Term Project Checkpoint C

Greg Roberts and Flavio Mota

MSDS 459 - Knowledge Engineering

May 25, 2025

Abstract

This paper reports on the design, implementation, and application of a Neo4j-based knowledge graph for the consumer discretionary sector. It describes the database schema and the current state of population, outlines the data ingestion pipeline that integrates Wikipedia, SEC filings, the yfinance API, Federal Statistical Business Indicators (FSBI), and Federal Reserve Economic Data (FRED) series, and demonstrates Cypher queries that support competitive intelligence questions about companies, industries, products, and macroeconomic context. The study shows how a graph representation can facilitate complex business insights and strategic decision-making.

Introduction

The consumer discretionary industry involves dynamic competition across overlapping product categories and geographic markets. Traditional relational databases often struggle to capture and query the multi-relational structure inherent in corporate-subsector-product relationships and their interaction with macroeconomic indicators. To address these challenges, this research implements a knowledge graph in Neo4j, enabling flexible representation of heterogeneous entities and efficient traversal of intricate relationships. This platform aims to answer management questions such as which firms share product offerings, how subsidiaries are structured, and how business performance correlates with economic trends.

Literature Review

Foundational work by Kejriwal, Knoblock, and Szekely (2021) describes semi-automated pipelines for relation extraction and graph construction, illustrating how knowledge graphs can bind heterogeneous data into a coherent structure. In financial applications, Yao et al. (2018) demonstrated the use of graph embeddings and traversal techniques for risk analysis, while Paulheim (2017) explored recommendation systems built on graph representations of corporate and market data. These studies highlight the benefits of graph queries for uncovering non-obvious correlations—such as linking consumer-spending anomalies to specific corporate events—and informed our choice of Neo4j for its flexible schema and rich query language.

Methods

Data ingestion and graph construction draw on four distinct data streams, each unified within Neo4j. First, textual sources are processed via Python: we scrape company

Wikipedia infoboxes and articles with BeautifulSoup—using a custom DocumentParser class to normalize fields such as name, CEO, headquarters, industry, products, and subsidiaries into structured dictionaries—and ingest SEC filings (10-K and 10-Q) obtained from the EDGAR API. The three most recent annual and quarterly filings per ticker are parsed for financial line items (revenue, net income, assets, liabilities) and subsidiary information, with each document represented as a node linked to its company.

Our ingestion workflow combines manual curation with automated pipelines:

- Manual curation: We maintain a CSV file of 10 consumer-discretionary companies (symbol, name, sector) and a JSON file defining product categories and subcategories.

Company	% Wtg
Amazon.com, Inc.	21.3%
Tesla, Inc.	16.0%
The Home Depot, Inc.	7.3%
Booking Holdings Inc.	4.8%
McDonald's Corporation	4.8%
The TJX Companies, Inc.	4.1%
Lowe's Companies, Inc.	3.6%
Starbucks Corporation	2.6%
O'Reilly Automotive, Inc.	2.3%
Chipotle Mexican Grill, Inc.	2.0%

Figure 1 - The Consumer Discretionary Select Sector SPDR Fund (XLY): https://finance.yahoo.com/quote/XLY/holdings/

Second, market data are captured through the yfinance client, which downloads daily closing prices to compute end-of-month returns stored as StockReturn nodes and sector index metrics from Fiserv's Small Business Index (FSBI) come as CSV exports containing sales and transaction volume levels, along with month-over-month and year-over-year percent changes; the FSBILoader module reads and normalizes these records into FSBIIndex and FSBIRecord nodes linked to their industries. Finally, macroeconomic context is incorporated by fetching time series specifically for the Leading Economic Index (LEI), Personal Consumption Expenditures (PCE), and the Consumer Price Index (CPI) from January 1, 2022 onward via the FRED REST API. In future work, the SEC dataset can be pulled from the Edgar API. Risk attributes as well as key financial ratios (debt-to-equity, ROA, etc.) can be derived from parsed SEC statements and attached to quarterly filing nodes. Each series is stored as an Indicator node with associated child records capturing observations by date.

All raw inputs (HTML, JSON, and CSV) are archived under data/raw, then passed through a lightweight parsing layer that converts them into canonical Python dictionaries before

feeding parameterized Cypher statements to Neo4j over the Bolt driver. This multi-modal pipeline produces a richly connected graph spanning textual descriptions, corporate fundamentals, sector signals, and economic time series, ready for downstream competitive-intelligence analysis.

Database Schema

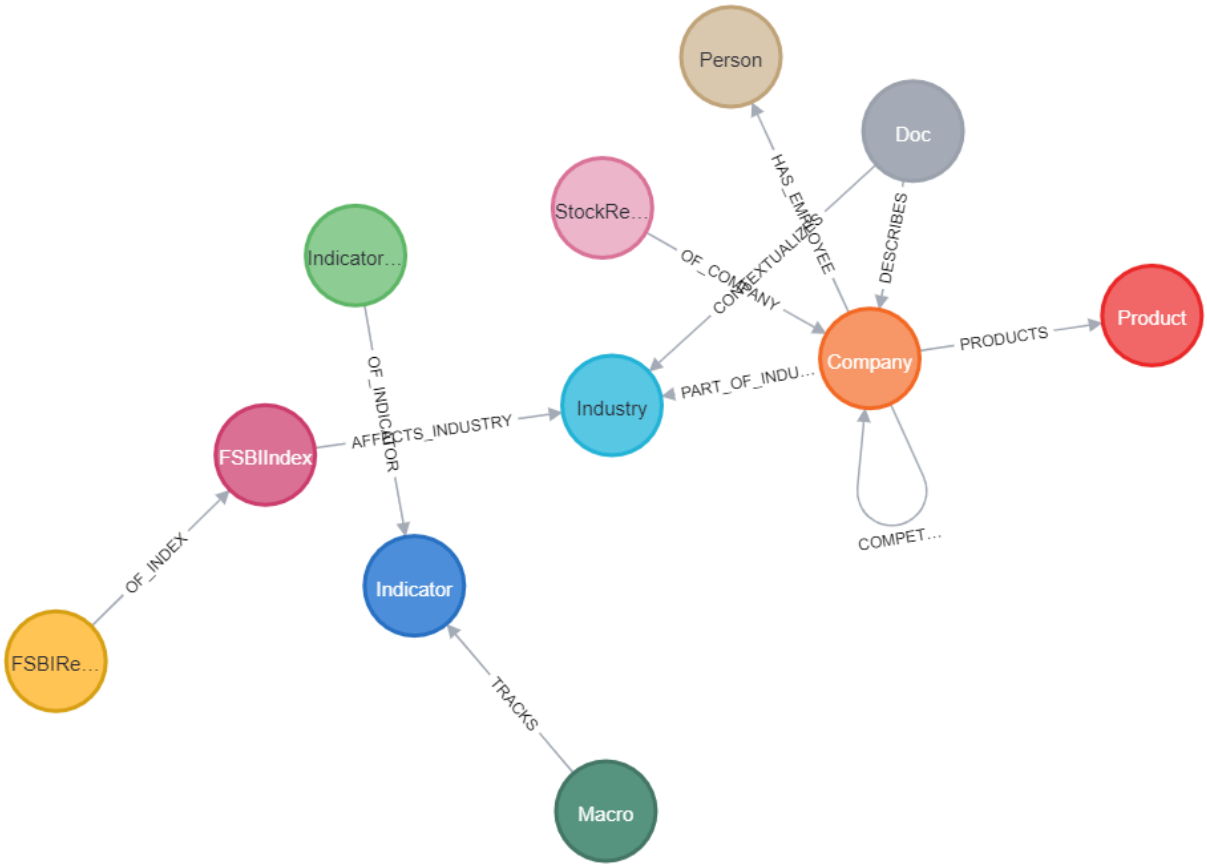

Figure 2 - Neo4j schema

The knowledge graph employs a rich schema of node labels, properties, and relationships to capture corporate, product, index, and macroeconomic entities.

Nodes in the graph are typed according to the following labels, each with a specific set of properties:

- Company: symbol, name, CEO, headquarters, marketCap, wikiURL, displayLabel, embedding, id, tags
- Document: id, text, source, tags, date, displayLabel, embedding

- FSBIIndex: index_id, seriesName (as name), valuesByDate (stored under value arrays), embedding, id, date
- FSBIRecord: sales_index, sales_mom_pct, sales_yoy_pct, transaction_index, trans_mom_pct, trans_yoy_pct, date, id
- Indicator: indicator_id, name, valuesByDate (value arrays), embedding, id, date
- IndicatorRecord: value, date, id, type
- Industry: name, displayLabel, id
- Person: name, role, id, displayLabel
- Product: name, displayLabel, id
- StockReturn: date, value, id, displayLabel

Collectively, property keys across all nodes include: date, displayLabel, embedding, id, index_id, indicator_id, name, role, sales_index, sales_mom_pct, sales_yoy_pct, symbol, tags, text, trans_mom_pct, trans_yoy_pct, transaction_index, type, and value.

Entities are interconnected through typed relationships that encode business context:

- PART_OF_INDUSTRY (Company → Industry)
- OFFERS / PRODUCTS (Company → Product)
- COMPETES_WITH (Company ↔ Company)
- SUBSIDIARY_OF (Company → Company)
- HAS_EMPLOYEE (Company → Person)
- OF_COMPANY (Person or Document → Company)
- OF_INDEX (FSBIRecord → FSBIIndex)
- OF_INDICATOR (IndicatorRecord → Indicator)
- AFFECTS_INDUSTRY (Indicator → Industry)

This schema allows storage of static attributes (e.g., company headquarters), time-series metrics (FSBI and FRED records), and multi-hop traversals across corporate, product, and economic dimensions.

The current knowledge graph has been populated with 11 company entities, 14 industry categories, 81 distinct products, 134 source documents, 10 FSBI index series, and the 3 macroeconomic indicators from FRED. This level of coverage reflects the integration of corporate, product, and economic signal data into a unified graph structure.

Results

Data Sources Identified

- Manually curated: A list of 10 consumer-discretionary companies; a hierarchical product taxonomy.
- Structured via APIs:
    - Consumer-spending series (FRED: Retail Trade, Food Services).
    - Monthly FSBI metrics (Fiserv).
    - Corporate financial metrics (Yahoo Finance).
- Unstructured via scraping:
    - Company metadata from Wikipedia (CEO, location, sector).
    - Filings metadata from EDGAR (document type, date, URL).

Information Collected

- A CSV table of 10 companies with symbol, name, and sector classification.
- JSON files for FSBI monthly values spanning January 2022 through April 2025.
- Scraped HTML pages stored under /data/wikipages/, each parsed into JSON records containing structured metadata fields.
- FRED economic data from an API providing signals for consumer related insights spanning January 2022 through April 2025.

Conclusions

The Neo4j knowledge graph effectively models companies, industries, products, corporate hierarchies, business metrics, and macroeconomic indicators. Its flexible schema and Cypher-driven queries facilitate answers to strategic questions relevant to competitive intelligence. Management can use this platform to identify overlapping product offerings, map competitor networks, analyze subsidiary structures, and integrate business metrics with economic trends.

The current prototype demonstrates how a partially curated, programmatically enriched knowledge graph can unify disparate data sources into a single analytical framework. By linking corporate metadata, market indices, consumer-spending trends, and document archives, the graph supports multidimensional queries—enabling competitive-intelligence analyses such as peer-group comparisons, sector-wide trend spotting, and risk-event correlations. This integrated model directly addresses the management problem of constructing a data-driven platform for stock recommendations in the consumer discretionary sector. Future work will expand company coverage, automate continuous EDGAR ingestion with NLP-based relation extraction, integrate sentiment and analyst-rating feeds, and deploy scheduled pipelines to keep the graph current. Ultimately, this knowledge graph will serve as a robust foundation for strategic decision support and predictive modeling.