International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 10 Issue: 03 | March - 2026 SJIF Rating: 8.659 ISSN: 2582-3930

A Comparative Study on Fake Job Post Prediction Using ML Technique

Mr. Suyog S. Dhoot[1], Darshan Patil[2], Ritesh Patil[3], Vikas Nikam[4], Yash Pawar[5]

[1] *Information Technology Department, K. K. Wagh Polytechnic, Nashik*
[2] *Information Technology Department, K. K. Wagh Polytechnic, Nashik*
[3] *Information Technology Department, K. K. Wagh Polytechnic, Nashik*
[4] *Information Technology Department, K. K. Wagh Polytechnic, Nashik*
[5] *Information Technology Department, K. K. Wagh Polytechnic, Nashik*

Abstract - The growth of online recruitment platforms has created many job opportunities, but it has also increased the number of fake job postings. Fraudulent advertisements often try to collect personal information or money from job seekers by offering false promises. To solve this problem, this paper presents FakeJobPrediction, a hybrid system for detecting fake job posts. The system combines a Deep Neural Network (DNN) model with a rule-based engine to classify job listings as real or fake.

The model is trained using the EMSCAD dataset from Kaggle along with additional synthetic data to improve performance. Natural Language Processing (NLP) techniques such as text cleaning, tokenization, stop-word removal, and TF-IDF vectorization are applied to process job descriptions. The DNN identifies complex text patterns, while the rule-based engine detects suspicious indicators like unrealistic salaries, vague content, and generic contact details.

The hybrid approach improves detection reliability and achieves an accuracy of 98.67% on the test dataset. The system is deployed using a web-based architecture that provides real-time predictions with confidence scores. FakeJobPrediction offers a practical and effective solution to protect job seekers from online recruitment fraud.

Key Words: Fake Job Prediction, Deep Neural Network (DNN), Rule-Based Engine, Natural Language Processing (NLP), EMSCAD Dataset, TF-IDF, Hybrid Classification Model, Online Recruitment Fraud.

1. INTRODUCTION

Online recruitment platforms have become a common way for companies to hire employees. However, the increase in online job postings has also resulted in a rise in fake job advertisements. These fraudulent postings are created to mislead applicants, collect personal information, or demand illegal payments. Because of this growing issue, there is a need for an automated system that can identify fake job posts accurately.

The proposed system, FakeJobPrediction, uses the Employment Scam Aegean Dataset (EMSCAD) available on Kaggle along with additional synthetic data for training. The system applies Natural Language Processing (NLP) techniques to analyze job-related information such as job descriptions, salary details, and company data. A Deep Neural Network (DNN) model is used to learn patterns from textual data, and a rule-based engine is integrated to detect predefined suspicious conditions. By combining machine learning with rule-based validation, the system classifies job postings as genuine or fraudulent with high accuracy and improved reliability.

2. LITERATURE SURVEY

Several researchers have investigated machine learning and natural language processing techniques for job fraud detection and similar text classification tasks. These studies highlight the importance of model selection and data preprocessing in achieving robust performance. Kumar and Reddy (2022) explored the use of classical decision tree classifiers and ensemble methods such as Random Forest for detecting fraudulent job postings. Although the Random Forest model outperformed the Decision Tree with an accuracy of approximately 90%, it demonstrated challenges in handling high-dimensional feature spaces due to the sparsity and complexity inherent in textual job descriptions.

Singh et al. (2021) employed Natural Language Processing (NLP) for preprocessing text data, including tokenization, stop-word removal, and Term Frequency Inverse Document Frequency (TF-IDF) vectorization. By applying Support Vector Machines (SVM) and Naïve Bayes classifiers, their study found that SVM achieved superior performance with an accuracy of around 92%, underscoring the significance of effective text

representation and feature extraction in classification tasks.

Zhang and Liu (2020) implemented a Deep Neural Network (DNN) architecture composed of multiple dense layers to capture complex semantic patterns from job description text. Their approach achieved an accuracy of approximately 97%, illustrating the strength of deep learning models in learning non-linear relationships and contextual information within high-dimensional text data.

Earlier, Patel and Shah (2019) evaluated algorithms such as K-Nearest Neighbors (KNN) and Decision Trees on a smaller dataset. While KNN demonstrated reasonable performance on limited data, it exhibited scalability and computational challenges when applied to larger datasets due to increased time complexity and sensitivity to feature dimensionality.

These studies collectively indicate that while traditional machine learning models can offer reasonable performance for binary text classification, deep learning–based approaches, particularly those capable of modeling higher-order text features, generally achieve better accuracy and robustness. Our work builds on these findings by integrating a hybrid approach combining a DNN model with a rule-based validation engine to further improve detection accuracy and system reliability.

3. PROPOSED METHODOLOGY

The proposed system, FakeJobPrediction, uses a hybrid approach that combines a Deep Neural Network (DNN) model with a rule-based validation engine to find and flag fraudulent job postings. This approach includes steps like data collection, preprocessing, feature extraction, model training, rule-based validation, and final prediction.

A. Data Collection

The main dataset for this research is the Employment Scam Aegean Dataset (EMSCAD). It contains labeled job postings that are marked as either genuine or fraudulent. To make the model more applicable to real situations, we added artificially created job postings to represent different fraud patterns. This combined dataset provides a better balance of both genuine and fraudulent job postings.

B. Data Preprocessing

Since job postings mainly consist of unstructured text data, we apply Natural Language Processing (NLP) techniques. The preprocessing steps include:

- Removing missing and duplicate values
- Normalizing text (changing it to lowercase and getting rid of punctuation)
- Tokenization
- Removing stop words
- Lemmatization
- Vectorization using TF-IDF transformation

These steps help eliminate noise and turn unstructured text into structured numerical data that's suitable for model training.

C. Feature Extraction

The key features used in the system are:

- Job title
- Company name
- Location
- Salary range
- Employment type
- Job description

We transform textual features into high-dimensional vectors using TF-IDF, while we encode structured features appropriately. The final feature matrix conveys both semantic and contextual information about job postings.

D. Deep Neural Network (DNN) Model

We implement a Deep Neural Network (DNN) to capture complex textual patterns. The architecture consists of:

- Input layer that matches TF-IDF feature dimensions
- Multiple dense hidden layers with ReLU activation
- Dropout layers to prevent overfitting
- Output layer with Sigmoid activation for binary classification

We train the model using binary cross-entropy loss and optimize it with the Adam optimizer. We evaluate its performance using accuracy, precision, recall, and F1-score metrics.

4. SYSTEM ARCHITECTURE

1. Overview

FakeJobPrediction follows a three-tier architecture:

1. Presentation Layer (Frontend)
2. Application Layer (Backend API)
3. Intelligence Layer (Hybrid Detection Engine)

The system integrates a Deep Neural Network (DNN) with a Rule-Based Engine to improve accuracy and reliability.

2. Architecture Diagram

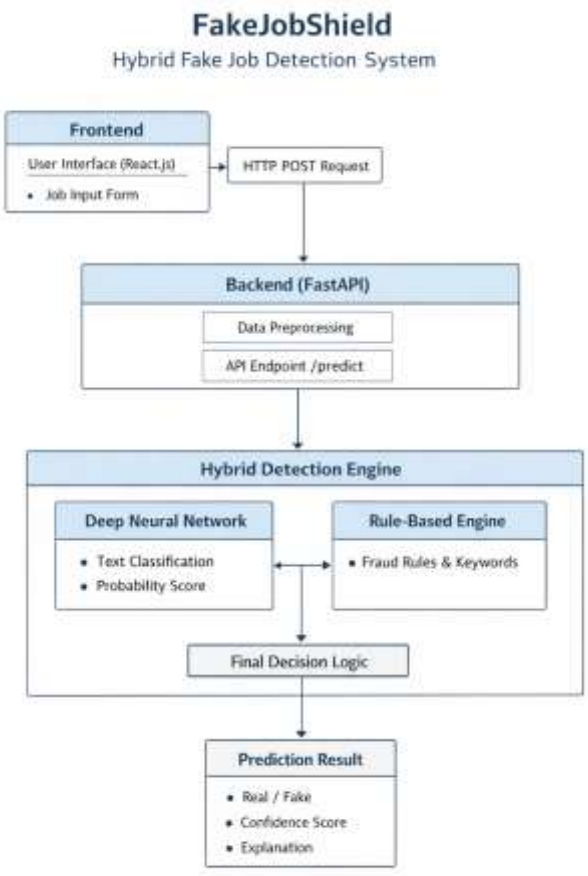


Fig-1:Architecture diagram of FakeJobPrediction

The architecture of the FakeJobPrediction system is designed as a hybrid web-based fake job detection framework that integrates a frontend interface, backend processing layer, and an intelligent hybrid detection engine. At the frontend level, developed using React.js, users enter job-related information such as job title, company name, location, salary, and job description through an interactive form. Once submitted, the frontend sends the collected data to the backend via an HTTP POST request. The backend, implemented using FastAPI, acts as the central processing unit of the system. It first performs data preprocessing on the received text, including cleaning, normalization, tokenization, and formatting to make the input suitable for analysis. The processed data is then passed to the hybrid detection engine, which forms the core of the system's intelligence.

The hybrid detection engine combines a Deep Neural Network (DNN)–based text classification model with a rule-based fraud detection module. The DNN analyzes linguistic patterns and contextual features within the job description to estimate the probability that a posting is fake or genuine, producing a confidence score. In parallel, the rule-based engine evaluates the job information against predefined fraud indicators such as unrealistic salary offers, suspicious keywords, missing company details, or abnormal job characteristics. The outputs from both the DNN and the rule-based module are then integrated through a final decision logic layer, which determines the overall classification. If either the DNN predicts a high likelihood of fraud or multiple fraud rules are triggered, the job posting is labeled as fake; otherwise, it is considered legitimate. Finally, the backend returns the prediction result—including classification (real or fake), confidence score, and explanation—to the frontend, where it is displayed to the user. This layered hybrid architecture enhances detection accuracy, reliability, and explainability while enabling real-time deployment as a scalable web application.

3. Component Description

A. Presentation Layer (Frontend)

Technology Used**:** React.js

Responsibilities:

- Accept job inputs:
  - Title
  - Company
  - Location
  - Salary
  - Description
- Validate required fields
- Send data to backend via REST API
- Display prediction result:
  - Real / Fake
  - Confidence score
  - Explanation

B. Application Layer (Backend)

Technology Used**:** FastAPI

Responsibilities:

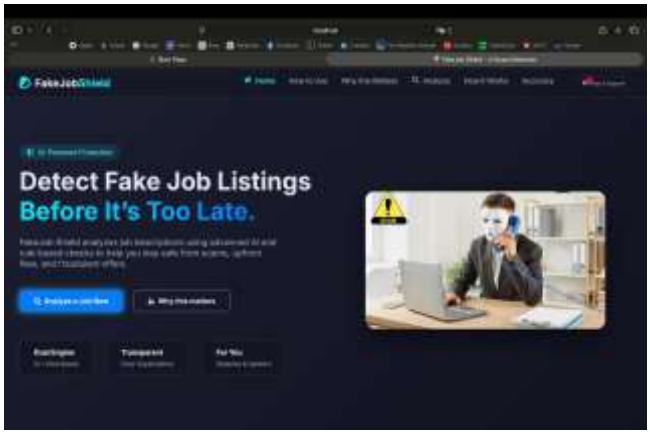
- Receive HTTP POST request (/predict)
- Validate input JSON
- Perform text preprocessing:
  - Lowercasing
  - Removing special characters
  - Tokenization
  - Padding sequences
- Forward processed data to Hybrid Detection Engine
- Combine outputs and return final result .

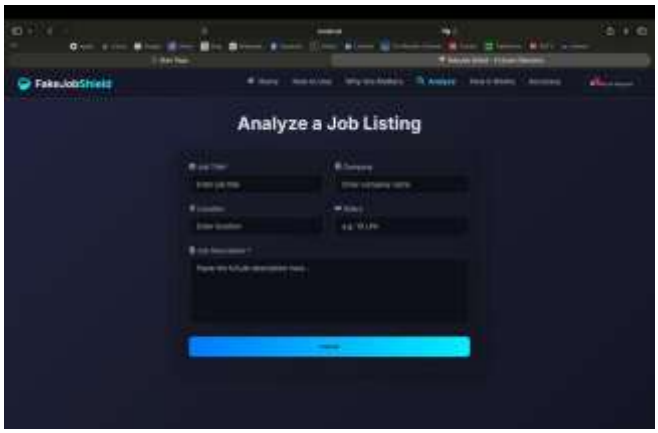
5. CONCLUSIONS & RESULT

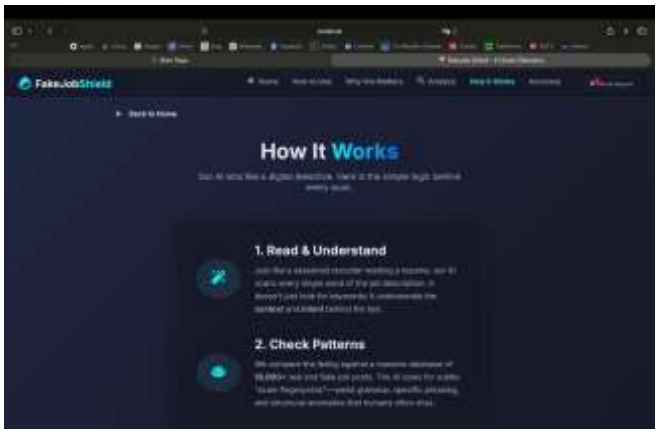
Fake job postings on online recruitment platforms pose significant risks to job seekers, including financial scams and data theft. FakeJobPrediction addresses this by integrating a Deep Neural Network (DNN) with a rule-based engine for robust detection.

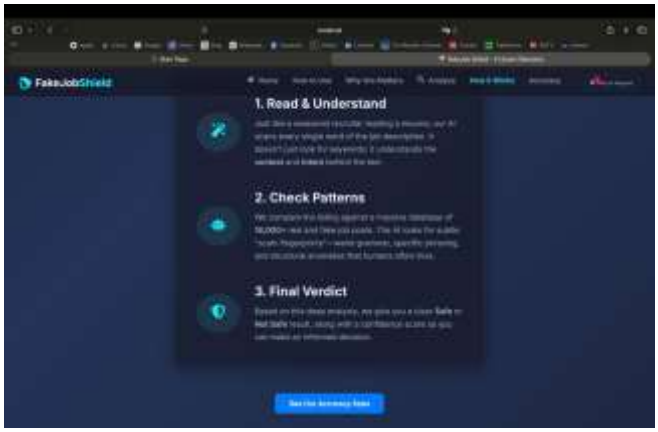
Trained on the Kaggle EMSCAD dataset augmented with synthetic examples, it employs NLP preprocessing (e.g., TF-IDF vectorization) to analyze job descriptions. The DNN uncovers subtle textual patterns, while rules flag anomalies like unrealistic salaries or vague requirements.

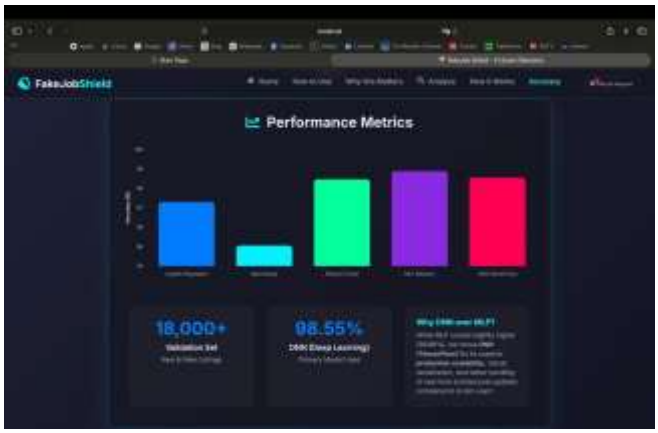
This hybrid model delivers 98.67% accuracy, outperforming standalone approaches through combined ML insight and deterministic checks. FakeJobPrediction thus offers a scalable, reliable shield against recruitment fraud.











ACKNOWLEDGEMENT

We would like to express our sincere gratitude to our project guide and faculty members of Department of Information Technology for their valuable guidance, continuous support, and encouragement throughout the development of the *FakeJobPrediction* project. Their insightful suggestions and technical expertise greatly contributed to the successful completion of this work.

We are also thankful to Kaggle for providing the EMSCAD dataset, which served as a crucial resource for training and evaluating the proposed fake job prediction model.

We extend our appreciation to our institution, K.K.Wagh polytechnic, Nashik, for providing the necessary infrastructure, computing facilities, and academic environment required to carry out this research effectively.

Finally, we express our gratitude to our friends and family for their constant motivation and support during the completion of this project.

REFERENCES

1. K. Shu, A. Sliva, S. Wang, J. Tang, and H. Liu, "Fake News Detection on Social Media: A Data Mining Perspective," *ACM SIGKDD Explorations Newsletter*, vol. 19, no. 1, pp. 22–36, 2017.
2. J. Pennington, R. Socher, and C. Manning, "GloVe: Global Vectors for Word Representation," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2014.
3. T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient Estimation of Word Representations in Vector Space," *arXiv preprint arXiv:1301.3781*, 2013.
4. I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016.
5. F. Chollet et al., "Keras," 2015. [Online]. Available: https://keras.io
6. J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," in *NAACL-HLT*, 2019.
7. Kaggle, "EMSCAD: Fake Job Postings Dataset," 2017. [Online]. Available:

https://www.kaggle.com/shivamb/real-or-fake-fake-jobposting-prediction

8. M. Sebastiani, "Machine Learning in Automated Text Categorization," *ACM Computing Surveys*, vol. 34, no. 1, pp. 1–47, 2002.
9. I. Witten, E. Frank, M. Hall, and C. Pal, *Data Mining: Practical Machine Learning Tools and Techniques*, 4th ed. Morgan Kaufmann, 2016.
10. R. Brachman and H. Levesque, *Knowledge Representation and Reasoning*. Morgan Kaufmann, 2004. (For Rule-Based Systems)