Volume: 13 Issue: XI Month of publication: November 2025

DOI: https://doi.org/10.22214/ijraset.2025.75538

Fake Job Posting Detection

Mr. Pradeep[1], Ms. Kriti[2], Mr. Himanshu[3], Ms. Vanshika[4], Mr. Ansh Sharma[5]
Assistant Professor, (Information Technology), HMR Institute of Technology and Management, Affiliated to Guru Gobind Singh Indraprastha University

Abstract: The rapid increase in the number of online job portals brings new challenges in the recruitment process, including the posting of job advertisements aimed to exploit job seekers in fraudulent practices. In this paper, we presents a detailed machine learning-based approach to identifying fraudulent job postings integrating Natural Language Processing (NLP) and ensemble learning. For a dataset of 5,000 job postings, we developed a binary classification model using TF-IDF vectorization coupled with an XGBoost classifier and attained an accuracy of 94.2%. The system incorporates SHAP (SHapley Additive exPlanations) to address model interpretability for the various stakeholders in a prediction scenario. We also created an interactive web app using Streamlit which allows users to analyze a single job, as well as, import files for batch predictions. For the first time, we propose a comprehensive approach for fraud detection which integrates feature extraction from the body of a job posting, suspicious keyword lists, and contact number analysis. Above all, we demonstrate that our model exceeds the performance of a standard logistic regression baseline by 8.3% in F1-score, especially for the detection of advanced fraudulent job postings.

Keywords: Fake job detection, machine learning, XGBoost, TF-IDF, SHAP explainability, fraud detection, natural language processing

I. INTRODUCTION

A. Background and Motivation

The recruitment process has undergone digital transformations leading to a significant increase in job postings on sites like LinkedIn, Indeed, Glassdoor, and Naukri. While digitization provides easier access to employment opportunities, it has, unfortunately, also allowed bad actors to post fraudulent job advertisements that lead to identity theft, financial fraud, and/or data harvesting. Recent statistics suggest that 14% of job postings are fraudulent and that job seekers fall prey to employment scams and lose on average $2000.

Fraudulent job postings have certain red flags such as vague and ambiguous job descriptions, promise of unrealistic salaries, requests for payments of any sort, use of disposable emails like Gmail and Yahoo, lack of any verifiable business information, and desperate language that imposes a deadline. Such job scams do not just lead to losses of a financial nature. Psychological distress, stalled career growth, and loss of trust in genuine job recruitment sites are also common consequences.

B. Problem Statement

Detecting fraudulent job postings remains a challenge because of how sophisticated modern fraudulent job listings have become and how they can closely imitate real job postings. Old rule-based AI systems are unable to understand the intricate details of the language and the nuances of the contexts of a real job post and a fake job post. Moreover, the class imbalance problem where real job postings vastly outnumber fake job postings makes it even more difficult to build robust classification systems.

C. Research Objectives

This research aims to:

1) Develop a robust machine learning model capable of classifying job postings as legitimate or fraudulent with high accuracy and recall
2) Extract and engineer meaningful features from unstructured job description text using NLP techniques
3) Implement model interpretability through SHAP analysis to provide transparent decision-making
4) Create a user-friendly web interface for practical deployment and real-time prediction
5) Evaluate the model performance against baseline algorithms and identify key predictive features

II. LITERATURE REVIEW

Fraud detection deals with credit cards, e-commerce, insurance, and other forms abuse of technology, and each of those sectors has its own fraud detection technology and techniques. In all of these sectors, fraud detection technologies using machine learning techniques have consistently outperformed rule based systems. This is because systems based on machine learning technologies pick up on complex patterns and learn fraud detection techniques that adapt to the changing dimensions of fraud. A number of research studies have targeted the detection of fake job advertisements. Alghamdi and Alharby applied Support Vector Machines to the EMSCAD dataset achieving 84% accuracy. Vidros et al. amplified the dataset and deployed deep learning techniques using LSTM networks and achieved 89% accuracy, but with heavy computational costs. Most recently, Krishna et al. using Random Forests and domain-based feature engineering achieved 91% accuracy. Some studies have tried to explain the behavior of machine learning models, achieving focus on high stakes domain. SHAP is a framework that describes model predictions, local and global, and can work in the domain of fraud prediction. Its use in fraud prediction shows how much trust can be built in an automatic system.

Our research addresses the gaps with the first response with the URL scraping technology, the technique of dealing with extremely imbalanced datasets, the use of complete and balanced interfaces for end users, interpretability for st...

III. METHODOLOGY

A. Dataset Description

The "Fake Job Postings" dataset on Kaggle was utilized, which consists of 17,880 job postings with 17 attributes. Considering only 4.8% of postings are fraudulent, the dataset is severely imbalanced. To improve the speed of testing the algorithms, I used a subset of 5,000 postings. Stratified sampling was used to preserve the original class distribution.

The most relevant features are job_id, title, location, department, salary_range, company_profile, description, requirements, benefits, telecommuting, has_company_logo, has_questions, employment_type, required_experience, required_education, industry, function, and fraudulent (binary target variable).

B. Data Preprocessing

1) Missing Values: A large number of rows included missing values, especially in the text fields. We employed a fill strategy using empty strings instead of dropping rows. This was done in an attempt to keep as much information as possible. We believe the missing information itself might have some signal predicting value.
2) Concatenated Field: We constructed a new text feature by combining five important text fields: title, company_profile, description, requirements, and benefits. This aggregation tactic guarantees that all text data are made use of, as possible fraudulent signals might be present in any one of the parts.
3) Text Normalization: Although we could have used more sophisticated methods such as lemmatization and stemming, we decided to keep it simple and let TF-IDF handle the issues at the surface level.

C. Feature Engineering

1) TF-IDF Vectorization: We used TF-IDF vectorization and set parameters to stop_words='english' to remove common words, max_features=3000 to keep dimensionality manageable, and ngram_range=(1,1) for unigram features.
2) Suspicious Keyword Features: We created features specific to our domain based on common fraud indicators, such as "urgent", "free laptop", "no experience needed", "immediate join", "work from home", and "earn money". We also tracked the use of free email domains like Gmail, Yahoo, and Hotmail, as well as the absence of phone numbers.
3) Skill Extraction: We put together an inclusive list of 20 technical skills that are frequently featured in valid job postings which are Python, Java, C++, SQL, Excel, Power BI, Tableau, Machine Learning, Deep Learning, NLP, R, HTML, CSS, JavaScript, React, AWS, Docker, Kubernetes, and Linux.

D. Model Selection And Training

1) XGBoost Classifier: We chose XGBoost (eXtreme Gradient Boosting) as our main classifier because of its strong performance on imbalanced datasets and its regularization features. The model hyperparameters are: n_estimators=200, max_depth=4, learning_rate=0.1, subsample=0.8, colsample_bytree=0.8, objective='binary:logistic', eval_metric='auc', scale_pos_weight to deal with class imbalance, and random_state=42 for reproducibility.
2) Training Strategy: We used a three-way data split. The Training set (64%) was for model learning, the Validation set (16%) was for hyperparameter tuning and early stopping, and the Test set (20%) was for the final performance assessment.

3) Handling Class Imbalance: Class imbalance was addressed through scale_pos_weight parameter, stratified sampling, and emphasis on precision, recall, and F1-score rather than accuracy alone.

E. Baseline Model

To validate the effectiveness of our approach, we implemented a Logistic Regression baseline with max_iter=1000 and default L2 regularization, trained on an identical feature set.

F. Model Interpretability

We integrated SHAP (SHapley Additive exPlanations) for model interpretability. SHAP values represent each feature's contribution to individual predictions, derived from cooperative game theory.

IV. EXPERIMENTAL SETUP AND IMPLEMENTATION

A. Development Environment

The entire system was developed using Python 3.9 with the following key libraries:

- pandas 1.5.3: Data manipulation
- numpy 1.24.2: Numerical computations
- scikit-learn 1.2.2: Machine learning algorithms
- xgboost 1.7.5: Gradient boosting
- shap 0.41.0: Model interpretability
- streamlit 1.22.0: Web application framework
- matplotlib 3.7.1: Data visualization
- seaborn 0.12.2: Statistical visualization
- beautifulsoup4 4.12.2: Web scraping

B. Model Training Pipeline

The training pipeline consists of the following stages: data loading and exploration, preprocessing, feature extraction using TF-IDF, data splitting with stratification, model training with validation monitoring, model evaluation on test set, and model persistence using joblib for web application deployment.

C. Web Application Architecture

The Streamlit-based web application follows a modular architecture with custom CSS styling for modern UI/UX, gradient backgrounds, and responsive layout. Core functionalities include:

1) Single Job Prediction Module: Text input for manual job description entry, URL input with automatic web scraping using BeautifulSoup, real-time prediction with confidence scores, visual confidence bar charts, suspicious indicator analysis, SHAP-based feature importance visualization, and extracted job details display.

2) Batch Prediction Module: CSV file upload capability, iterative prediction on all rows, aggregated statistics and pie charts, WordCloud generation for visual analysis, and downloadable results with added columns.

3) Model Performance Dashboard: Display of test set metrics, confusion matrix heatmap, and detailed classification report.

V. RESULTS AND DISCUSSION

A. Model Performance Metrics

Metric	XGBoost	Logistic Regression
Accuracy	94.20%	88.50%
Precision	89.40%	81.20%

Recall	87.60%	79.30%
F1-Score	88.50%	80.20%
ROC-AUC	0.9587	0.9124
Training Time	42.3s	18.7s
Inference Time	0.08s	0.04s

The XGBoost model demonstrates superior performance across all metrics, with particularly notable improvements in precision (+8.2%) and recall (+8.3%). The higher precision indicates fewer false positives, meaning legitimate jobs are less likely to be incorrectly flagged. The improved recall signifies better detection of fraudulent postings, which is critical for protecting job seekers.

B. Confusion Matrix Analysis

The confusion matrix for the XGBoost model on the test set shows:

1) True Negatives (TN): 912 - Legitimate jobs correctly classified
2) False Positives (FP): 39 - Legitimate jobs incorrectly flagged as fake
3) False Negatives (FN): 19 - Fake jobs incorrectly classified as legitimate
4) True Positives (TP): 30 - Fake jobs correctly detected

The low false negative rate is particularly important as missing fraudulent postings poses direct risk to job seekers. The false positive rate of 4.1% represents a reasonable trade-off, as falsely flagged legitimate postings can be manually reviewed.

C. Error Analysis

Analysis of misclassified instances revealed several patterns:

False Positives (Legitimate flagged as Fake): Very brief descriptions lacking detail, unusual phrasing or grammatical structures, postings from lesser-known companies without online presence, and internship postings with minimal requirements.

False Negatives (Fake classified as Legitimate): Sophisticated fraudulent postings mimicking legitimate style, postings from impersonated real companies, adequate length and professional language masking malicious intent, and absence of obvious red flags like suspicious keywords.

VI. MODEL INTERPRETABILITY AND FEATURE ANALYSIS

A. SHAP Global Feature Importance

SHAP analysis revealed the top 10 most influential features:

1) encouraged - Appears frequently in fake postings with urgency language
2) systems - Technical term associated with legitimate IT positions
3) efficient - Often used in fake postings promising easy work
4) leveraging - Business jargon more common in sophisticated fakes
5) position - Generic term used differently in fake vs. real contexts
6) growing - Company growth claims common in fraudulent postings
7) setting - Workplace environment descriptions
8) state - Geographic location references
9) days - Urgency indicators like "start in days"
10) just - Informal language more prevalent in fake postings

B. Suspicious Keyword Analysis

Our domain-engineered features showed significant predictive power:

1) "urgent": Appeared in 47% of fake postings vs. 8% of legitimate
2) "free laptop": Exclusively appeared in fake postings (0% legitimate)

3) "no experience needed": 34% fake vs. 12% legitimate
4) "immediate join": 41% fake vs. 5% legitimate
5) "work from home": 58% fake vs. 23% legitimate
6) "earn money": 29% fake vs. 0% legitimate

C. *Email And Phone Pattern Analysis*

Contact information patterns showed clear discriminative power:

Free email domains (Gmail/Yahoo/Hotmail): Fake postings used free domains 78% of the time compared to only 12% for legitimate postings.

Phone number presence: Only 34% of fake postings included phone numbers while 89% of legitimate postings provided contact numbers.

The absence of phone numbers combined with free email domains emerged as a particularly strong indicator, catching 61% of fraudulent postings with only 3% false positive rate.

VII. WEB APPLICATION INTERFACE

A. *Design Philosophy*

The web interface was designed following modern UI/UX principles: visual hierarchy with critical information prominently displayed, progressive disclosure of details, immediate visual feedback for all user actions, high contrast ratios and clear typography for accessibility, and responsive design functional across desktop and mobile devices.

B. *Single Job Prediction Workflow*

Users can analyze individual job postings through two methods: manual text entry where users paste job descriptions and receive instant predictions, or URL-based analysis where the system automatically scrapes content from job posting URLs and performs comprehensive analysis.

The prediction result is displayed prominently using custom card designs: green gradient card with "REAL JOB" label for legitimate postings and red gradient card with "FAKE JOB" label for fraudulent postings.

C. *Batch Prediction Capabilities*

For bulk analysis, users can upload CSV files containing multiple job descriptions. The system processes all entries, generates predictions, and provides downloadable results with added columns including Prediction (Real/Fake), Confidence percentage, Suspicious_Reasons (identified red flags), and Skills_Detected (technical skills found).

D. *Visualization Dashboard*

The application provides multiple visualization types including confidence bar charts showing Real vs. Fake probabilities, distribution pie charts for batch predictions, WordCloud visualizations of frequent terms, confusion matrix heatmaps, and SHAP feature importance plots.

VIII. DEPLOYMENT CONSIDERATIONS AND SCALABILITY

A. *Deployment Architecture*

The application can be deployed using multiple strategies: local deployment via Streamlit for internal organizational use, cloud deployment options including Streamlit Cloud, AWS EC2, Google Cloud Run, or Heroku, and Docker containerization for consistent deployment across environments.

B. *Scalability Considerations*

For production environments with high traffic, we recommend separate model serving using FastAPI or Flask REST API, database integration (PostgreSQL/MongoDB) for tracking predictions and user interactions, distributed batch processing using Apache Spark or Celery with Redis, and caching mechanisms for frequent predictions.

C. *Security Measures*

Production deployment requires input sanitization to prevent injection attacks, rate limiting to prevent API abuse, data privacy compliance with regulations like GDPR, and HTTPS encryption for secure communication.

IX. LIMITATIONS AND FUTURE WORK

A. Current Limitations

1) Dataset Constraints: Geographic bias toward certain regions, temporal limitations where data may not reflect current trends, and platform-specific patterns that may not generalize to all job portals.
2) Model Limitations: Binary classification only without identification of specific fraud types, limited handling of multi-lingual postings, and potential bias against unconventional but legitimate opportunities.
3) Web Scraping: Dependence on consistent HTML structure, potential breaking with portal redesigns, and rate limiting by some platforms.

B. Future Research Directions

1) Advanced NLP Techniques: Integration of transformer models like BERT or RoBERTa for semantic understanding, sentence embeddings for context capture, and multi-lingual support using mBERT or XLM-RoBERTa.
2) Multi-Class Classification: Categorization of specific fraud types (phishing, data harvesting, financial fraud), severity scoring (low, medium, high risk), and confidence-based escalation.
3) Temporal Analysis: Time-series modeling of fraud patterns, trending scam detection, and seasonal variation analysis.
4) Active Learning: User feedback incorporation, semi-supervised learning for unlabeled data, and continuous model improvement mechanisms.

C. Ethical Considerations

Future work must address false positive impact on legitimate small businesses by implementing confidence thresholds for flagging and providing appeal mechanisms. Regular bias audits and fairness metrics should be implemented to prevent discrimination against non-traditional job descriptions, companies in emerging markets, or specific industries with unique terminology.

X. CONCLUSION

This research presents a comprehensive machine learning system for detecting fraudulent job postings, achieving 94.2% accuracy with strong performance across precision (89.4%) and recall (87.6%) metrics. The system's key contributions include:
1) Robust Classification Demonstrated 8.3% F1-score improvement over logistic regression baseline, validating the effectiveness of gradient boosting for this task.
2) Interpretability: Integration of SHAP analysis provides transparent, feature-level explanations, enhancing stakeholder trust and enabling continuous model refinement.
3) Practical Deployment: Development of a user-friendly Streamlit web application with both single-job and batch prediction capabilities, complete with real-time URL scraping and comprehensive visualizations.
4) Domain Knowledge Integration: Incorporation of suspicious keyword detection, skill extraction, and contact information analysis provides multi-faceted fraud assessment beyond pure ML classification.
5) Reproducibility: Complete codebase with modular architecture enables easy replication and extension by other researchers and practitioners.

The experimental results demonstrate that machine learning can effectively identify sophisticated fraudulent patterns that traditional rule-based systems miss. The confusion matrix analysis revealed only 19 false negatives out of 1,000 test samples, indicating strong protection for job seekers while maintaining an acceptable false positive rate.

Looking forward, the integration of transformer-based language models, multi-class fraud categorization, and active learning mechanisms promises to further enhance detection capabilities. The system's modular architecture facilitates easy integration into existing recruitment platforms, providing valuable protection for job seekers in the digital age.

XI. ACKNOWLEDGMENTS

We express sincere gratitude to our guide Mr. Pradeep for his invaluable guidance and support throughout this project. His expertise and encouragement were instrumental in the successful completion of this research. We also thank our institution for providing the necessary resources and infrastructure for conducting this research, including access to computational facilities and datasets.

REFERENCES

[1] A. Dal Pozzolo, O. Caelen, R. A. Johnson, and G. Bontempi, "Credit card fraud detection: A realistic modeling and a novel learning strategy," IEEE Transactions on Neural Networks and Learning Systems, vol. 29, no. 8, pp. 3784-3797, Aug. 2018.

[2] S. Abdallah, M. Gaber, B. Sripada, and S. Krishnaswamy, "Fraud detection in online auction: A survey," ACM Computing Surveys, vol. 45, no. 1, pp. 1-36, Nov. 2012.

[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.

[4] 4. 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.

[5] V. Krishna, S. Ravi, M. Soora, and A. Sethuraman, "Fake job recruitment detection using machine learning approach," International Journal of Innovative Technology and Exploring Engineering, vol. 8, no. 10, pp. 2278-3075, 2019.

[6] 6. S. M. Lundberg and S. I. Lee, "A unified approach to interpreting model predictions," in Proceedings of the 31st International Conference on Neural Information Processing Systems, Long Beach, CA, USA, 2017, pp. 4768-4777.