SAMSUNG

Real-Time Big Data Processing with PySpark. NYC Taxi Trip Analysis

Big Data Course

Capstone Project
Final Report

For students (instructor review required)

Repository: https://github.com/QuangDuyReal/nyc-taxi-trip-analysis

Real-Time Big Data Processing with PySpark. NYC Taxi Trip Analysis

30/07/2025

GROUP 01

Đỗ Kiến Hưng
Nguyễn Văn Quang Duy

Content

1. Introduction

1.1. Background Information

Trong kỷ nguyên số, dữ liệu đã trở thành tài sản chiến lược, và khả năng khai thác các tập dữ liệu quy mô lớn (Big Data) là yếu tố quyết định lợi thế cạnh tranh của nhiều tổ chức. Bộ dữ liệu về các chuyến đi taxi tại New York City (NYC TLC Trip Record Data) là một ví dụ điển hình về thách thức và cơ hội này. Với hơn 1.5 tỷ bản ghi được tích lũy từ năm 2009 và dung lượng vượt quá 200GB, bộ dữ liệu này chứa đựng những thông tin sâu sắc về mô hình di chuyển, kinh tế đô thị và hiệu quả vận hành giao thông.

Tuy nhiên, việc xử lý một khối lượng dữ liệu khổng lồ như vậy đòi hỏi một cách tiếp cận vượt ra ngoài các công cụ truyền thống. Apache Spark, với API Python là PySpark, đã khẳng định vị thế là một framework xử lý dữ liệu phân tán hàng đầu, có khả năng xử lý hiệu quả cả dữ liệu lịch sử (batch) và dữ liệu thời gian thực (real-time). Dự án này được xây dựng để áp dụng sức mạnh của PySpark vào một bài toán thực tế, không chỉ để phân tích dữ liệu mà còn để xây dựng một hệ thống xử lý dữ liệu hoàn chỉnh, có cấu trúc và đáng tin cậy.

1.2. Motivation and Objective

Động lực chính của nhóm bắt nguồn từ mong muốn được áp dụng các kiến thức lý thuyết về Big Data đã học vào một dự án thực tế, có quy mô và độ phức tạp cao. Thay vì chỉ thực hiện các phân tích đơn lẻ, chúng tôi muốn xây dựng một pipeline dữ liệu (data pipeline) hoàn chỉnh, một sản phẩm cốt lõi trong công việc của một kỹ sư dữ liệu. Việc lựa chọn dữ liệu taxi NYC không chỉ vì quy mô lớn của nó, mà còn vì tiềm năng khai thác các insight hữu ích cho việc quy hoạch đô thị, tối ưu hóa dịch vụ và hỗ trợ các quyết định kinh doanh dựa trên dữ liệu.

Mục tiêu bao trùm của dự án là thiết kế và triển khai một pipeline xử lý dữ liệu lớn có khả năng mở rộng và chịu lỗi, sử dụng Apache PySpark với kiến trúc Medallion để phân tích dữ liệu chuyến đi taxi tại NYC.

Các mục tiêu cụ thể bao gồm:

- Xây dựng Pipeline theo kiến trúc Medallion (Bronze → Silver → Gold).
  - Bronze Layer. Nạp dữ liệu thô (raw data) từ nguồn mà không thay đổi.
  - Silver Layer. Làm sạch, xác thực, chuẩn hóa và làm giàu dữ liệu.
  - Gold Layer. Tổng hợp dữ liệu thành các bảng phân tích (aggregated tables) sẵn sàng cho việc báo cáo và khai thác.

- Đảm bảo chất lượng dữ liệu (Data Quality). Xây dựng một framework để kiểm tra tính hoàn chỉnh, hợp lệ, nhất quán và duy nhất của dữ liệu qua các lớp.

- Tích hợp Xử lý Luồng (Streaming). Thiết kế và triển khai một pipeline xử lý dữ liệu gần thời gian thực bằng Spark Structured Streaming, mô phỏng khả năng giám sát hoạt động kinh doanh liên tục.

- Tối ưu hóa và Đóng gói. Sử dụng Delta Lake để tăng cường độ tin cậy, phân vùng dữ liệu để tối ưu hiệu năng, và đóng gói ứng dụng để có thể thực thi độc lập.

1.3. Members and Role Assignments

Nhóm bao gồm 2 thành viên, với sự phân công vai trò thực hiện dự án.

- Đỗ Kiến Hưng (Data Engineer / Backend Data Processor). Giữ vai trò kỹ sư, chịu trách nhiệm thiết kế và triển khai các lớp xử lý dữ liệu phức tạp nhất của pipeline, nơi diễn ra các quá trình biến đổi và làm giàu dữ liệu cốt lõi.
    - Silver Layer. Phát triển logic làm sạch, xác thực và kỹ thuật đặc trưng (feature engineering).
    - Gold Layer. Xây dựng các bảng tổng hợp dữ liệu theo nghiệp vụ.
    - Streaming Pipeline. Thiết kế và triển khai pipeline xử lý dữ liệu gần thời gian thực.
    - Data Quality Framework. Xây dựng module kiểm tra chất lượng dữ liệu.

- Nguyễn Văn Quang Duy (Pipeline Orchestrator / Analytics Engineer). Chịu trách nhiệm về luồng đi của dữ liệu từ đầu vào, điều phối toàn bộ pipeline, và chuyển hóa dữ liệu đã xử lý thành các insight có thể hành động được.
    - Bronze Layer. Xây dựng quy trình nạp dữ liệu thô, hợp nhất schema và bổ sung metadata.
    - Main Pipeline. Viết kịch bản chính (main_pipeline.py) để điều phối và thực thi toàn bộ luồng xử lý.
    - Data Export. Phát triển chức năng xuất dữ liệu từ lớp Gold (PARQUET) sang định dạng CSV để phục vụ các công cụ BI.
    - Analysis & Visualization. Xây dựng notebook phân tích, tạo các biểu đồ trực quan hóa (plots) và thiết kế dashboard mẫu.

1.4. Schedule and Milestones

Dự án được thực hiện trong 8 tuần, với các cột mốc quan trọng được xác định.

- Giai đoạn 1. Khởi tạo và Thiết kế (Tuần 1-2). Hoàn thành việc thiết lập môi trường (Spark, Python, Delta Lake). Tải và khám phá dữ liệu. Hoàn thiện bản thiết kế kiến trúc Medallion và phân công nhiệm vụ chi tiết.

- Giai đoạn 2. Phát triển Pipeline Batch (Tuần 3-4). Hoàn thành mã nguồn cho lớp Bronze và Silver. Xây dựng và tích hợp framework Data Quality Checker.

- Giai đoạn 3. Phát triển Lớp Phân tích (Tuần 5-6). Hoàn thành mã nguồn cho lớp Gold. Xây dựng notebook phân tích đầu tiên để trực quan hóa và xác thực các kết quả tổng hợp.

- Giai đoạn 4. Phát triển Pipeline Streaming và Hoàn thiện (Tuần 7-8). Hoàn thành pipeline xử lý luồng. Tích hợp tất cả các thành phần vào main_pipeline.py. Chuẩn bị tài liệu kỹ thuật, báo cáo cuối kỳ và bài thuyết trình.

2. Project Execution

2.1. Simulated Scenario Description

Dự án mô phỏng một kịch bản thực tế của một công ty phân tích dữ liệu vận tải. Hệ thống được yêu cầu xây dựng một Data Lakehouse có khả năng.

1. Tiếp nhận dữ liệu lịch sử. Hàng tháng, công ty nhận được các file dữ liệu mới từ TLC. Pipeline phải có khả năng xử lý các file này một cách tự động, hợp nhất các cấu trúc dữ liệu có thể thay đổi theo thời gian, và cập nhật vào kho dữ liệu chính.

2. Xử lý dữ liệu gần thời gian thực. Hệ thống cần có một thành phần có khả năng giám sát hoạt động taxi gần như ngay lập tức, cung cấp các chỉ số vận hành (KPIs) trong các cửa sổ thời gian ngắn (ví dụ. mỗi 5-10 phút) để hỗ trợ các quyết định điều hành tức thì.

3. Cung cấp dữ liệu cho phân tích. Dữ liệu cuối cùng phải được tổ chức thành các bảng tổng hợp sạch, dễ hiểu, sẵn sàng để kết nối với các công cụ BI (như Power BI, Tableau) hoặc để các nhà phân tích dữ liệu (Data Analysts) sử dụng trực tiếp

2.2. Datasets Selection and Description

- Nguồn dữ liệu chính: NYC Yellow Taxi Trip Records (2023.1; 2024.1). Nguồn từ NYC Taxi & Limousine Commission (TLC) official website. Định dạng: Parquet.

- Bộ dữ liệu chứa thông tin chi tiết về từng chuyến đi của taxi vàng tại NYC, bao gồm thời gian và địa điểm đón/trả khách, quãng đường, chi phí, phương thức thanh toán, và số lượng hành khách.

- Dữ liệu có cấu trúc bán định dạng (semi-structured) và có thể có sự thay đổi về schema qua các năm. Dự án tuân thủ theo schema chính thức do TLC cung cấp. Các trường quan trọng được sử dụng bao gồm:
  tpep_pickup_datetime, tpep_dropoff_datetime, passenger_count, trip_distance, PULocationID, DOLocationID, fare_amount, tip_amount, total_amount.

2.3. Data Ingestion Pipeline

Tạo ra một nền tảng dữ liệu đáng tin cậy, nhất quán và có thể truy vết được. Quy trình thực thi như sau:

1. Thu thập và Hợp nhất Schema. Quy trình sử dụng spark.read.parquet() với tùy chọn mergeSchema=True. Điều này cho phép pipeline tự động xử lý các khác biệt về cấu trúc dữ liệu giữa các file từ các thời điểm khác nhau, tạo ra một schema thống nhất mà không làm mất dữ liệu.

2. Bổ sung Metadata. Sau khi đọc, hai loại metadata quan trọng được thêm vào mỗi bản ghi. Cột phân vùng (year, month) được trích xuất từ tpep_pickup_datetime, giúp tối ưu hóa đáng kể hiệu năng truy vấn ở các lớp sau. Cột siêu dữ liệu (ingestion_timestamp, source_file) ghi lại thời điểm dữ liệu được nạp và tên file nguồn, phục vụ cho việc kiểm toán, theo dõi nguồn gốc (data lineage) và gỡ lỗi.

3. Ghi dữ liệu vào Delta Lake. Dữ liệu sau khi xử lý được ghi vào một bảng Delta Lake trong lớp Bronze. Chế độ overwrite đảm bảo rằng mỗi lần chạy, pipeline sẽ tạo ra một phiên bản dữ liệu mới nhất và nhất quán, loại bỏ dữ liệu cũ, giúp quy trình có tính idempotent (chạy nhiều lần cho cùng kết quả). Phân vùng (partitionBy) giúp dữ liệu được lưu trữ vật lý trên đĩa theo cấu trúc thư mục year và month, tăng tốc độ đọc dữ liệu khi lọc theo thời gian.

2.4. Data Transformation Processing

Lớp Silver là nơi dữ liệu thô được "nấu chín" thành một tập hợp dữ liệu sạch, đã được làm giàu và sẵn sàng cho phân tích. Mục tiêu nghiệp vụ dùng để tinh chỉnh dữ liệu, loại bỏ các bản ghi không hợp lệ, và tạo ra các thuộc tính mới có giá trị cao. Quy trình thực thi bao gồm:

- Xác thực và Làm sạch Ban đầu. Loại bỏ các bản ghi có các trường quan trọng bị thiếu (NULL) hoặc không hợp lệ (ví dụ. trip_distance <= 0, fare_amount <= 0).
- Làm giàu và Kỹ thuật Đặc trưng (Feature Engineering). Tính toán các chỉ số quan trọng. trip_duration_minutes (thời lượng chuyến đi), speed_mph (tốc độ trung bình), tip_percentage (tỷ lệ tiền boa).
- Trích xuất các thuộc tính thời gian. pickup_hour, pickup_day_of_week, pickup_month, pickup_year từ tpep_pickup_datetime.
- Làm sạch Nâng cao và Loại bỏ Ngoại lệ (Outlier Removal). Áp dụng các bộ lọc dựa trên logic nghiệp vụ để loại bỏ các dữ liệu bất thường (ví dụ. trip_duration_minutes > 180, speed_mph > 100).

- Đánh giá chất lượng và Ghi Metadata. Tạo chỉ số quality_score để đánh giá mức độ tin cậy của từng bản ghi. Thêm cột processing_timestamp để ghi lại thời điểm xử lý.

Lớp Gold là đỉnh điểm của pipeline, nơi dữ liệu sạch được tổng hợp thành các bảng phân tích chiến lược. Mục tiêu nghiệp vụ nhằm cung cấp các bảng dữ liệu chất lượng cao, dễ hiểu, và tối ưu hóa cho các công cụ BI và người dùng cuối. Quy trình thực thi bao gồm:

- Kiểm tra dữ liệu nguồn. Trước khi tổng hợp, quy trình xác thực tính hợp lệ của DataFrame từ lớp Silver.
- Lưu trữ tối ưu. Các bảng Gold được lưu dưới định dạng Delta Lake và được phân vùng theo các chiều phù hợp (ví dụ. year, month, hour) để tối ưu hóa tốc độ truy vấn cho các báo cáo.
- Tổng hợp theo các Chiều Phân tích. Tạo ra một loạt các bảng tổng hợp, mỗi bảng tập trung vào một khía cạnh nghiệp vụ cụ thể.
    1. hourly_trip_analytics. Phân tích xu hướng theo giờ.
    2. location_hotspots. Xác định các "điểm nóng" về địa điểm đón khách.
    3. payment_analytics. Phân tích hành vi thanh toán.
    4. vendor_performance. Đánh giá hiệu suất của các nhà cung cấp dịch vụ.

2.5. Data Query and Insight

Sau khi pipeline hoàn tất, dữ liệu ở lớp Gold đã sẵn sàng để được truy vấn và khai thác. Quá trình này được thực hiện trong Jupyter Notebook (02_analytics_dashboard.ipynb). Mục tiêu nghiệp vụ nhằm chuyển đổi dữ liệu đã tổng hợp thành các thông tin có thể hành động được (actionable insights). Quy trình thực thi bao gồm

1. Kết nối và Đọc dữ liệu Gold. Notebook sử dụng Spark để đọc trực tiếp các bảng Delta từ lớp Gold.

2. Chuyển đổi sang Pandas. Để tận dụng hệ sinh thái trực quan hóa phong phú của Python, các Spark DataFrame nhỏ (kết quả của các truy vấn tổng hợp) được chuyển đổi thành Pandas DataFrame bằng phương thức: toPandas().

3. Trực quan hóa. Sử dụng các thư viện Matplotlib, Seaborn, và Plotly để tạo các biểu đồ (xem Phần 3.4).

4. Phân tích và Diễn giải. Mỗi biểu đồ được đi kèm với phần phân tích, rút ra các insight và hàm ý kinh doanh quan trọng, trả lời các câu hỏi như. "Giờ nào trong ngày có doanh thu cao nhất?", "Khu vực nào có nhu-cầu-cao-nhất?", "Phương thức thanh toán nào là phổ biến nhất?".

3. Results

3.1. Data Ingestion Scripts and Code

Quá trình nạp dữ liệu thô (Bronze Layer) và điều phối toàn bộ pipeline được quản lý bởi các kịch bản Python có cấu trúc rõ ràng.

src/bronze_layer.py. Chịu trách nhiệm thực hiện toàn bộ logic của lớp Bronze.
Hàm process_bronze_layer() đọc dữ liệu Parquet từ thư mục data/raw, tự động hợp nhất schema, bổ sung các cột metadata (ingestion_timestamp, source_file, year, month), và ghi kết quả vào bảng Delta Lake đã được phân vùng. Sử dụng mergeSchema để tăng tính bền vững của pipeline trước sự thay đổi của dữ liệu nguồn.

main_pipeline.py. Đóng vai trò là "nhạc trưởng" (orchestrator) của toàn bộ hệ thống. Script này khởi tạo SparkSession với các cấu hình cần thiết (ví dụ. kích hoạt hỗ trợ Delta Lake), sau đó gọi tuần tự các hàm xử lý cho từng lớp. Bronze, Silver, và Gold. Tích hợp logging chi tiết và cấu trúc try...except...finally để đảm bảo pipeline chạy một cách có kiểm soát, dễ gỡ lỗi và luôn giải phóng tài nguyên (spark.stop()) ngay cả khi có lỗi xảy ra.

3.2. Data Transformation Scripts and Code

Quá trình làm sạch, làm giàu và tổng hợp dữ liệu được thực hiện bởi các module riêng biệt, đảm bảo tính dễ bảo trì và mở rộng.

src/silver_layer.py. Chứa hàm process_silver_layer() để biến đổi dữ liệu từ Bronze thành Silver. Mã nguồn bao gồm các chuỗi phương thức .filter(), .withColumn() và các hàm từ pyspark.sql.functions để áp dụng các quy tắc nghiệp vụ, làm sạch và tạo ra các feature mới. Logic được chia thành các bước rõ ràng. validation, feature engineering, và advanced cleansing, giúp mã nguồn dễ đọc và dễ kiểm thử.

src/gold_layer.py. Chứa hàm process_gold_layer() nhận đầu vào là dữ liệu từ lớp Silver và tạo ra nhiều bảng tổng hợp. Mỗi bảng được tạo ra từ các phép toán groupBy() và agg() phức tạp để tính toán các chỉ số KPI theo các chiều phân tích khác nhau. Trả về một dictionary chứa các DataFrame đã được tổng hợp, cho phép main_pipeline dễ dàng quản lý và ghi nhiều bảng Gold một cách có hệ thống.

src/data_quality.py. Cung cấp lớp DataQualityChecker với các phương thức để thực hiện các kiểm tra chất lượng dữ liệu (tính hoàn chỉnh, hợp lệ, nhất quán). Đây là một thành phần có thể tái sử dụng, được tích hợp vào main_pipeline để đánh giá chất lượng dữ liệu sau lớp Silver, cung cấp một "cổng kiểm soát" trước khi dữ liệu được đưa vào lớp Gold.

3.3. Description and Sample of Transformed Datasets

Dữ liệu đã được chuyển đổi và tổ chức thành công qua 3 lớp với chất lượng tăng dần.

- Dữ liệu Lớp Bronze. Là bản sao của dữ liệu thô nhưng ở định dạng Delta Lake, đã được hợp nhất schema và bổ sung 4 cột metadata. Dữ liệu ở lớp này vẫn còn "nhiễu" và chưa được làm sạch.

	VendorID	tpep_pickup_datetime	...	Airport_fee	ingestion_timestamp
count	3.0	3	...	3.000000	3
mean	2.0	2024-02-01 00:00:43.666666	...	0.583333	2025-07-29 11:44:54.398417920
min	2.0	2024-02-01 00:00:17	...	0.000000	2025-07-29 11:44:54.398418
25%	2.0	2024-02-01 00:00:28	...	0.000000	2025-07-29 11:44:54.398417920
50%	2.0	2024-02-01 00:00:39	...	0.000000	2025-07-29 11:44:54.398417920
75%	2.0	2024-02-01 00:00:57	...	0.875000	2025-07-29 11:44:54.398417920
max	2.0	2024-02-01 00:01:15	...	1.750000	2025-07-29 11:44:54.398418
std	0.0	NaN	...	1.010363	NaN

- Dữ liệu Lớp Silver. Dữ liệu đã được làm sạch, xác thực và làm giàu. Các bản ghi không hợp lệ đã bị loại bỏ. Các cột mới như trip_duration_minutes, speed_mph đã được thêm vào.

	VendorID	tpep_pickup_datetime	...	quality_score	processing_timestamp
count	993513.000000	993513	...	993513.0	993513
mean	1.765797	2024-01-07 09:47:02.186108	...	1.0	2025-07-29 11:45:14.598440704
min	1.000000	2024-01-01 00:00:00	...	1.0	2025-07-29 11:45:14.598439
25%	2.000000	2024-01-04 14:39:11	...	1.0	2025-07-29 11:45:14.598438912
50%	2.000000	2024-01-07 03:14:11	...	1.0	2025-07-29 11:45:14.598438912
75%	2.000000	2024-01-10 14:44:46	...	1.0	2025-07-29 11:45:14.598438912
max	2.000000	2024-01-13 23:51:06	...	1.0	2025-07-29 11:45:14.598439
std	0.423500	NaN	...	0.0	NaN

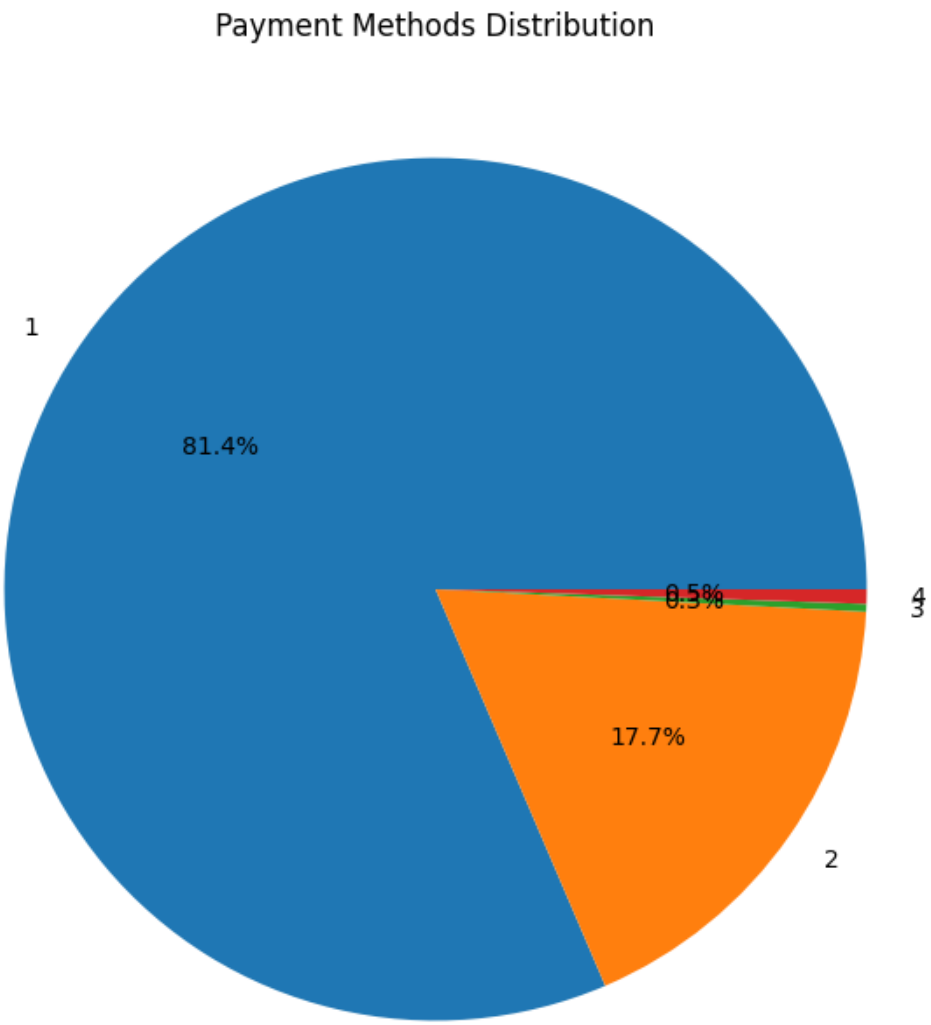
- Dữ liệu Lớp Gold. Bao gồm các bảng tổng hợp, sẵn sàng cho phân tích. Ví dụ, bảng hourly_trip_analytics chứa các chỉ số vận hành được tính toán theo từng giờ.

	pickup_hour	total_trips	avg_fare	avg_tip	...	total_revenue	max_fare	min_fare	processing_timestamp
count	24.000000	24.000000	24.000000	24.000000	...	2.400000e+01	24.000000	24.000000	24
mean	11.500000	111723.000000	19.090810	3.415876	...	3.068058e+06	570.440417	2.178333	2025-07-29 11:46:06.024773888
min	0.000000	12429.000000	16.442981	2.987693	...	4.062820e+05	292.610000	1.010000	2025-07-29 11:46:06.024774
25%	5.750000	62079.750000	17.801164	3.195237	...	1.741816e+06	453.272500	1.010000	2025-07-29 11:46:06.024773888
50%	11.500000	128594.500000	18.312622	3.429632	...	3.454927e+06	520.890000	1.010000	2025-07-29 11:46:06.024773888
75%	17.250000	159628.000000	19.348529	3.585804	...	4.293337e+06	699.302500	2.760000	2025-07-29 11:46:06.024773888
max	23.000000	191880.000000	27.862577	4.034232	...	5.312603e+06	940.930000	6.100000	2025-07-29 11:46:06.024774
std	7.071068	59766.723654	2.406652	0.267465	...	1.646398e+06	172.912298	1.656063	NaN

3.4. Data Visualization of Query Results

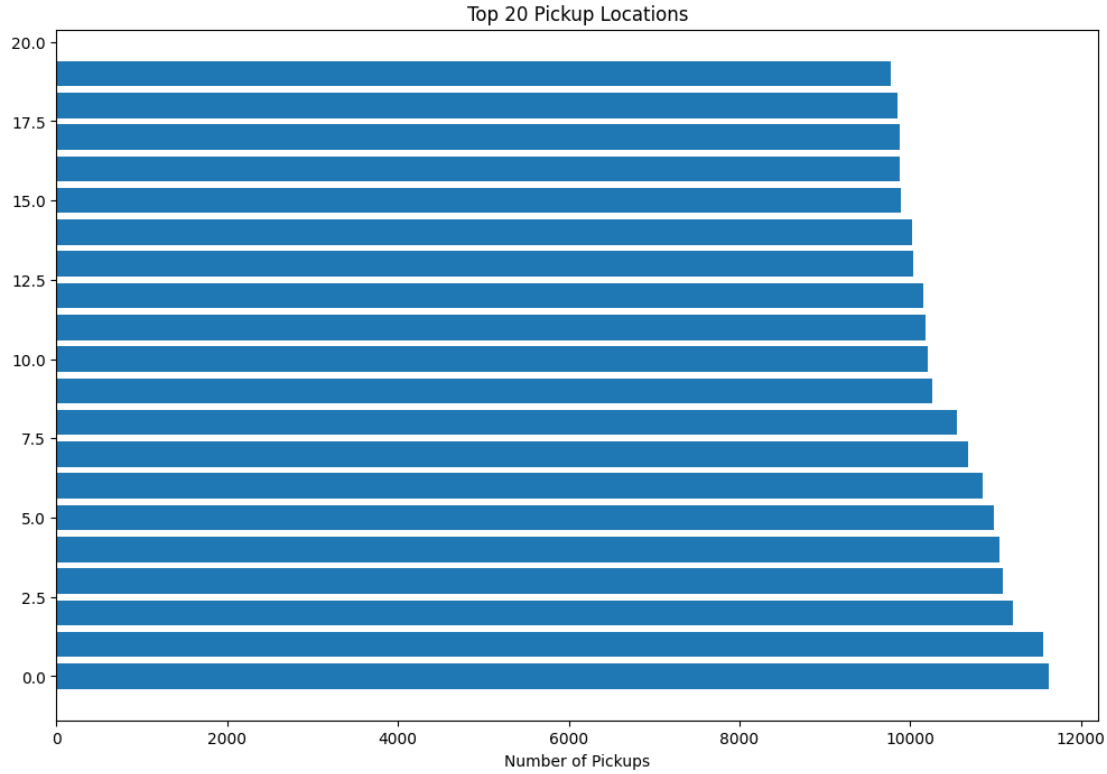
Các insight từ dữ liệu lớp Gold được trực quan hóa bằng các biểu đồ trong notebook notebooks/02_analytics_dashboard.ipynb. Các biểu đồ này không chỉ hiển thị dữ liệu mà còn kể một câu chuyện về hoạt động taxi tại NYC.

Biểu đồ 1. Phân bố Phương thức Thanh toán



- Biểu đồ cho thấy thanh toán bằng thẻ tín dụng (loại 1) chiếm đến 81.4%, trong khi tiền mặt (loại 2) chỉ chiếm 17.7%.
- Thói quen thanh toán không dùng tiền mặt đã trở thành tiêu chuẩn trong ngành taxi tại NYC.
- Các chiến lược kinh doanh, khuyến mãi, hoặc các dịch vụ tích hợp cần ưu tiên tuyệt đối cho các kênh thanh toán điện tử để tối ưu hóa trải nghiệm khách hàng và doanh thu.

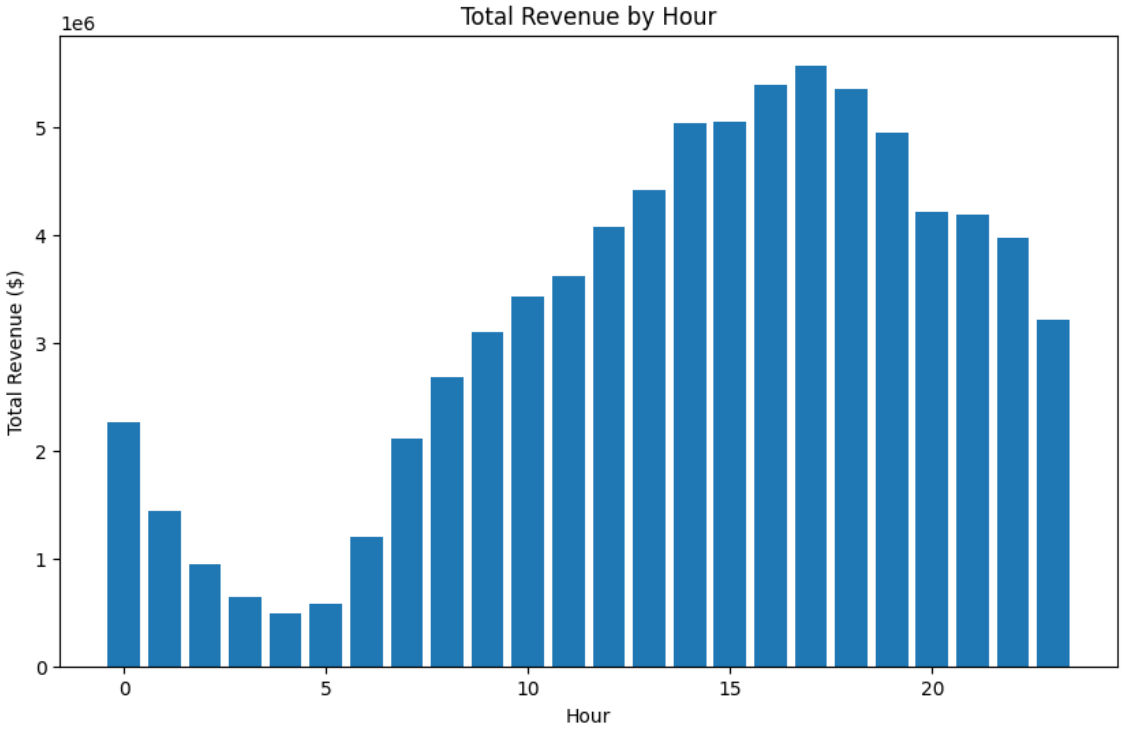
Biểu đồ 2. Top 20 Địa điểm Đón khách có Nhu cầu Cao nhất



Biểu đồ cột ngang thể hiện 20 khu vực có số lượt đón khách cao nhất. Lưu lượng ở các địa điểm hàng đầu rất cao và tương đối đồng đều, dao động trong khoảng từ 10,000 đến gần 12,000 lượt trong khoảng thời gian phân tích.

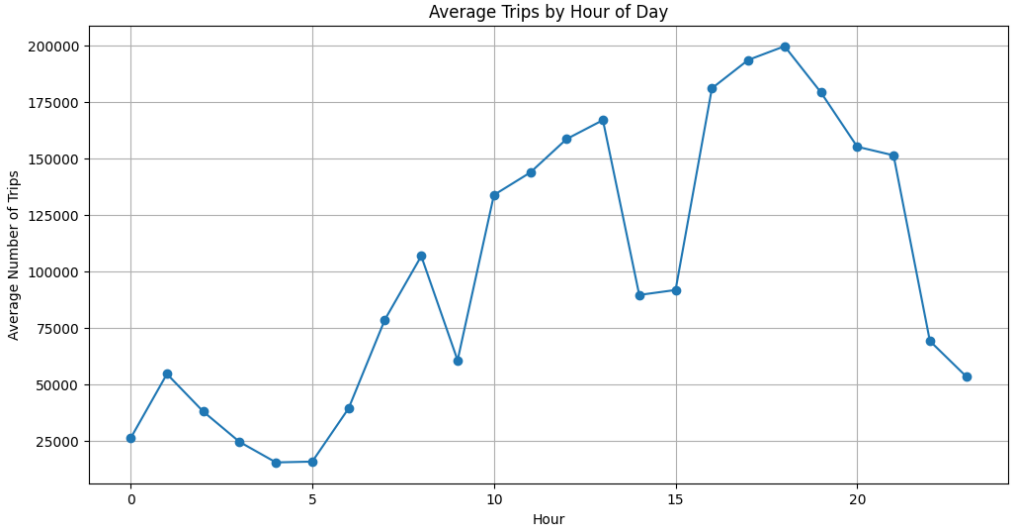
- Nhu cầu di chuyển bằng taxi tại NYC không tập trung vào một vài điểm cá biệt mà được phân bổ mạnh mẽ ở một nhóm các "điểm nóng" (hotspots) quan trọng, có thể là các trung tâm thương mại, khu văn phòng, hoặc các đầu mối giao thông công cộng lớn.
- Đây là những khu vực chiến lược. Các công ty taxi cần đảm bảo mật độ xe cao tại đây để giảm thời gian chờ của khách và tối đa hóa số chuyến đi. Đối với các nhà hoạch định chính sách đô thị, đây là những vị trí ưu tiên để xem xét cải thiện cơ sở hạ tầng giao thông và có thể là nơi lý tưởng để đề xuất đặt các trạm sạc nhanh cho taxi điện trong tương lai.

Biểu đồ 3. Tổng Doanh thu theo Khung giờ trong Ngày



- Doanh thu có mô hình biến động rất rõ rệt theo giờ. Doanh thu ở mức thấp nhất vào sáng sớm (4-5 giờ sáng), bắt đầu tăng mạnh từ sau 6 giờ sáng, và đạt đỉnh cao nhất trong khoảng thời gian từ 16.00 đến 19.00.
- Khung giờ tan tầm buổi chiều và bắt đầu các hoạt động buổi tối là "thời điểm vàng", mang lại nguồn doanh thu lớn nhất cho các tài xế và công ty taxi.
- Bất kỳ sự gián đoạn hoạt động nào trong khung giờ cao điểm này (ví dụ. hết xăng/pin, giao ca) đều gây ra thiệt hại kinh tế lớn nhất. Một chiến lược vận hành thông minh là khuyến khích tài xế nghỉ ngơi, bảo dưỡng hoặc sạc xe vào các giờ thấp điểm (như giữa buổi sáng hoặc đầu giờ chiều) để đảm bảo khả năng phục vụ tối đa trong khoảng thời gian sinh lời cao nhất.

Biểu đồ 4. Số chuyến đi Trung bình theo Khung giờ trong Ngày



- Biểu đồ đường cho thấy một mô hình nhu cầu hai đỉnh (bimodal) đặc trưng.

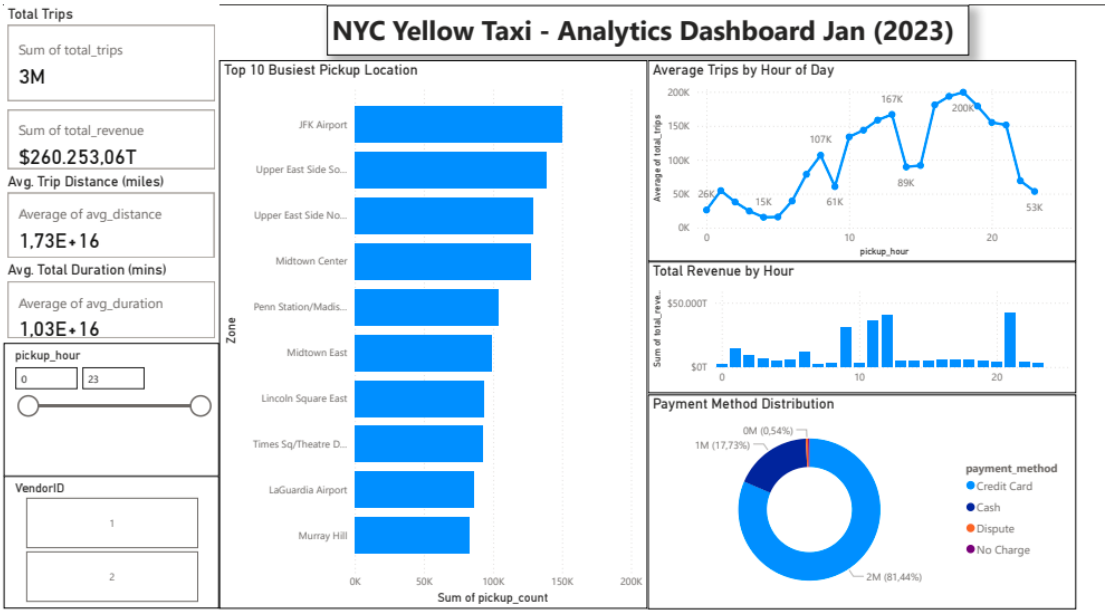
1. Đỉnh buổi sáng (Morning Rush). Một đỉnh nhỏ vào khoảng 8.00 - 9.00, tương ứng với giờ đi làm.
2. Đỉnh buổi tối (Evening Rush). Một đỉnh lớn hơn và kéo dài hơn, từ khoảng 18.00 đến 20.00, tương ứng với giờ tan làm và các hoạt động giải trí.

- Đặc biệt, có một điểm sụt giảm đột ngột vào khoảng 16.00, ngay trước khi bước vào giờ cao điểm nhất.

- Sự sụt giảm nhu cầu vào lúc 16.00 là một dấu hiệu mạnh mẽ cho thấy đây là thời điểm giao ca chính của các tài xế taxi. Họ thường kết thúc ca làm việc và bàn giao xe cho ca tiếp theo vào khoảng thời gian này.
- Thời điểm giao ca 16.00 là một "cơ hội vàng" bị bỏ lỡ nếu không được tối ưu hóa. Các công ty có thể đề xuất các điểm giao ca gần các "điểm nóng" buổi tối, hoặc đây cũng là thời gian lý tưởng nhất để các tài xế sạc pin cho xe điện, chuẩn bị năng lượng cho khung giờ hoạt động hiệu quả nhất sắp tới.

Tổng hợp Insight qua Dashboard Phân tích Toàn cảnh (January 2023)

Để tổng hợp các kết quả phân tích và cung cấp một góc nhìn quản trị toàn diện, một dashboard tương tác đã được xây dựng, thể hiện các chỉ số hiệu suất chính (KPIs) của tháng 1 năm 2023.



- Về quy mô hoạt động, trong tháng 1/2023, đã có 3 triệu chuyến đi được thực hiện, tạo ra tổng doanh thu hơn 260 triệu đô la.
- Về xu hướng, các mẫu hình được thể hiện trên dashboard (xu hướng theo giờ, phân bố địa điểm, phương thức thanh toán) hoàn toàn nhất quán với các phân tích chi tiết đã thực hiện trên các biểu đồ riêng lẻ.
- Về bộ lọc tương tác. Dashboard cho phép người dùng lọc dữ liệu theo pickup_hour và VendorID, giúp đi sâu vào các phân tích cụ thể hơn.

- Sức mạnh của các "Điểm nóng". Dashboard cho thấy Sân bay JFK là địa điểm đón khách số một, vượt trội so với các khu vực khác. Các sân bay (JFK, LaGuardia) và các khu vực trung tâm Manhattan (Upper East Side, Midtown, Penn Station) chiếm lĩnh hoàn toàn top 10, khẳng định vai trò là huyết mạch kinh tế và giao thông của thành phố.

- Mối tương quan giữa Nhu cầu và Doanh thu. Biểu đồ "Average Trips by Hour" và "Total Revenue by Hour" có hình dạng gần như tương đồng, xác nhận rằng doanh thu tăng/giảm trực tiếp theo nhu cầu di chuyển. Điều này nhấn mạnh tầm quan trọng của việc tối ưu hóa số lượng xe hoạt động trong giờ cao điểm.
- Hành vi người dùng được định hình rõ nét. Sự thống trị của thanh toán thẻ tín dụng (>81%) và nhu cầu di chuyển tăng vọt sau giờ làm việc cho thấy một tệp khách hàng chủ yếu là nhân viên văn phòng, khách du lịch và những người tham gia các hoạt động giải trí buổi tối.

- Chiến lược tập trung vào Sân bay. Các công ty taxi nên xây dựng các chiến lược ưu tiên cho khu vực sân bay, chẳng hạn như có đội xe chuyên dụng, tối ưu hóa thời gian chờ, và có thể áp dụng các mức giá đặc biệt cho các tuyến đường từ/đến sân bay.
- Tối ưu hóa "Giờ vàng". Dữ liệu từ dashboard cung cấp bằng chứng vững chắc để các nhà quản lý xây dựng chính sách khuyến khích tài xế hoạt động tích cực nhất trong khung giờ 17.00 - 20.00, ví dụ như áp dụng các chương trình thưởng theo số chuyến hoặc doanh thu.

4. Projected Impact

4.1. Accomplishments and Benefits

Dự án đã hoàn thành thành công mục tiêu cốt lõi là xây dựng một pipeline dữ liệu end-to-end, dựa trên kiến trúc Medallion, có khả năng xử lý và cung cấp các insight giá trị từ dữ liệu taxi NYC.

1. Xây dựng thành công Pipeline Dữ liệu End-to-End. Đã thiết kế và triển khai một pipeline hoàn chỉnh, từ việc nạp dữ liệu thô (Bronze), qua làm sạch (Silver), đến tổng hợp (Gold), chứng minh năng lực xây dựng một hệ thống dữ liệu có cấu trúc và đáng tin cậy.
2. Chuyển đổi Dữ liệu Thô thành Tài sản Phân tích. Pipeline đã biến đổi thành công dữ liệu thô, phức tạp thành các bộ dữ liệu sạch, đã được làm giàu và sẵn sàng cho phân tích (analysis-ready datasets). Điều này giúp loại bỏ rào cản kỹ thuật cho người dùng cuối và tăng tốc độ khai thác thông tin.
3. Cung cấp Insight Kinh doanh có thể Hành động được. Thông qua một dashboard tương tác, dự án đã cung cấp các insight kinh doanh quan trọng và dễ hiểu, hỗ trợ việc ra quyết định chiến lược. Cụ thể, dashboard đã làm rõ.
   - Xu hướng theo thời gian. Nhu cầu hoạt động đạt đỉnh vào 17.00 - 20.00.
   - Hành vi thanh toán. Thanh toán không dùng tiền mặt (thẻ tín dụng) chiếm ưu thế tuyệt đối với hơn 81%.
   - Các "điểm nóng" địa lý. Các sân bay (JFK, LaGuardia) và các trung tâm thương mại tại Manhattan là những khu vực có nhu cầu cao nhất.

Lợi ích mang lại bao gồm:

- Khả năng mở rộng (Scalability). Kiến trúc và công nghệ được lựa chọn (Spark, Parquet, Delta Lake) cho phép hệ thống dễ dàng xử lý khối lượng dữ liệu lớn hơn nhiều (hàng Terabyte) trong tương lai.
- Tính chịu lỗi (Fault-Tolerance). Việc sử dụng Spark và Delta Lake mang lại khả năng phục hồi sau lỗi, đảm bảo tính toàn vẹn của dữ liệu trong quá trình xử lý.
- Đảm bảo chất lượng dữ liệu (Data Quality). Kiến trúc Medallion, cùng với framework kiểm tra chất lượng, tạo ra một nền tảng dữ liệu đáng tin cậy.
- Nền tảng cho Quyết định Dựa trên Dữ liệu. Cung cấp một "Nguồn sự thật duy nhất" (Single Source of Truth) cho các nhà phân tích và quản lý, giúp họ đưa ra quyết định kinh doanh chính xác hơn.
- Tối ưu hóa Vận hành. Các insight về giờ cao điểm và địa điểm nóng giúp các công ty taxi tối ưu hóa việc điều phối tài xế, giảm thời gian chờ và tăng hiệu suất hoạt động.
- Cải thiện Trải nghiệm Khách hàng. Hiểu rõ hành vi khách hàng (ví dụ. thói quen thanh toán) giúp doanh nghiệp cải tiến dịch vụ và các hệ thống hỗ trợ.

4.2. Future Improvements

Mặc dù dự án đã đạt được các mục tiêu chính, vẫn còn nhiều cơ hội để cải tiến và mở rộng hệ thống trong tương lai.

Về hạn chế hiện tại,

- Xử lý luồng còn ở mức mô phỏng. Pipeline streaming hiện tại sử dụng file source để mô phỏng, chưa kết nối với một nguồn dữ liệu thời gian thực thực sự như Kafka.
- Thiếu các mô hình dự đoán. Dự án tập trung vào phân tích mô tả (descriptive analytics), chưa khai thác tiềm năng của phân tích dự đoán (predictive analytics).
- Triển khai trên môi trường local. Hệ thống chưa được triển khai trên một nền tảng đám mây, hạn chế khả năng mở rộng và tính sẵn sàng cao.

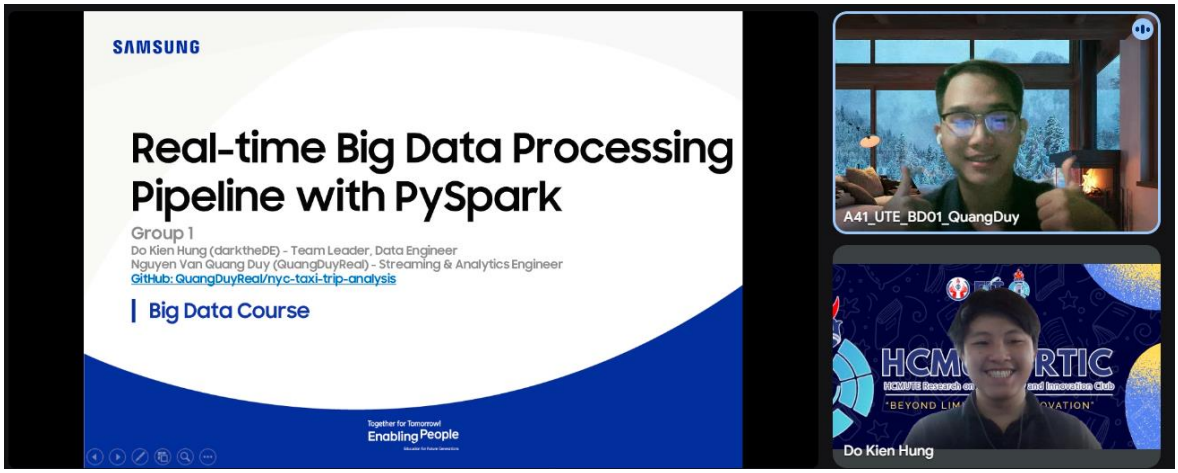
Về định hướng phát triển trong tương lai,

1. Nâng cấp Xử lý Luồng (Advanced Streaming). Tích hợp Apache Kafka làm nguồn dữ liệu đầu vào cho pipeline streaming.
   - Cho phép hệ thống xử lý dữ liệu thực sự trong thời gian thực (true real-time), cung cấp các insight tức thì và giảm độ trễ xuống mức tối thiểu.

2. Tích hợp Học máy (Machine Learning). Xây dựng và tích hợp các mô hình học máy vào pipeline.
   - Phát triển mô hình dự đoán thời gian của chuyến đi dựa trên thời gian trong ngày, địa điểm đón/trả và điều kiện giao thông.
   - Xây dựng mô hình dự đoán giá cước (fare amount).
   - Lợi ích. Cung cấp các tính năng thông minh cho ứng dụng đặt xe, giúp người dùng và tài xế có ước tính chính xác hơn.

3. Triển khai trên Nền tảng Đám mây (Cloud Deployment). Di chuyển và triển khai toàn bộ pipeline lên một nền tảng đám mây như AWS hoặc Azure.
   - Công nghệ liên quan có thể dùng AWS EMR/Glue, Azure Databricks, S3/Azure Data Lake Storage.
   - Tận dụng khả năng co giãn linh hoạt, tính sẵn sàng cao và hệ sinh thái dịch vụ phong phú của đám mây, giúp hệ thống sẵn sàng cho môi trường sản xuất (production-ready).

5. Team Member Review and Comment



NAME	REVIEW and COMMENT
Kiến Hưng	Dự án này là một cơ hội tuyệt vời để tôi áp dụng sâu các kiến thức về kiến trúc Medallion và tối ưu hóa Spark. Việc thiết kế các lớp Silver và Gold, đặc biệt là xử lý các quy tắc nghiệp vụ phức tạp, đã giúp tôi củng cố vững chắc kỹ năng xử lý dữ liệu. Thách thức lớn nhất là đảm bảo logic biến đổi dữ liệu vừa chính xác vừa hiệu quả về mặt hiệu năng. Tôi rất hài lòng với cấu trúc mã nguồn mà nhóm đã xây dựng được
Quang Duy	Vai trò của tôi trong dự án này là cầu nối giữa phần xử lý dữ liệu nền và kết quả phân tích cuối cùng. Tôi đã học được rất nhiều về cách điều phối một pipeline phức tạp, quản lý các dependencies giữa các bước và tầm quan trọng của việc xuất dữ liệu một cách sạch sẽ cho người dùng cuối. Thách thức thú vị nhất là làm sao để trực quan hóa một lượng lớn dữ liệu tổng hợp một cách hiệu quả nhất, giúp người xem nắm bắt ngay các điểm chính.

6. Instructor Review and Comment

CATEGORY	SCORE	REVIEW and COMMENT
IDEA	__/10	
APPLICATION	__/30	

RESULT	__/30	
PROJECT MANAGEMENT	__/10	
PRESENTATION & REPORT	__/20	
TOTAL	__/100	