FAKE JOB POSTING DETECTION

Presented by:

Neha Reddy Palnati 002337926

Krishnan Narayanan 002354016

INTRODUCTION

Job hunting today is more digital than ever. Platforms like LinkedIn and Indeed have made it incredibly easy to find opportunities with just a few clicks. But along with that convenience comes a serious downside: a rise in fake job postings. These scams can trick people into giving away personal details, paying bogus fees, or worse, falling into identity theft traps.

The usual ways of spotting these fake listings, like keyword filters or manual checks, just don’t cut it anymore. Scammers are getting smarter, and we need smarter tools to keep up. That’s where machine learning and deep learning come in. These AI-powered methods can dig through huge amounts of data, spot strange patterns, and catch the subtle red flags that people might miss, like salary offers or fake company info.

In this project, we dive into how artificial intelligence is being used to fight back against job scams. From simple models like logistic regression to powerful neural networks, we explore how these tools are being applied in the real world to make job hunting safer for everyone.

PROBLEM STATMENT

As the job market becomes increasingly digital, so do the tactics of cybercriminals exploiting it. Fake job postings have grown not just in number but also in complexity, often crafted to appear highly legitimate. These scams can lure even experienced job seekers, leading to serious consequences like data breaches, financial fraud, and compromised personal security. What makes the problem more challenging is the adaptability of these scams; they continuously evolve, making them difficult to detect using static, rule-based systems or manual reviews.

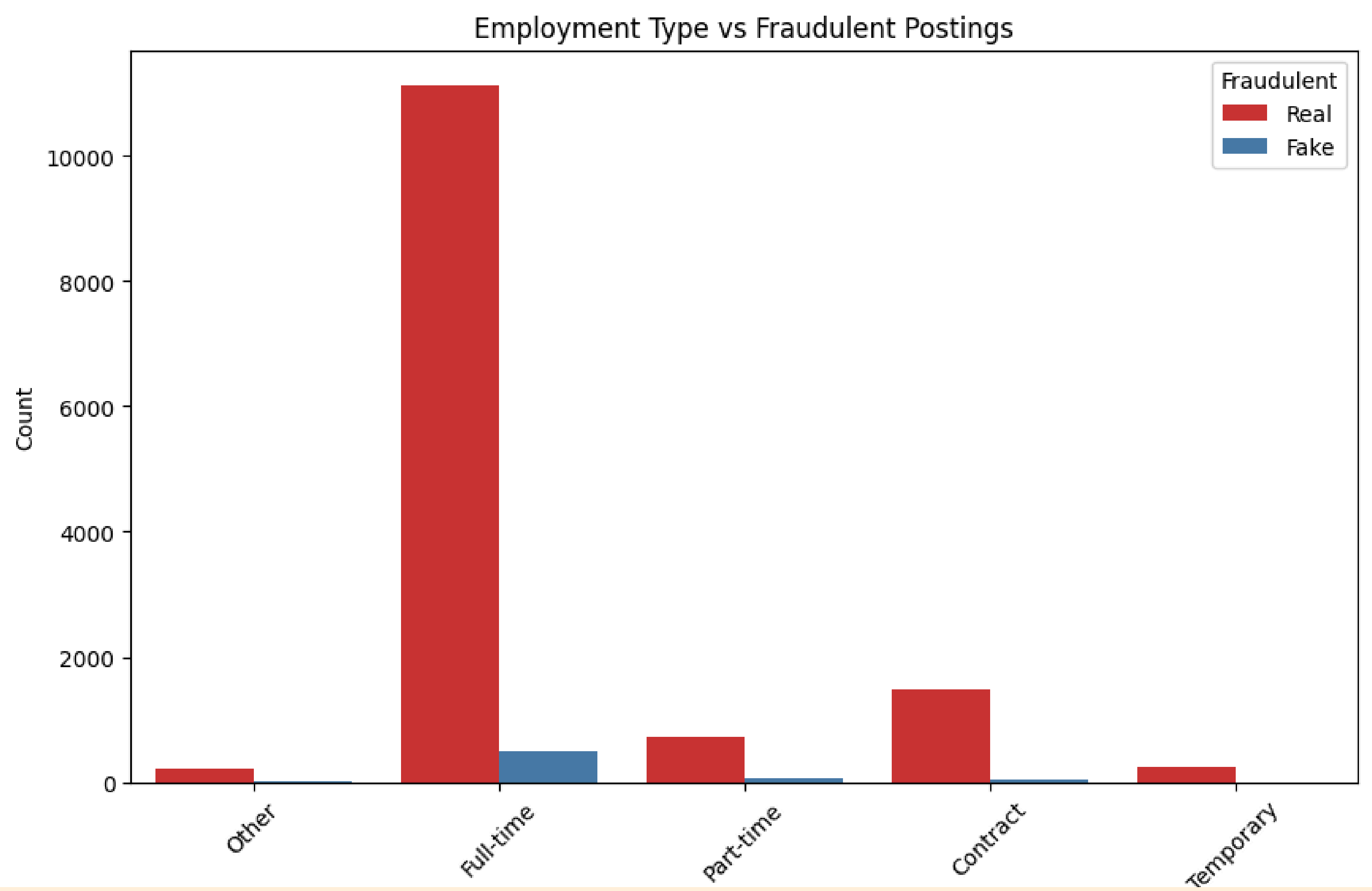
Despite efforts to flag suspicious listings, many fraudulent postings slip through the cracks due to their subtlety and sophistication. This growing threat calls for smarter solutions that can learn from patterns, detect nuanced anomalies, and scale across massive volumes of job data. Machine learning and deep learning models hold promise, but there's still a need to refine these technologies to make them more accurate, explainable, and resistant to manipulation.

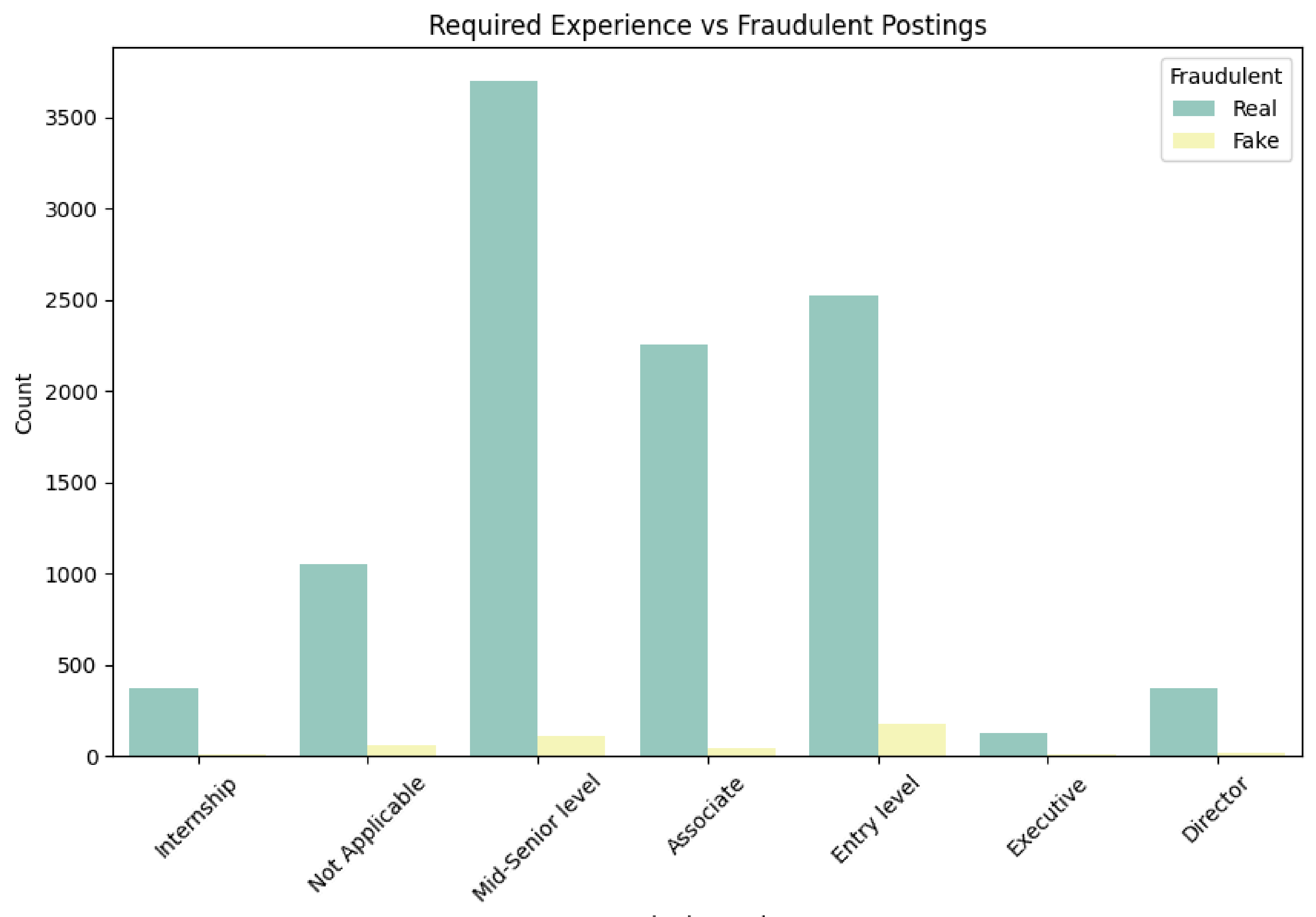
OBJECTIVE

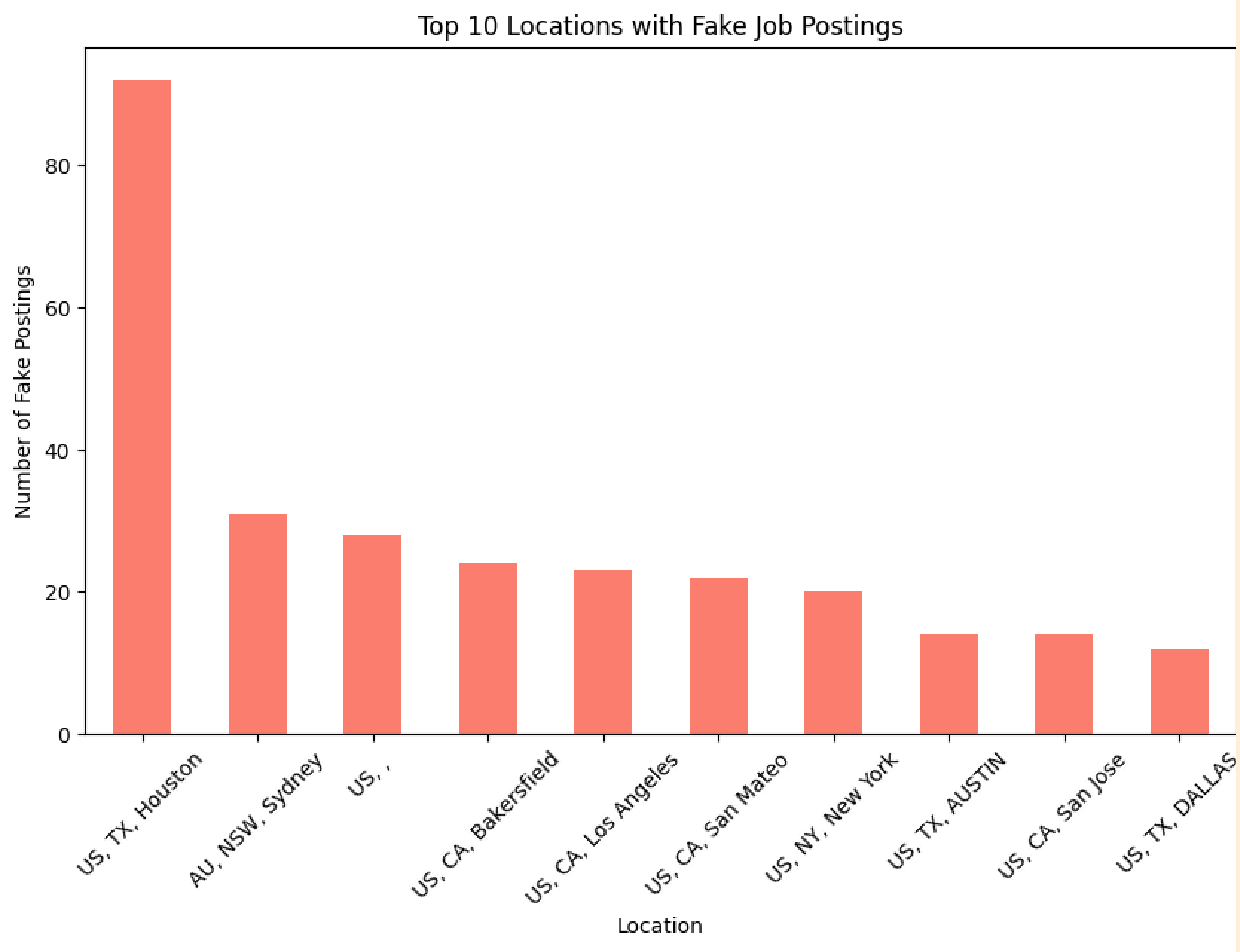
The primary objective of this project is to design and implement an intelligent system capable of detecting fake job postings across multiple platforms. We aim to harness the power of machine learning and deep learning techniques, including Long Short-Term Memory (LSTM) networks, Random Forests, XGBoost, and more, to identify and flag fraudulent job listings. By leveraging a combination of traditional and advanced models, our system aims to go beyond simple keyword detection, focusing on more nuanced patterns such as linguistic cues, metadata inconsistencies, and suspicious company profiles.

Through a multi-layered approach, the system will analyze job ads and classify them based on risk levels, helping job seekers avoid falling victim to scams. The model will be trained to detect fraudulent elements in various formats and across large datasets, ensuring it can effectively adapt to the continuously evolving tactics used by fraudsters. The goal is to provide a scalable solution for job platform administrators and users to automatically screen and remove fake job postings, ultimately improving the overall safety of online job hunting.

VISUALIZATIONS


Employment Type vs Fraudulent Postings


Required Experience vs Fraudulent Postings

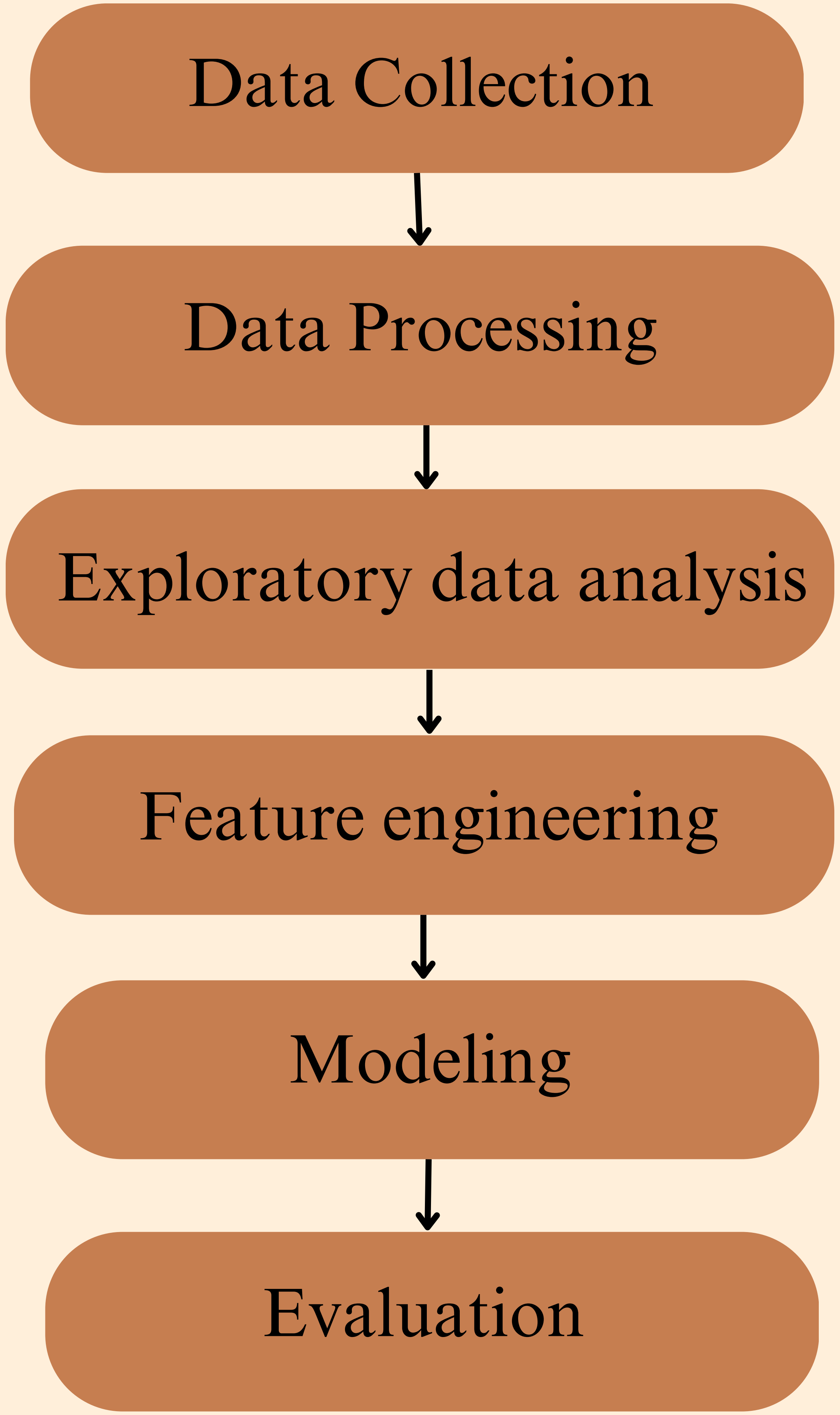

Top 10 Locations with Fake Job Postings

EXISTING WORK

A lot of research has been done on detecting fake job postings, and much of it revolves around using machine learning (ML) and deep learning (DL) techniques.

For example, Afzal et al. (2023) worked on improving feature selection and preprocessing by using methods like Chi-square, PCA, and SMOTE to balance the data, which helped them achieve a high accuracy rate of 99%. Similarly, Rofik et al. (2023) used GridSearchCV for fine-tuning SVM and Gradient Boosting models, getting accuracy rates of 98.88% and 98.08%, respectively. In the realm of machine learning, Suparna and Kumar (2023) compared different classifiers, and their deep neural network (DNN) performed the best, with 98% accuracy. Rajani et al. (2023) took it a step further, showing that ensemble classifiers, like Random Forest, were more effective than individual models in spotting fake job ads. When it comes to deep learning, Pillai (2023) used a Bi-LSTM model, which achieved an impressive 98.71% accuracy, showcasing how sequence-based models can be really powerful for this task. Tran et al. (2023) also explored deep learning using natural language processing (NLP) techniques, and their methods outperformed traditional ML approaches. While deep learning models, especially Bi-LSTMs, seem to perform better overall, they do need more computational resources. On the other hand, traditional machine learning models, when combined with smart feature selection and tuning, can still deliver strong results. A common challenge found in many studies is the issue of class imbalance, but techniques like SMOTE have been used effectively to address this.

FLOWCHART



Data Collection

Data Processing

Exploratory data analysis

Feature engineering

Modeling

Evaluation

METHODOLOGY

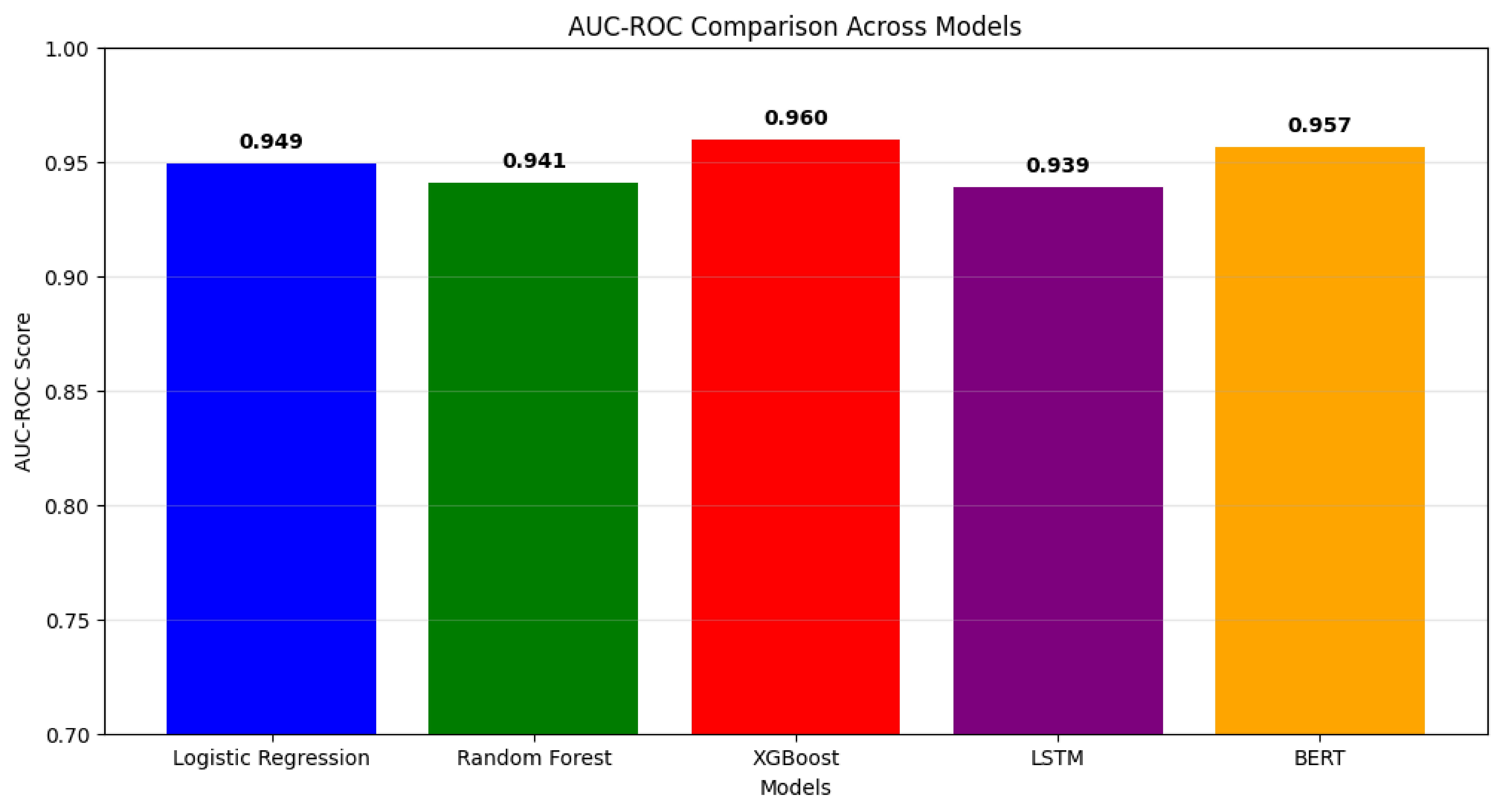
In this project, we followed a clear, step-by-step approach to identify fake job postings using machine learning (ML) and deep learning (DL) techniques.

We began by collecting a dataset from Kaggle, which included both real and fake job listings with key details such as job titles, descriptions, locations, and labels indicating whether the job was fraudulent. After gathering the data, we cleaned and prepared it by addressing missing values, removing unnecessary text, and converting the textual content into numerical data using methods like TF-IDF. We also created additional features, such as the length of the job description and whether the posting mentioned salary information. During the exploratory data analysis phase, we used visualizations and statistics to uncover patterns in the data, like common words in job descriptions, trends in job locations, and the typical length of postings. We then selected the most relevant features to train our models. We experimented with several machine learning algorithms, including Logistic Regression, Random Forest, XGBoost, and even deep learning models like Long Short-Term Memory (LSTM). To handle the class imbalance in the dataset, we applied techniques like SMOTE and class weighting. Finally, we assessed the performance of each model using metrics such as accuracy, precision, recall, F1-score, and AUC, helping us determine which model was best at identifying fraudulent job posts.

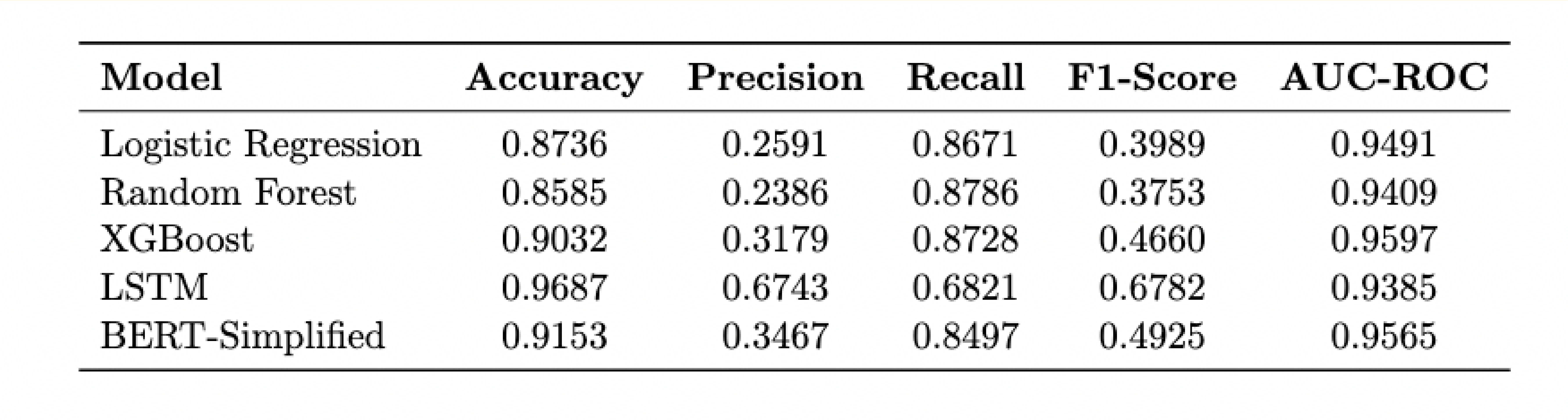
RESULTS AND CONCLUSION

In the fake job posting detection project, five models; Logistic Regression, Random Forest, XGBoost, LSTM, and a simplified BERT were evaluated on an imbalanced dataset using Accuracy, Precision, Recall, F1-Score, and AUC-ROC. XGBoost outperformed others with the highest AUC-ROC (0.9597), a strong Recall (0.8728) detecting 87.28% of fakes, and a balanced F1-Score (0.4660) with Precision (0.3179), minimizing false positives better than Random Forest. Random Forest had the highest Recall (0.8786) but the lowest Precision (0.2386) and F1-Score (0.3753), leading to many false positives. Logistic Regression (AUC-ROC: 0.9491, Recall: 0.8671, F1-Score: 0.3989) and BERT (AUC-ROC: 0.9565, Recall: 0.8497, F1-Score: 0.4925) performed well but trailed XGBoost. LSTM achieved the highest F1-Score (0.6782) and Precision (0.6743) but had a lower Recall (0.6821) and showed overfitting (training AUC-3: 0.9977 by Epoch 5). XGBoost is the best model for detecting fakes (high Recall) with strong overall performance (high AUC-ROC). Enhancing LSTM with more regularization, like increased Dropout or L2 regularization, could improve its Recall and reduce overfitting.

MODEL COMPARISION



EVALUATION METRICS



Model	Accuracy	Precision	Recall	F1-Score	AUC-ROC
Logistic Regression	0.8736	0.2591	0.8671	0.3989	0.9491
Random Forest	0.8585	0.2386	0.8786	0.3753	0.9409
XGBoost	0.9032	0.3179	0.8728	0.4660	0.9597
LSTM	0.9687	0.6743	0.6821	0.6782	0.9385
BERT-Simplified	0.9153	0.3467	0.8497	0.4925	0.9565

FUTURE WORK

Future steps include expanding the dataset with diverse job listings from various industries and regions for better generalization. Upgrading computational resources to support full BERT and deeper LSTM models could boost performance, while improving their interpretability with user-friendly explanations would enhance trust. Adding metadata and real-time detection would improve accuracy, and refining LSTM to reduce overfitting could make it more effective, possibly surpassing XGBoost. These improvements would make the system more reliable and practical.

THANK YOU