YZV411E - Big Data Analytics Project Proposal (E-commerce Purchase Prediction)

Berkay Türk *150220320* Umut Çalıkkasap *150210721*
Abdülkadir Külçe *150210322*

Abstract

Problem Definition: *E-commerce platforms generate massive amounts of user behavior data. The primary challenge is processing this large-scale data efficiently to predict user purchase intent. Single-node systems face memory and time constraints, making the analysis of complete datasets impractical. Distributed frameworks like Apache Spark overcome this by partitioning data and executing computations in parallel, allowing for the scalable development of accurate predictive models.*

Project Aim and Motivation

This project aims to build a machine learning model to predict whether a user will complete a purchase within a session. The 4.5 GB dataset, containing over 42 million user events, requires complex aggregations over millions of sessions, making a distributed engine like Apache Spark essential over single-node tools. The model is motivated by the real-world need for e-retailers to reduce cart abandonment and increase conversions by identifying high-intent users for targeted interventions. Our project will demonstrate the **benefit of distributed data processing** by performing these large-scale aggregations, a task infeasible on a single machine due to memory and CPU limitations.

Data

- Kaggle: *"E-commerce behavior data from multi-category store" (2019-Nov.csv)*
- **Structure:** Text-based behavioral data in CSV format; approximately 5.67 GB with 42.4M records. Features include `event time`, `brand`, `price`, `event type`, `product id`, `user id`, and `user session`, `category id`,`category code`.
- **Chunking Strategy:** Apache Spark will automatically partition the data across the cluster. We will perform parallel `groupBy` operations on `user session` for feature engineering.

Tools and Frameworks

- **Apache Spark:** Core engine for large-scale data processing, feature engineering, and model training using PySpark and Spark SQL.
- **Spark MLlib:** Used to train a classification model (e.g., Gradient-Boosted Trees) on the distributed dataset.
- **HDFS (Hadoop Distributed File System):** Fault-tolerant storage for raw and processed data accessible by the Spark cluster.

Expected Outcomes and Evaluation

The main outcome is a robust machine learning model capable of predicting purchase intent. The model's performance will be evaluated using standard classification metrics (Accuracy, Precision, Recall, AUC). Critically, we will demonstrate the system's scalability by comparing the processing runtime of our distributed Spark implementation against a single-node setup, visualizing the speed-up gains to illustrate the benefits of the distributed approach.

Progress Plan

The following steps are planned before the progress report:

- **Step 1. Environment Setup and Data Ingestion:** Set up a Spark/HDFS environment and ingest the dataset.
- **Step 2. Exploratory Data Analysis (EDA):** Perform initial analysis on the data using Spark SQL to understand distributions and formulate hypotheses.
- **Step 3. Data Preprocessing Pipeline:** Develop a PySpark script to clean and transform the data into a suitable format.
- **Step 4. Feature Engineering:** Implement a script to create session-based features (e.g., session duration, event counts, viewed products).
- **Step 5. Baseline Model Training:** Train and evaluate a baseline classification model using Spark MLlib to verify the end-to-end pipeline.