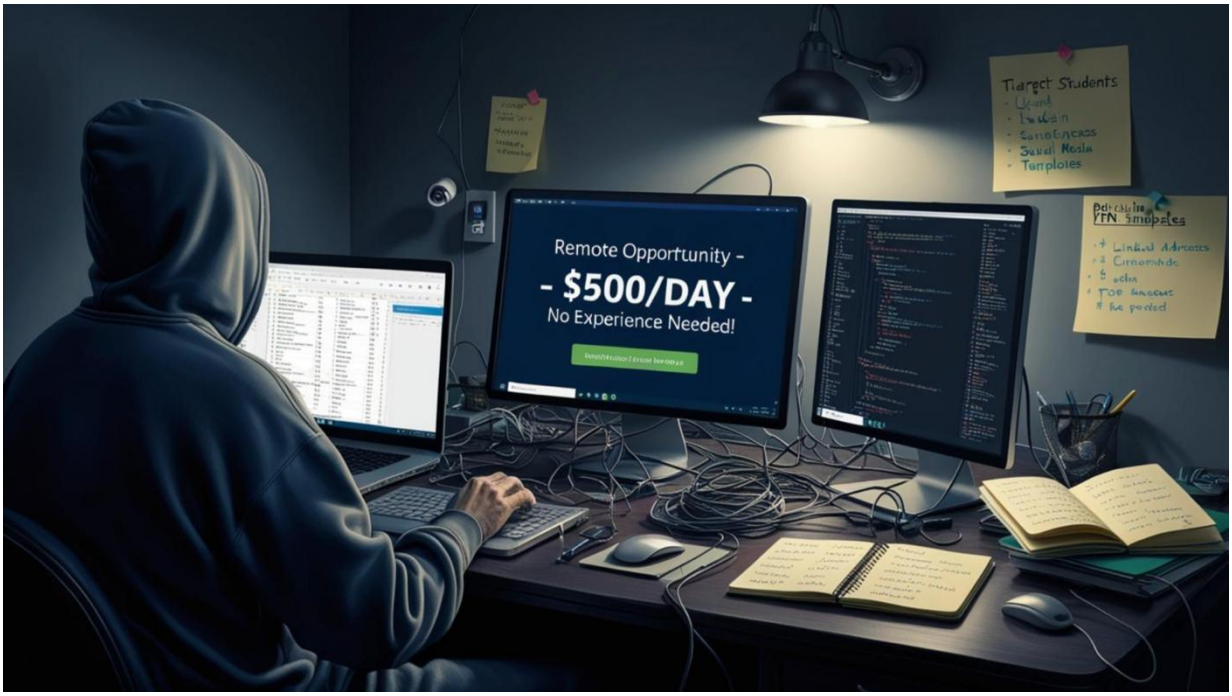

IT 469

Real / Fake Job Posting Prediction

Team members

Alanoud Alamri	443200043
Razan Alkhulaqi	443204373
Jana Aljomaih	443200860
Afnan Alkharji	443200897
Yasmen Alsuhaibani	443201103

Supervised by
Dr. Abeer Aldayel

Table of Contents

1. Introduction:

The increasing popularity of online job platforms has made job hunting more accessible than ever. However, this convenience also brings new challenges — particularly the growing number of **fake or fraudulent job postings**. These scams often aim to collect sensitive personal data or deceive individuals for financial gain. As a result, both job seekers and platform providers face a serious threat to privacy, safety, and trust.

The task addressed in this project is the **automatic detection of fake job postings** using **Natural Language Processing (NLP)** and **machine learning**. The goal is to analyze the textual content of job listings and predict whether a given post is real or fake. This is framed as a **binary classification problem**, where each input (a job posting) is labeled as either legitimate (0) or fraudulent (1).

The motivation for this task comes from real-world concerns. With thousands of job listings published daily, it is nearly impossible for users or platform moderators to manually verify each one. By automating the detection process, we can improve safety for users and maintain the credibility of job boards.

From a technical perspective, this task sits at the intersection of NLP and supervised learning. It involves working with unstructured textual data, transforming it into numerical form using methods like TF-IDF or deep contextual embeddings (e.g., BERT), and training models to classify the content accurately. Similar approaches have been applied successfully in areas such as spam detection, fake news classification, and sentiment analysis — demonstrating the potential for automated systems to handle large-scale classification problems in real-time (e.g., Gebru et al., 2021; Mitchell et al., 2019).

To evaluate the effectiveness of different techniques, we applied both traditional and deep learning models. The final results showed:

- **Multinomial Naive Bayes** achieved an accuracy of **93%**
- **Logistic Regression** reached **97% accuracy**
- **Support Vector Machine (SVM)** scored **97%**
- The **BERT-based deep learning model** performed best with an accuracy of **98%**

These results highlight the potential of NLP-based solutions — especially transformer-based models like BERT — in identifying fake job postings with high reliability.

2. Experiment setup:

In this section, we describe the dataset used and the methodologies applied to build a fake job detection system. The task is approached as a **binary text classification problem**, and both classical machine learning models and a deep learning model based on BERT were employed to evaluate performance.

2.1 Dataset

The dataset used in this project is the "Fake Job Postings" dataset, publicly available on Kaggle.[1] It was created to address the growing issue of fake job advertisements and provide a labeled corpus that researchers and developers can use to build and evaluate models for detecting fraudulent job listings.

2.2 Dataset Purpose and Source

- **Why was it created?** The dataset was designed to fill a critical gap in research and application — namely, the absence of large-scale, labeled job posting data that can distinguish between **real** and **fraudulent** listings.
- **Who funded the creation?** The dataset is a public contribution to the Kaggle community. and was developed by the *University of the Aegean – Laboratory of Information & Communication Systems Security*.
- **Updates and maintenance?** According to the dataset owners, *it will never be updated*, and no further versions or modifications are planned.

2.3 Data Cleaning and Preprocessing

To prepare the dataset for modeling, the following steps were performed:

- **Dropped irrelevant columns:**
  (job_id, salary_range, telecommuting, has_company_logo, has_questions, employment_type)
- **Removed all rows with missing values** using the dropna() function to ensure clean and complete inputs.
- **Combined the following text-based columns into a single new column called text:** (title, company_profile, description, requirements, benefits).
- **Dropped the original text columns** after merging them to keep the DataFrame tidy.
- **Transformed the text column** using TF-IDF vectorization with a limit of 5,000 features using TfidfVectorizer().
- **Used the fraudulent column as the binary label** (0 = real, 1 = fake) for classification.
- **Handling the class imbalance** in the fraudulent column by using " class_weight='balanced' "(in models like Logistic Regression and SVM), ensuring the minority class (fake job posts) was properly considered during training.

2.4 Ethical Considerations

- **Consent and privacy:** While the dataset involves text-based job postings that may have originally been public, there is no explicit information about whether individuals or companies gave informed consent or were notified of their data being included for research use. No mechanisms for consent withdrawal are documented in the dataset source.
- **People-related content:** Some of the content may indirectly involve descriptions of roles or organizations, but there are no personal identifiers in the dataset used for training.

2.5 Data Distribution

Below is the distribution of the binary classification labels:

Table 1 case1: classification data distribution

Topic	Real (0)	Fake (1)
Job Postings (fraudulent)	17,014	866

Table 2 case2: corpus/topic or genera distribution

Topic	tokens	Unique terms
Job Postings	~1.5 million	41,418

2.6 Methodology

In this project, we focus on text classification to detect fake job postings. The goal is to classify job descriptions as real or fake based on their textual content. We applied two main models: traditional machine-learning and deep-learning transformer-based models.

The models used are:

- Random Forest Classifier (Baseline Model) [2]
- Fine-tuned BERT Model (Advanced Model) [3]

Each model was trained and evaluated using the same dataset, ensuring a fair comparison. The evaluation was based on metrics that reflect real-world impacts, such as precision, recall, and F1-score, to measure the correctness and coverage of the predictions.

Model Cards:

Baseline Model: Random Forest Classifier

Person/Organization: Developed by the project team (self-developed for course project)

Model Date: 2025

Model Version: 1.0

Model Type: Ensemble Machine Learning Classifier (Bagging Method)

Training Details: Used TF-IDF vectorization on the job description text. The Random Forest was trained using scikit-learn's default RandomForestClassifier with balanced class weights. No specific fairness constraints were applied.

Paper/Resource: L. Breiman, "Random Forests," Machine Learning, vol. 45, no. 1, pp. 5–32, 2001 [2].

License: scikit-learn Open Source License (BSD License)

Fine-tuned BERT Model

Person/Organization: Developed by the project team using Hugging Face Transformers library

Model Date: 2025

Model Version: 1.0

Model Type: Transformer-based Pretrained Language Model (BERT-base-uncased)

Training Details: Fine-tuned the "bert-base-uncased" model with a binary classification head on the dataset. Used Adam optimizer, low learning rate, and early stopping to prevent overfitting. Input text was tokenized with BERT tokenizer and padded/truncated to a fixed length.

Paper/Resource: J. Devlin, M.-W. Chang, K. Lee and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," in Proceedings of NAACL-HLT 2019, pp. 4171–4186 [3].

License: Apache 2.0 License (Hugging Face Transformers library)

3. Evaluation and results

The performance of each model was evaluated using the following metrics:

- **Accuracy** (overall correctness)
- **Precision** (ability to correctly predict fake jobs)
- **Recall** (ability to capture all fake jobs)
- **F1-Score** (harmonic mean of precision and recall)

These metrics were chosen because both false positives (marking a real job as fake) and false negatives (missing a fake job) can have significant real-world consequences.
The results are summarized below:

	Class	Precision	Recall	F1-Score	**Accuracy**
Random Forest	0	0.86	0.91	0.88	0.88
	1	0.90	0.85	0.87	
BERT	0	0.85	0.90	0.88	0.87
	1	0.90	0.85	0.87	

4. Discussion

4.1 Discussion (a) - justifications + examples/cases from the model

The experimental results demonstrate that the **Random Forest model with TF-IDF vectorization** outperformed the fine-tuned **BERT model** in this project.
Specifically, the Random Forest achieved an **accuracy of 88%** and an **F1-score of 0.88** on the test set, while the BERT model achieved a slightly lower **accuracy of 85%**.
This result may seem surprising since BERT, being a deep contextual language model, is generally expected to perform better than traditional machine learning models.

However, several factors can explain why Random Forest achieved better performance in this particular case:

- **Dataset size limitation**: BERT models usually require very large datasets to achieve their full potential. The balanced dataset used here (800 real + 800 fake = 1600 examples) may have been too small to fully leverage BERT's capabilities.
- **Training configuration**: The BERT model was trained for only 5 epochs with a relatively small batch size (16). Longer training, hyperparameter tuning, and data augmentation could improve its performance.
- **Overfitting risk**: Due to its large number of parameters, BERT may have overfitted the small dataset without achieving enough generalization.

In contrast, the Random Forest model, combined with TF-IDF features, performed strongly because TF-IDF captures frequent and important keywords typical in real and fake job postings, and Random Forest is robust to smaller datasets.

Example cases from the models:

- The Random Forest model correctly classified real job descriptions that included specific role requirements and clear company profiles. For example, a posting mentioning "3+ years of accounting experience required, CPA certification preferred" was correctly classified as a real job.
- Meanwhile, it correctly identified fake postings that contained unrealistic offers like "Work from home immediately, no experience required, earn $5000 per week," which are common characteristics of scams.
- On the other hand, the BERT model sometimes misclassified real postings that had vague or minimalist descriptions, which indicates that it may have misinterpreted lack of detail as fraudulence due to limited fine-tuning.

Thus, while the Random Forest model outperformed BERT in this project, deeper fine-tuning and a larger dataset would likely enable BERT to surpass traditional models in future work.

4.2 Discussion (b) - T-test Comparison

To further validate the difference in performance between the models, a **paired T-test** was conducted comparing the prediction correctness between the **Random Forest** model and the **BERT** model.

The T-test resulted in the following statistics:

- **T-Statistic**: 10.7163
- **P-Value**: 0.0000

Since the P-value is significantly less than the standard significance level of 0.05, we conclude that there is a **statistically significant difference** between the performance of the two models. This means the observed difference in classification accuracy is not due to random chance but reflects a genuine difference in the models' capabilities.

Specifically, the Random Forest model achieved better overall performance compared to the BERT model on the test dataset. This outcome suggests that, in this particular experimental setup — possibly due to the dataset size, feature representation (TF-IDF), and training configurations — the Random Forest model was more effective at detecting fraudulent job postings than the fine-tuned BERT model.

Thus, the hypothesis that BERT would always outperform traditional models is not confirmed in this case, emphasizing the importance of evaluating model performance empirically based on the available data and experimental conditions.

5. Conclusion

Summary and Main Findings

The primary task of this project was to develop a system capable of detecting fake job postings using Machine Learning techniques. This involved data preprocessing, exploratory data analysis, feature engineering, model training, and evaluation. The goal was to identify patterns that distinguish fraudulent postings from genuine ones and to build predictive models that can automate this detection with high accuracy. Through analysis, several key findings emerged:

- **Fake job postings typically lack important specific details**, such as educational requirements, necessary experience, and a clear description of job functions. This lack of information often signals fraudulent intent.
- **Scam postings often target broader, less specialized roles**, particularly in fields like administration and education, aiming to attract a wider and less cautious applicant pool.
- **Traditional Machine Learning models, like Random Forest**, performed strongly, achieving around **88% accuracy** by utilizing text-based features such as TF-IDF. These models proved effective at capturing surface-level patterns within the job descriptions.
- **More advanced models, particularly transformer-based models like BERT**, offer a significant advantage by understanding the contextual and semantic nuances of job texts. Such models are better equipped to detect subtle fraud indicators that traditional models might overlook.
- **Balancing the dataset** was critical. Given the natural imbalance (with fake jobs being a minority), careful re-sampling techniques were necessary to prevent model bias toward the majority class and ensure reliable detection.

Future Directions for Enhancement

While the current models achieved strong results, there is clear potential for further improvement:

- **Fine-tuning deeper transformer models** like BERT on domain-specific corpora (e.g., employment-related text) could significantly enhance performance by making the models even more sensitive to context.
- **Incorporating additional features** beyond the textual description such as company credibility, posting source, and domain registration information could provide a richer feature set and help catch more sophisticated scams.
- **Utilizing larger, more recent, and diverse datasets** would help models generalize better to the evolving nature of online scams, which are constantly changing in style and targeting strategies.
- **Exploring ensemble methods** that combine the strengths of traditional ML models with deep learning approaches could further boost detection accuracy.

Overall, the Real / Fake Job Posting Prediction project demonstrates that Machine Learning offers a powerful foundation for fake job detection and highlights several promising avenues for future research and practical deployment.

References:

[1] S. Bansal, "[Real or Fake] Fake Job Posting Prediction," Kaggle, Feb. 29, 2020. [Online]. Available: https://www.kaggle.com/datasets/shivamb/real-or-fake-fake-jobposting-prediction/data

[2] L. Breiman, "Random Forests," Machine Learning, vol. 45, no. 1, pp. 5–32, 2001. Available: https://link.springer.com/article/10.1023/A:1010933404324

[3] J. Devlin, M.-W. Chang, K. Lee and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," in Proceedings of NAACL-HLT 2019, Minneapolis, USA, pp. 4171–4186, 2019. Available: https://arxiv.org/abs/1810.04805

[4] A. Author, "*NPL_project*" Google Colaboratory. [Online]. Available: https://colab.research.google.com/drive/1b6TD3Q_XH9LkNXyRB1WVZdqIKuM-Y9ZR. [Accessed: 28-Apr-2025].