**Project Title:** [**Retail Company**]

**Team Members:**

Harshavardhini A S[Team Leader]

Abarna S[Member]

Archana V[Member]

Sankari Malathi P[Member]

Thirunerai Selvi M[Member]

**Abstract:**

This project aims to develop a machine learning-powered product recommendation system for a retail company. The system will allow business users to upload training data, select features, and test the model through a user-friendly interface. Explanatory AI functionality will provide insights into model outcomes, and visual data analysis tools will simplify the understanding of recommendations.

**Project Overview:**

Our goal is to create a product recommendation system that leverages machine learning to suggest products to customers based on their preferences and behavior. This system will facilitate the training and testing of the model using various datasets, all within an intuitive user interface. Users will be able to upload training data, select relevant features, upload and preview test data, and receive explanations for model outcomes.

**Technologies Used:**

- Programming Language: Python  
- Machine Learning Libraries: TensorFlow, Scikit-Learn  
- Frameworks: Flask (for the UI)  
- Docker for containerization  
- MLOps tools (e.g., MLflow, Kubeflow)  
- Explanatory AI libraries (e.g., SHAP, Lime)  
- Data visualization libraries (e.g., Matplotlib, Seaborn)

**Data Collection and Preprocessing:**

Data will be collected from various sources, including customer behavior, purchase history, and product attributes. Data cleaning procedures will be applied to handle missing data, and preprocessing techniques will be used to transform the data for modeling.

**Model Architecture:**

We will implement a collaborative filtering model, which is well-suited for product recommendations. The system will employ user-based and item-based collaborative filtering techniques to provide personalized recommendations.

**Training Process:**

The model will be trained using historical data. We will use collaborative filtering algorithms to generate recommendations. Hyperparameters will be optimized, and the training process will involve cross-validation to ensure robust model performance.

**Evaluation Metrics:**

The performance of the recommendation system will be evaluated using metrics such as precision, recall, and F1-score to measure the accuracy of the recommendations.

Results and Discussion:

The project will provide business users with a user-friendly interface to upload training and test data, select features, and receive recommendations. Explanatory AI functionality will explain why specific recommendations are made, enhancing user trust in the system. Visual data analysis tools will be integrated to visualize data patterns and model outcomes, making it easier for business users to understand the recommendations.

Deployment:

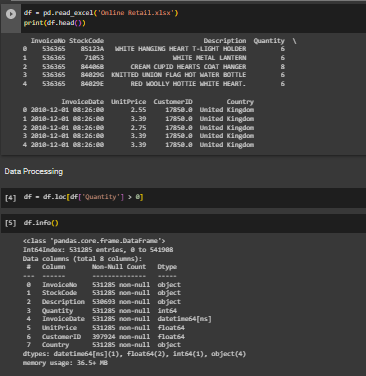
The machine learning model and the user interface will be deployed in a containerized environment. Users will interact with the system through a web-based interface powered by Flask. API endpoints will facilitate model interactions.

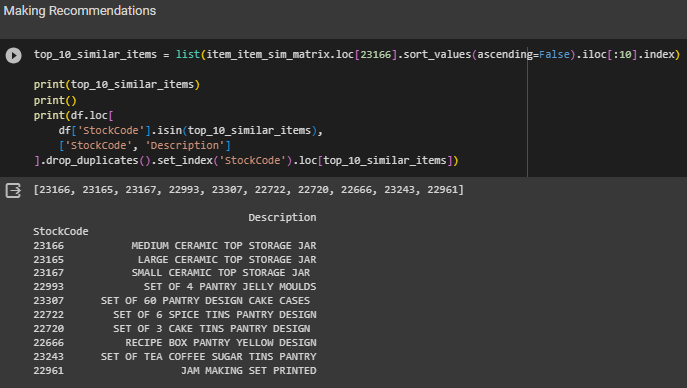
Deepsphere.AI

**Instructions for Running the Project:**

1. Install the required dependencies using the provided `requirements.txt`.  
2. Set up the Docker environment for model deployment.  
3. Start the Flask-based user interface by running the designated script.  
4. Upload training and test data through the UI.  
5. Select features for the model.  
6. Receive product recommendations and explanations for each recommendation.  
7. Utilize visual data analysis tools for a deeper understanding of the data and recommendations.

**Code Snippets:**

Here are some code snippets that illustrate key aspects of the implementation, including data preprocessing, model training, and explanatory AI:



**Conclusion:**

This project aims to provide the retail company with a robust product recommendation system that can be easily maintained, improved, and understood by business users. The integration of MLOps, explanatory AI, and data visualization tools enhances the usability and interpretability of the system.

**References:**

<https://github.com/dipanjanS/practical-machine-learning-with-python/blob/master/notebooks/Ch08_Customer_Segmentation_and_Effective_Cross_Selling/Online%20Retail.xlsx>

**Acknowledgments:**

Thank any mentors, team members, or individuals who contributed to your project's success.