Abstract

This report details the development of a machine learning pipeline designed to detect fraudulent job postings by integrating Machine Learning Operations (MLOps), Automated Machine Learning (AutoML), and HuggingFace technologies. Leveraging the "Real or Fake Job Posting Prediction" dataset from Kaggle, the project emphasizes automation, scalability, and ethical considerations in deploying a robust model for real-world applications.

Detecting Fraudulent Job Postings

AIG130 Project 2

Group 5: Aliyyah Jackhan, Mohammed Aadil, Jonathan Chacko & Masoud Masoori

Table of Contents

[Introduction 2](#_Toc194141381)

[Problem Statement 2](#_Toc194141382)

[Pipeline Architecture and Design 2](#_Toc194141383)

[Data Ingestion and Preprocessing 2](#_Toc194141384)

[Data Ingestion 2](#_Toc194141385)

[Data Cleaning & Preprocessing 2](#_Toc194141386)

[Feature Engineering 3](#_Toc194141387)

[Text Vectorization 3](#_Toc194141388)

[Meta-feature Extraction 3](#_Toc194141389)

[Model Selection with AutoML 3](#_Toc194141390)

[Model Training and Evaluation 4](#_Toc194141391)

[Training Strategy 4](#_Toc194141392)

[Evaluation Metrics 4](#_Toc194141393)

[MLOps Integration: CI/CD and Model Monitoring 4](#_Toc194141394)

[Continuous Integration/Deployment (CI/CD) 4](#_Toc194141395)

[Model Monitoring and Maintenance 4](#_Toc194141396)

[Implementation Steps 5](#_Toc194141397)

[Ethical and Performance Considerations 5](#_Toc194141398)

[Conclusion and Future Work 5](#_Toc194141399)

# Introduction

The rise of online job platforms has resulted in an increase in fraudulent job postings, which pose significant risks to job seekers. Identifying these deceptive postings accurately is essential for protecting users. This project aims to develop an automated solution by integrating MLOps practices, AutoML tools, and HuggingFace's Natural Language Processing (NLP) capabilities to effectively detect fraudulent job postings.

# Problem Statement

Fake job postings can mislead job seekers, waste resources, and damage the reputation of platforms. By automating the detection of fraudulent job descriptions, this project helps maintain data quality and protect users. *The Kaggle Dataset we are using to classify job postings as either real or fraudulent, contains about 18,000 job postings with approximately 800 labeled as fraudulent. The dataset includes features such as job descriptions, company profiles, and employment types, providing a comprehensive basis for model training.*

**Dataset =** [*https://www.kaggle.com/datasets/shivamb/real-or-fake-fake-jobposting-prediction*](https://www.kaggle.com/datasets/shivamb/real-or-fake-fake-jobposting-prediction)

# Pipeline Architecture and Design

## Data Ingestion and Preprocessing

### Data Ingestion

To ensure data integrity and reproducibility, the dataset will be imported and version-controlled using tools such as:

* **Git and GitHub** for collaboration and tracking changes.
* **Data Version Control (DVC)** to manage dataset versions, enabling efficient experimentation and rollback.
* **Kaggle API** for automated data retrieval, ensuring up-to-date information.

### Data Cleaning & Preprocessing

To prepare the dataset for model training, several preprocessing steps will be applied:

* **Handling Missing Values**: Missing entries will be either imputed using statistical methods or explicitly labeled as "Unspecified."
* **Text Cleaning**:
  + Removal of HTML tags, special characters, and punctuation.
  + Elimination of stopwords to reduce noise.
  + Standardization via **lemmatization**, converting words to their base form.
* **Categorical Encoding**:
  + One-hot encoding for nominal categorical features (e.g., employment type, industry).
  + Ordinal encoding for ordered categorical attributes where applicable.

## Feature Engineering

### Text Vectorization

Since job descriptions are a critical component, text-based features will be transformed into numerical representations using:

* **Traditional NLP Techniques**:
  + **TF-IDF (Term Frequency-Inverse Document Frequency)** to weigh words based on importance.
  + **Count Vectorization** for frequency-based representation.
* **Advanced Deep Learning Embeddings**:
  + Leveraging **Hugging Face transformer models** like BERT or RoBERTa to extract semantic and contextual information.
  + Using **word embeddings** such as Word2Vec or FastText to enhance text representation.

### Meta-feature Extraction

Beyond textual data, additional features will be engineered to improve model performance:

* **Numerical Features**:
  + Salary range (if available), normalized for consistency.
  + Length of job descriptions, measuring verbosity.
* **Categorical Features**:
  + Employment type (e.g., full-time, part-time, contract) converted into machine-readable formats.
  + Company profile attributes such as company size, industry type, and website presence.

## Model Selection with AutoML

To automate and optimize model selection, AutoML frameworks will be employed:

* **AutoML Platforms**:
  + **H2O.ai**: Enables automatic feature selection and hyperparameter tuning.
  + **TPOT**: Genetic algorithm-based pipeline optimization.
* **Candidate Models**:
  + Traditional Machine Learning:
    - **Logistic Regression** for baseline comparisons.
    - **Random Forest & Gradient Boosting (XGBoost, LightGBM)** for balanced performance.
  + Deep Learning:
    - **LSTMs & Transformer-based models (BERT, DistilBERT)** for contextual understanding.

## Model Training and Evaluation

### Training Strategy

* The dataset will be split into **training (80%) and testing (20%) subsets**.
* **Addressing Class Imbalance**:
  + **SMOTE (Synthetic Minority Over-sampling Technique)** will be applied to generate synthetic fraudulent job postings and mitigate class imbalance.
  + **Cost-sensitive learning** to penalize misclassification of fraudulent listings.

### Evaluation Metrics

To ensure comprehensive model assessment, multiple evaluation metrics will be considered:

* **Accuracy**: General performance indicator.
* **Precision**: To measure false positive reduction.
* **Recall**: Key metric for detecting fraudulent postings.
* **F1-score**: Balances precision and recall, crucial for imbalanced datasets.
* **ROC-AUC Score**: Evaluates classification probability distributions.

## MLOps Integration: CI/CD and Model Monitoring

### Continuous Integration/Deployment (CI/CD)

To streamline the development and deployment cycle:

* **GitHub Actions or Jenkins** will automate model training and deployment.
* **Docker & Kubernetes** for scalable deployment, ensuring easy integration into real-world applications.
* **Cloud Platforms (AWS, GCP, Azure)** for hosting models with serverless functions.

### Model Monitoring and Maintenance

Once deployed, continuous monitoring will be implemented:

* **Real-time Model Performance Tracking**:
  + **Prometheus & Grafana** for live dashboards.
  + **MLflow** to log experiments and track model drift.
* **Data Drift Detection**:
  + Regular retraining schedules to adapt to evolving fraudulent patterns.
  + **Alert systems** to detect sudden spikes in misclassified fraudulent postings.

# Implementation Steps

***Step 1: Data Preprocessing using AutoML***

Employ AutoML tools to automate data preprocessing tasks, including feature selection and transformation, ensuring optimal data quality for model training.​

***Step 2: Integration of HuggingFace for Text Analysis***

Leverage HuggingFace's transformer models to extract rich, contextual embeddings from job descriptions, enhancing the model's ability to discern subtle patterns indicative of fraudulent postings.​

# Ethical and Performance Considerations

* **Model Performance:** Strive for a balance between detecting fraudulent postings (sensitivity) and correctly identifying legitimate ones (specificity), minimizing false positives and negatives.​
* **Ethical Considerations:** Ensure the model does not inadvertently discriminate against specific job categories or companies. Maintain transparency in the model's decision-making process to build trust with users.​

# Conclusion and Future Work

This project demonstrates the integration of MLOps, AutoML, and HuggingFace technologies to build an efficient, scalable, and robust machine learning pipeline for detecting fraudulent job postings. The approach not only automates the model selection process but also leverages state-of-the-art NLP methods to extract meaningful insights from unstructured text.

**Future Enhancements:**

* Expand the pipeline to incorporate additional external data sources for improved feature richness.
* Enhance model monitoring and retraining protocols.
* Explore additional transformer architectures and ensemble strategies to further boost performance.