Phase 2: True and Fake News detection using Natural Language Processing

Significant Program:

```
import pandas as pd
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import LabelEncoder
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import accuracy_score, classification_report

# Load your dataset
data = pd.read_csv('true.csv', encoding='latin-1')

# Show available columns
print("Columns:", data.columns.tolist())

# Select relevant columns (drop completely empty/unnecessary ones)
data = data[['Country Killed', 'Organization', 'Medium', 'Job', 'Coverage',
          'Freelance', 'Local/Foreign', 'Source of Fire', 'Type of Death',
          'Taken Captive', 'Threatened', 'Tortured']].copy()

# Drop rows with missing 'Threatened' values
data.dropna(subset=['Threatened'], inplace=True)

# Fill other missing values with "Unknown"
data.fillna("Unknown", inplace=True)

# Encode categorical columns
label_encoders = {}
for col in data.columns:
   le = LabelEncoder()
   data[col] = le.fit_transform(data[col].astype(str))
   label_encoders[col] = le

# Split into features (X) and target (y)
X = data.drop(columns=['Threatened'])
y = data['Threatened']

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

# Train classifier
model = RandomForestClassifier(n_estimators=100, random_state=42)
model.fit(X_train, y_train)
```

```
# Predictions
y_pred = model.predict(X_test)

# Evaluation
print(f"Accuracy: {accuracy_score(y_test, y_pred) * 100:.2f}%")
print("Classification Report:\n", classification_report(y_test, y_pred))

# Predict for new data (example)
sample = X.iloc[0:1]  # Use first row as a sample
prediction = model.predict(sample)
threat_status = label_encoders['Threatened'].inverse_transform(prediction)[0]
print(f"Prediction for sample journalist: {threat_status}")
```

Obtained Output:

Columns: ['Date', 'Name', 'Sex', 'Country Killed', 'Organization', 'Nationality', 'Medium', 'Job', 'Coverage', 'Freelance', 'Local/Foreign', 'Source of Fire', 'Type of Death', 'Impunity (for Murder)', 'Taken Captive', 'Threatened', 'Tortured', 'Unnamed: 17', 'Unnamed: 18', 'Unnamed: 19', 'Unnamed: 20', 'Unnamed: 21', 'Unnamed: 22', 'Unnamed: 23', 'Unnamed: 24', 'Unnamed: 25']
Accuracy: 81.43%
Classification Report:

	precision	recall	f1-score	support
1	0.83	0.91	0.87	163
4	0.70	0.52	0.60	63
5	1.00	1.00	1.00	11
accuracy			0.81	237
macro avg	0.84	0.81	0.82	237
weighted avg	0.81	0.81	0.81	237

Prediction for sample journalist: No