Labor Trafficking Job Post Detector
Final Project Report

Francisco Terán

Sathwik Harapanahalli

December 2, 2025

Contents

1 Introduction

Online job postings are frequently used to recruit workers, but they can also be exploited for deceptive or fraudulent purposes linked to labor trafficking. Many victims are targeted through misleading advertisements that offer unrealistic benefits, require immediate relocation, or exert control over identity documents.

This project develops a prototype tool that uses Natural Language Processing (NLP) and machine learning to automatically detect potential risks in job advertisements. Although not a legal or diagnostic system, the tool provides early indicators that can help researchers, analysts, and organizations examine suspicious recruitment patterns.

2 Problem Statement

Labor trafficking often begins with misleading or coercive job advertisements. Manual inspection is time-consuming, inconsistent, and does not scale.

Goal: Build an automated text-analysis system capable of flagging potentially high-risk job postings by learning linguistic patterns associated with deceptive recruitment behaviors.

3 Project Objectives

3.1 Minimum Viable Product (MVP)

- Load a dataset of job posts.

- Preprocess and clean the text.

- Extract TF-IDF features.

- Train a simple classifier (Logistic Regression).

- Output predictions and confidence scores to a CSV file.

3.2 Target Product

- Improved preprocessing and feature extraction.

- Multiple classification model evaluations.

- Keyword-frequency summaries.

- Clearer structure and better modularization.

3.3 Reach Version

- Flask web interface for real-time prediction.

- API endpoint for programmatic access.

- Highlighting suspicious phrases for interpretability.

- Potential integration of additional datasets or NLP models.

4 Dataset

The project uses the publicly available *Fake Job Postings* dataset. It includes fields such as:

- Job title

- Company profile

- Job description

- Requirements

- Benefits

- Fraudulent label (0 = legitimate, 1 = suspicious)

A cleaned dataset, `cleaned_posts.csv`, is generated by merging and normalizing text fields.

5 Methodology

5.1 Preprocessing

Implemented in `src/preprocess.py`:

- Lowercasing

- Removal of punctuation and symbols

- Merging text fields into one unified string

- Whitespace normalization

5.2   Feature Extraction

TF-IDF vectorization with:

- Unigrams and bigrams (1–2 grams)

- Maximum 5000 features

- English stopwords removed

5.3   Model Training

Implemented in `src/train.py`:

- Logistic Regression classifier

- Model saved using `joblib`

- Vectorizer persisted for prediction

5.4   Batch Prediction

Implemented in `src/predict.py`:

- Loads the saved model and vectorizer

- Generates labels and probability scores

- Outputs `predictions.csv`

5.5   Web Application

Implemented in `src/app.py`, with frontend in `templates/index.html`. Features include:

- Text submission form

- High- vs. low-risk classification

- Confidence score display

- Highlighted suspicious phrases

6 System Architecture

User Input $\rightarrow$ Flask API $\rightarrow$ TF-IDF Vectorizer $\rightarrow$ ML Model $\rightarrow$ Prediction Output

All components run locally with minimal dependencies.

7 Results

The model identifies multiple red-flag linguistic patterns, including:

- Unrealistic salaries

- Required onsite living

- Passport confiscation or document retention

- Urgent relocation demands

Predictions include:

- Binary classification (0 = low risk, 1 = high risk)

- Confidence probability score

The web interface enhances interpretability by highlighting suspicious phrases using hand-crafted rules.

8 Discussion

This system demonstrates that simple NLP features and classical machine learning models can meaningfully detect suspicious linguistic patterns in job postings. While not a forensic or legal tool, its interpretability and modular pipeline make it highly suitable for research and education.

Future improvements may incorporate:

- Deep learning or transformer-based embeddings

- Multilingual detection

- Heatmaps of risk indicators

- Sector-specific analytics

9 Technologies Used

- Python

- pandas

- scikit-learn

- Flask

- joblib

- HTML/CSS/JavaScript

- Regex-based text cleaning

10 Project Structure

```
project/

 data/
    fake_job_postings.csv
    cleaned_posts.csv

 outputs/
    jobpost_model.joblib
    tfidf_vectorizer.joblib
    predictions.csv

 src/
    preprocess.py
    train.py
    predict.py
    app.py

 templates/
    index.html

 report/
    final_report.pdf
```

```
requirements.txt
README.md
```

11 Authors and Contributions

Francisco Terán

- Data preprocessing

- Model development

- Web interface implementation

- Documentation

Sathwik Harapanahalli

- Dataset integration

- Model implementation

- Repository organization

Both authors contributed equally to design, experimentation, and writing.

12 Conclusion

This project successfully demonstrates an end-to-end machine learning pipeline that can analyze job postings and identify potential labor trafficking risks. It integrates preprocessing, modeling, batch predictions, and a real-time interactive interface into a coherent and reproducible workflow.

13 References

- Scikit-learn Documentation: `https://scikit-learn.org`

- Flask Documentation: `https://flask.palletsprojects.com/`

- Fake Job Postings Dataset (Kaggle)

- Research literature on labor trafficking indicators