Detecting Fake Job Postings Using NLP

Aamid Mohsin, Christian Kevin Sidharta, Marvin Roopchan

Abstract

Job scams pose a significant threat to job seekers, leading to financial loss and identity theft. This research aims to develop a machine learning model to detect fraudulent job postings using the Employment Scam Aegean Dataset (EMSCAD), which contains 17,880 real-life job ads. We investigate the effectiveness of natural language processing (NLP) techniques and machine learning algorithms in distinguishing between legitimate and fraudulent job listings. By leveraging text-processing methods such as tokenization, feature extraction, and sentiment analysis, combined with classification models, we seek to identify linguistic markers and patterns indicative of scam job postings. The study provides insights into the key characteristics of fraudulent listings and evaluates the model's performance in detecting scams, contributing to the development of automated screening tools for online job platforms.

1 Introduction

1.1 Motivation

A report by the Wall Street Journal revealed that one in five job listings is either fake or not intended to be filled by job posters (WSJ, 2024). Many job seekers on average have to apply to dozens of positions, only to receive no interviews or offers. Even on trusted job hiring platforms such as LinkedIn, many fall victim to fraudulent postings, often 'ghost jobs' - listings for positions that are non-existent or already filled, and more malicious scams designed to exploit desperate job seekers, particularly in difficult economic conditions. To protect job seekers and improve the reliability of job platforms, it is important to distinguish legitimate job postings from fraudulent ones. Effective detection methods can help filter out misleading listings, ensuring that job seekers encounter genuine opportunities, reducing the risk of exploitation.

1.2 Objective

Our objective will be to develop a machine learning model to detect fraudulent job postings, using the Employment Scam Aegean Dataset (EMSCAD), a publicly available dataset containing 17,880 real-life job ads.

We will aim to answer the following question:
How effectively can NLP techniques and machine learning algorithms distinguish between legitimate and fraudulent job postings?

By leveraging text-processing techniques and classification models, we seek to identify patterns and linguistic markers that differentiate scam job postings from legitimate ones.

Our reviewers noted concerns about the novelty of our work. Although similar models have been explored in previous research, no study has compared different models and assessed their impacts on accuracy. Our study offers new insights by conducting a comprehensive comparison of feature representations (BoW vs. TF-IDF) in conjunction with various balancing techniques, highlighting optimal configurations for this specific task.

1.3 Initial Assumptions

Our assumptions include that with the use of NLP preprocessing, feature extraction, and a supervised machine learning approach using Naïve Bayes model, we will achieve higher accuracy in detecting fraudulent job postings compared to baseline classification models, including Logistic Regression and existing benchmarks. This paper will detail our research and findings.

1.4 Key Terminology

The following terms are defined for the purpose of this paper:

Supervised ML Algorithms: A category of machine learning algorithms that learn from labeled data to make predictions or classifica-

tions. In this study, we specifically utilize Naive Bayes, along with Logistic Regression.

NLP: A computational process that enables computers to understand, manipulate, and interpret human language.

Legitimate Job Posting: A job listing that is from a verified company looking to hire for a particular role. Contains a clear job description, legitimate and detailed information about the company hiring, realistic qualifications and requirements, realistic salary offers.
The listing does not contain malicious links or requests for sensitive personal information (SIN, bank details, etc.). The listing also may provide a legitimate job application process.

Fraudulent Job Posting: A job listing with malicious intent designed to deceive job seekers for fraudulent purposes (identity theft, financial fraud, phishing, etc.) or 'ghost jobs' (positions already filled/never intended to be filled)

2 Methods

2.1 Data Preprocessing

Drawing on standard text-processing procedures (Jurafsky & Martin, 2024) for classification tasks, we will use the following steps:

Text Cleaning and Normalization

- **Lowercasing**: Converting all characters to lowercase reduces the dimensionality caused by spelling variants in uppercase vs. lowercase.
- **Removing Punctuation and Special Characters**: Eliminates noise that could confuse the model when generating features.
- **Tokenization**: We will segment text into tokens (words, symbols) using NLTK, allowing us to handle each token separately.
- **Stop Word Removal**: We will drop high-frequency, low-information words (e.g., "the," "is," "and") to reduce noise.
- **Lemmatization**: We will convert words into their base or dictionary form (e.g., "running," "ran," "runs" $\rightarrow$ "run") to reduce inflectional variations and group similar terms.

Handling Class Imbalance

Because there are many more legitimate job postings than fraudulent ones in EMSCAD, we will evaluate techniques that mitigate class imbalance:

- Oversampling the minority (fraud) class (using SMOTE). SMOTE (Synthetic Minority Over-sampling Technique) generates synthetic samples for the minority class rather than simply duplicating existing samples, helping to avoid overfitting while balancing class distribution.

We will compare these methods to see which yields the best balance between detecting fraudulent posts and maintaining high overall accuracy.

2.2 Feature Extraction

To effectively classify text, as highlighted by Jurafsky and Martin (2024), we need to represent text meaningfully. Therefore, we will explore various feature extraction methods:

Bag-of-Words (BoW)

- Unigram Model: A baseline where each unique token is a feature and its value is the token count (or presence/absence).
- This approach aligns with the classic text classification framework described in (Jurafsky & Martin, 2024), particularly when discussing Naïve Bayes.

TF-IDF (Term Frequency – Inverse Document Frequency)

- We will assign more weight to terms that are frequent in a particular document but rare across the dataset to highlight discriminative words (e.g., "scam," "wire transfer," "urgent").
- TF-IDF has been recommended in literature (Jurafsky & Martin, 2024) to improve on raw counts by de-emphasizing overly common terms.
- TF-IDF was implemented using unigrams and bigrams to capture individual terms and short phrases indicative of fraudulent behaviour. This is mentioned in (Jurafsky & Martin, 2024).

2.3 Model Selection

Multinomial Naïve Bayes (Primary Model)

We will rely on a standard implementation (e.g., scikit-learn's MultinomialNB), which is well-suited for text classification tasks and frequently

Model	Accuracy	Precision	Recall	F1 Score
Bag-of-Words				
Multinomial Naïve Bayes	0.9706	0.6734	0.7630	0.7154
Logistic Regression	0.9851	0.9286	0.7514	0.8306
Neural Network	0.9804	0.7909	0.8092	0.8
TF-IDF				
Multinomial Naïve Bayes	0.9516	0.0	0.0	0.0
Logistic Regression	0.9709	1.0	0.3988	0.5702
Bag-of-Words with Oversampling				
Multinomial Naïve Bayes	0.9692	0.6318	0.8728	0.7330
Logistic Regression	0.9855	0.8623	0.8324	0.8471
Neural Network	0.9880	0.9577	0.7861	0.8635
TF-IDF with Oversampling				
Multinomial Naïve Bayes	0.9818	0.7596	0.9133	0.8294
Logistic Regression	0.9897	0.9416	0.8382	0.8869

Table 1: Performance comparison of models using Bag-of-Words and TF-IDF features, with and without oversampling. Metrics include Accuracy, Precision, Recall, and F1 Score.

cited in (Jurafsky & Martin, 2024) for its simplicity and efficiency.

- **Smoothing**: We will implement Laplace smoothing to handle zero probabilities, as discussed in (Jurafsky & Martin, 2024). To ensure that no zero probabilities occur during the calculations.

Baseline Comparisons

- **Logistic Regression**: A standard linear model that often performs well in text classification. We will compare its performance to Naïve Bayes in terms of accuracy, precision, recall, and F1 score.
- **Neural Network Deep Learning Model**: We implemented a simple neural network to explore potential gains in performance and compare with our other models.

2.4 Training and Validation (Cross-Validation)

To optimize the training and validation processes of our models, the following data partitioning strategies were used:

Data Partitioning

We will first split the dataset into training (80%) and test (20%) subsets, maintaining the original class distribution in both sets to prevent bias. This separation ensures that our final evaluation uses completely unseen data. We will split the dataset into training and test subsets (e.g., 80% training, 20% test).

k-Fold Cross-Validation

We will use k-fold cross-validation (5 folds) to obtain reliable performance metrics. In each iteration, we train on $k-1$ folds and validate on the remaining fold. We average the performance metrics (accuracy, precision, recall, F1) across all folds to get a robust estimate of how well our model generalizes.

Hyperparameter Settings

- Naïve Bayes will be configured with $\alpha = 1$, which corresponds to laplace smoothing.
- Logistic Regression will be configured with the default regularization (C=1, moderate strength) and a maximum of 1000 iterations.
- Deep Learning will be configured with 4-layer architecture with ReLU activations and sigmoid output, Adam optimization, early stopping (patience=3), and class weights to address data imbalance.

2.5 Evaluation Metrics

Jurafsky and Martin emphasize the importance of not just accuracy, but also precision and recall when dealing with skewed classes. Hence, we will measure accuracy, precision, recall, f1 score and the confusion matrix for model evaluation. Given that false negatives (failing to detect fraud) can be particularly harmful, we will pay special attention to recall and F1 scores.

In this paper, we demonstrate that NLP approaches can effectively distinguish between legitimate and fraudulent job postings. Our experiments

reveal that the performance of these methods varies depending on the machine learning model and feature vector used.

3 Results

3.1 Performance Metrics

The results in Table 1 indicate that Logistic Regression consistently outperforms Multinomial Naïve Bayes (MNB) when using both Bag-of-Words (BoW) and TF-IDF features without oversampling. Specifically, with BoW and no oversampling, Logistic Regression achieved the highest accuracy (0.9851), precision (0.9286), and F1 score (0.8306) among the three models. Although the Neural Network also performed competitively (F1=0.8), its metrics were slightly lower than those of Logistic Regression in this case.

When oversampling was applied (using SMOTE) to address class imbalance, all models generally showed improvements in recall, particularly for the minority (fraudulent) class, an essential outcome in fraud detection scenarios where failing to identify a scam can have many negative repercussions. With oversampled BoW, the Neural Network yielded the highest F1 score (0.8635), followed by Logistic Regression (0.8471) and then MNB (0.7330). This suggests that the Neural Network's ability to learn and predict can yield significant improvement in comparison to the other models when a more balanced training set is used.

Turning to TF-IDF with oversampling, Logistic Regression had the best F1 score (0.8869), alongside a high accuracy (0.9897) and precision (0.9416). The Neural Network could not be fully trained on this feature set due to insufficient system resources (RAM), and thus results are not shown in the table. However, it is plausible that, with more computational resources, the Neural Network might have matched or surpassed Logistic Regression's performance on over-sampled TF-IDF, given its success with over-sampled BoW.

Finally, while Multinomial Naïve Bayes benefits from simplicity and efficiency, it generally underperformed relative to the other models across most configurations. This is reflected in lower precision scores often below 0.70 without oversampling suggesting that MNB sometimes misclassified legitimate postings as fraudulent. Nonetheless, it remains a strong baseline for text classification tasks due to its computational speed and ease of interpretation.

To validate our results, we used 5-fold cross-validation. This process revealed no significant deviations from our original findings.

In summary:

- Logistic Regression performed best for BoW with no oversampling, beating MNB and the Neural Network on accuracy, precision, recall, and F1.
- Oversampling consistently boosted recall and F1 across all models, with the Neural Network taking the lead for oversampled BoW, and Logistic Regression retaining top performance for oversampled TF-IDF.
- Neural Network performance was limited for TF-IDF due to hardware constraints; with more RAM, it could potentially surpass Logistic Regression in this setting.

These observations underscore the importance of selecting both appropriate feature representations (BoW vs. TF-IDF) and class-balancing strategies (oversampling) when developing robust fraud detection systems.

3.2 Linguistic Characteristics of Job Listings

Across all models and feature extraction methods, several recurring patterns emerge. Fraudulent lisitngs, as identified by both Naive Bayes and Logistic Regression, frequently emphasize terms related to "data entry", "administrative" tasks, and immediate financial incentives like "earn" and "wage". This suggests a focus on quick, "too good to be true" opportunities, which may entice those in desperate financial circumstances. Furthermore, misspelled or odd terms such as "rohan", "accion", and "aker", which appear prominently in Logistic Regression results, may suggest a lack of professionalism, which can be considered a red flag as legitimate companies typically prioritize polished and error free communication. The presence of "link" and "link url" also highlights the potential for external, potentially malicious websites present in the listing that the user is encouraged to click.

In contrast, non-fraudulent listings consistently feature terms associated with professional settings and skill sets, such as "team", "client", "marketing", "digital", and "web". These listings also often include terms like "experience", "skill", and "service", indicating a focus on qualifications and professional development.

Overall, fraudulent listings tend to rely on generic

job titles and financial promises, while non-fraudulent listings emphasize specific skills, and professional requirements.

4 Limitations

Despite the promising results, there are several limitations present within our study:

- **Feature Representation**: The current feature extraction methods, Bag-of-Words and TF-IDF capture term frequency and importance, but lack the ability to model semantic relationships. Future work should explore word embeddings, like Word2Vec, to improve representation and potentially enhance model performance, particularly for Naïve Bayes.
- **N-gram Range**: Our feature extraction primarily utilized unigrams and bigrams. Future work could explore the inclusion of trigrams to capture more complex phrase patterns and potentially improve model performance.
- **Feature Scope**: Our study focused solely on performing model predictions using text-based columns within the EMSCAD dataset. The exclusion of categorical features like location, insdustry, and required experience, as well has binary features such as telecommuting and company logo presence, represents a significant limitation. Future work should investigate the impact of incorporating these features.
- **Resource Constraints**: For the TF-IDF feature set with oversampling, the Neural Network could not be fully trained due to insufficient system memory (RAM). The neural network implementation required approximately 100GB RAM for TF-IDF features, exceeding our available resources. Future work could explore more memory-efficient architectures or cloud-based solutions. As a result, its performance metrics are not reported, and we suspect that with additional computational resources, the Neural Network might have surpassed Logistic Regression on over-sampled TF-IDF data.
- **Ghost Job Postings**: Due to their indistinguishable nature from legitimate listings in terms of language and structure, these postings are characterized by employers lack of genuine intent to hire. As such, our study, reliant on textual and feature-based analysis, may struggle to detect these postings. Future research could explore the integration of real-time data such as company hiring activity, to better detect these postings.

5 Societal Implications:

Detecting fraudulent job postings is not only a technical challenge but also has significant social implications. By reducing the prevalence of scam job listings, these models can protect job seekers from financial and personal harm. However, it is crucial to balance the detection accuracy with the risk of misclassifying legitimate job postings, which can inadvertently harm reputable companies and genuine job opportunities. Hence, the development of robust, interpretable, and fair models is essential to ensure that automated screening tools contribute positively to the online job market.

In summary, our study confirms that NLP and machine learning provide powerful tools for detecting job scams. While current models yield promising results, addressing the limitations identified here will be key to deploying effective, real-world solutions.

6 References

Amruth Jithraj V. R. (2020). Recruitment Scam Dataset. https://www.kaggle.com/datasets/amruthjithrajvr/recruitment-scam

Jurafsky, D. Martin, J. H. (2024). Naive Bayes, Text Classification, and Sentiment. https://web.stanford.edu/ jurafsky/slp3/

The Wall Street Journal. (2024). Ghost Jobs: The Rise of Fake Job Listings. https://www.wsj.com/lifestyle/careers/ghost-jobs-2c0dcd4e

Pramod M. Mathapati, A.S. Shahapurkar, K.D. Hanabaratti, (2017). Sentiment Analysis using Naïve Bayes Algorithm. International Journal of Computer Sciences and Engineering, 5(7), 75-77.

H. Tabassum, G. Ghosh, A. Atika and A. Chakrabarty, "Detecting Online Recruitment Fraud Using Machine Learning," 2021 9th International Conference on Information and Communication Technology (ICoICT), Yogyakarta, Indonesia, 2021, pp. 472-477, doi: 10.1109/ICoICT52021.2021.9527477.