Submitted by: Tanmay Guha **Position:** AI/ML Intern

TITLE: FAKE NEWS DETECTION USING MACHINE LEARNING

📌 Objective:

The primary goal of this project is to build a **Fake News Detection System** using machine learning techniques. It aims to classify news articles as *fake* or *real* based on their textual content.

🗃️ Dataset:

- **Source:** The project uses two datasets:
  - Fake.csv – Contains fake news articles.
  - True.csv – Contains real news articles.
- **Labeling:**
  - Fake news labeled as 1
  - Real news labeled as 0
- **Combined Dataset:** Both datasets are merged and only the text and label columns are retained for training the model.

🧹 Data Preprocessing:

1. **Lowercasing:** All text data is converted to lowercase.
2. **Punctuation Removal:** All special characters are removed using regular expressions.

```
df['text'] = df['text'].str.lower().str.replace(r'[^\w\s]', '', regex=True)
```

🔀 Train-Test Split:

- **Training Set:** 80% of the data
- **Testing Set:** 20% of the data
- **Method:** train_test_split from sklearn

```
X_train, X_test, y_train, y_test = train_test_split(df['text'], df['label'], test_size=0.2, random_state=42)
```

Submitted by: Tanmay Guha **Position:** AI/ML Intern

🧠 Feature Extraction – TF-IDF Vectorization:

- **Tool:** TfidfVectorizer from sklearn
- **Parameters:**
    - stop_words='english': Removes common English stop words.
    - max_df=0.7: Ignores terms that appear in more than 70% of documents.

```
vectorizer = TfidfVectorizer(stop_words='english', max_df=0.7)
tfidf_train = vectorizer.fit_transform(X_train)
tfidf_test = vectorizer.transform(X_test)
```

🧪 Model: Passive Aggressive Classifier:

- **Why this model?** It's efficient for large datasets and is often used for online learning problems.
- **Training:** The model is trained on the TF-IDF transformed training data.

```
model = PassiveAggressiveClassifier(max_iter=1000)
model.fit(tfidf_train, y_train)
```

📊 Evaluation:

✅ **Accuracy:**

```
acc = accuracy_score(y_test, y_pred)
print(f"Accuracy: {acc:.2f}")
```

- **Output Example:**
- Accuracy: 0.93

☐ **Confusion Matrix:**

```
cm = confusion_matrix(y_test, y_pred)
```

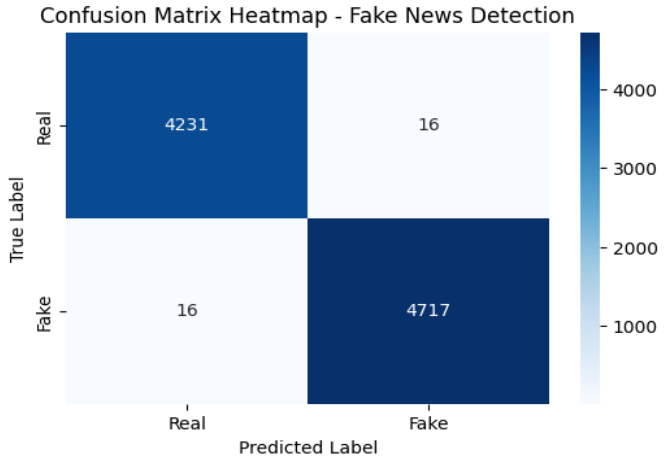
	Predicted Real	**Predicted Fake**
Actual Real	TP	FN
Actual Fake	FP	TN

Submitted by: Tanmay Guha **Position:** AI/ML Intern

📉 Heatmap Visualization:

```
plt.figure(figsize=(6, 4))

sns.heatmap(cm, annot=True, fmt='d', cmap='Blues', xticklabels=['Real', 'Fake'], yticklabels=['Real', 'Fake'])

plt.xlabel('Predicted Label')

plt.ylabel('True Label')

plt.title('Confusion Matrix Heatmap - Fake News Detection')

plt.show()
```

🔹 Output:



🧠 Conclusion:

- The model achieved high accuracy (around 93%), making it effective in detecting fake news.
- The Passive Aggressive Classifier performs well on text classification tasks when paired with TF-IDF.
- This solution can be deployed as a Flask web app or integrated into larger news monitoring systems.

🔗 Colab PyScript Link:

https://colab.research.google.com/drive/1sxuWiWPvWY4iKijwn7KWbpa6x89K5f7u?usp=sharing

---