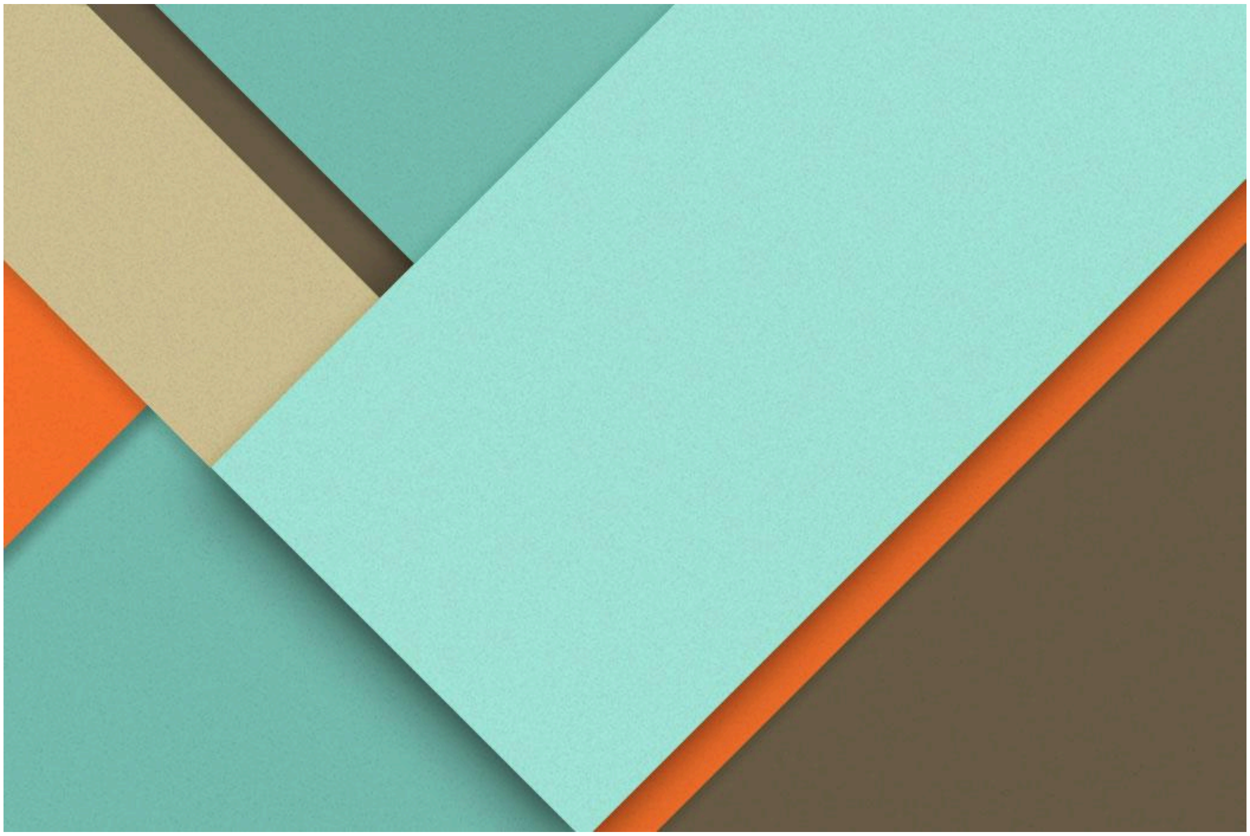

Amazon Co-Purchasing Network Analysis using Neo4j

Team CheeseCake

Omkar Bhandare

Shivam Choudhury

Tanishka Rahate

Ayush Mundada

Jadhav Udaykiran

Overview

The project leverages the Stanford SNAP Amazon co-purchasing dataset to build a comprehensive graph database in *Neo4j* that models complex product relationships and customer behaviours. By implementing *cypher* queries and advanced visualization techniques, we aim to extract actionable consumer cum business intelligence.

Goals

1. Develop a robust graph data model that accurately represents the multi-dimensional relationships between products, categories, groups, and customers
2. Implement advanced graph algorithms, including *PageRank*, community detection, and centrality measures to identify influential products and natural clusters among them
3. Create a *lite*-recommendation engine that leverages both explicit and implicit relationships with frequency weighting
4. Build an interactive visualization dashboard that allows for intuitive exploration of the product network

Specifications

Data: Stanford SNAP Amazon Dataset (v2025)

Analysis: PageRank, Louvain Community Detection

Database: Neo4j Desktop, GDS Library, APOC Library

Backend: Python, Flask, Neo4j Python Driver

Frontend: HTML, CSS, JavaScript, D3.js

Weekly Work Plan

I. Week 1: Data Modeling and Ingestion

a. Finalize graph schema design with appropriate indexes
b. Develop ETL pipeline for data preprocessing and cleaning
c. Import node and relationships with batch processing

II. Week 2: ALgorithms Implementation

a. Configure GDS Library and necessary plugins
b. Implement *PageRank* and centrality measures for product influence scoring
c. Apply community detection algorithms to identify product clusters
d. Develop custom Cypher queries for complex relationship analysis

III. Week 3: API and Visualization Development

a. Develop an interface for the end user
b. Configure Flask API endpoints for Neo4j access
c. Develop interactive network visualization components
d. Create a dashboard layout with filtering capabilities

IV. Week 4: Refinement, Testing and Documentation

a. Refine all the components
b. Develop comprehensive documentation and usage examples
c. Final presentation with Documentation