Sessionization and Funnel Analysis of Website Clickstreams using PySpark

Muralii Krishnan Thirumalai
Department of Electrical and Computer Engineering, Rutgers University
mt1171@scarletmail.rutgers.edu

Abstract:
This project aims to design a scalable data analytics pipeline for website clickstream data using PySpark. The main goal is to process and analyze large volumes of user interaction logs to derive behavioral insights such as session length, page depth, and conversion funnel metrics. The pipeline will simulate a modern data engineering stack with layered storage (Bronze, Silver, Gold), and demonstrate event-time processing, windowing, and sessionization logic on Databricks or Google Colab.

1. Introduction

Every click, search, or page visit on a website generates valuable behavioral data known as clickstream logs. Analyzing these logs helps identify user engagement patterns and conversion drop-offs. However, due to the high volume and velocity of such data, single-machine tools struggle to scale. PySpark, with its distributed processing and SQL-like DataFrame API, enables large-scale computation efficiently, making it ideal for this project.

2. Methodology

The project will use a public clickstream dataset (e.g., from Kaggle) and follow a three-stage design:

- **Data Ingestion & Cleaning:** Read raw CSV or JSON logs with `spark.read`, convert timestamps, handle null values, and partition by `user_id`.
- **Sessionization:** Use Spark's `Window.partitionBy("user_id").orderBy("event_time")` with `lag()` to compute inter-event gaps. New sessions begin when inactivity exceeds 30 minutes, generating unique session IDs.
- **Funnel Analysis:** Map URLs to funnel stages (home → search → product → cart → checkout) and measure user progression, drop-offs, and average session metrics.

All processed data will be stored in structured layers (Bronze for raw, Silver for cleaned, Gold for aggregated) using Delta Lake for reliability and query efficiency.

3. Expected Results

Deliverables will include:

1. A PySpark pipeline capable of sessionizing and aggregating web events.
2. Computed KPIs such as average session duration, page depth, and funnel conversion rates.
3. Visual reports using Matplotlib or Plotly showing user navigation and drop-off trends.

4. Conclusion

This project provides practical experience in data engineering and analytical reasoning using big data technologies. It bridges concepts from distributed computing, event-time analytics, and behavioral data interpretation—ideal for advanced coursework in data systems or applied analytics.