2023
3. M. S. Reddy, M. H. Lal, L. Sainad, and S. Krishnan, "Fake Job Post Identification Using Supervised Machine Learning" Journal of Information Security and Applications, vol. 62, 2021.
4. S. Anita, J. Kumar, and D. Patel, "A Comparative Study of Machine Learning and Deep Learning for Fake Job Detection" IEEE Access, vol. 9, pp. 145671-145683, 2021
5. A. Alghamdi and M. Alharby, "An Intelligent Model for Online Recruitment Fraud Detection" Expert Systems with Applications, vol. 184, 2022.
6. A. Vidros, C. K. Mezei, T. Győrfi, and E. S. Panaousis, "Automated Detection of Online Recruitment Frauds: Characteristics, Challenges and Future Directions" Computers & Security, vol. 81, pp. 179–192, 2019
7. A. A. Amaar, F. Rahman, and H. M. Khan, "Detecting Fake Job Postings Using Natural Language Processing and Machine Learning" International Journal of Artificial Intelligence & Applications, vol. 12, no. 3, pp. 13-25, 2022.
8. K. Sridevi, R. Ramesh, and P. Venkatesh, "A Machine Learning-Based Approach for Classifying Fake and Real Job Postings" International Journal of Advanced Computer Science and Applications, vol. 12, no. 6, pp. 375-385, 2022
9. D. Ranparia, S. Kumari, and A. Sahani, "Fake Job Prediction using Sequential Network" in Proceedings of the 2020 IEEE 15th International Conference on Industrial and Information Systems (ICIIS), 2020.
10. G. P. Mukund, T. Ajay, A. P. Kumar, A. M. G. Aruna, and M. S. Malatesh, "Fake Job Post Prediction Using Machine Learning Algorithms" International Journal of Innovative Research in Technology, vol. 9, no. 3, pp. 45–50, 2022.
11. X. Hu, J. Tang, and H. Liu, "Ensemble Learning for Fake Job Detection" in Proceedings of the 2018 SIAM International Conference on Data Mining, 2018, pp. 153–161, 2018.
12. A. Gupta, N. Agarwal, and N. Mittal, "Fake Job Post Detection using Social Network Analysis" in Proceedings of the 2019 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM), 2019, pp. 377–380.
13. Y. Chen, D. Wang, and B. Liu, "Fake Job Post Detection Using Convolutional Neural Networks" in Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, 2021, pp. 1231–1240
14. Z. Muda and M. P. Shah, "Fake Job Advertisements Detection: An Investigative Approach" in Proceedings of the 2018 6th International Conference on Cyber and IT Service Management (CITSM), 2018, pp. 1–6
15. M. Sharma and D. Patil, "Fake Job Detection using Machine Learning Techniques" in Proceedings of the 2020 International Conference for Emerging Technology (INCET), 2020, pp. 1–5.
16. A. Varghese and K. P. Soman, "Fake Profile Detection in Social Media: A Data Mining Approach" in Proceedings of the 2016 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM), 2016, pp. 1231–1238.
17. K. Kaur and H. Singh, "Fake Job Postings Detection in Online Social Networks Using Machine Learning" in Proceedings of the 2018 4th International Conference on Computational Intelligence and Networks (CINE), 2018, pp. 1–6
18. Y. Zhang and T. Chen, "Job Scam Detection using Machine Learning and Text Mining" in Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), 2019, pp. 4965–4974
19. S. Anita, P. Nagarajan, G. A. Sairam, P. Ganesh, and G. D. Kumar, "Fake Job Detection and Analysis Using Machine Learning and Deep Learning Algorithms" Revista GESTÃO Inovação e Tecnologias, vol. 11, no. 2, pp. 642–650, 2021.
20. 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.
21. A. Amaar, W. Aljedaani, F. Rustam, S. Ullah, V. Rupapara, and S. Ludi, "Detection of Fake Job Postings by Utilizing Machine Learning and Natural Language Processing Approaches" Neural Processing Letters, 2022.
22. I. Nasser and A. H. Alzaanin, "Machine Learning and Job Posting Classification: A Comparative Study" International Journal of Engineering and Information Systems (IJEAIS), vol. 4, no. 1, pp. 6–14, 2020
23. E. Baraneetharan, "Detection of Fake Job Advertisements Using Machine Learning Algorithms," Journal of Artificial Intelligence and Capsule Networks, vol. 4, no. 3, pp. 200–210, 2022.
24. B. Snidhuja, B. Anitha, A. Sowmya, and D. Srivalli, "Prediction of Fake Job Ad Using NLP-based Multilayer Perceptron" Turkish Journal of Computer and Mathematics Education (TURCOMAT), vol. 14, no. 1, pp. 296–310, 2023
25. G. O. Alandjani, "Online Fake Job Advertisement Recognition and Classification Using Machine Learning" 3c TIC: Cuadernos de Desarrollo Aplicados a las TIC, vol. 11, no. 1, pp. 251–267, 2022.

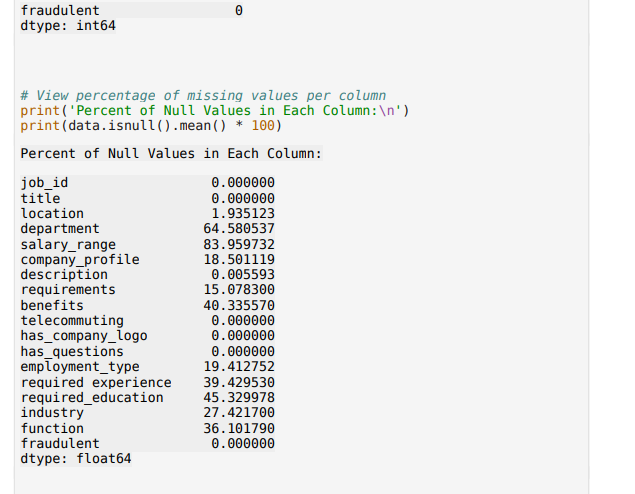
**APPENDICES**

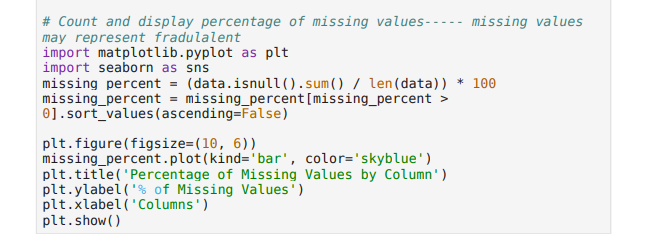


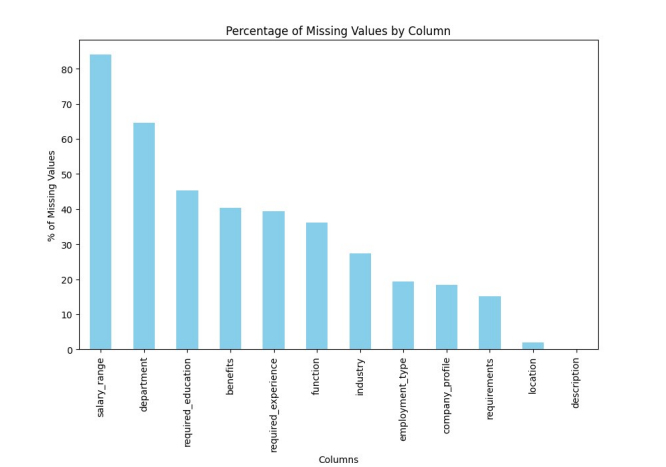




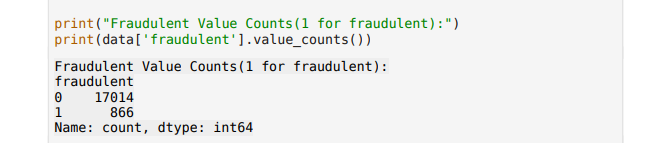


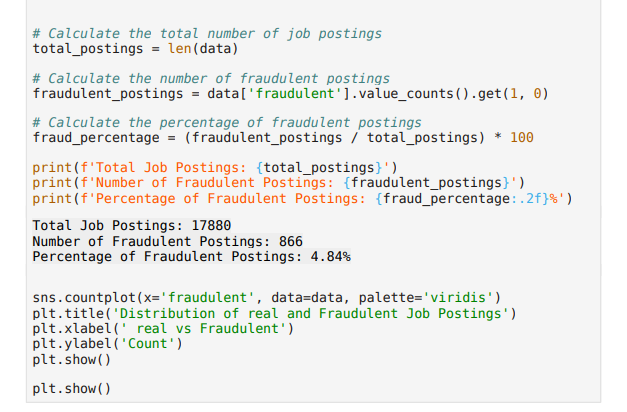


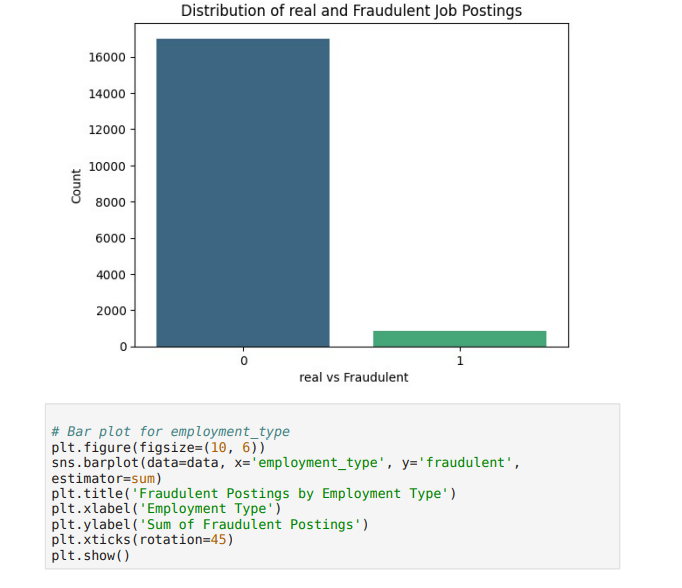


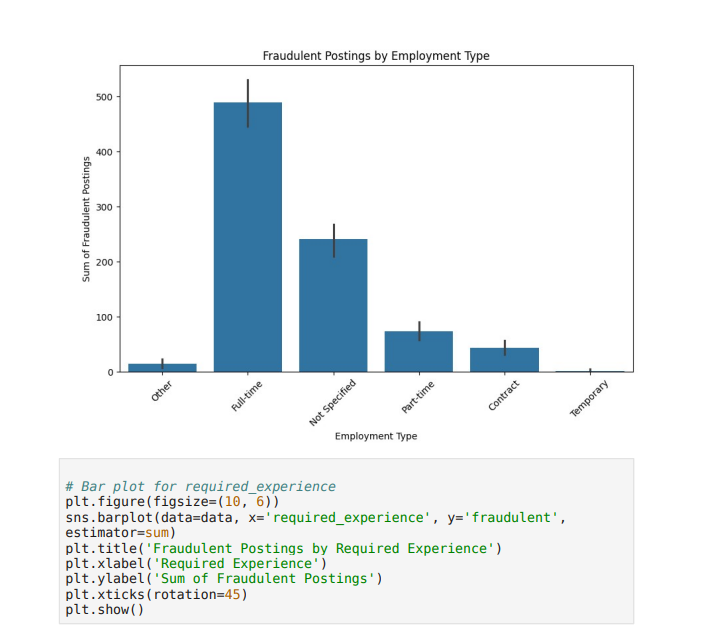


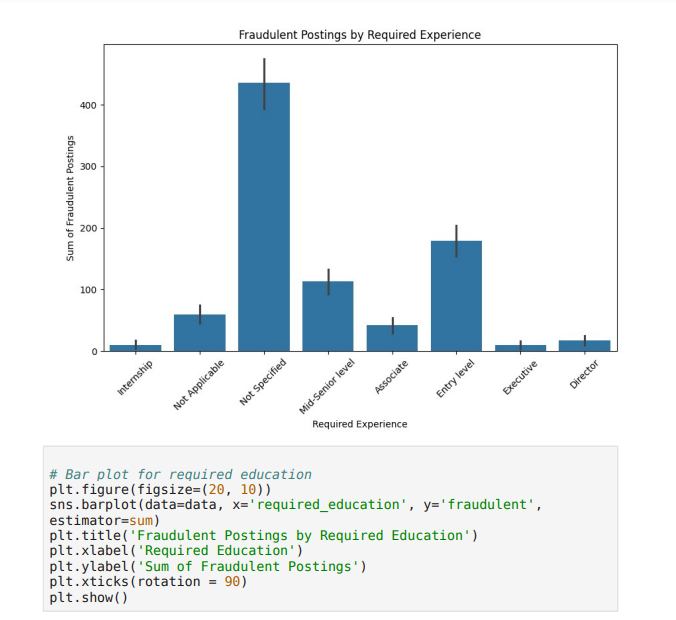




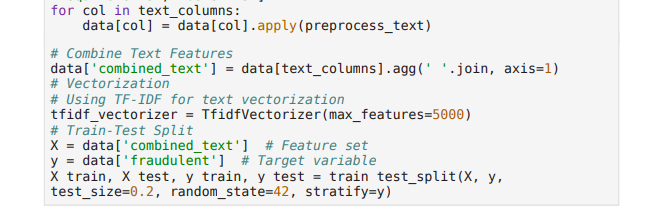


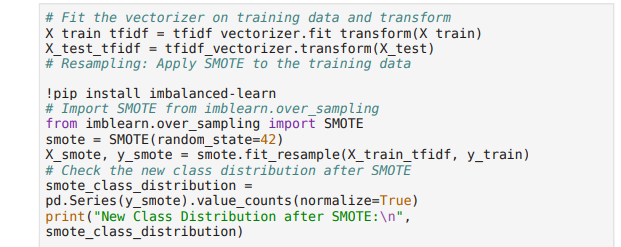


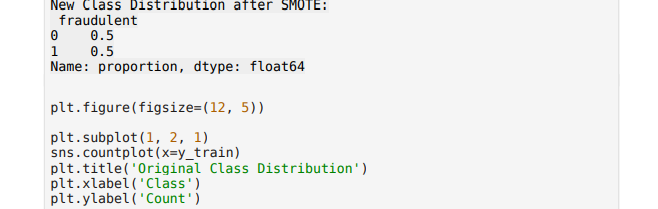


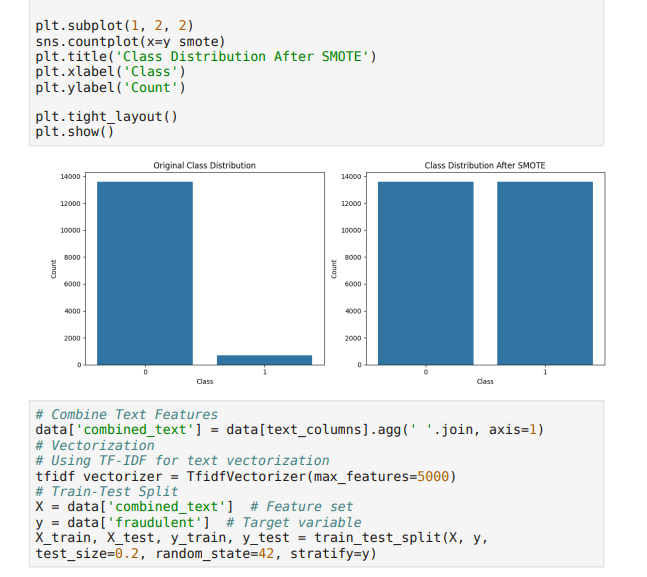




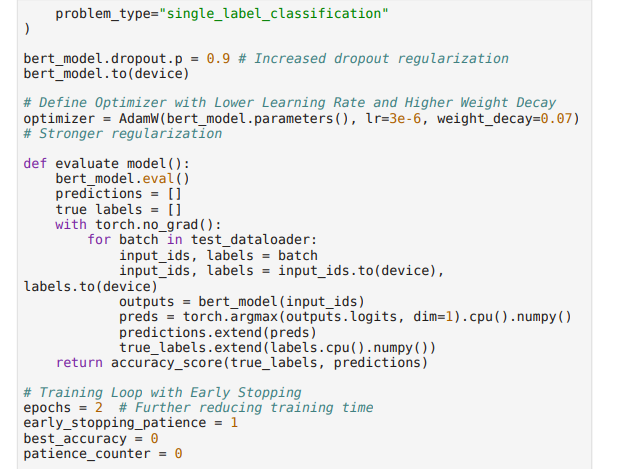


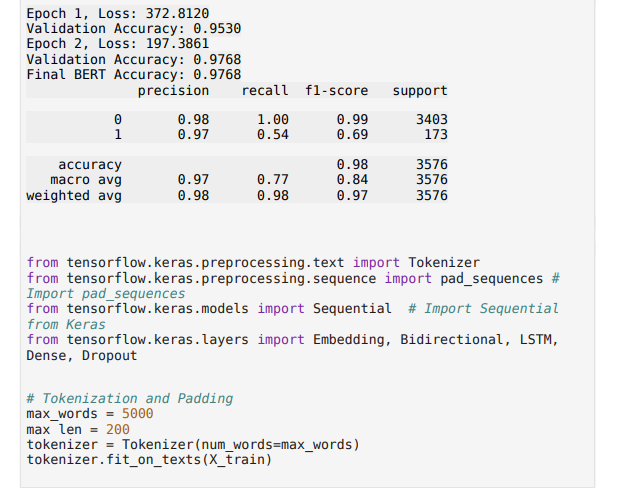
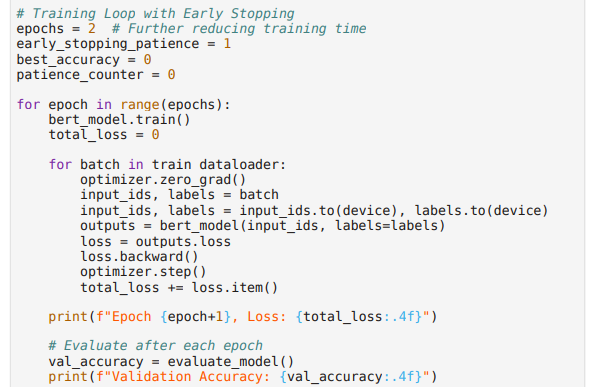


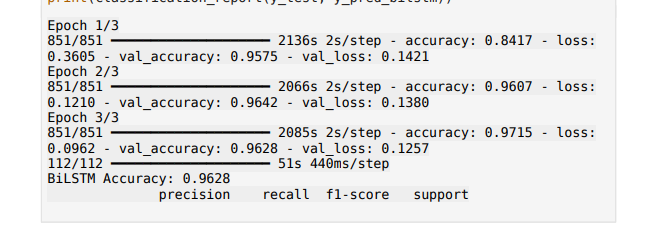
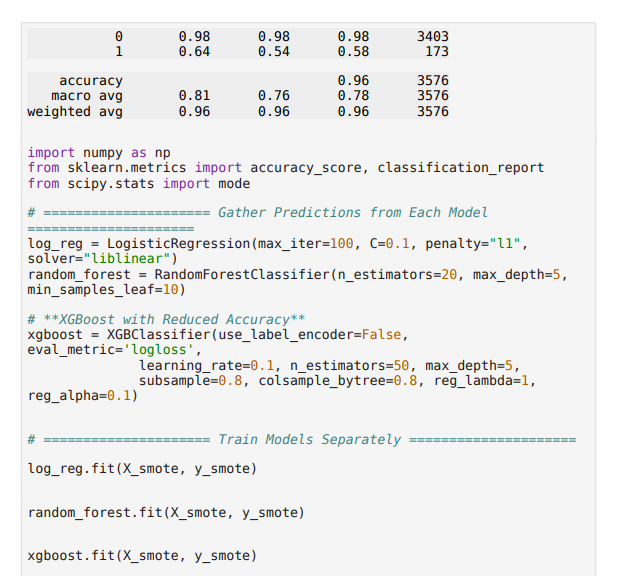
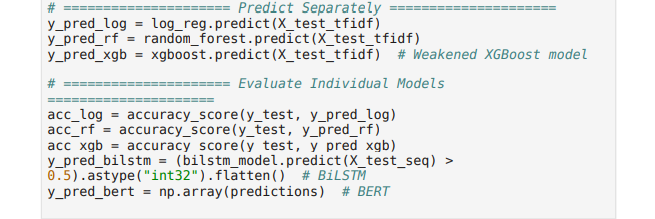
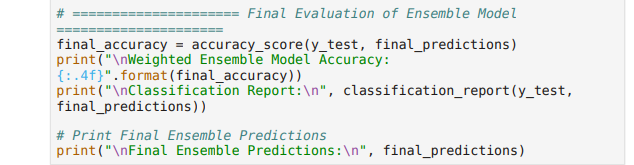
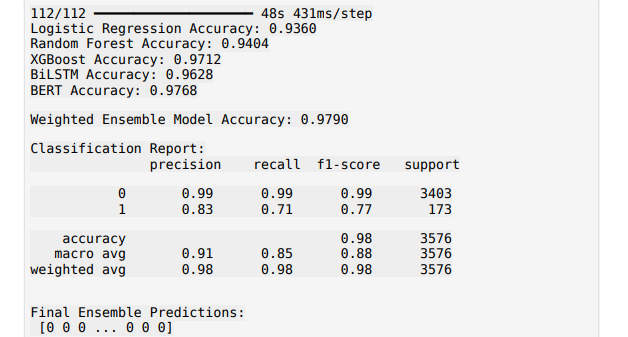
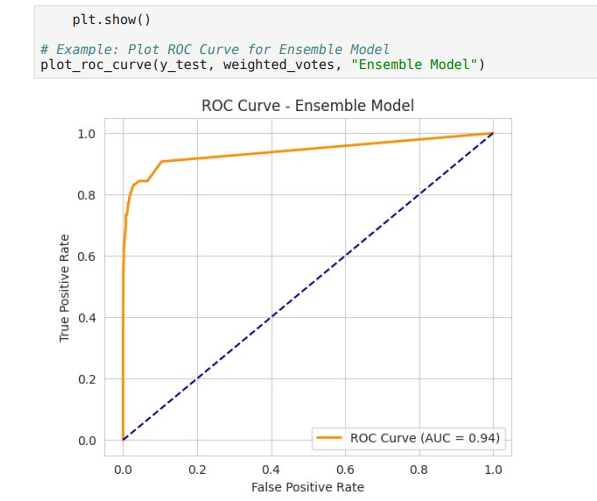
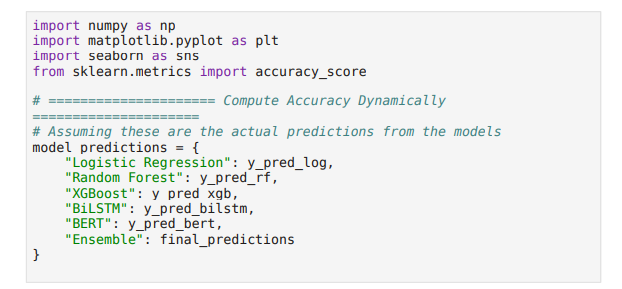
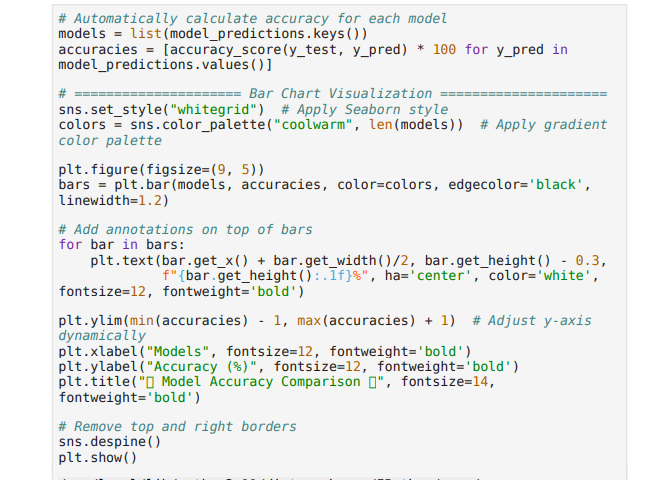
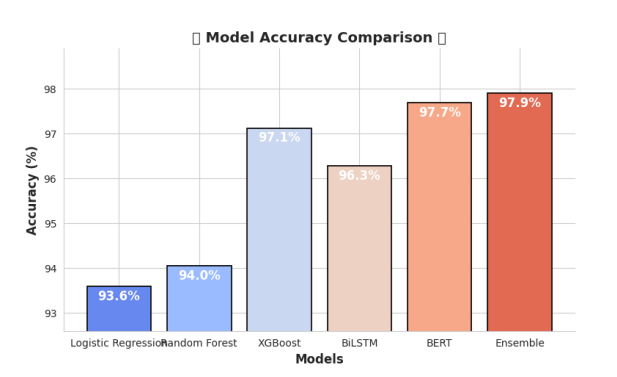
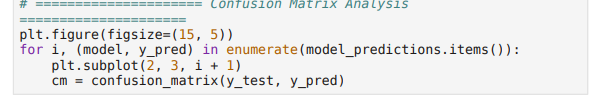
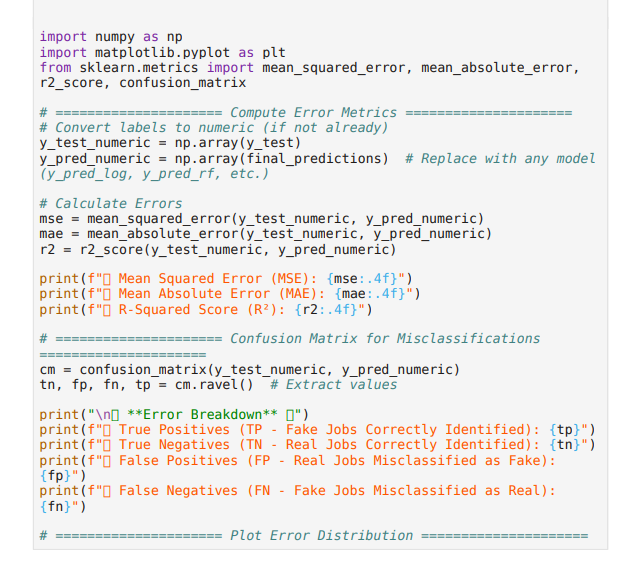
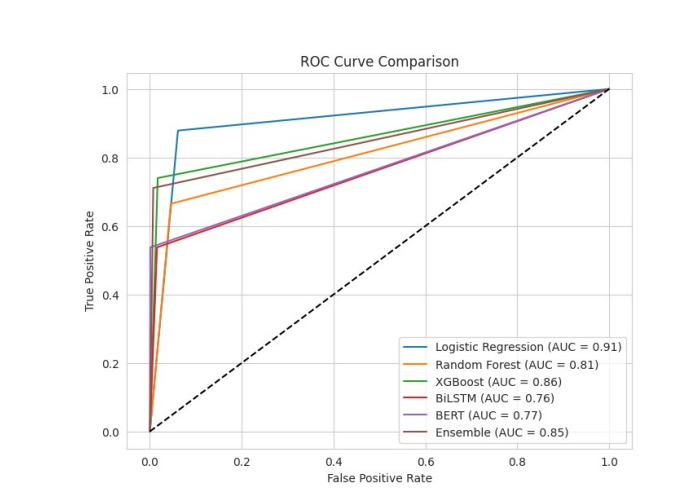
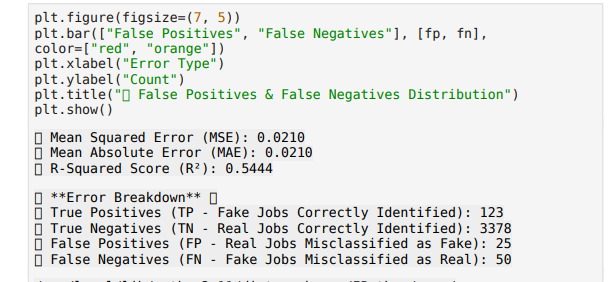


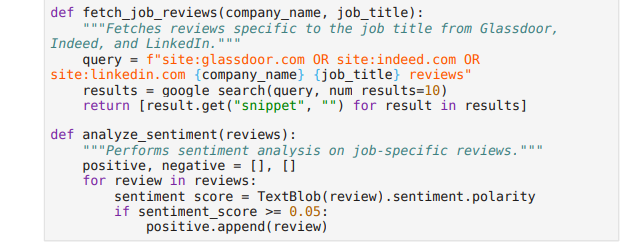
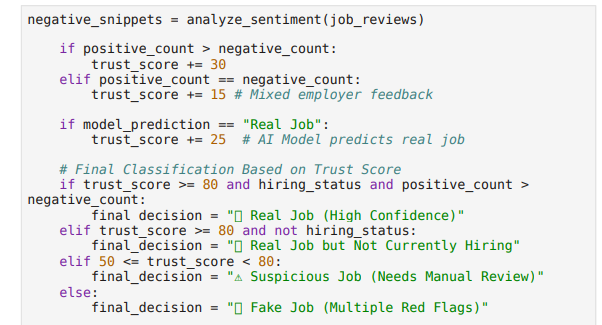
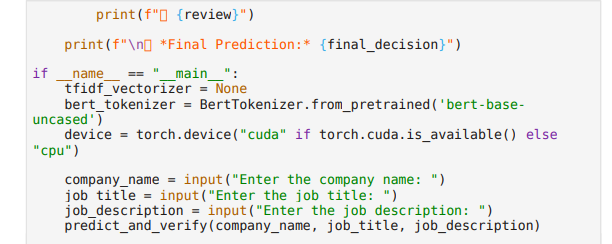






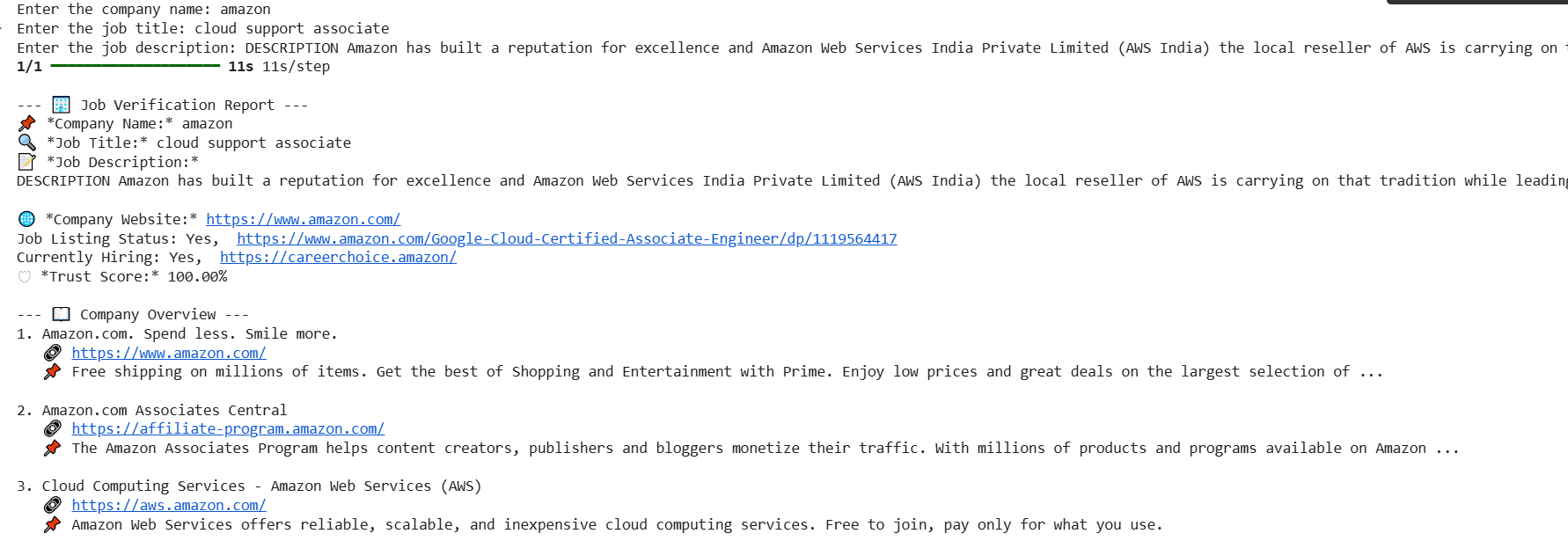


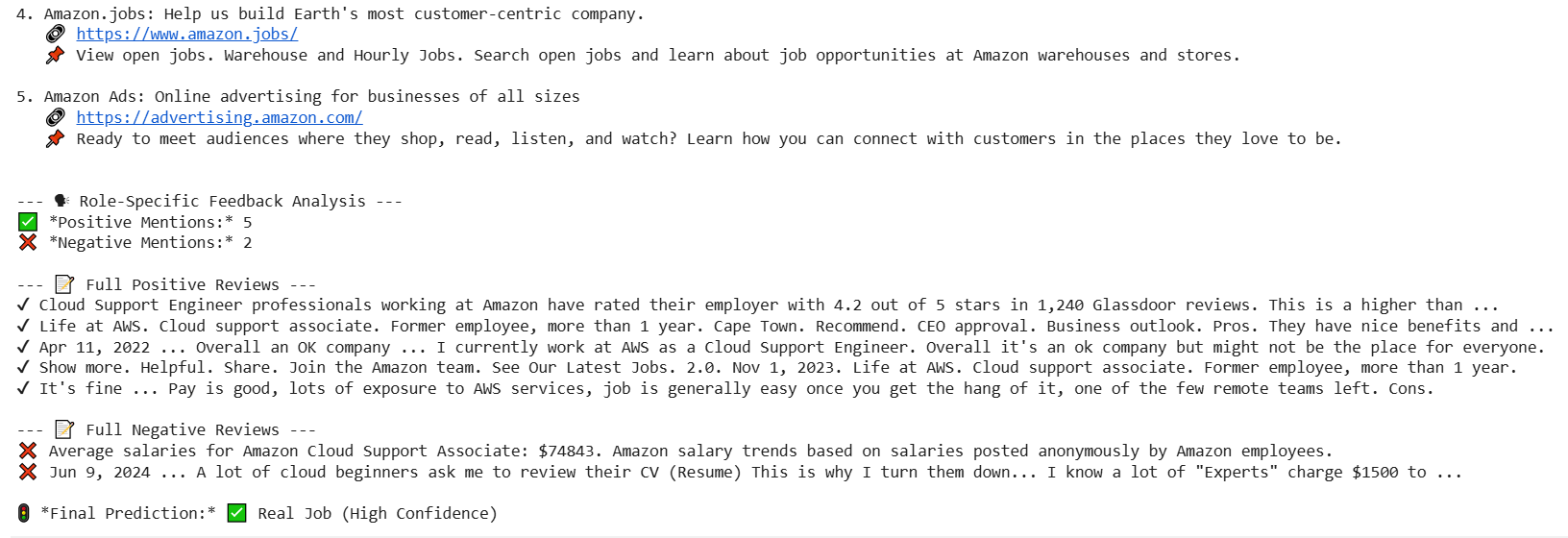
               

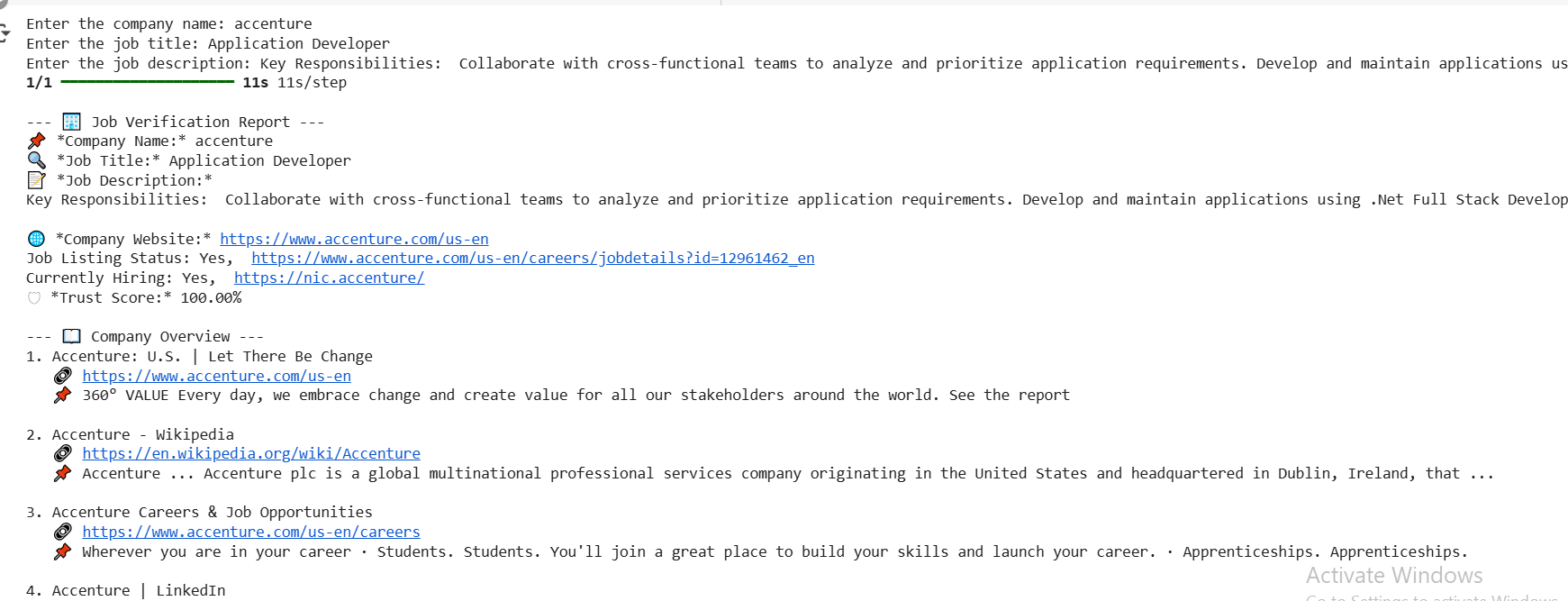
**RESULT**

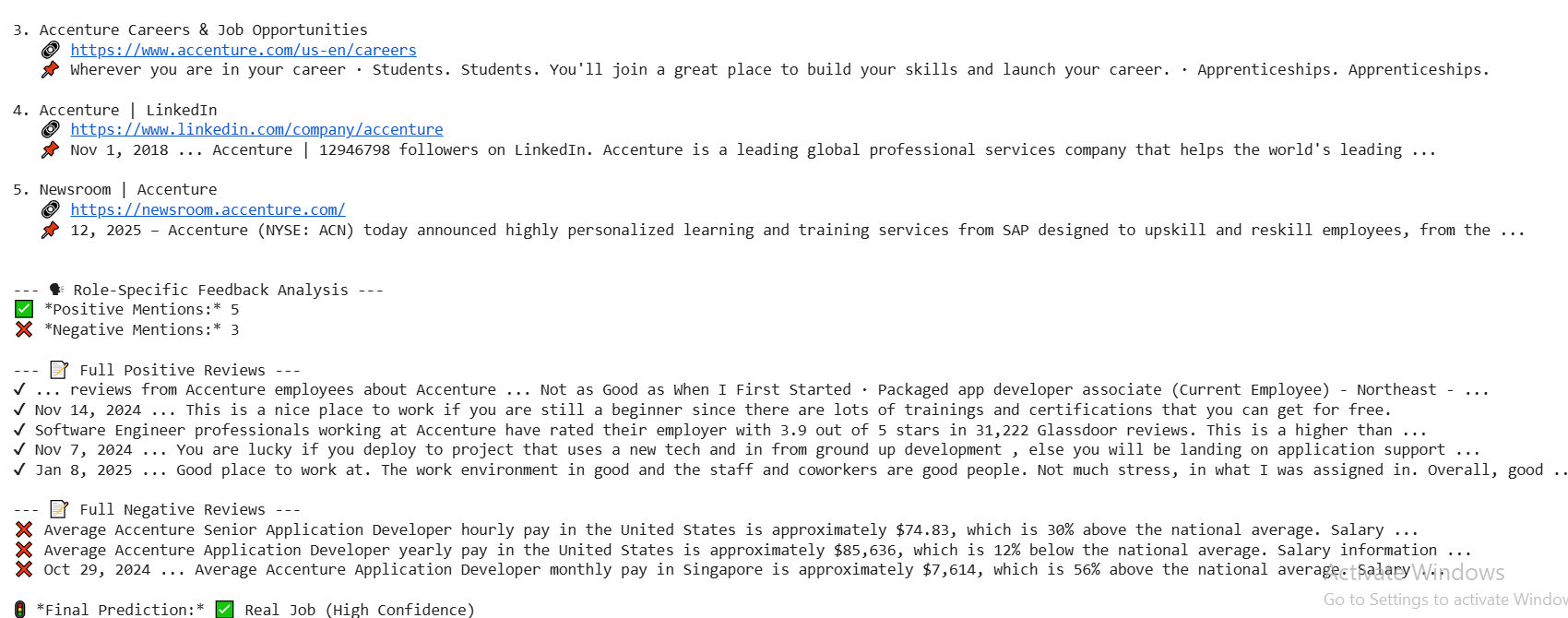
**1. AMAZON CLOUD SUPPORT ASSOCIATE – REAL JOB AND CURRENTLY HIRING**



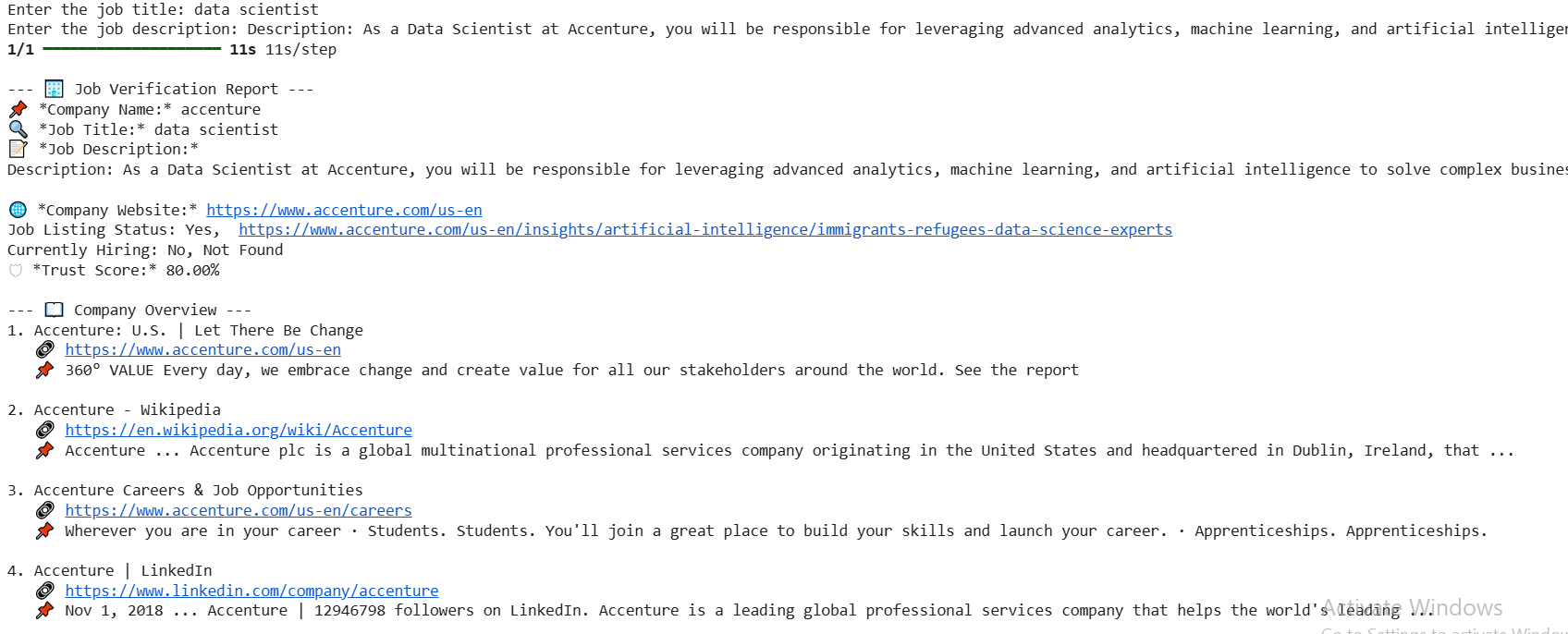


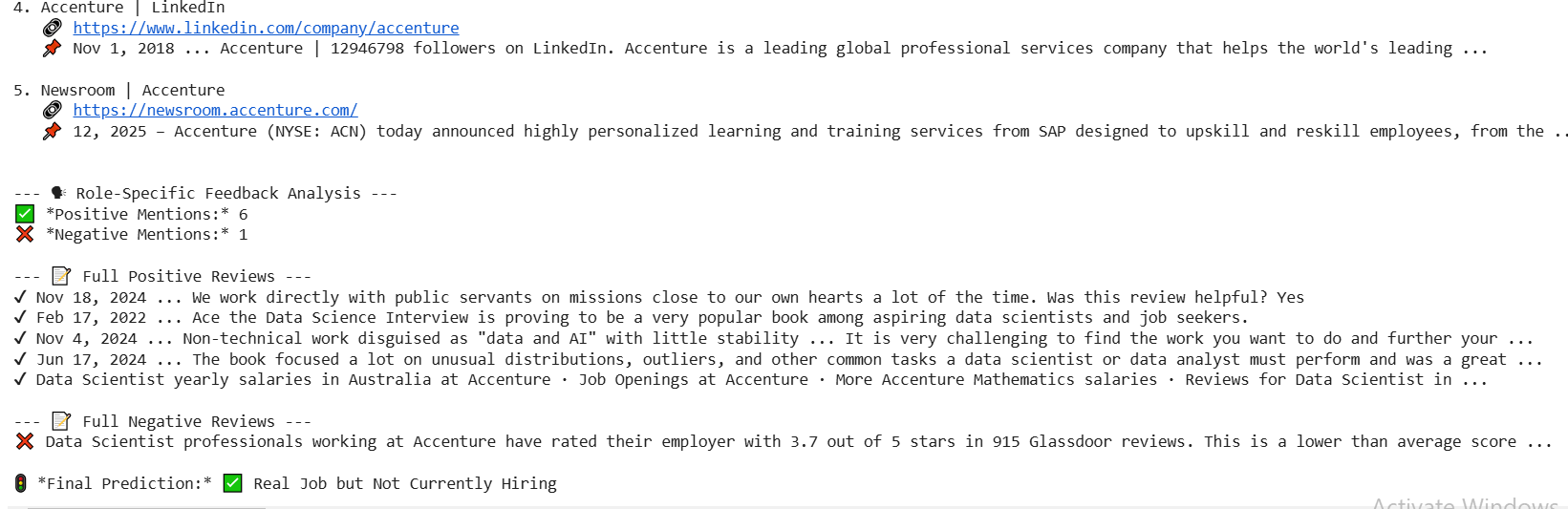
**2. ACCENTURE APPLICATION DEVELOPER – REAL JOB AND CURRENTLY HIRING**





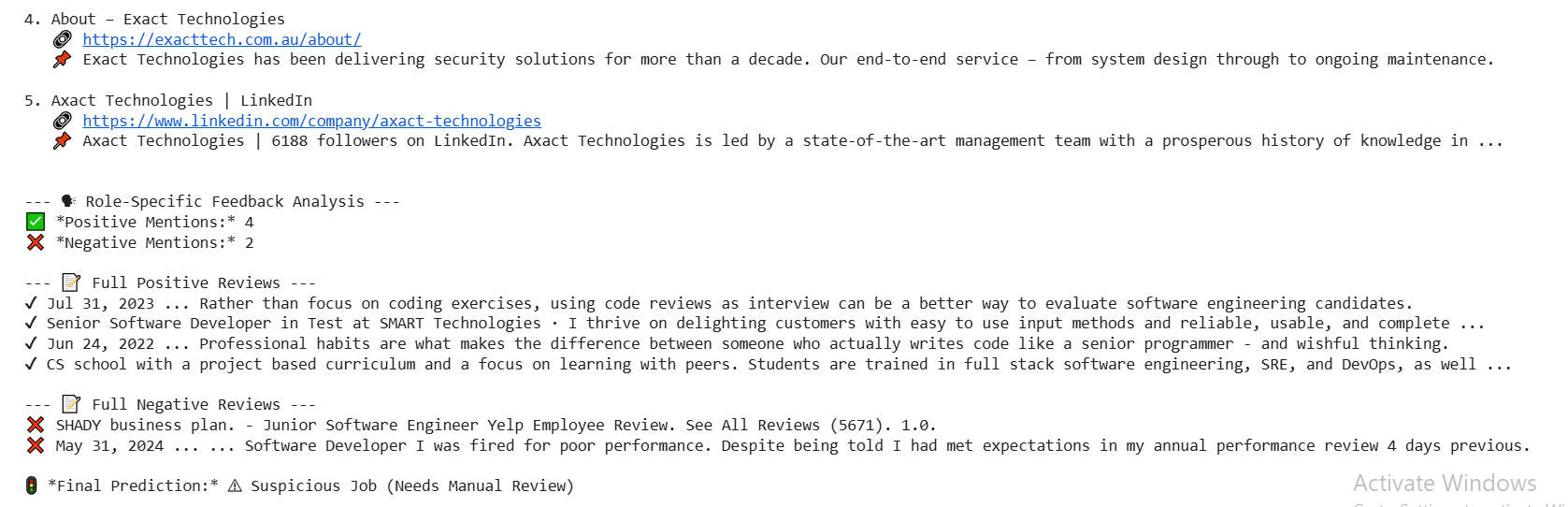
1. **ACCENTURE DATA SCIENTIST – REAL JOB AND CURRENTLY NOT HIRING**





**4. AXACT TECHNOLOGIES JUNIOR SOFTWARE DEVELOPER – FAKE JOB**





**5. THIRUMOOLAR SOFTWARE AI DEVELOPER – FAKE JOB**

