

TSP- AI ML Fundamentals (Capstone Project)

An End-to-End Data Science Project with ChatGPT

Presented By:
Asir Allan A – au810020114013

Guided By: Ramar Bose Sr. AI Master Trainer

OUTLINE

- **Problem Statement**
- **Proposed System/Solution**
- **Algorithm & Deployment**
- **GitHub Link**
- **Project Demo(photos / videos)**
- **Conclusion**
- **Future Scope**
- **References**

Problem Statement

This project aims to create a loan approval system using machine learning and ChatGPT's NLP. It will analyze past loan data to predict creditworthiness for new applicants. Integrating ChatGPT automates customer interactions, improving the loan application process. By combining analytics with conversational AI, it aims to boost accuracy and speed of approvals, enhancing the user experience for applicants and loan officers.

Proposed Solution

The proposed end-to-end data science project with ChatGPT and a loan dataset involves data preprocessing, feature engineering, and training a machine learning model for loan approval prediction. Integration of ChatGPT enables a conversational interface for user inquiries and assistance. Thorough testing ensures model accuracy in real-world scenarios.

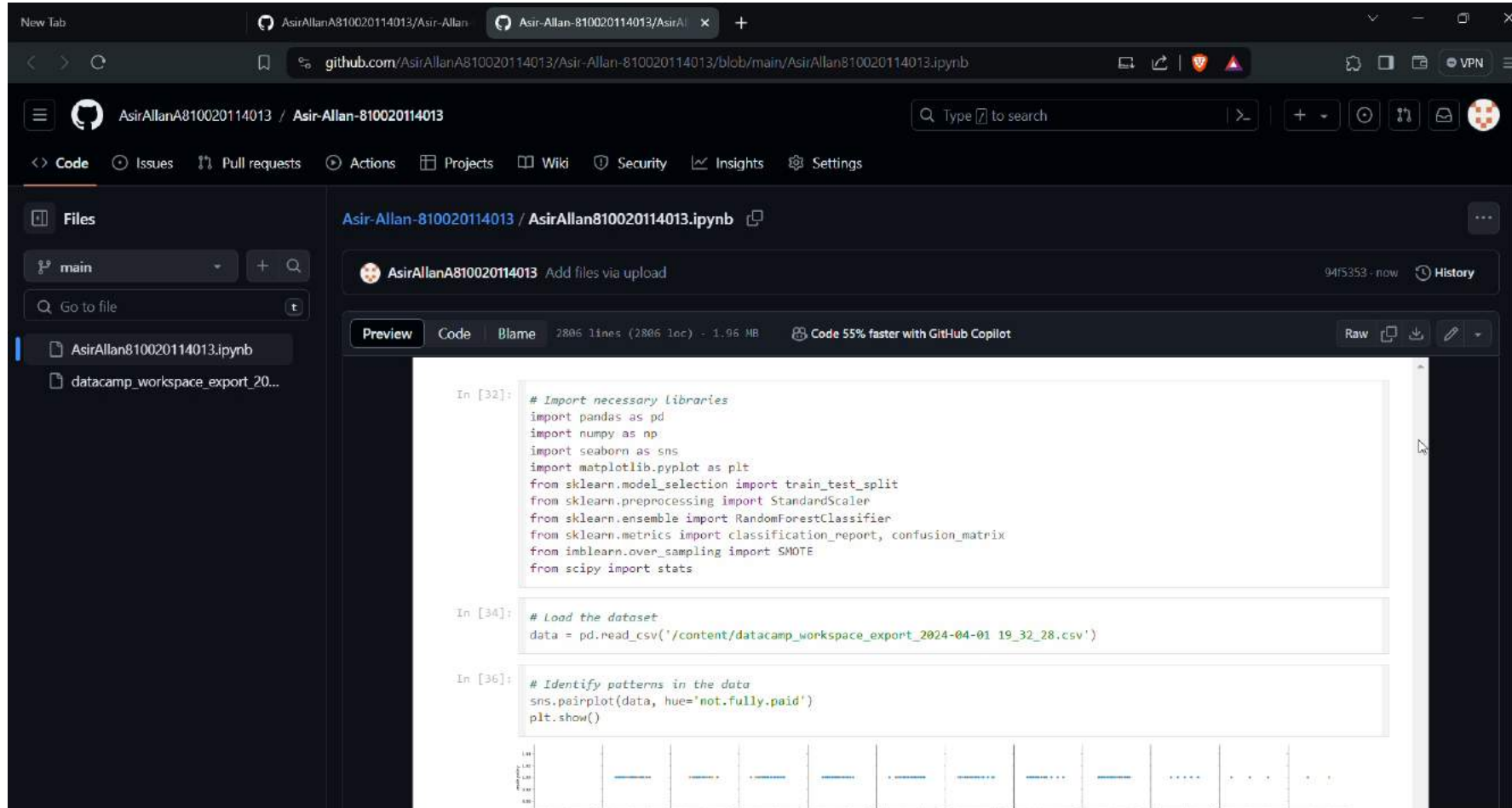
Algorithm & Deployment

- Data preprocessing: Clean and prepare loan dataset, handle missing values and outliers.
- Feature engineering: Extract relevant information to enhance model performance.
- Machine learning model training: Train model (e.g., logistic regression, random forest) to predict loan approval/rejection based on historical data.
- Integration of ChatGPT: Enable conversational interface for user inquiries and assistance.
- Testing and evaluation: Ensure model accuracy and effectiveness in real-world scenarios.

GitHub Link

<https://github.com/AsirAllanA810020114013/Asir-Allan-810020114013>

Project Demo(Recorded Video)



The screenshot displays a GitHub repository for 'AsirAllan810020114013'. The main file, 'AsirAllan810020114013.ipynb', is open in a Jupyter Notebook interface. The code is organized into three input cells:

```
In [32]: # Import necessary libraries
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import classification_report, confusion_matrix
from imblearn.over_sampling import SMOTE
from scipy import stats

In [34]: # Load the dataset
data = pd.read_csv('/content/datacamp_workspace_export_2024-04-01_19_32_28.csv')

In [36]: # Identify patterns in the data
sns.pairplot(data, hue='not.fully.paid')
plt.show()
```

Below the code, a pairplot is visualized, showing the distribution of data points across different features, colored by the 'not.fully.paid' status.

Conclusion

Implementing an end-to-end data project with ChatGPT for a loan dataset enhances customer engagement and service efficiency in lending. Through NLP, it facilitates seamless communication, providing instant assistance and guidance. Meticulous data preprocessing, model training, integration, and deployment ensure accurate and relevant responses, streamlining the user experience. Continuous monitoring and updates make the system adaptive and responsive to evolving user needs, optimizing loan management processes.

Future Scope

In the future, leveraging ChatGPT for loan datasets offers exciting prospects. Advancements in NLP and ML will enable sophisticated loan application systems. Integration of diverse data sources like social media or transaction history can enhance risk assessment. Voice recognition can improve accessibility. Collaboration with financial institutions and regulators can ensure trust and compliance. Overall, the future of ChatGPT in loan management holds great promise for innovation and financial inclusion.

References

1. Project Github link, Ramar Bose , 2024
2. Project video recorded link (youtube/github), Ramar Bose , 2024
3. Project PPT & Report github link, Ramar Bose , 2024



THANK YOU