# Detecting Fraudulent Job Postings Using Machine Learning Algorithms

**OUAQUIF DIKRA*1**
***1Second Year Data Engineering Student,**
**dikra.ouaquif@etu.uae.ac.ma**
**Supervisor: Pr. KHAMJANE AZIZ**

## Abstract

Many job seekers fall prey to scam job postings which are hard to identify and this problem greatly needs solving. The model that is created within this research is able to flag a job advertisement for fraud using a fully automated machine learning self-taught model which utilizes the ML powered data scraped from Kaggle that contains around 17000 entries. The dataset scraped includes company details, titles, description, and salary, but has a target variable that is seriously highly skewed with utmost 5 pct fraud and the rest 95 pct legit. After a thorough exploratory data analysis (EDA), cleaning the data and visualizing, feature engineering + encoding the categorical variables + cleaning and vectorizing for the text were used to further improve interactions with the model. The SMOTE technique was used as a means to combat the class imbalance, afterwards the dataset was split for model training. Multiple classifiers like Logistic Regression, Random Forest, XGBoost, and even Naive Bayes were used to assess the model. Based on the analysis done its clear that lightgmb is the most accurate as it flags the fraudulent posts with an accuracy of 98 percent even while looking for fraudulent job postings. With online platforms being exposed to such models it can lower the risks of scam postings being used.

**Keywords**: Fraud Detection, Machine Learning, LightGBM, SMOTE, Feature Engineering, Classification

## I.Introduction

In the context of frauds related to online recruitment, one of the most recurring and glaring issues that has emerged in the past dealing training is employment scams. As the hiring process shifts more towards the use of online recruitment instruments, there is a great impetus on those seeking employment opportunities, as firms are inclined to make advertisements on jobs with ease. This ease of access is however taken advantage of the conmen who in numerous ways deal in fake job postings, thus duping individuals looking for jobs, primarily for monetary purposes or for impersonation. Such scams do not only affect the unemployed but the decent reputation of many companies is damaged too. It has become vital to unearth these so called ‘job advertisements as a means to protect users and uphold the integrity of the many recruitment websites. [1] A broad View of the problem scenario requires a structured representation of the set of tasks deemed fraudulent such as wage or employment advertisements. To obtain this information, a procedure that deals with the classification of job advertisements is performed and this allows sorting of fake and valid postings while notifying users. This split can be made in two parts. That is, Single Classifier Based Prediction and Ensemble Classifiers Based Prediction. Such models are validated on an annotated dataset and benchmarked using specificity, sensitivity. F1-score and other performance parameters. This will be accompanied by a diagrammatic representation of the procedure, followed by a description of the method layout involving these processes, including feature extraction and model fitting.

## II. Related work

Scams related to employment have advanced to level of being referred to as fraudulent job postings, and have become a big concern in the recruitment industry, especially being able to find work on the internet. Numerous studies exist showing the possibility of being able to use techniques from the domains of machine learning, and data mining for spotting fake job listings. In this segment, we give a compendium of major research and methodologies that have been reported earlier to solve the issue of job scams and related issues.

### 1.Fraud Detection in Online Job Postings

Dutta and Bandyopadhyay proposed a machine learning approach for identifying fraudulent job postings by applying multiple classification algorithms. They categorized classifiers into single and ensemble-based models and found that ensemble classifiers, such as Random Forest and AdaBoost, outperformed individual classifiers. The study emphasized the importance of preprocessing, such as data cleaning and feature selection, in improving model accuracy. Their work showed that machine learning models, especially ensemble methods, can effectively differentiate between legitimate and fake job offers, making it a valuable tool for job seekers and recruiters [2].

### 2.Fraud Detection in Online Platforms

A comparable method became applied to discover fraudulent activities in different online systems, together with e-trade web sites and social media. Alghamdi and Alharby explored the usage of gadget studying models to locate fraud in online transactions, that specialize in purchaser evaluations and product listings. Their research demonstrated that strategies like Random Forest, XGBoost, and Naive Bayes should detect fraudulent patterns with high precision. While their work ordinarily focused on product listings, the techniques they advanced are applicable to the activity recruitment domain, mainly in detecting faux job posts [3].

### 3.Email and Review Spam Detection

Machine studying strategies had been widely used in detecting junk mail in diverse forms, including e mail spam and assessment unsolicited mail. Dada et al. Performed an intensive review on the use of gadget getting to know for email spam detection, highlighting the success of classifiers like Naive Bayes and SVM. Similarly, techniques used for detecting faux evaluations, which include function extraction and classification fashions, percentage similarities with task scam detection, as each require the identity of patterns in text statistics that distinguish legitimate content from fraudulent cloth [4]

### Summary of the Related Work

These studies collectively show the ability of system learning fashions, specifically ensemble strategies, in detecting fraudulent process postings and different styles of on-line fraud. Previous paintings has demonstrated that information preprocessing, function engineering, and balancing strategies like SMOTE are vital for enhancing model performance. The use of classifiers like Random Forest, XGBoost, and Naive Bayes has validated effective throughout diverse domains, consisting of task posting fraud detection. However, demanding situations including class imbalance and the need for greater state-of-the-art models still continue to be, which motivates the want for in addition studies in this location.

## III.Methodology:

This study aims to pinpoint fraudulent job postings. Such fake job advertisements should not be advertised at all as it will assist job seekers to focus on real ones. For this purpose, a dataset from Kaggle is used in which information whether a job appears or does not appear suspicious is provided.

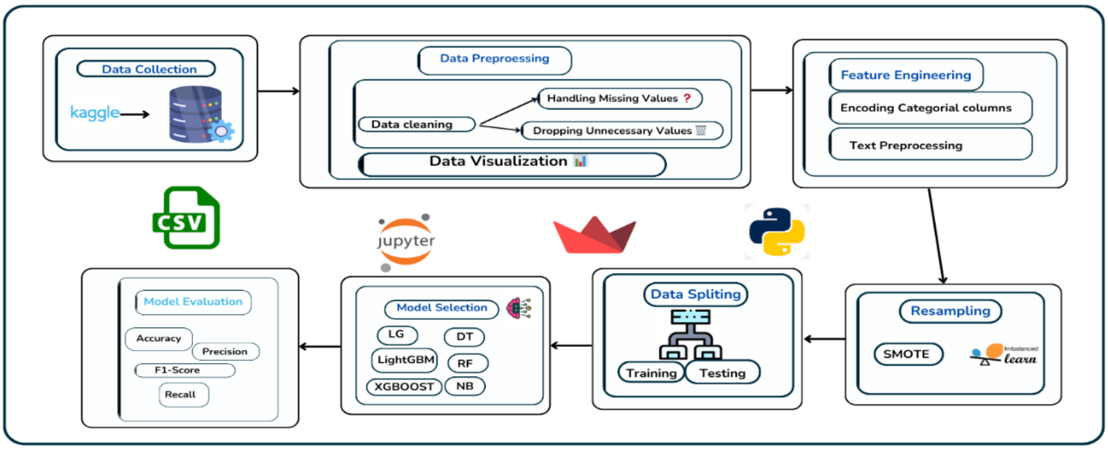
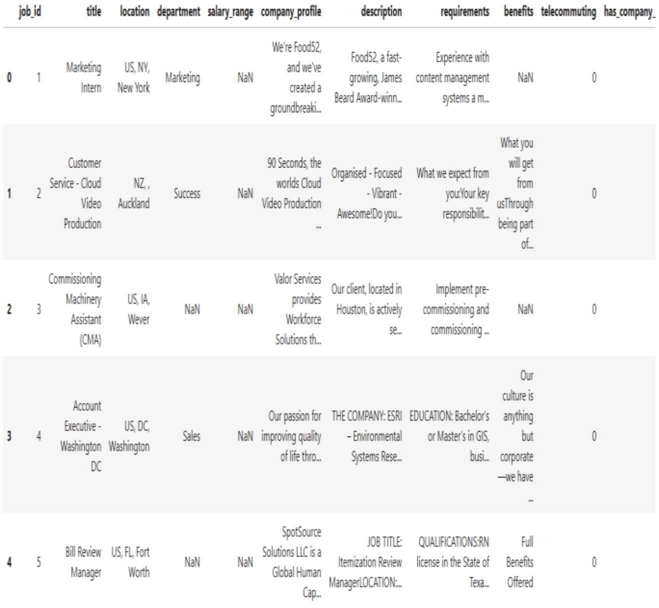


Fig 1: Architecture

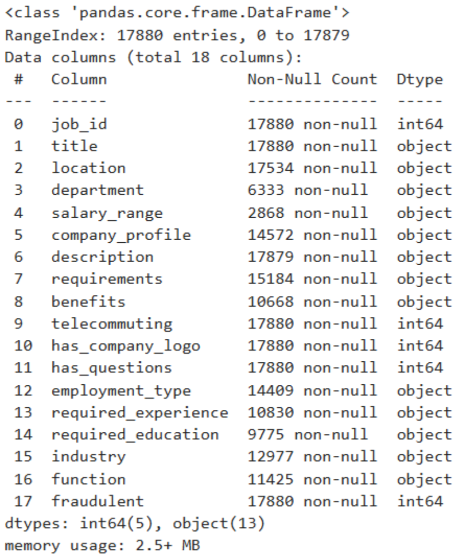
## A. Data Collection

For this research, we used fake job postings data set from Kaggle which is freely available. It contains details such as company names, salaries, job locations, titles and descriptions. The binary representation of the variable distinguishes between genuine and fake job postings (0 for genuine,1 for fake posting).In this portion of our work, we employed the dataset in order to pre-process data to obtain significant features and subsequently, interrogate the dataset for the purpose of machine learning with the end goal of classifying if job posting as fake or real. The dataset schema is depicted in Figure 2 where attributes of the dataset and their respective data types are illustrated.



| | job_id | title | location | department | salary_range | company_profile | description | requirements | benefits | telecommuting | has_company_ |
|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | 1 | Marketing Intern | US, NY, New York | Marketing | NaN | We're Food52, and we've created a groundbreaki... | Food52, a fast-growing, James Beard Award-winn... | Experience with content management systems a m... | NaN | 0 | |
| **1** | 2 | Customer Service - Cloud Video Production | NZ, , Auckland | Success | NaN | 90 Seconds, the worlds Cloud Video Production ... | Organised - Focused - Vibrant - Awesome!Do you... | What we expect from you:Your key responsibilit... | What you will get from usThrough being part of... | 0 | |
| **2** | 3 | Commissioning Machinery Assistant (CMA) | US, IA, Wever | NaN | NaN | Valor Services provides Workforce Solutions th... | Our client, located in Houston, is actively se... | Implement pre-commissioning and commissioning ... | NaN | 0 | |
| **3** | 4 | Account Executive - Washington DC | US, DC, Washington | Sales | NaN | Our passion for improving quality of life thro... | THE COMPANY: ESRI – Environmental Systems Rese... | EDUCATION: Bachelor's or Master's in GIS, busi... | Our culture is anything but corporate—we have ... | 0 | |
| **4** | 5 | Bill Review Manager | US, FL, Fort Worth | NaN | NaN | SpotSource Solutions LLC is a Global Human Cap... | JOB TITLE: Itemization Review ManagerLOCATION:... | QUALIFICATIONS:RN license in the State of Texa... | Full Benefits Offered | 0 | |

r



```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 17880 entries, 0 to 17879
Data columns (total 18 columns):
 #   Column               Non-Null Count  Dtype
---  ------               --------------  -----
 0   job_id               17880 non-null  int64
 1   title                17880 non-null  object
 2   location             17534 non-null  object
 3   department           6333 non-null   object
 4   salary_range         2868 non-null   object
 5   company_profile      14572 non-null  object
 6   description          17879 non-null  object
 7   requirements         15184 non-null  object
 8   benefits             10668 non-null  object
 9   telecommuting        17880 non-null  int64
 10  has_company_logo     17880 non-null  int64
 11  has_questions        17880 non-null  int64
 12  employment_type      14409 non-null  object
 13  required_experience  10830 non-null  object
 14  required_education   9775 non-null   object
 15  industry             12977 non-null  object
 16  function             11425 non-null  object
 17  fraudulent           17880 non-null  int64
dtypes: int64(5), object(13)
memory usage: 2.5+ MB
```

Fig 2: Data Overview

## B. Dataset Processing

To begin with a crucial step in a machine learning process, models should always be applied after conducting Exploratory Data Analysis (EDA), as this process is essential in determining the integrity of the dataset. Some of the issues that EDA identified include missing data, outliers, and interactions between the explanatory variables and the response variable. To facilitate the process of observing the distribution and relationship among variables, histograms, box plots, and correlation heatmaps were employed. It was noticed during the EDA that a few columns had a lot of missing data while several other columns had a weak correlation with the target variable. These insights became the backbone for deciding on the steps necessary for data cleansing and preparation for model training.

### Handling Missing Values:

### Data Cleaning:

We removed the columns with a high percentage of missing values (Features with weak correlation to the target variable, such as job_id, location, telecommuting, and others). While remaining columns with missing values were appropriately imputed. Textual columns were filled with a placeholder, *'**Missing**'*, while categorical columns were filled with *'**Not Specified**'.*

### Dropping Unnecessary Columns:

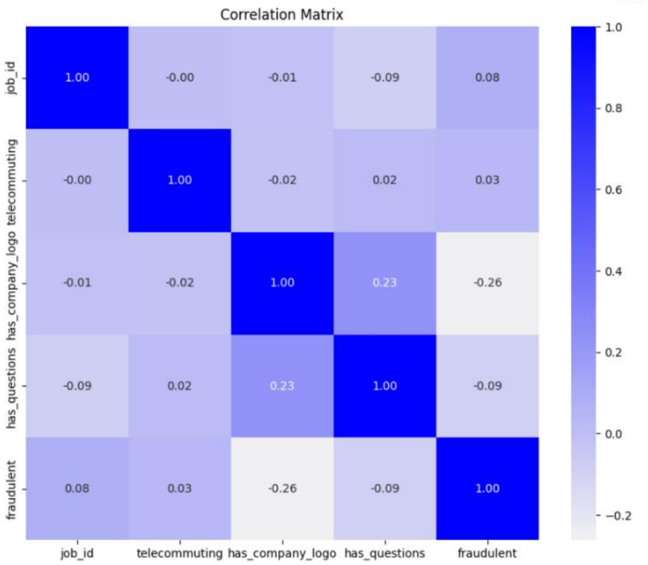


Fig 3: Correlation Matrix

### Removing Duplicates:

We removed duplicate rows to ensure data consistency also to prevent biases in model training.

## C. Feature Engineering:

In this step we aim to transform the raw of the dataset to features that could be more effectively used by machine learning models.

### 1.Encoding Categorical Variables:

**-One Hot Encoding:** for columns with low cardinality for example: employment_type,required_experience to present them as binary features.

**-Target Encoding:** used for columns with high cardinality such as required education, industry,function to map categories to the target variable's mean values,impeoving the model's ability to capture the influence of these features.

### 2.Text Preprocessing:

### 2.1 Text Cleaning:

- We converted the text to lowercase.
- We removed **Digits** and **punctuation** by using regular expressions.
- We removed **stop words** (like: and,the..) to reduce noise.
- **Lemmatization** was applied to reduce the word to their basic forms.

### 2.2 Combining text columns:

We combined all the relevant textual columns into new single column named 'Company' to have efficient text processing.

### 2.3 Text vectorization:

**TF_IDF(Term frequency-Inverse Document frequency)** changed int implemented to the newly created 'Company' column-IDF transforms the textual content into numerical characteristic vectors, shooting the importance of each phrase relative to the whole dataset even as minimizing the impact if not unusual words. The TF-IDF matrix turned into restrained to the top 500 capabilities to reduce dimensionality and enhance processing performance.

## D. Data Set Balancing:

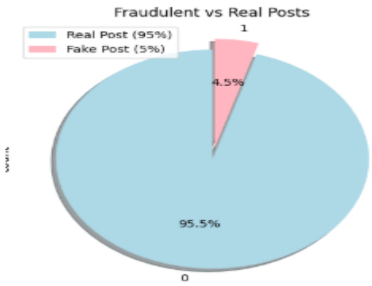


Fig 4: Job Postings Distribution Before Balancing

In this examine, the dataset suffered from an imbalance elegance distribution, which means that the number of valid task postings become drastically better than the fraudulent ones. This imbalance should cause biased model performance, where in the set of rules may additionally expect the bulk elegance (legitimate process postings) with high accuracy while failing to stumble on the minority class (fraudulent activity postings). To address this trouble, we carried out data balancing techniques to make sure that the gadget studying model ought to learn from both lessons similarly and make more accurate predictions for fraudulent jo postings. We used the **SMOTE** (Synthetic Minority Over-sampling Technique) set of rules, which generates artificial examples of the minority class by means of interpolating between present times. This method helped to create a greater balanced distribution of the goal lessons, presenting the model with enough examples of fraudulent activity postings.

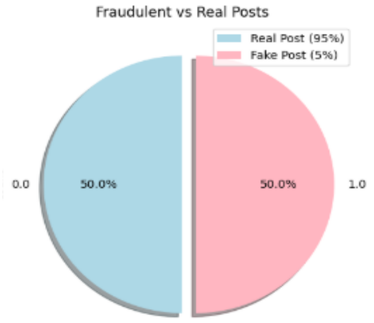


Fig 5: Job Postings Distribution After Balancing

## E. Data split

We split the dataset into training and testing sets using train_test_split from sklearn.The training set is used to train the machine learning modle,and the testing set is used to evaluate the performance of the machine learning model on unseen data.(the 80% of the dataset is used for training while 20% of the dataset is used to test the model).

# IV. Model Selection and Building

Serval machine learning models were selected on this study to classify task transfer as deceptive or legitimate. The model selection was based on the ability to deal with the imbalanced dataset and complex relationships between variables. Hyperparameter tuning was performed using techniques such as Grid Search,Random Search,Cross-validation and AdaBoost to determine the optimal parameters for each algorithm before submitting the final model this process ensured that the models were optimized for performance which is excellent..

The models used:

### 1.Logistic Regression:

Logistic Regression is a simple and effective Linear model that is commonly used for binary classification problems. It estimates the probability that given job posting belongs to a particular class, making it easy to interpret. This model was used to serve as a baseline and compare the performance of more complex models. The key hyperparameters tuned where:

**-c: Con**trols the strength of regularization. Smaller values imply stronger regularization.

**- Solver:** Options like liblinear and saga were tested for convergence speed and performance

$$P(y=1|X)=\frac{1}{1+e^{-(w^TX+b)}}$$

- $X$ is the feature vector.
- $w$ is the weight vector.
- $b$ is the bias term.
- $e$ is the base of the natural logarithm.

### 2.Random Forest:

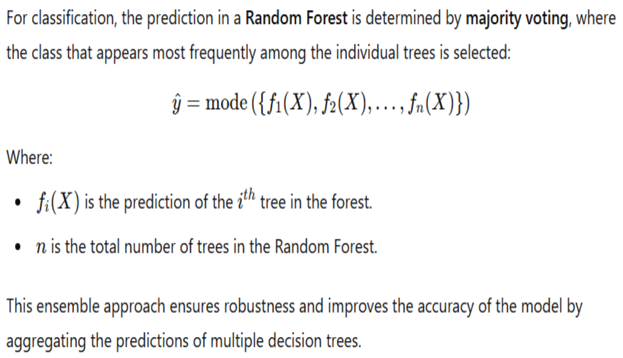
Random Forest is a method of ensemble learning that build various decision trees and combine the individual results from the trees to enhance the accuracy of the classification.it is widely used for its self-regulating overfitting and the fact that it is compatible with both numerical and categorical data.in this research,Random Forest was used for identifying complex relationships in the data set. Important Hyperparameters include:

**-n_estimators:** The number of trees in the forest. The higher it is the more accurate it becomes but it takes longer to compute. This was set at an optimal.

**-max_depth**: Tree depth is limited so that overfitting does not occur. An optimal of 10 was found in the grid search.

**-max_features:**It decides the number of features that are taken into account while splitting. The sqrt method seemed to be the most useful.

**-min_samples and min_samples_leaf:** Controlled the minimum samples that were necessary in order to undergo splits and leaf nodes for a better model generalization.



For classification, the prediction in a **Random Forest** is determined by **majority voting**, where the class that appears most frequently among the individual trees is selected:

$$\hat{y} = \text{mode}(\{f_1(X), f_2(X), \ldots, f_n(X)\})$$

Where:

- $f_i(X)$ is the prediction of the $i^{th}$ tree in the forest.
- $n$ is the total number of trees in the Random Forest.

This ensemble approach ensures robustness and improves the accuracy of the model by aggregating the predictions of multiple decision trees.

## 3.XGBoost (Extreme Gradient Boosting):

**Xgboost** is an updated technique that applies gradient boosting ensemble to several weak learners(often trees) in order to create one decent classifier.it is particularly effective when it comes to addressing the issue of embalced datasets as well as that of high accuracy, therefore ,it serves as a robust model for identifying fraudulent job advertising. The key hyperparametrs tuned where:

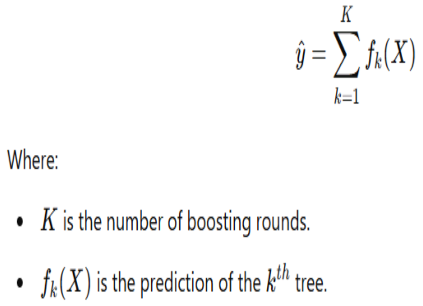
**-learning-rate**: Determines the size of steps taken in the boosting process. Value of 0.1 was found to be a good compromise between speed and accuracy

**-n_estimators**: Number of boosting iterations.150 was found to be optimal value.

**-max_depth:**Tree depth restriction to avoid overfitting.6 was established to be optimal trough cross-validation.

**-subsample and colsample_bytree:** These parameters restricted the proportion of trees per sample and the number of features per tree for enhancing the model.

**-scale_pos_weight:** This parameter altered the positive class weight to address imbalanced class distributions.



$$\hat{y} = \sum_{k=1}^{K} f_k(X)$$

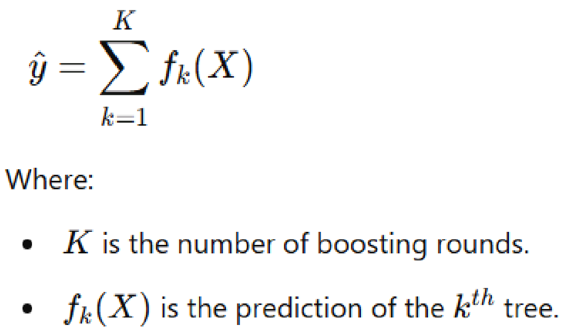
Where:

- $K$ is the number of boosting rounds.
- $f_k(X)$ is the prediction of the $k^{th}$ tree.

## 4.LightGBM:

**LightGBM** is a framework for gradient boosting that is optimized to be highly efficient and faster when working with data that is large in size. The reason it was selected is due its efficiency in dealing with high dimensional data and it has a fast-training speed in comparison to the common methods of gradient boosting. The scaling advantage of LightGBM makes it appropriate for this project. The key hyperparameters tune where:

**-learning_rate:** A magnitude of 0.05 was found to be reasonably good in setting the rate of convergence and perf.



$$\hat{y} = \sum_{k=1}^{K} f_k(X)$$

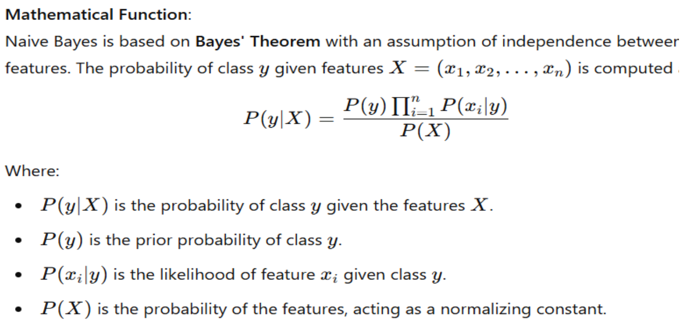
Where:

- $K$ is the number of boosting rounds.
- $f_k(X)$ is the prediction of the $k^{th}$ tree.

**5.Naive Bayes:**

Naive Bayes is a probabilistic classifier that applies Bayes’ Theorem with strong (naive) independence assumptions. It is particularly well-suited for text classification tasks, such as classifying job descriptions as legitimate or fraudulent, making it an essential model in this research. Key considerations included:

**-alpha**: The smoothing parameter to handle zero probabilities. An optimal value of 1 was used to stabilize the model for sparse text data.



**Mathematical Function**:

Naive Bayes is based on **Bayes' Theorem** with an assumption of independence between features. The probability of class $y$ given features $X = (x_1, x_2, \ldots, x_n)$ is computed

$$P(y|X) = \frac{P(y)\prod_{i=1}^{n} P(x_i|y)}{P(X)}$$

Where:

- $P(y|X)$ is the probability of class $y$ given the features $X$.
- $P(y)$ is the prior probability of class $y$.
- $P(x_i|y)$ is the likelihood of feature $x_i$ given class $y$.
- $P(X)$ is the probability of the features, acting as a normalizing constant.

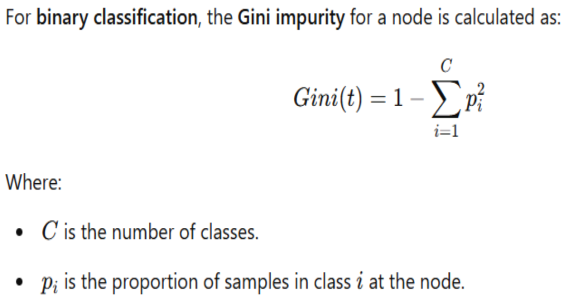
**6.Decision Tree:**

The Decision Tree model was used to split the data into subsets based on feature values, eventually creating a tree-like structure for classification. It is interpretable and can handle both categorical and numerical data, making it useful for understanding how different job posting features contribute to classifying them as fraudulent or legitimate. Key hyperparameters were:

**-max_depth**: Limited to 8 to reduce overfitting.

**-min_samples_split** and **min_samples_leaf**: Controlled the minimum samples required for splits and leaf nodes, respectively, to balance depth and accuracy.

**-criterion**: Tested both Gini and entropy to determine the best impurity measure. The gini index provided better results.



For **binary classification**, the **Gini impurity** for a node is calculated as:

$$Gini(t) = 1 - \sum_{i=1}^{C} p_i^2$$

Where:

- $C$ is the number of classes.
- $p_i$ is the proportion of samples in class $i$ at the node.

Each of these models was trained and evaluated using the best parameters identified through the aforementioned tuning techniques. The results of each model were compared to determine which performed best at classifying fraudulent job postings.

# V. Model Evaluation Metrics

When evaluating the performance of machine learning models,it is important to use several metrics measuring performance.These metrics should include additional variables apart from the general accuracy and should also identify other modes of incorrect classifactions that may hinder the model.Below,a few metrics that are predominantly used to evaluate the model of this problem are presented.

**Accuracy:** is one of the most commonly used metrics and represents the ratio of correct predictions to the total number of instances. Accuracy is one of the common metrics to use but it does have its weaknesses especially in cases of imbalanced dataset. A model could still attain a considerable accuracy by accommodating the correct class to the majority class while providing unfavorable attention to the little-known class when it is predominant in nature, This is the case when detecting fake posts.

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

Where TP = True Positives, TN = True Negatives, FP = False Positives, and FN = False Negatives.

**Precision:** The proportion of actual positive instances in the prediction set is the measure of precision. In this ratio false positives who are negative instance that have been incorrectly classified as positives are significant

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

**-Recall (Sensitivity):**Recall is simply the measure of all actual positive instances compared to the true positive results. This metric is particularly useful in scenarios where negative results shouldn't outweigh the positive results.

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

**-F1-Score:** The harmonic mean of precision and recall

$$\text{F1-Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

**-AUC-ROC:** AUC represents the likelihood of a model distinguishing between positive and negative classes. The higher the AUC, the better the model.

**-Confusion Matrix:** A matrix showing the number of true positives, false positives, true negatives, and false negatives. It gives a complete picture of how well the model performs, especially in classification tasks.

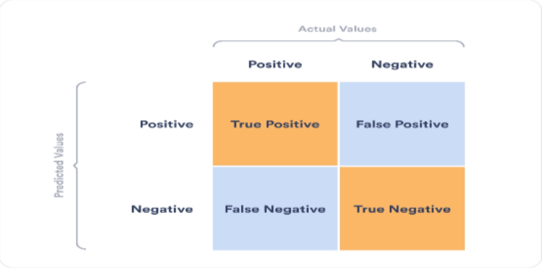


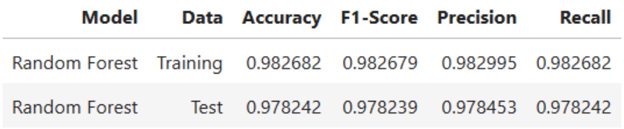
Fig 6 : Confusion Matrix

Once the metric is defined, it is vital to justify the selection of metric in respect of the severity it holds in answering the problem posed. For example, if the aim the task is to identify fake posts, recall may be periodized over precision. The rationale for such an approache is thaht not only fake posts but a lot of non-fake posts can be marked as fake which can prove to be quite deleterious. In this case, the goal is to reduce the maximum amount of fake posts that can go undectected.The reasoning behind this remains quite simple: in situations such as this, missing a fake post is worse than labeling a real post even if it is falsely as fake.Xerros o omission or errors of commission are more sensitive and consequential to the undertaken task than errors of inclusion. False negatives are much more deleterious in situations where content moderation has to prevent the spread of fake or malicious posts in contrast, such forms of false inclusion are merely grievances to users whom theses features falsely flag rather than gravely interfering with the system's purpose of preventing fake content.

# VI. Model Evaluation and comparaison:

## 1.Model Evaluation:

### A. Random Forest:



| Model | Data | Accuracy | F1-Score | Precision | Recall |
|---|---|---|---|---|---|
| Random Forest | Training | 0.982682 | 0.982679 | 0.982995 | 0.982682 |
| Random Forest | Test | 0.978242 | 0.978239 | 0.978453 | 0.978242 |

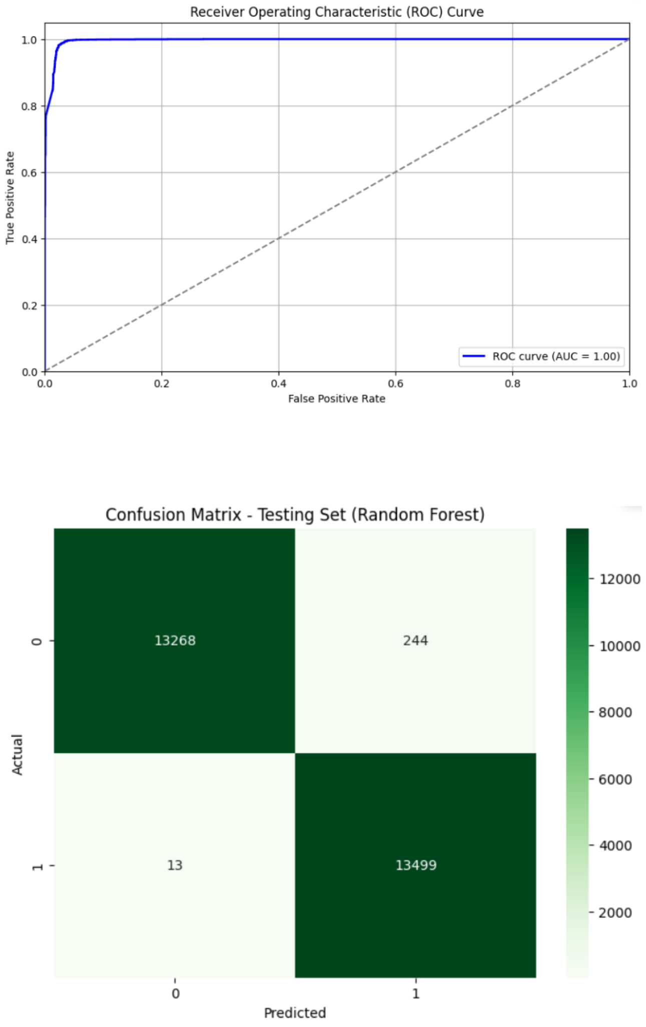
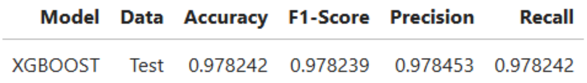


Figure 7 : Random Forest Model Performance : Confusion Matrix, ROC Curve, and Metrics

The accuracy of Random Forest model is about 97.8%, also this model achieves a high value across all the metrics as shown. Which means it is well-suited to the detect both real and fake job offer.

## B. XGBoost:



| Model | Data | Accuracy | F1-Score | Precision | Recall |
|---|---|---|---|---|---|
| XGBOOST | Test | 0.978242 | 0.978239 | 0.978453 | 0.978242 |

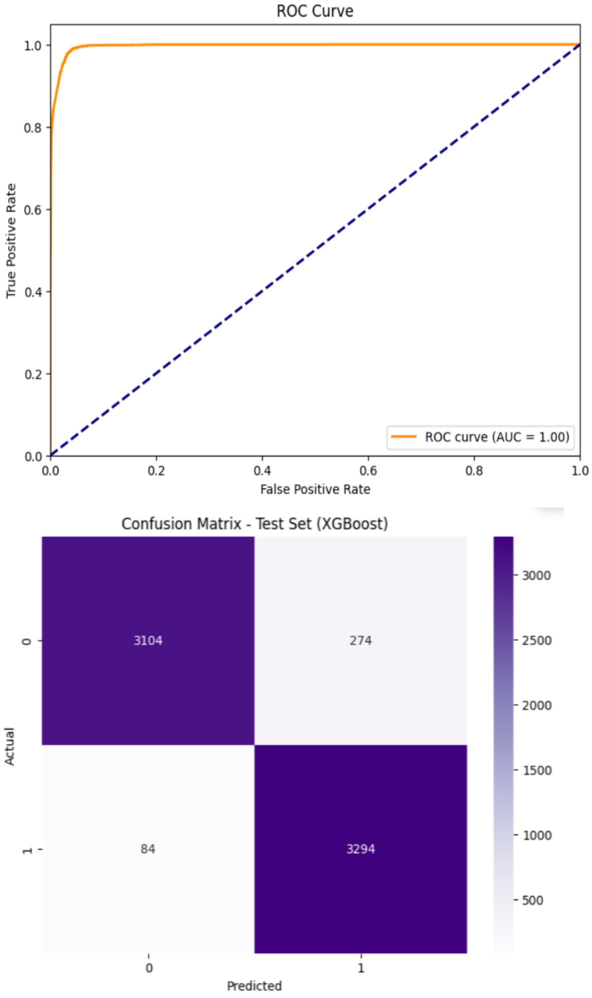
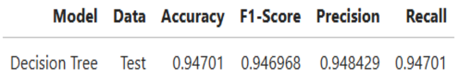


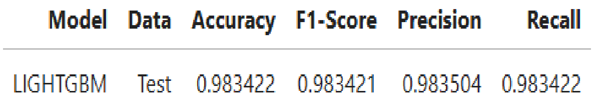
Fig 8 : XGBOOST Model Performance : Confusion Matrix, ROC Curve, and Metrics

The accuracy of **XGBOOST** is 97.8%.This model's high recall and precision indicative his effective ability tin classifying job offers.



| Model | Data | Accuracy | F1-Score | Precision | Recall |
|---|---|---|---|---|---|
| Decision Tree | Test | 0.94701 | 0.946968 | 0.948429 | 0.94701 |

## C.LightGBM:



| Model | Data | Accuracy | F1-Score | Precision | Recall |
|---|---|---|---|---|---|
| LIGHTGBM | Test | 0.983422 | 0.983421 | 0.983504 | 0.983422 |

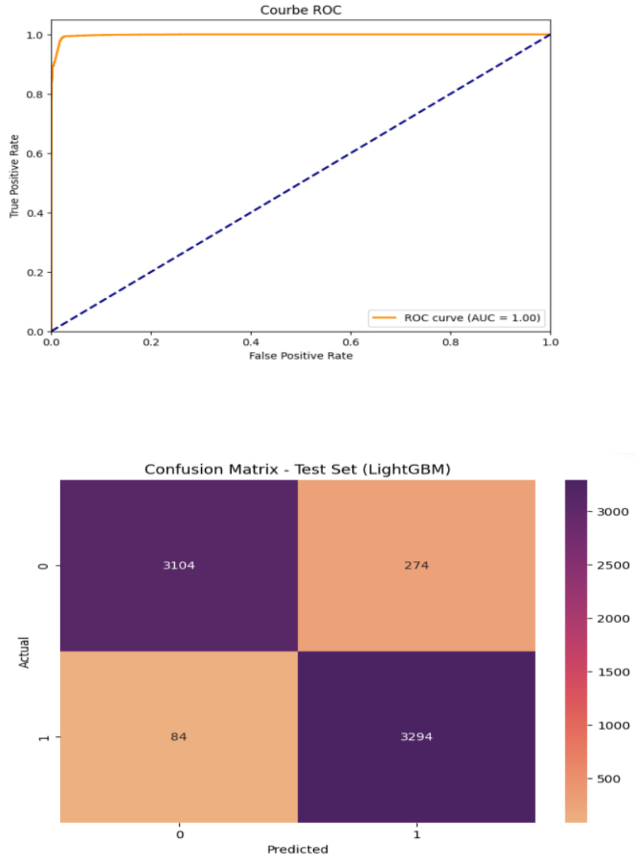
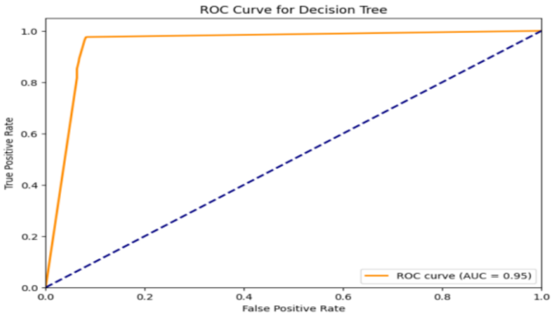


Fig 9: lightGBM Model Performance Confusion Matrix, ROC

**The LightGBM** is the most effective model in terms of recall and precision.

## D.Decision Tree:



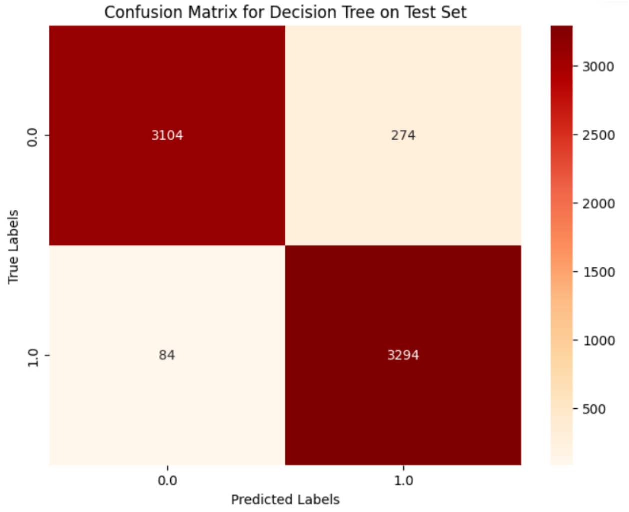
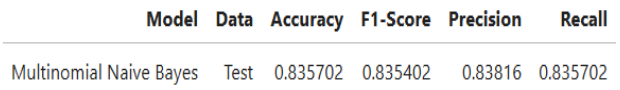


Figure 10 : Decision Tree Model Performance : Confusion Matrix, ROC Curve, and Metrics

## E. Multinomial Naïve Bayes :



| Model | Data | Accuracy | F1-Score | Precision | Recall |
|---|---|---|---|---|---|
| Multinomial Naive Bayes | Test | 0.835702 | 0.835402 | 0.83816 | 0.835702 |

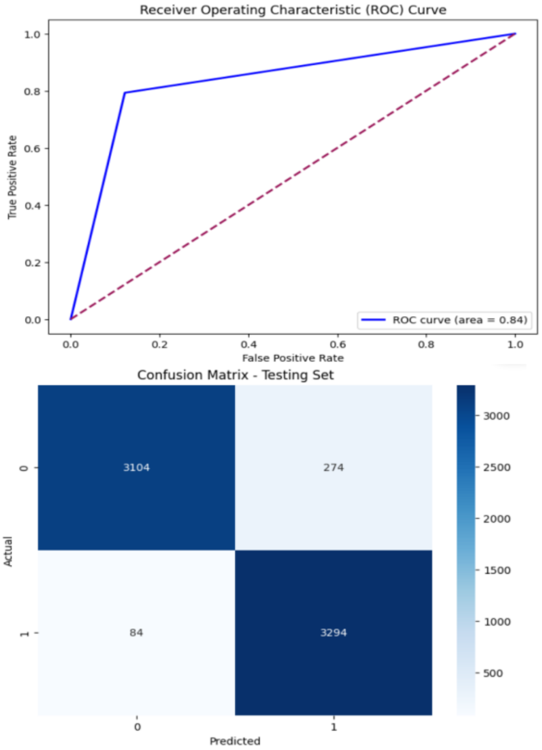
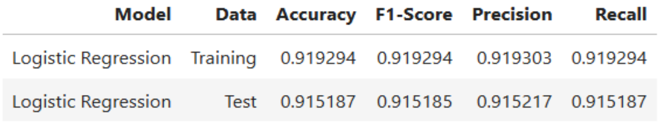


Figure 11 : Naïve Bayes Model Performance : Confusion Matrix, ROC

## F. Logistic Regression:



| Model | Data | Accuracy | F1-Score | Precision | Recall |
|---|---|---|---|---|---|
| Logistic Regression | Training | 0.919294 | 0.919294 | 0.919303 | 0.919294 |
| Logistic Regression | Test | 0.915187 | 0.915185 | 0.915217 | 0.915187 |

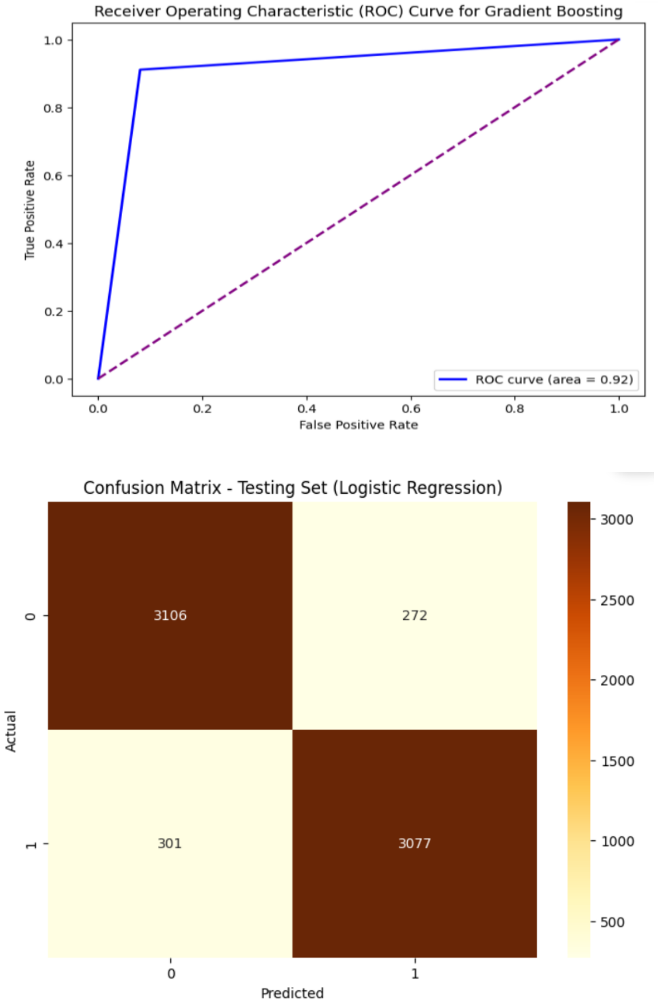
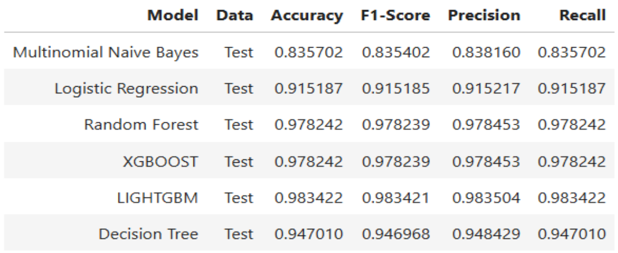
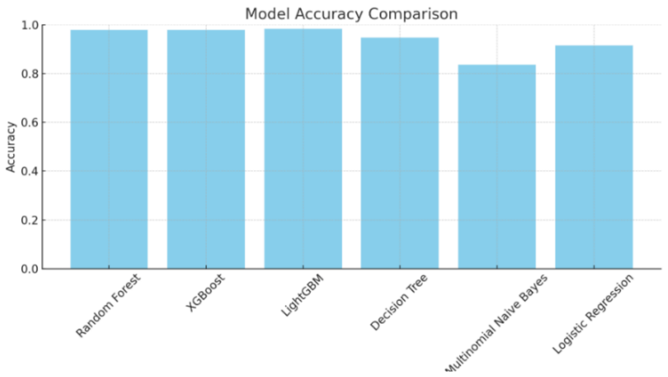


Figure 11: Logistic Regression Model Performance:Confusion Matrix, ROC Curve, and Metrics

## 2.Model Comparison:



| Model | Data | Accuracy | F1-Score | Precision | Recall |
|---|---|---|---|---|---|
| Multinomial Naive Bayes | Test | 0.835702 | 0.835402 | 0.838160 | 0.835702 |
| Logistic Regression | Test | 0.915187 | 0.915185 | 0.915217 | 0.915187 |
| Random Forest | Test | 0.978242 | 0.978239 | 0.978453 | 0.978242 |
| XGBOOST | Test | 0.978242 | 0.978239 | 0.978453 | 0.978242 |
| LIGHTGBM | Test | 0.983422 | 0.983421 | 0.983504 | 0.983422 |
| Decision Tree | Test | 0.947010 | 0.946968 | 0.948429 | 0.947010 |



The Decision Tree Classifier shows promising effects whilst in comparison to the Multinomial Naive Bayes and Logistic Regression classifiers. While its performance is robust, specially in terms of accuracy, it does now not outperform tree-based totally fashions like Random Forest, XGBoost, or LightGBM. These ensemble models, which combine more than one choice bushes, deliver appreciably better effects. The experimental effects imply that ensemble classifiers, which includes Random Forest and XGBoost, outperform man or woman classifiers in terms of accuracy, precision, do not forget, and F1-rating, as proven in the consequences. Among the ensemble fashions, LightGBM provides the first-rate overall performance, achieving an accuracy of ninety eight.34%, with excessive precision, bear in mind, and F1-score. The Random Forest and XGBoost models also carry out exceptionally properly, with accuracy scores of ninety seven.Eighty two%. These models are well-ideal for the faux job offer detection challenge because of their balanced metrics, but LightGBM slightly edges out the others in standard performance. Although Multinomial Naive Bayes and Logistic Regression provide affordable consequences, their performance is notably lower in comparison to the tree-based totally models. In precise, Multinomial Naive Bayes struggles with decrease accuracy and F1-ratings, making it less appropriate for this mission.

In end, LightGBM may be seemed because the simplest version for detecting faux task gives, accompanied intently by using Random Forest and XGBoost. These fashions have confirmed no longer best excessive accuracy but additionally sturdy precision and take into account, making them dependable alternatives for deployment in this state of affairs.

# VII. Discussion of the Result

The analysis of different types of devices learning models for fake offers detection indeed shows that LightGBM, Random Forest and XGBoost outperform the other classifiers Multinomial Naive Bayes and Logistic Regression. All these 3 ensemble models, which are based on decision trees, are best at capturing complex non-linear relationships within the data, which is a crucial factor in the detection of fraudulent activity postings. In particular, LightGBM reached the best accuracy, precision, recall and F1-score among all models. This higher performance can be largely explained by its ability to deal with large volumes of datasets and its ability to use gradient boosting that allows it to concentrate on hard-to-classified instances and improve them. Likewise, the ensemble method of combining several weak learners (decision trees) also benefits Random Forest and XGBoost since this promotes more generalization and lowers the chances of overfitting as compared to single classifiers.

These models have a great advantage in working with unbalanced datasets. In the case of fake offers detection, the proportion of fake offers is likely to be smaller compared to real willing to pay offers, leading to an imbalance problem. Random Fo ensemble techniques such

## VIII. Model Limitation

these models are expected to behave well in detecting fake assignments, but some situations may test their performance. Immediately,If new data contains unstructured or noisy information (e.g., irrelevant words, misspellings, or fraudulent language), models such as Random Forest and XGBoost may struggle to maintain accuracy in a higher position. This can be mitigated by more data preprocessing or advanced natural language processing techniques.If new spurious tasks appear that do not match previously observed patterns, the model may struggle to recognize these new patterns. This is because tree-based models, although robust, rely heavily on patterns found in the training data. One solution could be to periodically retrain the models with updated data or to use online learning to adjust new models in real time.

## IX. Conclusion

We have evaluated several machine learning models for the task of detecting fake job offers. The results demonstrate that LightGBM, Random Forest, and XGBoost outperform simpler models such as Multinomial Naive Bayes and Logistic Regression, delivering higher accuracy, precision, recall, and F1-score. Among the ensemble methods, LightGBM stands out as the most effective model, achieving the highest overall performance due to its ability to handle large datasets efficiently and effectively manage class imbalance. Despite their strong performance, the tree-based models have certain limitations, such as computational expense and sensitivity to noisy or sparse data. While LightGBM is the most computationally efficient among the ensemble models, all three models require careful tuning to achieve optimal results in real-world scenarios. The findings suggest that tree-based ensemble methods are highly suitable for fake job offer detection, given their robustness and ability to generalize well. However, these models should be continuously updated and fine-tuned to adapt to emerging patterns in fake job offers. In conclusion, LightGBM can be considered the best model for this task, though Random Forest and XGBoost also offer strong alternatives depending on specific system requirements and performance needs.

## References

[1] S. Dutta and S. K. Bandyopadhyay, "Fake Job Recruitment Detection Using Machine Learning Approach," International Journal of Engineering Trends and Technology (IJETT), vol. 68, no. 4, pp. 48-53, 2020. doi: 10.14445/22315381/IJETT-V68I4P209.

[2] S. Dutta and S. K. Bandyopadhyay, "Fake Job Recruitment Detection Using Machine Learning Approach," International Journal of Engineering Trends and Technology (IJETT), vol. 68, no. 4, pp. 48-53, Apr. 2020. doi: 10.14445/22315381/IJETT-V68I4P209.

[3] B. Alghamdi and F. Alharby, "An Intelligent Model for Online Recruitment Fraud Detection," Journal of Information Security, vol. 10, no. 3, pp. 155–176, 2019. doi: 10.4236/jis.2019.103009.

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