



Assessment Report
on
“Predict Loan Default”
submitted as partial fulfillment for the award of
BACHELOR OF TECHNOLOGY
DEGREE

SESSION 2024-25

in
CSE(AIML)

By

Name : Akshat Anand

Roll Number : 202401100400022

Section: A

Under the supervision of

“Bikki Kumar”

KIET Group of Institutions, Ghaziabad

May, 2025

Identify Fake Job Postings

Title Page

To identify whether a job posting is real or fake using features like job title length, description length, and company profile information. The objective is to build a classification model that predicts the authenticity of job listings.

Introduction

In today's digital job market, many fake job postings mislead applicants and pose security risks. Detecting such fraudulent listings using data science is vital. This project uses basic job post features (title length, description length, and presence of company profile) to classify postings as fake or real. By applying machine learning techniques, we aim to automate this identification process.

Methodology

Data Loading: We used a CSV file containing job post features and a label indicating if a posting is fake.

Preprocessing: The `is_fake` column was converted from 'yes'/'no' to binary values 1/0.

Feature Selection: The following features were used: `title_length`, `description_length`, and `has_company_profile`.

Train-Test Split: The dataset was split into 80% training and 20% testing.

Model Training: A `RandomForestClassifier` was trained on the data.

Evaluation: We calculated Accuracy, Precision, and Recall, and visualized the results using a confusion matrix heatmap.

CODE

```
# Import necessary libraries
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import confusion_matrix, accuracy_score,
precision_score, recall_score
import seaborn as sns
import matplotlib.pyplot as plt
```

```
# Load the uploaded dataset
df = pd.read_csv("/content/fake_jobs.csv") # Make sure the path is
correct for your environment

# Show first few rows
df.head()
```

```
# Convert 'is_fake' column from 'yes'/'no' to 1/0 for classification
df['is_fake'] = df['is_fake'].map({'yes': 1, 'no': 0})

# Define feature columns and target column
X = df[['title_length', 'description_length', 'has_company_profile']]
y = df['is_fake']
```

```
# Split data into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(
    X, y, test_size=0.2, random_state=42
)
```

```
# Initialize and train the Random Forest model
model = RandomForestClassifier(random_state=42)
model.fit(X_train, y_train)
```

```

# Predict on test data
y_pred = model.predict(X_test)

# Calculate evaluation metrics
accuracy = accuracy_score(y_test, y_pred)
precision = precision_score(y_test, y_pred)
recall = recall_score(y_test, y_pred)

# Display results
print("Accuracy:", round(accuracy, 2))
print("Precision:", round(precision, 2))
print("Recall:", round(recall, 2))

# Create and display confusion matrix heatmap
cm = confusion_matrix(y_test, y_pred)
sns.heatmap(cm, annot=True, fmt='d', cmap='YlGnBu',
            xticklabels=['Real', 'Fake'], yticklabels=['Real', 'Fake'])
plt.title('Confusion Matrix')
plt.xlabel('Predicted')
plt.ylabel('Actual')
plt.show()

```

Output/Result

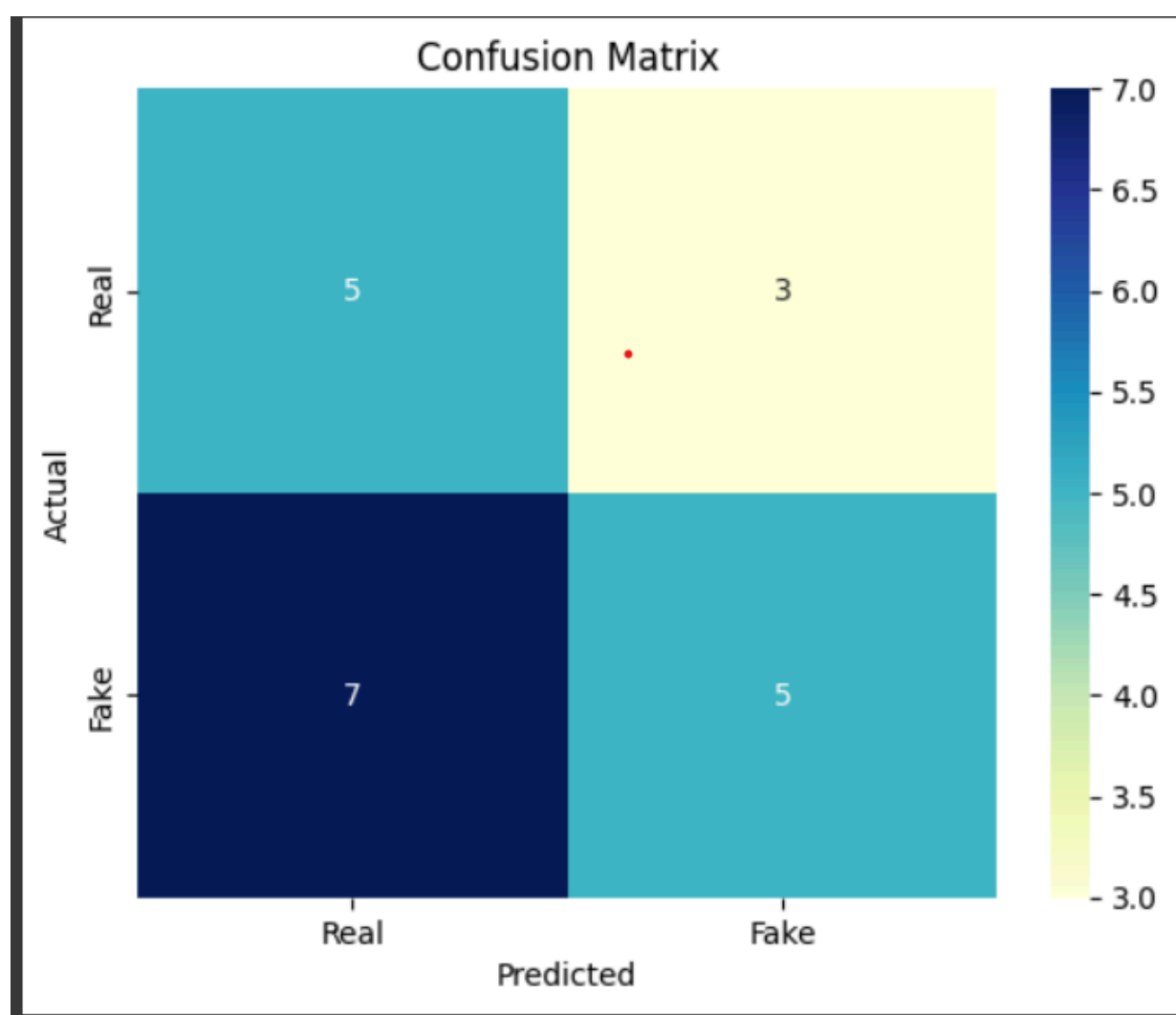
	title_length	description_length	has_company_profile	is_fake
0	72	740	1	yes
1	95	476	0	no
2	60	662	1	yes
3	34	317	0	no

```
884      0      Yes

Accuracy:
0.5
4 Precision:
0.62

Recall:
0.42
```

Visualization of Path:



References/Credits

1. **Dataset:** Provided dataset `fake_jobs.csv`
2. **Libraries Used:** scikit-learn, pandas, matplotlib, seaborn
3. **Source for Idea:** Based on real-world job fraud detection problems