**ĐẠI HỌC UEH**

**TRƯỜNG CÔNG NGHỆ VÀ THIẾT KẾ**

**KHOA CÔNG NGHỆ THÔNG TIN KINH DOANH**

****

**ĐỒ ÁN MÔN HỌC**

**DỮ LIỆU LỚN VÀ ỨNG DỤNG**

**ĐỀ TÀI: Ứng dụng Big Data để phân khúc khách hàng trên**

**bộ dữ liệu mô phỏng giao dịch Walmart**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **GVHD** | | **:** | TS. Nguyễn Mạnh Tuấn | |
|  | **Mã LHP** | | **:** | INF509079 | |
|  | **Nhóm thực hiện** | | **:** | Nhóm 05 | |
|  | **Thành viên** | | **:** |  | |
|  | | Thái Bảo An | | | 31211027641 |
|  | | Hà Gia Hiến | | | 31211021362 |
|  | | Đặng Nhật Huy | | | 31211027641 |
|  | | Nguyễn Tân Niên | | | 31211027661 |
|  | | Nguyễn Hoàng Minh Trí | | | 31211023349 |

***TP. Hồ Chí Minh, ngày 29 tháng 5 năm 2024***

# LỜI CẢM ƠN

Trước khi bắt đầu phần trình bày nội dung đồ án, chúng em muốn bày tỏ lòng biết ơn sâu sắc đến Thầy Nguyễn Mạnh Tuấn - giảng viên hướng dẫn môn học Dữ liệu lớn và ứng dụng.

Thầy đã chia sẻ với chúng em những kiến thức quý báu và hỗ trợ tận tình suốt thời gian húng em theo học môn này. Chúng em cũng muốn gửi lời cảm ơn đến khoa và các thầy cô đã tạo điều kiện tốt nhất cho chúng em thực hiện đề tài và hoàn thiện đồ án này.

Cuối cùng, nhóm chúng em xin chúc sức khỏe và thành công đến tất cả quý thầy cô trong khoa Công Nghệ Thông Tin Kinh Doanh cũng như Thầy Nguyễn Mạnh Tuấn, để thầy cô tiếp tục truyền đạt kiến thức và truyền cảm hứng cho thế hệ tương lai. Xin chân thành cảm ơn các thầy cô rất nhiều!

# 

# LỜI NÓI ĐẦU

Trong một thời đại mà sự sống còn của doanh nghiệp ngày càng phụ thuộc vào khả năng nắm bắt thông tin và biến nó thành lợi ích cạnh tranh, big data đã trở thành một vũ khí quan trọng không thể thiếu. Các doanh nghiệp thông minh không chỉ nhận thức được giá trị của dữ liệu lớn mà còn biết cách khai thác nó để tạo ra lợi ích thực tiễn. Dữ liệu lớn không chỉ là một khái niệm mà là trung tâm của cuộc cách mạng công nghệ, mang lại cơ hội vô tận và thách thức không nhỏ. Việc hiểu và sử dụng hiệu quả dữ liệu lớn không chỉ là một yếu tố quan trọng mà còn là chìa khóa để thúc đẩy sự phát triển bền vững trong mọi lĩnh vực, từ kinh doanh đến y tế, từ giáo dục đến khoa học.

Đồ án này mở đầu với một tư duy đơn giản nhưng phản ánh sự phức tạp của thế giới kinh doanh hiện đại: "Những doanh nghiệp nào đã thành công trong việc ứng dụng Big Data và làm thế nào họ đã làm điều đó?". Nhóm đã bắt đầu nghiên cứu này với mục tiêu tìm hiểu, phân tích và đúc kết các trường hợp thành công của các doanh nghiệp, từ đó thu thập những bài học quý giá cũng như trực tiếp ứng dụng big data vào thực tiễn để phát triển cái nhìn sâu rộng về cách big data đang thay đổi bức tranh kinh doanh hiện nay.

Bối cảnh của đề tài này không chỉ là về sự phát triển của big data mà còn là về sự thay đổi sâu sắc trong cách mà doanh nghiệp tiếp cận và sử dụng thông tin. Bằng cách tìm hiểu các công nghệ, các phương pháp và các chiến lược mà các doanh nghiệp ứng dụng thành công, nhóm hy vọng đồ án này sẽ đem lại cái nhìn tổng thể và sâu sắc về vai trò của big data trong kinh doanh hiện đại và cung cấp những bài học thực tiễn cho những ai quan tâm đến lĩnh vực này.

# MỤC LỤC

[LỜI CẢM ƠN 2](#_Toc167908761)

[LỜI NÓI ĐẦU 3](#_Toc167908762)

[MỤC LỤC 4](#_Toc167908763)

[DANH MỤC HÌNH ẢNH 7](#_Toc167908764)

[DANH MỤC BẢNG BIỂU 8](#_Toc167908765)

[CHƯƠNG 1: TỔNG QUAN 9](#_Toc167908766)

[1.1. Giới thiệu về đề tài 9](#_Toc167908767)

[1.1.1. Lý do chọn đề tài 9](#_Toc167908768)

[1.1.2. Mục tiêu nghiên cứu 10](#_Toc167908769)

[1.1.3. Phạm vi nghiên cứu 10](#_Toc167908770)

[1.2. Giới thiệu về doanh nghiệp 11](#_Toc167908771)

[1.2.1. Sơ lược về Walmart 11](#_Toc167908772)

[1.2.2. Sơ lược về Uber 11](#_Toc167908773)

[CHƯƠNG 2: BIG DATA TRONG DOANH NGHIỆP 12](#_Toc167908774)

[2.1. Cách Walmart áp dụng Big Data 12](#_Toc167908775)

[2.2. Cách Uber áp dụng Big Data 13](#_Toc167908776)

[2.3. Công nghệ sử dụng 16](#_Toc167908777)

[2.3.1. Các công nghệ được Uber sử dụng 16](#_Toc167908778)

[2.3.2. Các công nghệ được Walmart sử dụng 18](#_Toc167908779)

[2.4. Tìm hiểu về các công nghệ đã chọn 19](#_Toc167908780)

[2.4.1. Công nghệ 1: Apache Hbase 19](#_Toc167908781)

[2.4.2. Công nghệ 2: Apache Spark 23](#_Toc167908782)

[CHƯƠNG 3: CƠ SỞ LÝ THUYẾT 28](#_Toc167908783)

[3.1. Dữ liệu lớn 28](#_Toc167908784)

[3.2. *PySpark* 29](#_Toc167908785)

[3.3. Tổng quan các thư viện sử dụng 29](#_Toc167908786)

[3.3.1. Thư viện Pandas 29](#_Toc167908787)

[3.3.2 Thư viện Plotly 30](#_Toc167908788)

[3.4. Mô hình RFM 30](#_Toc167908789)

[CHƯƠNG 4: TRIỂN KHAI 32](#_Toc167908790)

[4.1. Giới thiệu về bộ dữ liệu 32](#_Toc167908791)

[4.1.1. Mô tả bộ dữ liệu 32](#_Toc167908792)

[4.1.2. Các thuộc tính 32](#_Toc167908793)

[4.2. Môi trường lập trình 33](#_Toc167908794)

[4.3. Tiền xử lý 33](#_Toc167908795)

[4.3.1. Mô tả dữ liệu 33](#_Toc167908796)

[4.3.2. Biến đổi cột *Price* 34](#_Toc167908797)

[4.3.3. Kiểm tra các giá trị thiếu (missing values) 34](#_Toc167908798)

[4.3.4. Kiểm tra cột *Price* 35](#_Toc167908799)

[4.3.5. Kiểm tra giá trị trùng lặp. 36](#_Toc167908800)

[4.3.6. Loại bỏ những hóa đơn bị hủy 37](#_Toc167908801)

[4.3.7. Thêm thuộc tính *TotalPrice* 37](#_Toc167908802)

[4.3.8 Tổng hợp script xử lý tiền xử lý dữ liệu 37](#_Toc167908803)

[4.4. Một số truy vấn cơ bản 39](#_Toc167908804)

[4.4.1. Tổng số hóa đơn theo ngày 39](#_Toc167908805)

[4.4.2. Top 10 khách hàng chi tiêu nhiều nhất 40](#_Toc167908806)

[4.4.3. Những số liệu trên từng quốc gia (Số lượng khách hàng, số hóa đơn, tổng tiền chi tiêu) 41](#_Toc167908807)

[4.4.4. Những sản phẩm được mua nhiều nhất. 42](#_Toc167908808)

[CHƯƠNG 5: PHÂN KHÚC KHÁCH HÀNG 43](#_Toc167908809)

[5.1. Thực hiện phân khúc 43](#_Toc167908810)

[5.1.1. Tính điểm Recency 43](#_Toc167908811)

[5.1.2. Tính điểm Frequency 45](#_Toc167908812)

[5.1.3. Tính điểm Monetary 46](#_Toc167908813)

[5.1.4. Phân khúc khách hàng 47](#_Toc167908814)

[5.2. Biểu diễn trực quan 49](#_Toc167908815)

[CHƯƠNG 6: KẾT LUẬN VÀ ĐỀ XUẤT 50](#_Toc167908816)

[6.1. Kết quả đạt được 50](#_Toc167908817)

[6.2. Đề xuất 51](#_Toc167908818)

[6.2.1. Hạn chế còn tồn động 51](#_Toc167908819)

[6.2.2. Hướng phát triển tiếp theo 52](#_Toc167908820)

[TÀI LIỆU THAM KHẢO 53](#_Toc167908821)

[PHỤ LỤC 54](#_Toc167908822)

[Bảng phân công 54](#_Toc167908823)

[Chương 2.4. Tìm hiểu về các công nghệ đã chọn 54](#_Toc167908824)

[Các đường link 55](#_Toc167908825)

# 

# DANH MỤC HÌNH ẢNH

[Hình 1.1: Dự báo doanh thu thị trường Big Data 7](#_heading=h.35nkun2)

[Hình 1.2.1: Logo Walmart 9](#_heading=h.3j2qqm3)

[Hình 1.2.2: Logo Uber 10](#_heading=h.4i7ojhp)

[Hình 2.1: Mối quan hệ trong nền tảng Big Data của Uber 14](#_heading=h.3as4poj)

[Hình 2.2: Kiến trúc kỹ thuật và hệ sinh thái tiếp thị trực tuyến của Walmart 16](#_heading=h.49x2ik5)

[Hình 2.3: WordCloud 25](#_heading=h.23ckvvd)

[Hình 3.1: Mô hình RFM 28](#_heading=h.1v1yuxt)

# 

# DANH MỤC BẢNG BIỂU

# 

[Bảng 3.1: Ý nghĩa thang điểm RFM 29](#_heading=h.4f1mdlm)

[Bảng 3.2: Tiêu chí phân khúc khách hàng dựa trên điểm RFM 30](#_heading=h.2u6wntf)

[Bảng 4.1: Thuộc tính bộ dữ liệu “Online Retail II” 31](#_heading=h.37m2jsg)

[Biểu đồ 4.1: Tổng số hóa đơn theo ngày 37](#_heading=h.sqyw64)

[Biểu đồ 4.2: Top 10 khách hàng chi tiêu nhiều nhất 38](#_heading=h.1rvwp1q)

[Biểu đồ 4.3: Top 10 sản phẩm được mua nhiều nhất 39](#_heading=h.1664s55)

[Biểu đồ 5.1: Tỷ lệ các phân khúc khách hàng của Walmart 47](#_heading=h.48pi1tg)

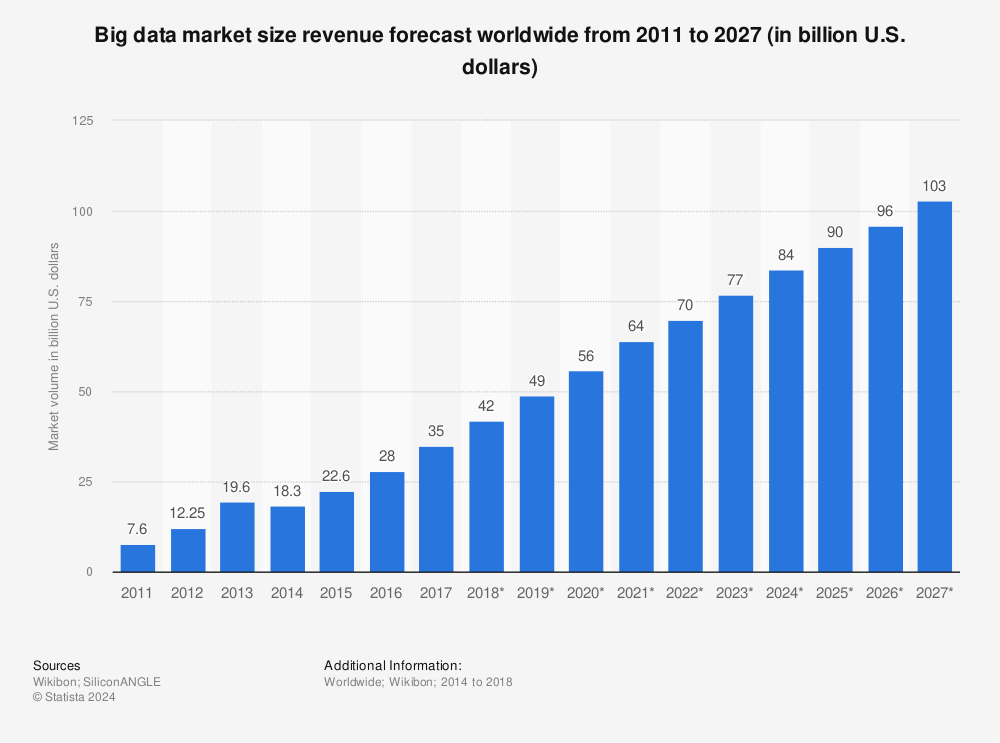
# 

# CHƯƠNG 1: TỔNG QUAN

## 1.1. Giới thiệu về đề tài

### 1.1.1. Lý do chọn đề tài

Ngày nay, xã hội ngày càng phát triển, nhu cầu của con người từ đó mà ngày càng tăng đòi hỏi doanh nghiệp phải cung cấp các sản phẩm cần thiết đến cho những khách hàng đang thực sự cần loại sản phẩm đó. Trong thập kỷ vừa qua, công nghệ Big Data đã chứng kiến sự phát triển vượt bậc trên khắp thế giới, từ đó mở ra những cơ hội mới và đột phá trong việc phân tích và dự đoán hành vi của khách hàng. Big Data không chỉ cho phép các doanh nghiệp thu thập và lưu trữ lượng dữ liệu khổng lồ mà còn giúp họ hiểu sâu hơn về nhu cầu và sở thích của khách hàng thông qua các công cụ phân tích tiên tiến.



#### Hình 1.1: Dự báo doanh thu thị trường Big Data

Walmart, một trong những tập đoàn bán lẻ lớn nhất thế giới, là điển hình tiêu biểu của việc áp dụng Big Data vào hoạt động kinh doanh. Với hàng triệu giao dịch mỗi ngày, Walmart sở hữu một lượng dữ liệu khách hàng khổng lồ và đa dạng. Việc ứng dụng Big Data vào phân tích hành vi khách hàng không chỉ giúp Walmart nâng cao hiệu quả trong việc cung cấp sản phẩm mà còn cải thiện trải nghiệm mua sắm của khách hàng. Bằng cách hiểu rõ hơn về thói quen mua sắm và nhu cầu của khách hàng, Walmart có thể tối ưu hóa quản lý hàng tồn kho, đưa ra các chương trình khuyến mãi phù hợp, và nâng cao chất lượng dịch vụ. Với nguồn thông tin từ khách hàng dồi dào, ta có thể sử dụng và tiến hành các phân tích với kết quả trực quan, thực tế nhất.

Đề tài này sẽ tập trung vào việc khai thác lượng dữ liệu khách hàng khổng lồ của Walmart, từ đó dựa vào các nguồn dữ liệu này để tiến hành việc phân tích hành vi mua sắm của khách hàng, giúp cải thiện việc cung cấp sản phẩm đến cho đúng khách hàng. Việc nghiên cứu này sẽ giúp chúng ta hiểu rõ hơn về khách hàng, phân loại được các đối tượng khách hàng để có những đề xuất cung cấp các sản phẩm thiết yếu.

### 1.1.2. Mục tiêu nghiên cứu

Nghiên cứu này nhằm đạt được hai mục tiêu chính:

* Phân khúc khách hàng tại Walmart, trong đó bao gồm khám phá các xu hướng mua sắm, sở thích sản phẩm và các yếu tố ảnh hưởng đến quyết định mua hàng của khách hàng và xác định các nhóm khách hàng dựa trên số liệu các hóa đơn.
* Đưa ra các đề xuất để cải thiện việc cung cấp sản phẩm thiết yếu dựa trên kết quả phân tích để tối ưu hóa quản lý hàng tồn kho và phân phối sản phẩm. Ngoài ra, bằng cách hiểu rõ các phân khúc khách hàng, doanh nghiệp có thể thiết kế các chiến lược tiếp thị nhắm vào từng phân khúc cụ thể, từ đó tối ưu hóa ngân sách tiếp thị và cải thiện hiệu quả chiến dịch tiếp thị.

### 1.1.3. Phạm vi nghiên cứu

Phạm vi nghiên cứu của đồ án này tập trung vào việc phân tích dữ liệu khách hàng và giao dịch của Walmart, được cung cấp từ nền tảng Kaggle. Cụ thể:

* Dữ liệu khách hàng: Bao gồm thông tin về nhân khẩu học, lịch sử mua sắm và các tương tác khác với hệ thống của Walmart.
* Dữ liệu sản phẩm: Gồm thông tin chi tiết về các sản phẩm được mua, danh mục sản phẩm, giá cả và khuyến mãi liên quan.
* Dữ liệu giao dịch: Chứa các bản ghi chi tiết về từng giao dịch mua bán, bao gồm thời gian, địa điểm và số lượng sản phẩm.

Nghiên cứu sẽ sử dụng các phương pháp phân tích dữ liệu hiện đại để khai thác các thông tin quan trọng từ các bộ dữ liệu này, từ đó đưa ra các kết luận và đề xuất có giá trị cho Walmart trong việc cải thiện dịch vụ và cung cấp sản phẩm thiết yếu cho khách hàng.

## 1.2. Giới thiệu về doanh nghiệp

### 1.2.1. Sơ lược về Walmart



#### Hình 1.2.1: Logo Walmart

Walmart Inc. là một tập đoàn bán lẻ đa kênh của Mỹ, hoạt động mạng lưới siêu thị (còn được gọi là siêu trung tâm), cửa hàng giảm giá và cửa hàng tạp hóa tại Hoa Kỳ, có trụ sở tại Bentonville, Arkansas. Doanh nghiệp này được thành lập bởi hai anh em Sam và James “Bud” Walton tại Rogers, Arkansas vào năm 1962 và được công ty đăng ký theo Luật Tổng hợp Delaware vào ngày 31 tháng 10 năm 1969. Ngoài ra, Walmart còn sở hữu và vận hành các cửa hàng Sam’s Club.

Hiện nay, Walmart có 10.586 cửa hàng và câu lạc bộ tại 24 quốc gia, hoạt động dưới 46 tên khác nhau. Walmart là công ty lớn nhất thế giới tính theo doanh thu, theo danh sách Fortune Global 500 vào tháng 10 năm 2022. Walmart cũng là nhà tuyển dụng tư nhân lớn nhất thế giới với 2,2 triệu nhân viên.

### 1.2.2. Sơ lược về Uber

****

#### Hình 1.2.2: Logo Uber

Uber, thành lập vào năm 2009, được sáng lập bởi Garrett Camp, Oscar Salazar và Travis Kalanick, đã trở thành một biểu tượng của nền kinh tế chia sẻ và đã thay đổi cách chúng ta di chuyển. Ý tưởng ban đầu của họ là tạo ra một nền tảng giúp người dùng dễ dàng và tiết kiệm chi phí khi đặt xe.

Hiện nay, Uber hoạt động tại khoảng 70 quốc gia và 10.500 thành phố trên toàn thế giới. Đây là công ty dẫn đầu trong lĩnh vực chia sẻ xe với hơn 150 triệu người dùng hàng tháng và 6 triệu tài xế và người giao hàng. Tính đến năm 2023, Uber đã thực hiện hơn 47 tỷ chuyến đi kể từ khi thành lập.

Ngoài việc giúp người dùng tìm cách di chuyển từ điểm A đến điểm B, Uber còn cung cấp dịch vụ đặt thức ăn, giúp người dùng tiếp cận dịch vụ y tế, tạo ra giải pháp đặt hàng vận chuyển hàng hóa và hỗ trợ doanh nghiệp cung cấp trải nghiệm du lịch cho nhân viên. Hiện tại Uber cũng đã chiếm 67% thị phần dịch vụ gọi xe tại Mỹ vào năm 2019 và đã trở thành lựa chọn hàng đầu cho nhiều nhu cầu vận chuyển khác nhau.

# CHƯƠNG 2: BIG DATA TRONG DOANH NGHIỆP

## 2.1. Cách Walmart áp dụng Big Data

Walmart nhận thức sớm về sức mạnh của phân tích dữ liệu và đang xây dựng đám mây riêng lớn nhất thế giới, có khả năng xử lý 2,5 petabyte dữ liệu mỗi giờ. Bước ngoặt đến vào năm 2004 khi Giám đốc thông tin Linda Dillman phát hiện ra tiềm năng của phân tích dữ liệu thông qua việc phân tích dữ liệu một cách toàn diện. Điều này đã giúp Walmart nắm bắt nhu cầu tiêu dùng trong những tình huống khó khăn, nhưng ví dụ như việc nhận ra sự tăng trưởng trong việc bán bánh quy dâu tây trước cơn bão. Walmart mở rộng bộ phận dữ liệu lớn và phân tích của mình và công bố kế hoạch thành lập đám mây dữ liệu riêng lớn nhất thế giới vào năm 2015.

Dữ liệu lớn đã giúp Walmart giải quyết các thách thức cụ thể, như tối ưu hóa chiến lược giá cả và cung cấp các sản phẩm phù hợp đúng thời điểm và địa điểm. Một trong những sáng kiến đột phá của Walmart là Data Café, một trung tâm phân tích hiện đại.

Chi tiết kỹ thuật:

* Data Café kết nối thông tin từ 200 nguồn khác nhau, bao gồm dữ liệu khí tượng, dữ liệu kinh tế, dữ liệu Nielsen, dữ liệu viễn thông, dữ liệu mạng xã hội, giá xăng, và cơ sở dữ liệu sự kiện địa phương.
* Các thuật toán của Walmart được thiết kế để nhanh chóng xử lý các bộ dữ liệu lớn và đa dạng này, tìm ra giải pháp trong thời gian thực.

Hệ thống Data Café đã giảm đáng kể thời gian từ việc xác định vấn đề đến việc đề xuất giải pháp, từ hàng tuần xuống còn khoảng 20 phút trung bình. Cơ sở dữ liệu giao dịch thời gian thực của Walmart, bao gồm 40 petabyte dữ liệu, là điểm mấu chốt của hạ tầng phân tích của họ, được hỗ trợ bởi các công nghệ như Hadoop, Spark và Cassandra.

Walmart đã thành công trong việc sử dụng Big Data để giải quyết nhiều thách thức trong ngành bán lẻ. Tại trung tâm phân tích dữ liệu thời gian thực của họ, họ đã thu thập và phân tích hàng tỷ giao dịch từ hàng loạt cửa hàng trên toàn thế giới. Bằng cách này, họ có thể theo dõi và hiểu rõ hơn về các mô hình mua sắm của khách hàng, từ đó đưa ra các quyết định kinh doanh đúng đắn và chiến lược marketing mục tiêu hơn.

Walmart không chỉ dừng lại ở việc thu thập dữ liệu từ giao dịch mua bán, họ còn tích hợp dữ liệu xã hội để dự đoán hành vi tiêu dùng. Nhờ đó, họ có thể đưa ra các dự đoán về xu hướng mua sắm và sở thích của khách hàng, từ đó tinh chỉnh và cá nhân hóa các chiến lược kinh doanh, quảng cáo và sản phẩm.

Ngoài ra, Walmart đã sử dụng dữ liệu để cải thiện quản lý hàng tồn kho bằng cách giảm lượng hàng tồn đọng trong các cửa hàng mà vẫn đảm bảo cung cấp đủ số lượng sản phẩm cho từng cửa hàng. Họ cũng tối ưu hóa chuỗi cung ứng để giảm chi phí và tăng hiệu suất.

Cuối cùng, nhờ vào việc có được dữ liệu bán hàng thời gian thực, các quản lý của Walmart hiện có thể đưa ra quyết định dựa trên dữ liệu chính xác và kịp thời, giúp cải thiện hiệu quả kinh doanh và tăng cường trải nghiệm của khách hàng.

## 2.2. Cách Uber áp dụng Big Data

Uber đã trải qua một hành trình đầy đổi mới và tiến bộ trong việc áp dụng nguyên tắc của dữ liệu lớn thông qua các thế hệ khác nhau:

*Thế hệ 1: Lập nền tảng (Trước năm 2014)*

Uber bắt đầu với một lượng dữ liệu hạn chế được lưu trữ trong các cơ sở dữ liệu truy vấn và xử lý trực tuyến truyền thống như MySQL và PostgreSQL. Việc truy cập dữ liệu phân tán, không có cái nhìn toàn cầu nào có sẵn. Khi Uber mở rộng quy mô toàn cầu, nhu cầu về việc truy cập dữ liệu trung tâm trở nên rõ ràng, dẫn đến việc tạo ra kho dữ liệu phân tích đầu tiên. Người dùng dữ liệu được phân loại thành các nhóm hoạt động thành phố, các nhà khoa học dữ liệu và các nhóm kỹ thuật, mỗi nhóm có nhu cầu cụ thể về việc truy cập và phân tích dữ liệu. Kho dữ liệu phân tích, được hỗ trợ bởi Vertica, tổng hợp dữ liệu từ các nguồn khác nhau vào một nơi và cung cấp một giao diện SQL tiêu chuẩn cho người dùng.

*Thế hệ 2: Nắm bắt Hadoop (Sau năm 2014)*

Để vượt qua các hạn chế của thế hệ đầu tiên, Uber dịch chuyển sang một kiến ​​trúc dựa trên Hadoop, giới thiệu một hồ dữ liệu cho việc tiếp nhận dữ liệu nguyên thô. Các công nghệ như Presto, Apache Spark và Apache Hive được tích hợp để cung cấp các tùy chọn truy vấn linh hoạt và xử lý các loại dữ liệu khác nhau. Quy trình mô hình hóa và biến đổi dữ liệu đã được chuyển sang Hadoop, giảm chi phí và cải thiện tính mở rộng. Định dạng Parquet đã được áp dụng cho hiệu suất lưu trữ. Việc mở rộng tỷ lệ ngang đã được nhấn mạnh, làm cho tất cả các dịch vụ dữ liệu trong hệ sinh thái có thể mở rộng để đáp ứng nhu cầu tăng trưởng.

*Thế hệ 3: Bền vững lâu dài (Từ đầu năm 2017 về sau)*

Với việc kích thước dữ liệu và nhu cầu của người dùng tăng lên, Uber tập trung vào giải quyết các hạn chế về tính mở rộng và giảm thiểu độ trễ dữ liệu. Các thách thức như giới hạn về khả năng mở rộng của HDFS, tính mới mẻ của dữ liệu, hỗ trợ cho các hoạt động cập nhật/xóa và ETL và mô hình hóa nhanh hơn đã được xác định và nhắm vào cải thiện. Các giải pháp bao gồm việc mở rộng HDFS, tái kiến ​​trúc các đường ống dữ liệu để tiếp nhận từng phần của dữ liệu một cách tăng dần và giải quyết các hạn chế của Hadoop và Parquet để hỗ trợ các hoạt động cập nhật. Mục tiêu là đảm bảo tăng trưởng và bền vững lâu dài của nền tảng Dữ liệu lớn, đồng thời đáp ứng nhu cầu dữ liệu thời gian thực của hoạt động Uber

Với những yêu cầu đã nêu ở trên, họ đã phát triển Hadoop Upserts và Incremental (Hudi), một thư viện mã nguồn mở dựa trên Spark cung cấp một lớp trừu tượng trên HDFS và Parquet để hỗ trợ các thao tác cập nhật và xóa cần thiết. Hudi có thể được sử dụng từ bất kỳ Spark job nào, có khả năng mở rộng theo chiều ngang và chỉ dựa vào HDFS để hoạt động. Do đó, bất kỳ nền tảng Big Data nào cần hỗ trợ các hoạt động cập nhật/xóa cho dữ liệu lịch sử đều có thể tận dụng Hudi. Hudi cho phép truy xuất dữ liệu đã thay đổi một cách tăng dần và cải thiện hiệu suất truy vấn, giảm độ trễ dữ liệu và cung cấp khả năng truy cập nhanh chóng và tin cậy vào dữ liệu nguyên thô và mô hình trong Hadoop.

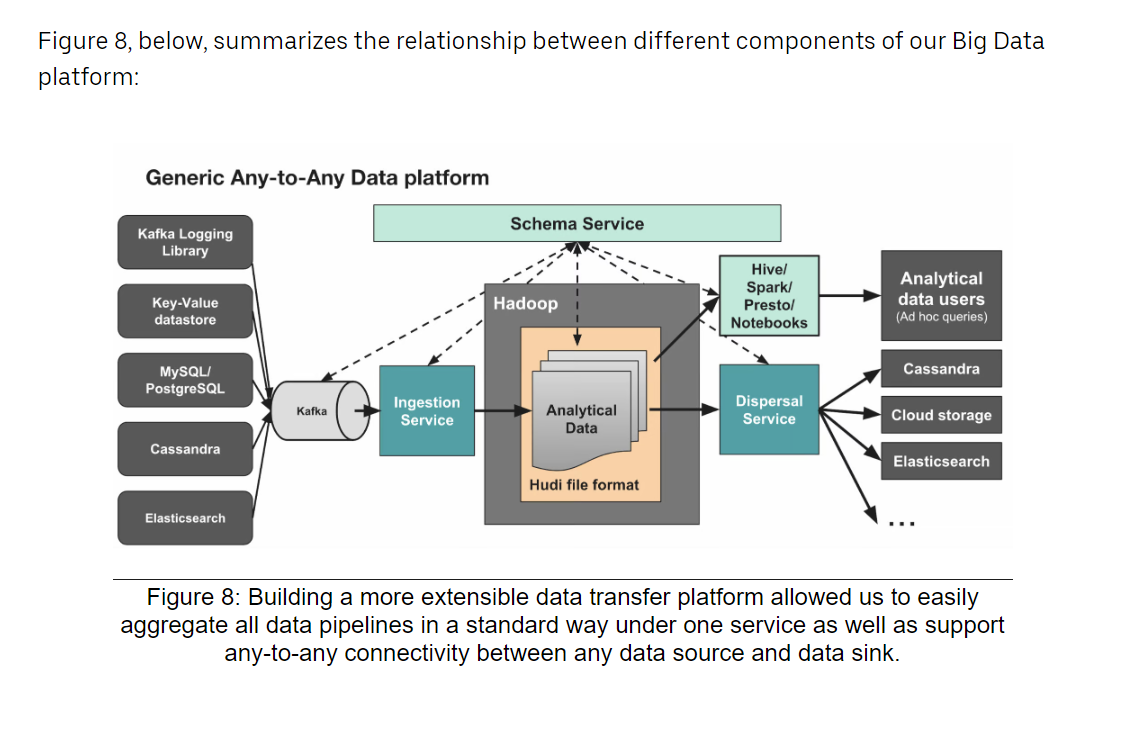
Nền tảng Big Data của Uber cũng tích hợp Marmaray để tiếp nhận thay đổi từ các kho dữ liệu nguồn và áp dụng chúng lên dữ liệu hiện tại trong Hadoop sử dụng thư viện Hudi. Điều này giúp giảm bớt tài nguyên tính toán và cải thiện độ tin cậy của dữ liệu nguyên thô. Hudi cung cấp hai chế độ xem khác nhau cho các bảng Hadoop để người dùng có thể lựa chọn phù hợp với nhu cầu của họ. Uber cũng đang tập trung vào việc cải thiện chất lượng dữ liệu, độ trễ dữ liệu và hiệu suất, cũng như tính mở rộng và đáng tin cậy của nền tảng Big Data của họ. Điều này phản ánh cam kết của công ty đối với việc ra quyết định dựa trên dữ liệu và sự phát triển của các công nghệ dữ liệu lớn.

Uber đã sử dụng Big Data một cách chi tiết và sâu rộng để cải thiện dịch vụ và tối ưu hóa trải nghiệm người dùng. Họ dự đoán nhu cầu và cung cấp dựa trên dữ liệu từ mỗi chuyến đi, điều chỉnh số lượng xe cần thiết và đáp ứng nhanh chóng với thị trường thay đổi. Đặc biệt, việc sử dụng dữ liệu để xác định giá cước động giúp cân bằng cung và cầu, đồng thời khuyến khích tài xế ra đường vào những thời điểm cần thiết. Uber cũng tối ưu hóa lộ trình bằng cách xác định các đường đi tốt nhất và phân tích giao thông đô thị để cung cấp dịch vụ ở các khu vực ít được phục vụ. Họ lưu trữ và giám sát dữ liệu từ mỗi chuyến đi để dự đoán nhu cầu và thiết lập giá cước. Với hơn 100 petabyte dữ liệu, Uber quản lý nền tảng Big Data dựa trên Apache Hadoop từ năm 2014 để đảm bảo độ tin cậy, khả năng mở rộng và dễ sử dụng của dữ liệu, và hiện đang tập trung vào việc tăng tốc độ và hiệu quả của nền tảng này.

Tổng quát, những thách thức như sự kiện kiểm soát quy phạm và tranh cãi về giá cả làm nổi bật sự phức tạp của việc thực hiện các chiến lược dựa trên dữ liệu trong các ngành công nghiệp được quy định chặt chẽ. Tuy nhiên, sự thành công của Uber thể hiện tiềm năng biến đổi của dữ liệu lớn khi được sử dụng một cách hiệu quả để đáp ứng nhu cầu của khách hàng, tối ưu hóa hoạt động và thúc đẩy đổi mới.

## 2.3. Công nghệ sử dụng

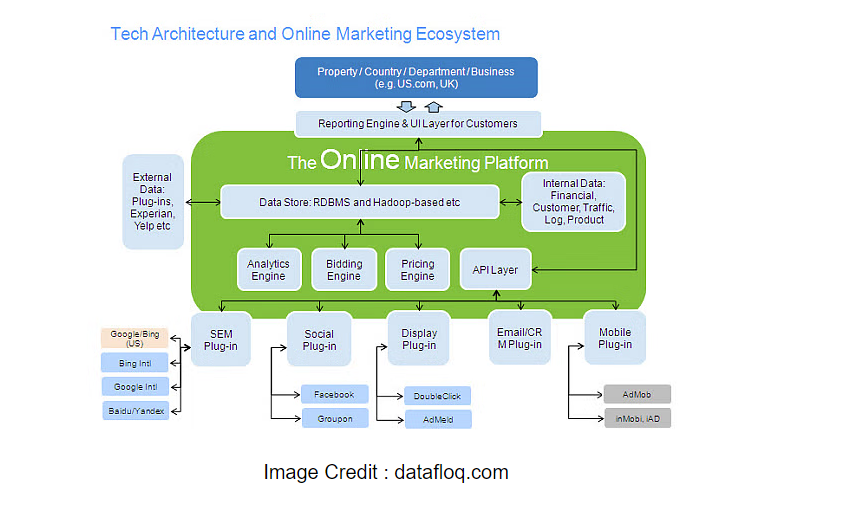
### 2.3.1. Các công nghệ được Uber sử dụng



#### Hình 2.1: Mối quan hệ trong nền tảng Big Data của Uber

* **Thư viện Ghi nhật ký Kafka:** Thành phần này có trách nhiệm thu thập nhật ký từ các dịch vụ khác nhau trong Uber. Những nhật ký này rất quan trọng cho việc giám sát, gỡ lỗi và phân tích dữ liệu.
* **Kho dữ liệu Kiểu Khóa-Giá trị:** Đây là các cơ sở dữ liệu được thiết kế để lưu trữ dữ liệu dưới dạng cặp khóa-giá trị. Loại lưu trữ này thường được sử dụng cho bộ nhớ đệm và các tình huống cần truy xuất dữ liệu nhanh chóng.
* **MySQL/PostgreSQL:** Đây là các hệ thống quản lý cơ sở dữ liệu quan hệ được biết đến với khả năng xử lý các truy vấn phức tạp và dữ liệu giao dịch. Uber sử dụng các hệ thống này cho dữ liệu có cấu trúc cần các khả năng của cơ sở dữ liệu quan hệ.
* **Cassandra:** Một cơ sở dữ liệu NoSQL mã nguồn mở nổi tiếng với khả năng xử lý lượng lớn dữ liệu trên nhiều máy chủ hàng hóa mà không có điểm lỗi đơn lẻ, đảm bảo tính sẵn có cao.
* **Elasticsearch:** Một công cụ tìm kiếm dựa trên thư viện Lucene, được Uber sử dụng cho tìm kiếm toàn văn, phân tích dữ liệu thời gian thực và các ứng dụng khác cần khả năng tìm kiếm nhanh.
* **Kafka:** Đóng vai trò như trung tâm cho các luồng dữ liệu thời gian thực. Kafka được sử dụng để xuất bản, đăng ký, lưu trữ và xử lý các dòng bản ghi theo thời gian thực, điều này rất cần thiết cho quá trình ra quyết định thời gian thực trong hoạt động của Uber.
* **Dịch vụ Tiếp nhận:** Thành phần này chịu trách nhiệm tiêu thụ dữ liệu từ Kafka và nhập nó vào hệ thống để xử lý hoặc lưu trữ thêm.
* **Hadoop:** Được sử dụng để lưu trữ và xử lý các tập dữ liệu lớn trong môi trường tính toán phân tán. Trong bối cảnh này, nó có vẻ như được kết nối với dịch vụ tiếp nhận, cho thấy nó xử lý và lưu trữ dữ liệu lớn cho mục đích phân tích.
* **Định dạng tệp Hudi:** Hudi là một khuôn khổ quản lý dữ liệu nguồn mở được sử dụng bởi Uber để đơn giản hóa việc xử lý dữ liệu gia tăng và phát triển ống dẫn dữ liệu trên Hadoop. Nó cung cấp các khả năng như quản lý lưu trữ dữ liệu và phục vụ dữ liệu gần thời gian thực, cho phép Uber xử lý cập nhật và chèn vào hệ thống một cách hiệu quả. Điều này giảm đáng kể độ trễ dữ liệu, từ 24 giờ xuống còn dưới một giờ cho dữ liệu đã được mô hình hóa, nâng cao khả năng phản hồi của các ứng dụng dựa trên dữ liệu của họ.
* **Dịch vụ Schema:** Có thể là dịch vụ quản lý tổ chức và quản trị các schema dữ liệu trong hệ sinh thái của Uber, đảm bảo dữ liệu trên các hệ thống khác nhau tương thích và đạt chuẩn.
* **Hive/Spark/Presto/Notebooks:** Các công nghệ này được sử dụng để truy vấn và phân tích dữ liệu. Hive và Spark được sử dụng cho xử lý batch, Presto dùng để truy vấn dữ liệu một cách nhanh chóng trực tiếp từ kho dữ liệu, và Notebooks dùng cho công việc khoa học dữ liệu tương tác.
* **Dịch vụ Phân tán:** Có thể là hệ thống được sử dụng để phân phối dữ liệu đến các điểm cuối hoặc dịch vụ khác nhau, dựa trên nhu cầu của các người dùng dữ liệu khác nhau.
* **Người dùng Dữ liệu Phân tích:** Đại diện cho các bên liên quan hoặc hệ thống tiêu thụ dữ liệu đã được xử lý cho mục đích phân tích, có thể bao gồm việc tạo báo cáo hoặc hiểu biết dựa trên dữ liệu lịch sử và thời gian thực.
* **Lưu trữ Đám mây:** Được sử dụng để lưu trữ dữ liệu trên các nền tảng đám mây, điều này cung cấp khả năng mở rộng, độ tin cậy và khả năng truy cập toàn cầu.
* **Cassandra:** Xuất hiện trong sơ đồ kết nối với dịch vụ phân tán, cho thấy vai trò của nó trong việc cung cấp khả năng cơ sở dữ liệu phân tán để xử lý lượng lớn dữ liệu phân bổ trên nhiều máy chủ.

### 2.3.2. Các công nghệ được Walmart sử dụng



#### Hình 2.2: Kiến trúc kỹ thuật và hệ sinh thái tiếp thị trực tuyến của Walmart

**Hadoop:** được sử dụng để lưu trữ lượng dữ liệu khổng lồ từ nhiều nguồn khác nhau như cửa hàng, bộ phận trực tuyến và các đơn vị kinh doanh. Hệ thống này giúp lưu trữ dữ liệu phân tán, giảm thiểu rủi ro mất mát dữ liệu và tăng cường tính sẵn sàng. Ngoài ra, Hadoop còn được sử dụng để xử lý và phân tích khối lượng lớn dữ liệu một cách nhanh chóng và hiệu quả, giúp tối ưu hóa quá trình ra quyết định và cải thiện hiệu suất hoạt động kinh doanh

**Cassandra:** cho phép Walmart thực hiện các hoạt động ghi và đọc dữ liệu nhanh chóng, tối ưu hóa quy trình quản lý và phân tích dữ liệu. Ngoài ra, Cassandra có khả năng chịu lỗi khá cao, đảm bảo cho dữ liệu luôn bảo mật và sẵn sàng, giúp Walmart hoạt động liên tục mà không bị gián đoạn

**Spark:** dùng để xử lý và phân tích dữ liệu trong bộ nhớ, tăng tốc độ xử lý và giảm thời gian chờ đợi, từ đó cải thiện hiệu quả hoạt động, ngoài ra Spark còn hỗ trợ xử lý dữ liệu thời gian thực, cho phép Walmart phản ứng nhanh chóng với các thay đổi trên thị trường và tối ưu hóa các quyết định kinh doanh ngay lập tức

## 2.4. Tìm hiểu về các công nghệ đã chọn

### 2.4.1. Công nghệ 1: Apache Hbase

Nằm trong hệ sinh thái của Hadoop, Apache HBase là một cơ sở dữ liệu phân tán, có khả năng mở rộng cao, được thiết kế để lưu trữ và quản lý dữ liệu lớn (big data). HBase là một dự án của Apache Software Foundation và là một phần của hệ sinh thái Hadoop. Nó được lấy cảm hứng từ Bigtable của Google, một hệ thống lưu trữ dữ liệu phân tán nổi tiếng.

*Cách thức hoạt động và ứng dụng trong Big Data.*

*- Kiến trúc cơ bản:*

* Table: Dữ liệu trong HBase được tổ chức dưới dạng bảng (table), nhưng không giống như trong các cơ sở dữ liệu quan hệ truyền thống, bảng trong HBase có thể có số lượng cột rất lớn.
* Column Families: Các cột được nhóm lại thành các column family, đây là đơn vị lưu trữ và quản lý dữ liệu cơ bản trong HBase.
* Rows: Dữ liệu được lưu trữ dưới dạng hàng (rows), mỗi hàng có một khóa dòng (row key) duy nhất.
* Cells: Giao điểm của hàng và cột gọi là cell, nơi lưu trữ dữ liệu thực tế cùng với dấu thời gian (timestamp).

*- Cách thức hoạt động:*

1. Bảng được chia thành các phần nhỏ hơn gọi là region. Mỗi region chứa một tập hợp các hàng liên tiếp.

2. Các region được quản lý bởi các region server, chúng chịu trách nhiệm xử lý các yêu cầu đọc và ghi dữ liệu. Khi một region phát triển quá lớn, nó sẽ được chia nhỏ để duy trì hiệu suất.

3. HBase Master sẽ điều phối hoạt động của cluster HBase, bao gồm việc phân chia và phân phối regions tới các region server, và quản lý metadata của bảng.

4. HDFS Integration: HBase sử dụng HDFS làm hệ thống lưu trữ cơ bản, lưu trữ các tệp dữ liệu (HFiles) và các bản ghi write-ahead log (WAL).

5. Sử dụng Apache Zookeeper để điều phối và quản lý trạng thái của các node trong cluster HBase, đảm bảo tính nhất quán và tính sẵn sàng.

*Ứng dụng của HBase trong Big Data*

Real-time Data Processing: HBase hỗ trợ các thao tác đọc và ghi dữ liệu nhanh chóng, giúp nó phù hợp với các ứng dụng yêu cầu xử lý dữ liệu theo thời gian thực, như hệ thống quản lý quảng cáo trực tuyến, hệ thống phát hiện gian lận, và phân tích dữ liệu trực tiếp.

Scalable Data Storage: Khả năng mở rộng ngang của HBase cho phép lưu trữ và quản lý lượng dữ liệu rất lớn, lên đến hàng petabyte, mà không làm giảm hiệu suất. Điều này làm cho HBase trở thành lựa chọn lý tưởng cho các hệ thống quản lý dữ liệu lớn.

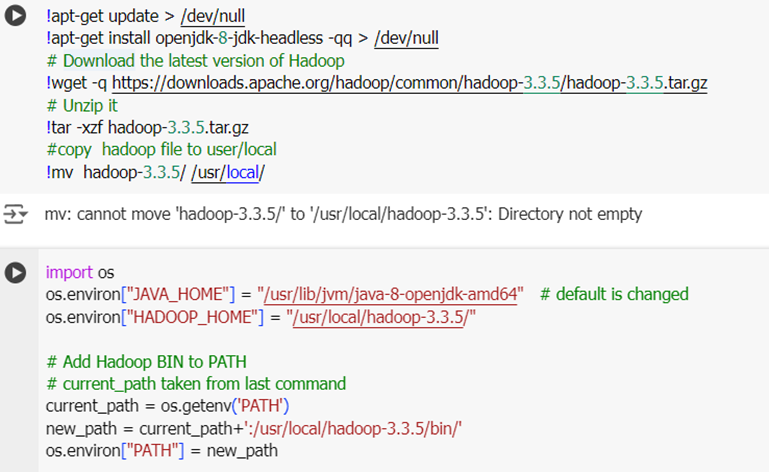
· Time-series Data: HBase là lựa chọn tốt cho các ứng dụng lưu trữ dữ liệu chuỗi thời gian (time-series data) nhờ vào khả năng quản lý dữ liệu có dấu thời gian.

· Data Warehouse: HBase có thể được sử dụng như một phần của hệ thống kho dữ liệu (data warehouse) để lưu trữ và truy vấn các tập dữ liệu lớn, hỗ trợ các quy trình ETL (Extract, Transform, Load).

· Integration với hệ sinh thái Hadoop: HBase tích hợp chặt chẽ với các công cụ khác trong hệ sinh thái Hadoop như Apache Hive, Apache Pig, và Apache Spark, hỗ trợ các phân tích dữ liệu phức tạp và xử lý dữ liệu phân tán.

*Tiến hành cài đặt và demo tạo bảng dữ liệu với Hbase:*

- Đầu tiên, nhóm cài đặt và setup môi trường cho Hadoop:

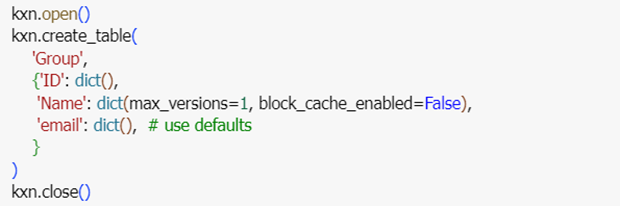


- Tiếp theo, nhóm tiếp tục cài đặt và setup môi trường cho Hbase:





- Sau khi đã cài đặt xong, nhóm bắt đầu demo tạo 1 bảng dữ liệu có tên là Group với 3 thuộc tính là ID, Name và email:



- Tiếp theo, nhóm sử dụng lệnh put để thêm các dòng dữ liệu vào bảng:



- Cuối cùng, nhóm sử dụng lệnh scan và vòng lặp để in ra kết quả dữ liệu của bảng đã tạo:



### 2.4.2. Công nghệ 2: Apache Spark

Apache Spark là một hệ thống xử lý phân tán mã nguồn mở được sử dụng cho các khối công việc dữ liệu lớn. Nó sử dụng bộ nhớ đệm trong và thực thi truy vấn tối ưu để thực hiện các truy vấn phân tích nhanh chóng trên dữ liệu bất kỳ kích thước nào. Spark cung cấp các API phát triển bằng Java, Scala, Python và R, và hỗ trợ việc tái sử dụng mã cho nhiều khối công việc khác nhau, bao gồm xử lý hàng loạt, truy vấn tương tác, phân tích thời gian thực, học máy và xử lý đồ thị.

Cách thức hoạt động và ứng dụng trong Big Data:

- Spark được tạo ra để giải quyết hạn chế của MapReduce bằng cách thực hiện xử lý trong bộ nhớ, giảm số bước trong một công việc và tái sử dụng dữ liệu qua nhiều phép toán song song. Với Spark, chỉ cần một bước để đọc dữ liệu vào bộ nhớ, thực hiện các phép toán và ghi kết quả trở lại, dẫn đến thời gian thực thi nhanh hơn đáng kể. Spark cũng tái sử dụng dữ liệu bằng cách sử dụng bộ nhớ cache để tăng tốc độ các thuật toán học máy mà liên tục gọi một hàm trên cùng một tập dữ liệu. Việc tái sử dụng dữ liệu được thực hiện thông qua việc tạo ra các DataFrame, một trừu tượng trên Resilient Distributed Dataset (RDD), là một tập hợp các đối tượng được lưu trong bộ nhớ và được sử dụng lại trong nhiều phép toán Spark.

*Ứng dụng của Apache Spark trong Big Data:*

* Batch Processing: Spark có khả năng xử lý khối lượng dữ liệu lớn với tốc độ nhanh chóng nhờ vào tối ưu hóa việc quản lý bộ nhớ và lập lịch tác vụ. Nó thích hợp cho các tác vụ ETL (Extract, Transform, Load), xử lý dữ liệu lớn và phân tích dữ liệu.
* Stream Processing: Spark Streaming cho phép xử lý dữ liệu theo thời gian thực, như xử lý luồng sự kiện từ cảm biến IoT, dữ liệu clickstream từ trang web, và các hệ thống phát hiện gian lận.
* Machine Learning: MLlib cung cấp các công cụ và thuật toán để xây dựng và triển khai các mô hình machine learning phân tán, như phân loại, hồi quy, clustering, và recommendation.
* Interactive Data Analysis: Spark SQL và DataFrames cho phép người dùng thực hiện các phân tích dữ liệu tương tác và truy vấn dữ liệu một cách dễ dàng và hiệu quả.
* Graph Processing: GraphX cho phép xử lý và phân tích các cấu trúc dữ liệu dạng đồ thị, giúp trong các ứng dụng như phân tích mạng xã hội, phân tích lộ trình, và tối ưu hóa mạng.

Tiến hành cài đặt Spark và demo xử lý, tạo Word Cloud cho 1 file dữ liệu văn bản:

- Đầu tiên, nhóm sẽ tiến hành cài đặt Spark:

|  |
| --- |
|  |



- Tiếp theo, nhóm download bộ dữ liệu từ drive, lưu nó vào biến textRDD và hiển thị dữ liệu văn bản:

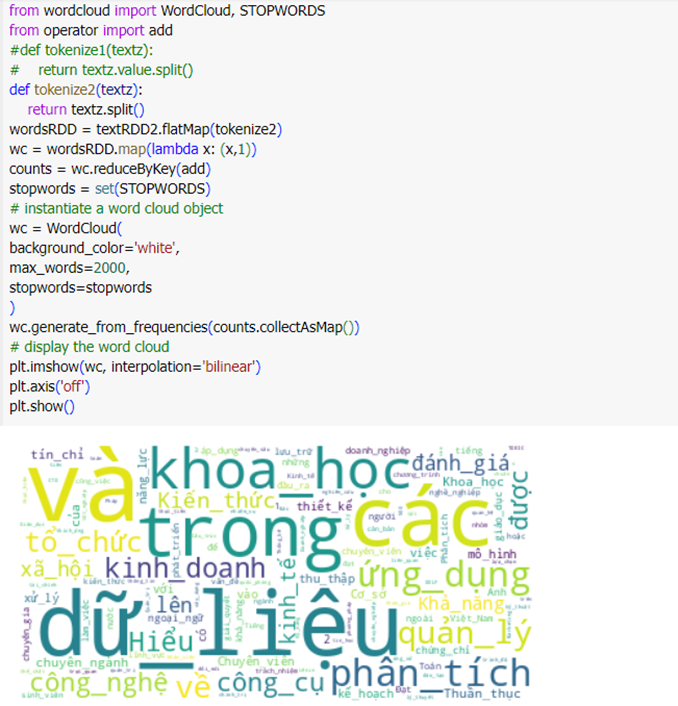




- Tiếp đến, nhóm thực hiện tiền xử lý dữ liệu văn bản và đưa nó vào 1 file mới:



- Cuối cùng, nhóm sử dụng Spark RDD để tính WordCount và hiển thị Word Cloud:





#### Hình 2.3: WordCloud

Hình WordCloud đã cho thấy kết quả tính toán rất chính xác của Spark khi đưa ra kết quả cho thấy các từ lớn như “dữ liệu,” “khoa học,” “đánh giá,” và “quản lý” xuất hiện nhiều nhất trong bài viết nói về khoa học dữ liệu. Spark đã cho thấy được sự hiệu quả và chính xác của mình khi có thể giúp tạo ra một WordCloud cung cấp cái nhìn nhanh về những chủ đề được nhắc đến nhiều nhất và có thể giúp người xem nhanh chóng nắm bắt được thông tin quan trọng.

# CHƯƠNG 3: CƠ SỞ LÝ THUYẾT

## 3.1. Dữ liệu lớn

Big Data là một thuật ngữ dùng để mô tả khối lượng dữ liệu lớn đến mức không thể xử lý hiệu quả bằng các công cụ quản lý dữ liệu truyền thống. Đây là một lĩnh vực đang phát triển nhanh chóng với khả năng phân tích và xử lý các tập dữ liệu lớn để tìm ra thông tin hữu ích, xu hướng và mẫu hình.

Lý thuyết Big Data cung cấp một bộ nguyên tắc tổng quát giải thích cơ sở, kiến thức và phương pháp được sử dụng trong thực hành khoa học dựa trên dữ liệu. Nó bao gồm các khái niệm như:

· **Dữ liệu có cấu trúc**: Dữ liệu này có thể được lưu trữ, truy cập và xử lý dưới dạng định dạng cố định. Ví dụ về dữ liệu có cấu trúc bao gồm các bảng trong cơ sở dữ liệu quan hệ.

· **Dữ liệu không có cấu trúc**: Dữ liệu này không có hình thức hoặc cấu trúc cụ thể, thường là một nguồn dữ liệu đa dạng bao gồm văn bản, hình ảnh, video, v.v.

· **Dữ liệu bán cấu trúc**: Dữ liệu này nằm giữa dữ liệu có cấu trúc và không có cấu trúc, có thể chứa cả hai loại dữ liệu.

Big Data còn được định nghĩa bởi ba “V”:

· Volume (Khối lượng): Số lượng lớn dữ liệu cần được xử lý.

· Velocity (Tốc độ): Tốc độ cao mà dữ liệu được tạo ra và cần được xử lý.

· Variety (Đa dạng): Loại hình thông tin của dữ liệu rất đa dạng.

Big Data Analytics là quá trình thu thập, kiểm tra và phân tích lượng lớn dữ liệu để khám phá xu hướng thị trường, thông tin chi tiết và mẫu hình có thể giúp các công ty đưa ra quyết định kinh doanh tốt hơn. Các doanh nghiệp như Walmart hay Uber cũng đã áp dụng thành công big data để có thể thấu hiểu được khách hàng, tối ưu hóa chiến lược kinh doanh vào đưa ra quyết định sáng suốt để dẫn đến sự thành công.

## 3.2. *PySpark*

Pyspark là thư viện Python giúp tương tác với Apache Spark, một nền tảng mã nguồn mở dùng để xử lý dữ liệu lớn song song và theo cụm. PySpark thích hợp cho những người đã quen thuộc với Python và các thư viện như Pandas để phân tích và xây dựng các tiến trình xử lý dữ liệu lớn hơn.

Apache Spark là công cụ tính toán phân bố dữ liệu trên nhiều máy chủ, còn PySpark cung cấp API cho phép Python làm việc với Spark. Dữ liệu chính được dùng trong PySpark là DataFrame, tương tự như DataFrame trong R và Pandas.

Điểm khác biệt chính giữa PySpark DataFrame và Pandas DataFrame là cách thực thi các tác vụ. PySpark sẽ trì hoãn các tác vụ cho đến khi cần thiết, giúp tránh việc phải nạp toàn bộ dữ liệu vào bộ nhớ. Trong khi đó, Pandas sẽ thực hiện các tác vụ ngay lập tức.

PySpark bao gồm nhiều thư viện con hữu ích như:

* **PySparkSQL:** cho phép phân tích dữ liệu lớn bằng các câu lệnh SQL.
* **MLlib:** cung cấp các thuật toán học máy cho phân loại, dự đoán, phân cụm, lọc cộng tác, giảm kích thước và tối ưu hóa.
* **GraphFrames:** thư viện xử lý đồ thị hiệu quả trên nền PySpark.

Mặc dù dễ dùng hơn Spark, PySpark cũng có hạn chế như khó khăn trong việc gỡ lỗi do thông báo lỗi bao gồm cả lỗi từ Java và Python. Ngoài ra, Spark còn phức tạp hơn và tốn nhiều tài nguyên hơn các lựa chọn khác như Ray và Dask.

## 3.3. Tổng quan các thư viện sử dụng

### 3.3.1. Thư viện Pandas

Thư viện Pandas trong Python là một thư viện mã nguồn mở hỗ trợ mạnh mẽ trong việc thao tác và phân tích dữ liệu. Đây là một công cụ quan trọng cho việc xử lý dữ liệu trong ngôn ngữ lập trình Python. Pandas được sử dụng rộng rãi cả trong nghiên cứu và phát triển các ứng dụng về khoa học dữ liệu nhờ khả năng linh hoạt và hiệu quả của nó.

Thư viện này sử dụng cấu trúc dữ liệu chính là DataFrame, cho phép dễ dàng quản lý và thao tác dữ liệu dạng bảng. Pandas cung cấp nhiều chức năng xử lý dữ liệu mạnh mẽ, từ lọc, nhóm, đến tính toán thống kê và vẽ biểu đồ

### 3.3.2 Thư viện Plotly

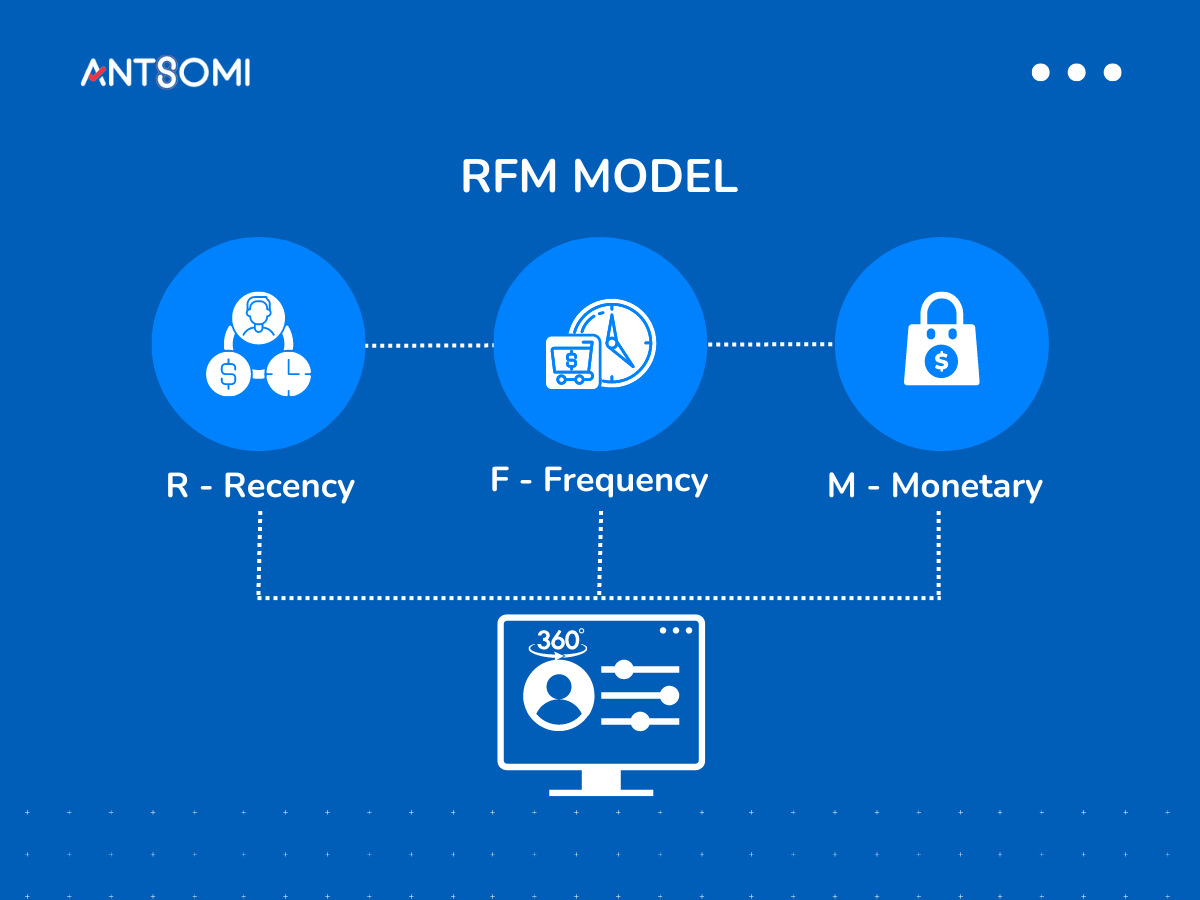
Plotly (plotly.py) là thư viện Python mã nguồn mở cho phép tạo các đồ họa tương tác trên nền tảng trình duyệt. So với các thư viện đồ hoạ khác trong Python như Matplotlib, Seaborn thì plotly nổi bật với khả năng tạo ra các biểu đồ tương tác mạnh mẽ, các đồ thị chất lượng cao

Thư viện plotly cung cấp khả năng xử lý nhanh, dễ dàng sử dụng và có khả năng kết hợp tốt với các thư viện khác như Pandas.

## 3.4. Mô hình RFM

Mô hình RFM là một phương pháp phân tích khách hàng dựa trên ba yếu tố chính:

* **R** – Recency (Thời gian mua hàng gần nhất): Khoảng thời gian kể từ lần giao dịch gần nhất của khách hàng.
* **F** – Frequency (Tần suất mua hàng): Số lần giao dịch khách hàng đã thực hiện trong khoảng thời gian nhất định.
* **M** – Monetary (Giá trị tiền cho mỗi lần mua hàng): Tổng giá trị của các giao dịch mà khách hàng đã thực hiện trong một khoảng thời gian nhất định.

****

#### Hình 3.1: Mô hình RFM

Trong đồ án này, với mỗi thuộc tính (Recency, Frequency, Monetary), nhóm sẽ đánh giá khách hàng theo thang điểm từ 1 (thấp nhất) đến 3 (tốt nhất) dựa trên các giao dịch của họ.

|  |  |  |  |
| --- | --- | --- | --- |
| **Yếu tố \ Điểm** | **1** | **2** | **3** |
| **Recency** | Mua hàng từ lâu | Mua hàng một thời gian gần đây | Vừa mới mua hàng gần đây |
| **Frequency** | Hiếm khi mua | Thỉnh thoảng mua | Mua thường xuyên |
| **Monetary** | Tiêu ít tiền cho các giao dịch | Tiêu một khoảng vừa phải cho các giao dịch | Tiêu nhiều tiền cho các giao dịch |

#### Bảng 3.1: Ý nghĩa thang điểm RFM

Dựa vào bảng trên, ta thấy khách hàng lý tưởng sẽ có mức điểm 3-3-3 cho cả ba yếu tố. Tiếp theo nhóm sẽ phân khúc khách hàng dựa trên các tập hợp nhóm điểm này:

|  |  |  |
| --- | --- | --- |
| **Phân khúc khách hàng** | **Nhóm điểm R-F-M** | **Đặc điểm** |
| Khách hàng trung thành | 333, 332, 323, 233 | Khách hàng trong nhóm này thường xuyên mua hàng (F cao), mua gần đây (R cao), và chi tiêu lớn (M cao). Họ thể hiện sự trung thành và ủng hộ với Walmart. |
| Khách hàng tiềm năng | 331, 322, 313, 232, 223, 213 | Khách hàng có tiềm năng lớn để trở thành khách hàng trung thành. Họ có thể đã mua sản phẩm gần đây hoặc mua hàng thường xuyên, nhưng chi tiêu của họ thường chưa đạt đến mức cao nhất. |
| Khách hàng thường | 321, 312, 311, 231, 222, 221, 212, 211 | Nhóm này bao gồm những khách hàng có tần suất mua hàng và số tiền chi tiêu ở mức trung bình. Họ mua hàng đều đặn nhưng không thường xuyên hoặc không chi tiêu nhiều. |
| Khách hàng đã mất | 133, 132, 131, 123, 122, 121, 113, 112, 111 | Nhóm này bao gồm những khách hàng mà Walmart đã đánh mất, có thể vì họ mất quan tâm hoặc chuyển sang các đối thủ cạnh tranh. Đặc điểm nổi bật của họ là không mua hàng trong một khoảng thời gian dài (R = 1). |

#### Bảng 3.2: Tiêu chí phân khúc khách hàng dựa trên điểm RFM

# 

# CHƯƠNG 4: TRIỂN KHAI

## 4.1. Giới thiệu về bộ dữ liệu

### 4.1.1. Mô tả bộ dữ liệu

Bộ dữ liệu “Online Retail II” bao gồm tất cả các giao dịch xảy ra từ đầu tháng 12 năm 2009 đến đầu tháng 12 năm 2011 cho một công ty bán lẻ trực tuyến không có cửa hàng trực tiếp tại Vương quốc Anh. Công ty này chủ yếu bán các món quà độc đáo phù hợp cho mọi dịp và nhiều khách hàng của công ty là các nhà bán bán buôn. Để đáp ứng theo nhu cầu của đồ án này, nhóm sẽ xem bộ dữ liệu này như một bộ dữ liệu giao dịch của Walmart.

Bộ dữ liệu được cung cấp bởi Dr. Daqing Chen, Giám đốc Khóa học: MSc Data Science tại Trường Kỹ thuật, Đại học London South Bank, London SE1 0AA, UK.

### 4.1.2. Các thuộc tính

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Thuộc tính** | **Kiểu dữ liệu** | **Ý nghĩa** |
| 1 | InvoiceNo | String | Số hoá đơn |
| 2 | StockCode | String | Mã hàng hoá |
| 3 | Description | String | Mô tả hàng hoá |
| 4 | Quantity | Integer | Số lượng hàng hóa |
| 5 | InvoiceDate | String | Ngày in hoá đơn |
| 6 | UnitPrice | Float | Giá mỗi đơn vị hàng hóa |
| 7 | CustomerID | Integer | Mã khách hàng |
| 8 | Country | String | Quốc gia |

#### Bảng 4.1: Thuộc tính bộ dữ liệu “Online Retail II”

## 4.2. Môi trường lập trình

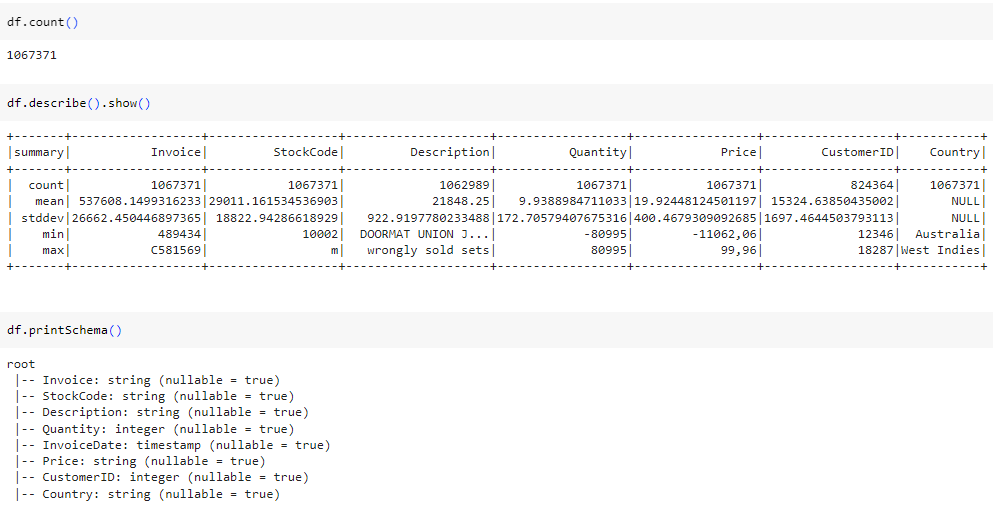
Trong đồ án này, nhóm lập trình trên môi trường Google Colab. Google Colaboratory là môi trường miễn phí do Google cung cấp, cho phép người dùng viết và chạy code Python trong trình duyệt.

Một trong những điểm mạnh của Colab là nó cung cấp các tài nguyên phần cứng mạnh mẽ như GPU và TPU, hỗ trợ rất tốt cho các tác vụ tính toán nặng nề. Thêm vào đó, Colab tích hợp liền mạch với Google Drive, giúp ta dễ dàng lưu trữ và chia sẻ dự án của mình. Nền tảng này cũng hỗ trợ nhiều thư viện Python phổ biến và cung cấp một môi trường lập trình tương tác, thích hợp cho việc viết code, kiểm thử và trực quan hóa dữ liệu.

## 4.3. Tiền xử lý

### 4.3.1. Mô tả dữ liệu

Đưa ra mô tả tổng quan về dữ liệu, bao gồm thông tin về số lượng bản ghi (rows), số lượng cột (columns), kiểu dữ liệu của từng cột, và sự phân bố của các giá trị trong các cột đó. Ảnh dưới cho thấy dữ liệu có 1,067,371 dòng và 8 cột.



*Mô tả dữ liệu và kiểu dữ liệu của các cột.*



*5 dòng đầu của bộ dữ liệu*

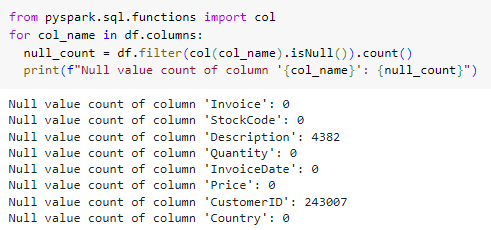
### 4.3.2. Biến đổi cột *Price*

Để thống nhất ký tự trong số nguyên và số thập phân, nhóm sử dụng dấu “.” làm ký tự ngăn cách phần thập phân và phần nguyên; sử dụng dấu “,” để ngăn cách hàng nghìn. Cụ thể, nhóm chuyển từ dấu “,” sang dấu “.” trong cột *Price*.



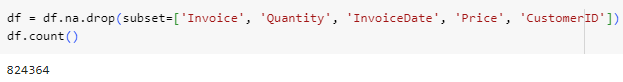
### 4.3.3. Kiểm tra các giá trị thiếu (missing values)

Xác định các cột có giá trị thiếu và phân tích tỷ lệ các giá trị bị thiếu để quyết định phương pháp xử lý (loại bỏ, thay thế, hoặc giữ nguyên). Có thể thấy 2 cột tồn tại null là cột *Description* và *CustomerID*, tuy nhiên cột *Description* cho phép những giá trị thiếu.



*Kiểm tra giá trị thiếu ở các cột.*

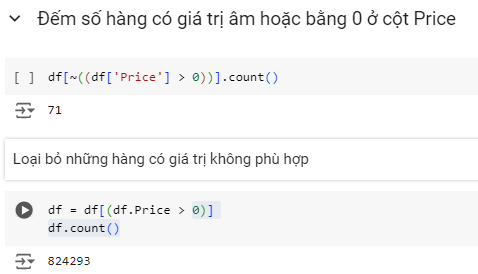
Vì thế nhóm không thực hiện drop những hàng có giá trị này, mà chỉ tiến hành drop những hàng có null ở những cột còn lại. Sau khi drop, DataFrame còn 824,364 dòng.



*Fill các cột còn lại, sau đó kiểm tra còn nhiêu dòng trong DataFrame.*

### 4.3.4. Kiểm tra cột *Price*

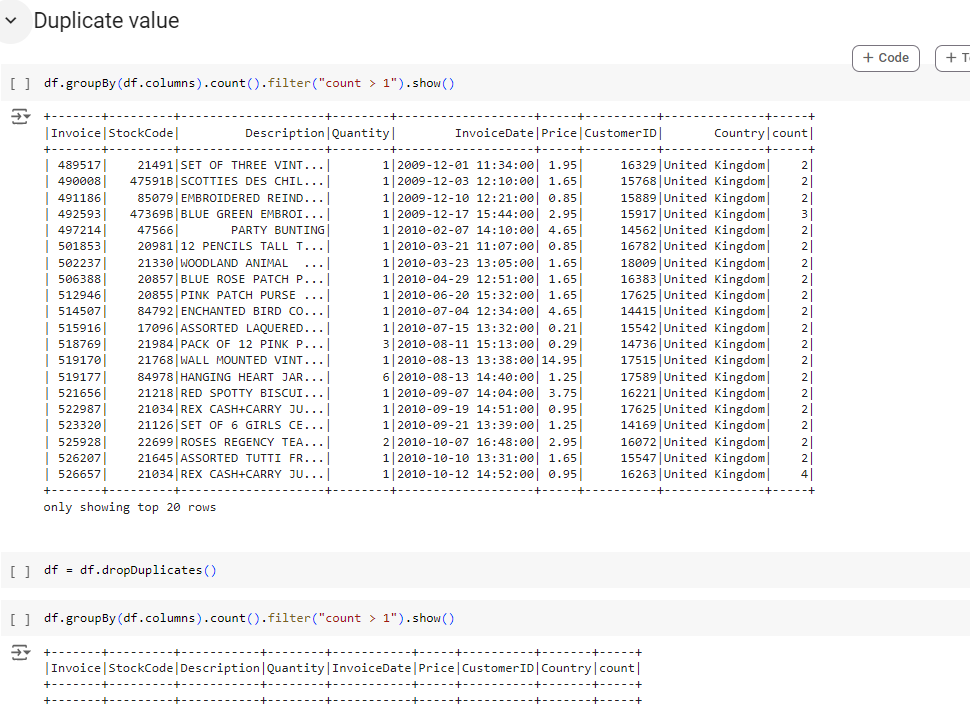
Cột *Price* có những giá trị âm, ta cần kiểm tra những giá trị này. Bộ dữ liệu có 71 giá trị âm ở cột *Price*, ta tiến hành loại bỏ những giá trị này khỏi DataFrame.



*Kiểm tra những giá trị không phù hợp ở Price, sau đó loại bỏ chúng.*

### 4.3.5. Kiểm tra giá trị trùng lặp.

Ngoài kiểm tra giá trị trùng lặp, nhóm dùng dropDuplicates() của thư viện Pandas để loại bỏ những giá trị không phù hợp này.



*Kiểm tra sau đó loại bỏ giá trị trùng lặp*

### 4.3.6. Loại bỏ những hóa đơn bị hủy

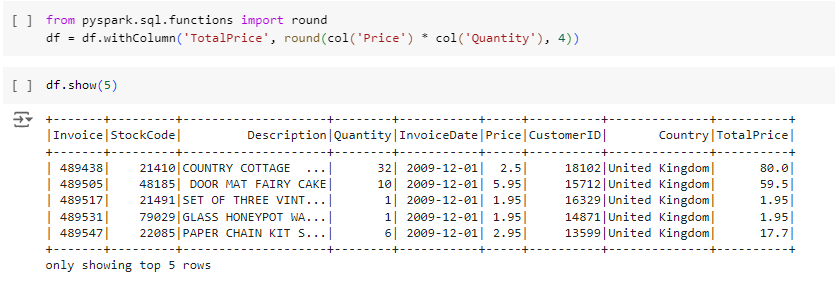
Trong những hóa đơn này, có những hóa đơn mà khách hàng đã hủy. Ta cần loại bỏ những đơn hàng này trong DataFrame để tiện hơn trong việc phân tích.



*Dùng filter để loại bỏ những Invoice có giá trị “C”*

### 4.3.7. Thêm thuộc tính *TotalPrice*

Ngoài ra, DataFrame cần một thuộc tính có thể tính tổng giá mỗi đơn giá của mỗi hàng hóa. Ta thêm vào đây một cột có tên là TotalPrice, bằng cách lấy tích của cột *Price* và cột *Quantity.*

**

*DataFrame sau khi thêm cột TotalPrice.*

### 4.3.8 Tổng hợp script xử lý tiền xử lý dữ liệu

- Thực hiện tổng hợp các bước tiền xử lý trên thành 1 hàm, với nhiệm vụ làm sạch dữ liệu.



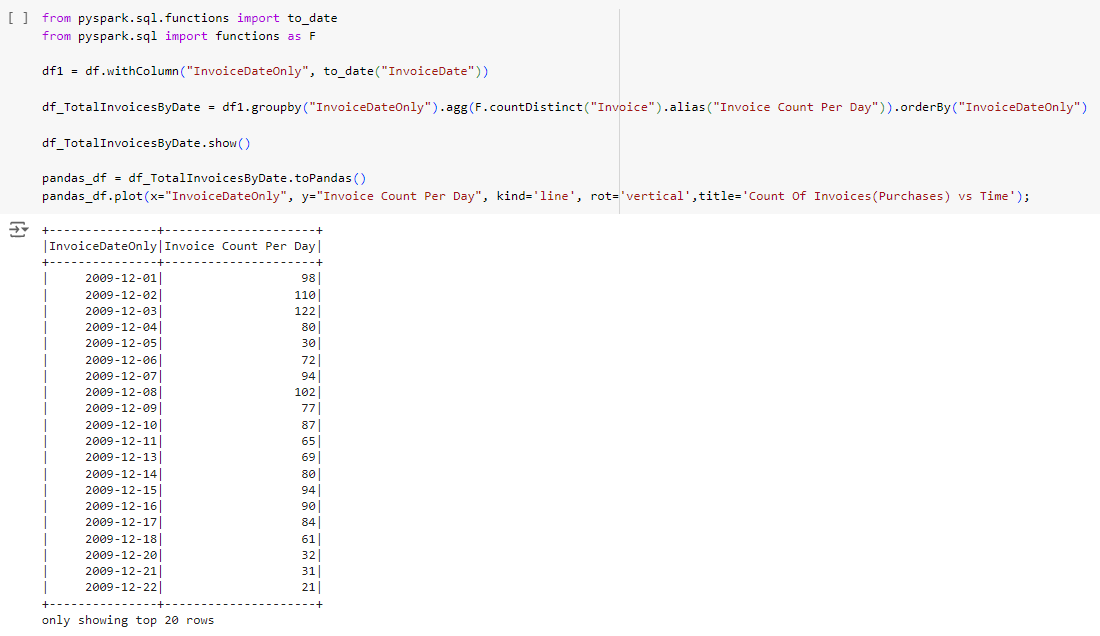
- Thêm vào cuối hàm trên dòng lệnh lưu trữ file dữ liệu sau khi được làm sạch bằng phương thức repartition(1).write.csv, với tham số 1 sẽ hợp nhất tất cả các phân chia thành một phần duy nhất. Bước này nhằm mục đích lưu bộ dữ liệu sau khi xử lý, thuận tiện cho các tác vụ phân tích về sau.

## 4.4. Một số truy vấn cơ bản

Trước khi tiến hành truy vấn, nhóm tạo một Bảng Spark có tên df bằng DataFrame đã qua tiền xử lý.



### 4.4.1. Tổng số hóa đơn theo ngày

****

*Truy vấn tổng số hóa đơn theo ngày bằng SparkSQL*

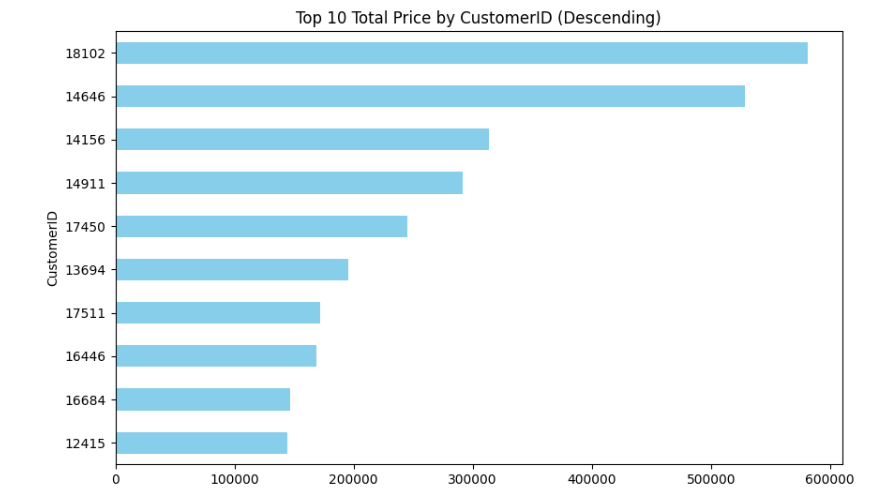
****

#### Biểu đồ 4.1: Tổng số hóa đơn theo ngày

### 4.4.2. Top 10 khách hàng chi tiêu nhiều nhất

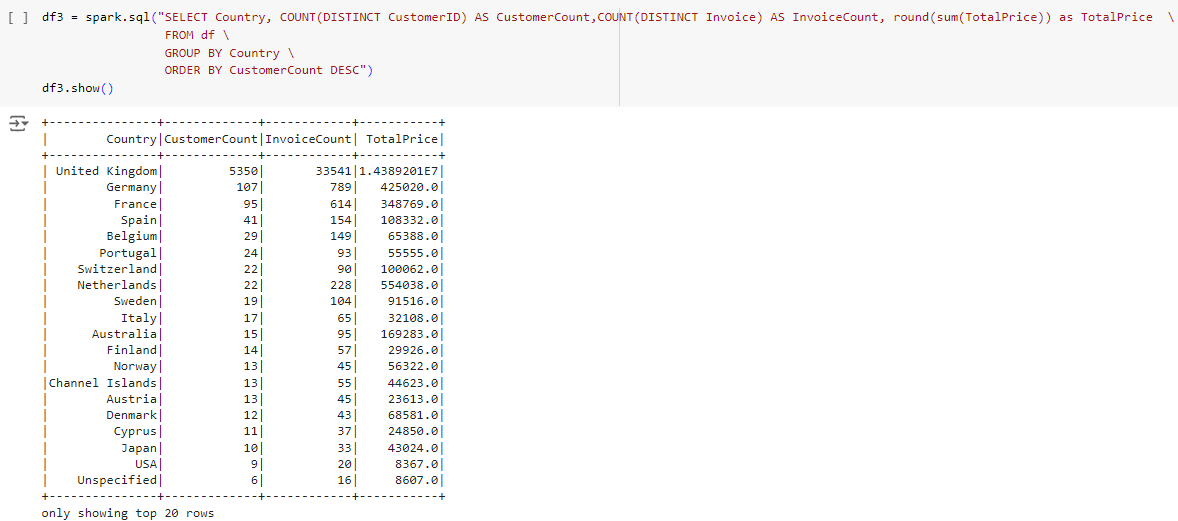
****

*Truy vấn những khách hàng chi tiêu nhiều nhất bằng SparkSQL*

****

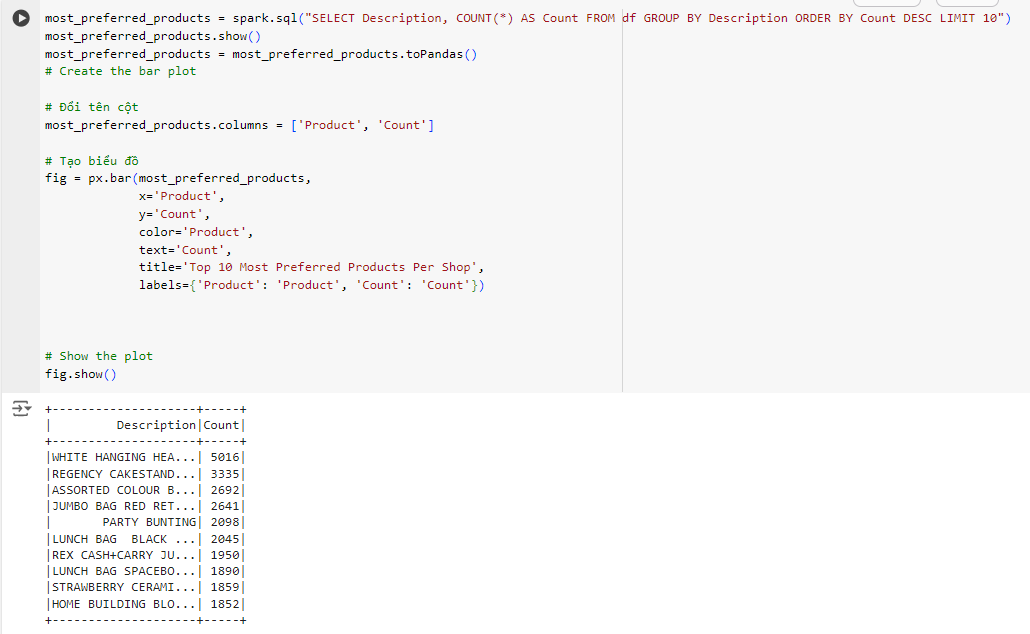
#### Biểu đồ 4.2: Top 10 khách hàng chi tiêu nhiều nhất

### 4.4.3. Những số liệu trên từng quốc gia (Số lượng khách hàng, số hóa đơn, tổng tiền chi tiêu)



*Truy vấn những số liệu trên từng quốc gia, sắp xếp theo số khách hàng từ cao đến thấp bằng SparkSQL.*

### 4.4.4. Những sản phẩm được mua nhiều nhất.



*Truy vấn Top 10 sản phẩm được mua nhiều nhất.*

|  |
| --- |
|  |

#### Biểu đồ 4.3: Top 10 sản phẩm được mua nhiều nhất

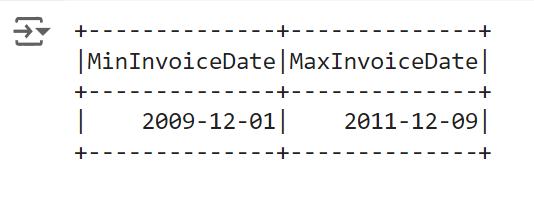
# CHƯƠNG 5: PHÂN KHÚC KHÁCH HÀNG

## 5.1. Thực hiện phân khúc

### 5.1.1. Tính điểm Recency

Để thực hiện tính điểm Recency cho khách hàng, trước tiên ta cần phải biết ngày đầu tiên (min) và ngày cuối cùng (max) trong bộ dữ liệu.

|  |
| --- |
| from pyspark.sql.functions import min, max  spark = SparkSession.builder.appName("InvoiceDateMinMax").getOrCreate()  *# Tìm ngày min và max trong bộ dữ liệu* result = df.agg(min("InvoiceDate").alias("MinInvoiceDate"), max("InvoiceDate").alias("MaxInvoiceDate")) result.show() |

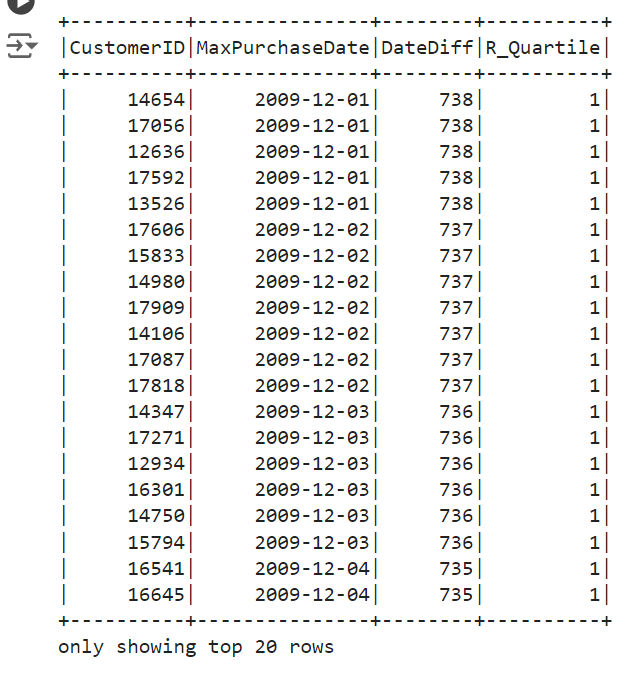
Output: ****

Vì ngày cuối cùng là 12/09/2011, ta sẽ lấy đây là ngày làm mốc để tính giá trị DateDiff. Ví dụ một khách hàng có hóa đơn tại Walmart vào ngày 11/09/2011 thì DateDiff của họ sẽ bằng 1, tức là họ đã mua hàng 1 ngày trước.

Tiếp theo, nhóm sẽ áp dụng hàm **ntile**, một hàm quan trọng vì nó được sử dụng để phân chia dữ liệu thành các nhóm có cùng số lượng phần tử, và cũng được sử dụng để tính toán điểm Frequency và Monetary.

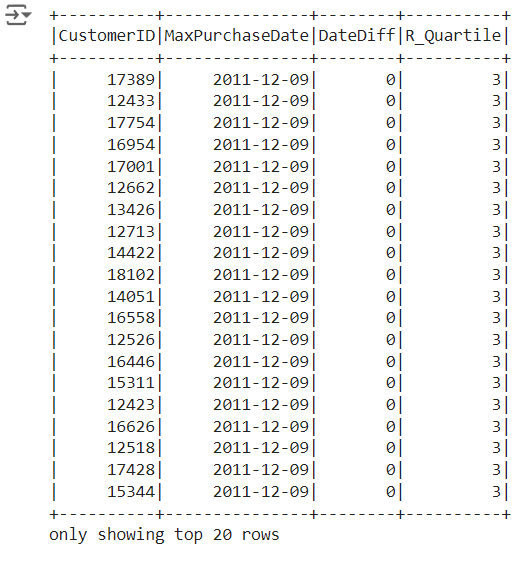
Hàm ntile sẽ chia dữ liệu thành các nhóm tương đương về kích thước, dựa trên số lượng quantile được chỉ định. Ví dụ trong đồ án này, nhóm sử dụng ntile(3).over, nó sẽ chia dữ liệu thành 3 nhóm có số lượng phần tử gần như bằng nhau, tức là chia thành 3 phân vị. Điều này giúp nhóm tính điểm theo các phân đoạn có cùng tỉ lệ phân bố.

|  |
| --- |
| from pyspark.sql import SparkSession from pyspark.sql.functions import max as spark\_max, datediff, lit from pyspark.sql.window import Window from pyspark.sql.functions import ntile  spark = SparkSession.builder.appName("RFMRecency").getOrCreate()  *# Tính lần mua gần nhất của mỗi khách hàng* most\_recent\_purchase = df.groupBy("CustomerID").agg(spark\_max("InvoiceDate").alias("MaxPurchaseDate"))  *# Tính DateDiff giữa ngày đó và 12/09/2011 (ngày cuối cùng trong bộ dữ liệu)* reference\_date = "2011-12-09" recency\_df = most\_recent\_purchase.withColumn("DateDiff", datediff(lit(reference\_date), "MaxPurchaseDate"))  *# Tính điểm R cho mỗi khách hàng dựa vào phân vị* window\_spec = Window.orderBy(recency\_df["DateDiff"].desc()) recency\_df = recency\_df.withColumn("R\_Quartile", ntile(3).over(window\_spec))  recency\_df.show() |

Output: ****

Ta thấy những khách hàng mà đã mua từ rất lâu (hơn 700 ngày trước) đều được cho điểm R là 1. Để xác thực rằng ta đã chấm điểm đúng, ta tiến hành kiểm tra xem những khách hàng có hóa đơn gần đây.

|  |
| --- |
| recency\_df.orderBy("DateDiff", ascending=True).show() |

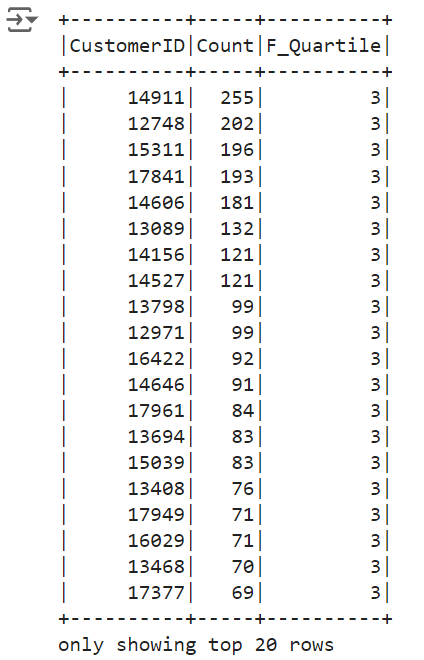
Output: ****

Ta thấy tất cả các khách vừa mua hàng ngày hôm ấy đều được cho điểm 3, từ đó có thể xác nhận các khách hàng đã được chấm điểm đúng.

### 5.1.2. Tính điểm Frequency

Trước tiên ta sẽ tính số lần khách mua hàng dựa vào số lượng hóa đơn, sau đó áp dụng hàm ntile.

|  |
| --- |
| *# Đếm số lần khách hàng mua hàng bằng cách đếm số lượng hóa đơn* frequency\_df = df.groupBy("CustomerID").agg(countDistinct("InvoiceDate").alias("Count"))  *# Tính điểm F cho mỗi khách hàng dựa vào phân vị* window\_spec = Window.orderBy(frequency\_df["Count"].asc()) frequency\_df = frequency\_df.withColumn("F\_Quartile", ntile(3).over(window\_spec))  frequency\_df.orderBy("Count", ascending=False).show() |

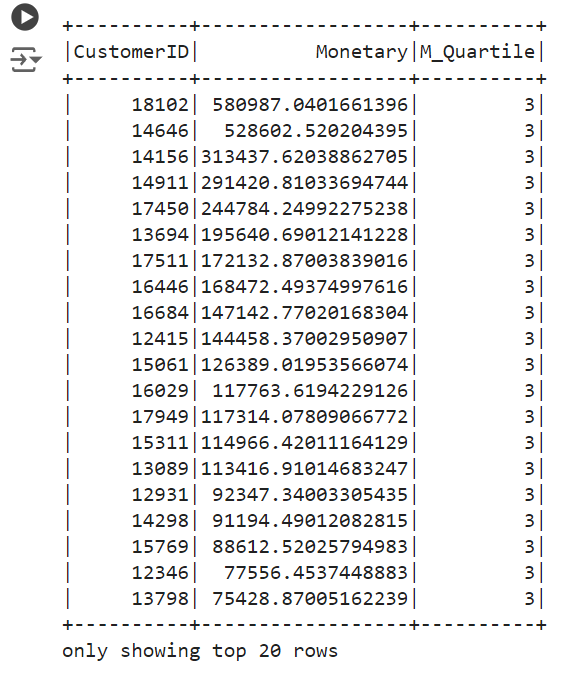
Output: 

Khách hàng mang ID “14911” là người mua hàng thường xuyên nhất với 255 hóa đơn, và tất nhiên họ đã được cho điểm 3 cho yếu tố F.

### 5.1.3. Tính điểm Monetary

Trước tiên ta sẽ tính tổng giá trị tất cả các hóa đơn của khách hàng bằng việc cộng cột “TotalPrice”. Sau đó áp dụng hàm ntile như thường.

|  |
| --- |
| from pyspark.sql.functions import sum as spark\_sum  *# Tính tổng số tiền mỗi khách hàng đã trả* monetary\_df = df.groupBy("CustomerID").agg(spark\_sum("TotalPrice").alias("Monetary"))  *# Tính điểm M cho mỗi khách hàng dựa vào phân vị* window\_spec = Window.orderBy(monetary\_df["Monetary"].asc()) monetary\_df = monetary\_df.withColumn("M\_Quartile", ntile(3).over(window\_spec))  monetary\_df.orderBy("Monetary", ascending=False).show() |

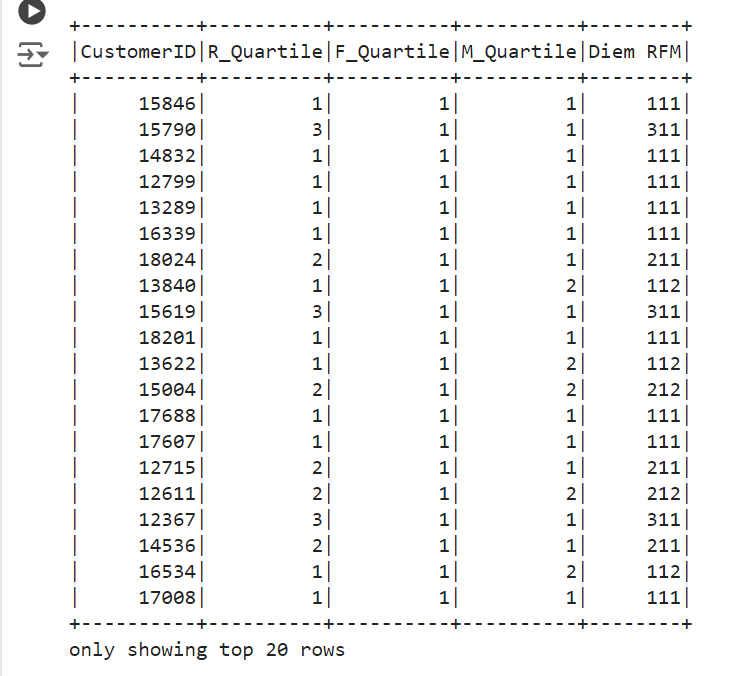
Output: 

### 5.1.4. Phân khúc khách hàng

Ta sẽ join 3 bảng vừa tạo để tính điểm: recency\_df; frequency\_df; monetary\_df dựa trên “CustomerID”.

Tạo một table mới là combined\_df với cột “Diem RFM” để gộp 3 yếu tố này thành một cột theo thứ tự R-F-M.

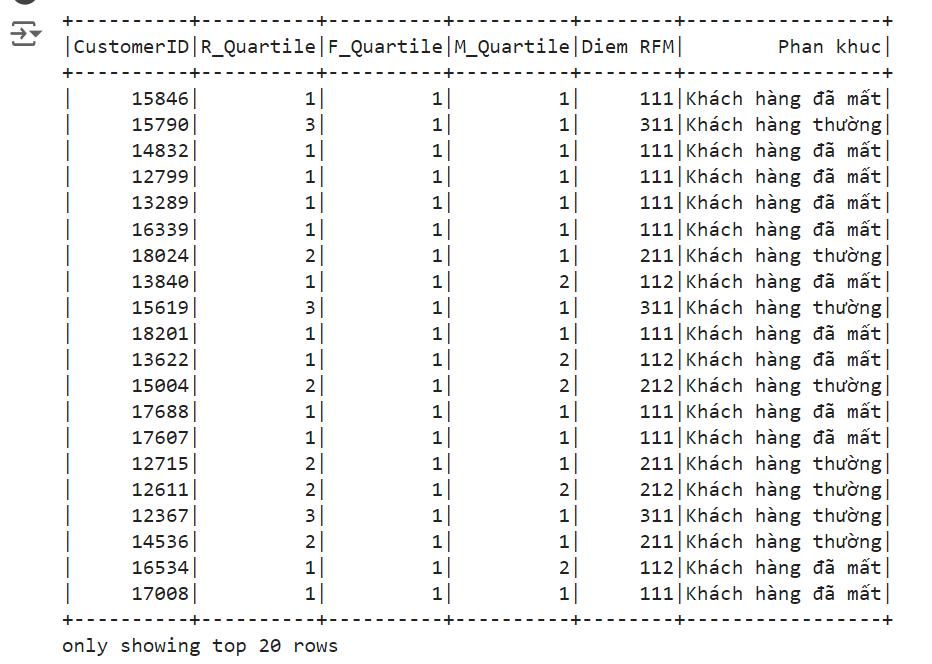
|  |
| --- |
| *from pyspark.sql.functions import concat\_ws  # Kết hợp 3 bảng lại combined\_df = recency\_df.join(frequency\_df, "CustomerID", "inner") \  .join(monetary\_df, "CustomerID", "inner") \   combined\_df = combined\_df.select("CustomerID", "R\_Quartile", "F\_Quartile", "M\_Quartile") combined\_df = combined\_df.withColumn("Diem RFM", concat\_ws("", combined\_df["R\_Quartile"], combined\_df["F\_Quartile"], combined\_df["M\_Quartile"]))  combined\_df.show()* |

Output: 

Bước tiếp theo, ta dựa vào bảng Tiêu chí trong chương 3 để phân khúc khách hàng, tạo một cột mới là “Phan khuc”.

|  |
| --- |
| from pyspark.sql.functions import when  *# Dựa vào bảng Tiêu chí phân khúc khách hàng dựa trên điểm RFM* tt = ["333", "332", "323", "322", "233", "232", "223"] tn = ["331", "321", "313", "312", "222", "213"] t = ["311", "231", "221", "212", "211"] m = ["133", "132", "131", "123", "122", "121", "113", "112", "111"]  *# Tạo một cột mới để phân khúc khách hàng* combined\_df = combined\_df.withColumn("Phan khuc",  when(combined\_df["Diem RFM"].isin(tt), "Khách hàng trung thành")  .when(combined\_df["Diem RFM"].isin(tn), "Khách hàng tiềm năng")  .when(combined\_df["Diem RFM"].isin(t), "Khách hàng thường")  .when(combined\_df["Diem RFM"].isin(m), "Khách hàng đã mất")  .otherwise(None))  combined\_df.show() |

Output:

****

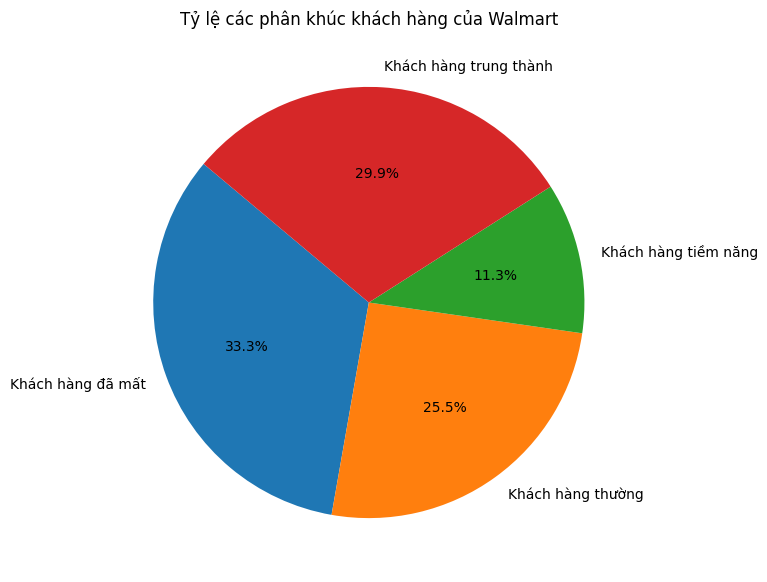
Từ đây nhóm đã phân khúc khách hàng theo cụm một cách thành công.

## 5.2. Biểu diễn trực quan

Sau khi từng khách hàng đã được phân loại vào một cụm nào đó, ta tiến hành đếm và tính tỷ lệ của các cụm khách hàng, rồi biểu diễn bằng thư viện pyplot.

|  |
| --- |
| import matplotlib.pyplot as plt  *# Đếm số lượng của từng phân khúc khách hàng* phan\_khuc = combined\_df.groupBy("Phan khuc").count()  *# Quy về dataframe pandas và vẽ biểu đồ* phan\_khuc\_pd = phan\_khuc.toPandas()  plt.figure(figsize=(10, 7)) plt.pie(phan\_khuc\_pd['count'], labels=phan\_khuc\_pd['Phan khuc'], autopct='%1.1f%%', startangle=140) plt.title('Tỷ lệ các phân khúc khách hàng của Walmart') plt.show() |

Kết quả cuối cùng ta thu được biểu đồ sau đây:

****

#### Biểu đồ 5.1: Tỷ lệ các phân khúc khách hàng của Walmart

# CHƯƠNG 6: KẾT LUẬN VÀ ĐỀ XUẤT

## 6.1. Kết quả đạt được

Qua quá trình thực hiện đồ án này, nhóm đã đạt được các kết quả như sau:

* Cung cấp tổng quan kiến thức về Big Data, cách Big Data đang được áp dụng vào các doanh nghiệp, cụ thể ở đây là Walmart và Uber. Ngoài ra, nhóm cũng tìm hiểu thêm được các công nghệ mà 2 doanh nghiệp trên sử dụng để khai thác lượng dữ liệu vô cùng lớn và dồi dào này.
* Nhóm đã tiến hành demo một số ví dụ cơ bản như tạo table, gắn giá trị vào table, tạo mask, ... dựa trên 2 công nghệ Apache HBase và Apache Spark.
* Nhóm đã tiến hành phân tích bộ dữ liệu giao dịch “Online Retail II” để phân khúc khách hàng dựa trên mô hình RFM thành 4 cụm: khách hàng trung thành, khách hàng tiềm năng, khách hàng thường và khách hàng đã mất.
  + Để thực hiện thành công việc phân khúc, nhóm đã phải kết hợp giữa kiến thức kinh doanh thực tế (mô hình RFM, tính điểm yếu tố, gán nhãn) và kiến thức kỹ thuật về phân tích dữ liệu nói chung, dữ liệu lớn nói riêng (sử dụng PySpark, truy vấn SQL, sử dụng hàm tích hợp, áp dụng phân vị, biểu diễn trực quan dữ liệu).

## 6.2. Đề xuất

### 6.2.1. Hạn chế còn tồn động

Bài đồ án của nhóm chắc chắn không thể tránh khỏi những sai sót. Và trong quá trình thực hiện, nhóm chúng em đã tự nhận diện được một số hạn chế và khó khăn như sau:

* Việc crawl data trực tiếp sẽ khó, một phần là vì bản chất của dữ liệu giao dịch: rất nhạy cảm. Các giao dịch sẽ chứa những thông tin cá nhân riêng tư của khách hàng, họ mua gì, ở đâu, khi nào, … Do đó, việc cào trực tiếp từ các hệ thống của Walmart là không khả thi do tính bảo mật cao của loại dữ liệu giao dịch. Bộ dữ liệu “Online Retail II” cũng không thể mô phỏng chính xác như Walmart, từ mô hình, kích cỡ, thuộc tính, tới các giá trị.
* Nhóm chưa triển khai các mô hình học máy (machine learning) để phân tích, dự báo dữ liệu, hay thậm chí sử dụng học máy không giám sát để phân cụm khách hàng.
* Như trên, mặc dù mô hình RFM rất hữu ích trong việc phân khúc khách hàng, nhưng nó vẫn có những giới hạn nhất định như chỉ xem xét 3 yếu tố để tính điểm, phân cụm theo một tiêu chuẩn cứng nhắc, nên cũng chưa thể cung cấp một cái nhìn toàn diện về tất cả các yếu tố ảnh hưởng đến hành vi của khách hàng.
* Nhóm chỉ mới sử dụng PySpark và một số công cụ cơ bản để phân tích dữ liệu, chưa áp dụng các công nghệ Big Data khác như Hadoop, Kafka hoặc các hệ thống xử lý dữ liệu thời gian thực.

### 6.2.2. Hướng phát triển tiếp theo

Sau khi đã có một cái nhìn tổng quan về tỷ lệ của các cụm khách hàng, nhóm có thể:

* Đưa ra những chiến lược kinh doanh mới dựa trên phân khúc khách hàng chẳng hạn như tạo các chương trình khách hàng thân thiết, cung cấp các ưu đãi đặc biệt để duy trì và tăng cường mối quan hệ, hay áp dụng các chiến dịch giành lại khách hàng đã mất như cung cấp các ưu đãi hấp dẫn để thu hút họ quay lại.
* Triển khai các mô hình học máy như hồi quy tuyến tính hay cây quyết định để dự báo tổng doanh thu trong những năm tiếp theo (2012, 2013, …).
* Crawl data từ các nguồn khác thay vì chỉ dựa vào dữ liệu có sẵn, nhóm thu thập dữ liệu từ các nguồn công khai khác như mạng xã hội, trang web đánh giá sản phẩm, và các báo cáo thị trường để có cái nhìn toàn diện hơn về hành vi và xu hướng của khách hàng.
* Sử dụng các công nghệ Big Data khác như Apache Kafka / Spark Streaming để xử lý và phân tích dữ liệu thời gian thực.

# 

# TÀI LIỆU THAM KHẢO

1. Dm-Teamseo. (2024). *Inside Walmart’s Decision Hub: A Data-Infused Success Story*. AtliQ Technologies. <https://www.atliq.com/inside-walmarts-decision-hub-a-data-infused-success-story/>
2. Marr, B. (2021). *Walmart: Big Data analytics at the world’s biggest retailer*. Bernard Marr. <https://bernardmarr.com/walmart-big-data-analytics-at-the-worlds-biggest-retailer/>
3. Goswami, C. (2024). *Success Story of Uber - How It Scaled Heights in Mobility Sector?* StartupTalky. <https://startuptalky.com/uber-story/>
4. Uber Blog (2018). *Uber’s Big Data Platform: 100+ Petabytes with Minute Latency | Uber Blog*. <https://www.uber.com/en-VN/blog/uber-big-data-platform/>
5. Stats in Mind (2022). *The Amazing Ways Uber Is Using Big Data to Improve User Experience*. <https://statsinmind.my/uber-big-data-analytics/>
6. Amazon Web Services, Inc (2024). *What is Spark? - Introduction to Apache Spark and Analytics - AWS*.. <https://aws.amazon.com/what-is/apache-spark/>
7. Tiao, S. (2024). *What is Big Data?* <https://www.oracle.com/big-data/what-is-big-data/>
8. Swan, M. (2018). *Big Data Theory*. Springer eBooks. <https://doi.org/10.1007/978-3-319-32001-4_508-1>
9. Bernard Marr. (2021). *Walmart: Big Data analytics at the world’s biggest retailer*

[*https://bernardmarr.com/walmart-big-data-analytics-at-the-worlds-biggest-retailer/*](https://bernardmarr.com/walmart-big-data-analytics-at-the-worlds-biggest-retailer/)

1. Optimove (2023). *RFM Segmentation, Analysis & Model Marketing | Optimove.* [*https://www.optimove.com/resources/learning-center/rfm-segmentation*](https://www.optimove.com/resources/learning-center/rfm-segmentation)

# PHỤ LỤC

## Bảng phân công

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Họ tên** | **Công việc được giao** | **Mức độ hoàn thành** |
| **1** | Thái Bảo An | Chương 1.2: Giới thiệu về doanh nghiệp  Chương 2.1: cách Uber áp dụng Big Data  Chương 2.2: cách Walmart áp dụng Big Data Chương 2.4. Tìm hiểu về các công nghệ đã chọn Chương 3.1: Dữ liệu lớn  Chương 3.2: PySpark | 100% |
| **2** | Hà Gia Hiến | Chương 3.4: Mô hình RFM  Chương 5: Phân khúc khách hàng  Chương 6: Kết luận  Lập trình phân khúc | 100% |
| **3** | Đặng Nhật Huy | Chương 4.1: Giới thiệu về bộ dữ liệu  Chương 4.3: Tiền xử lý  Lập trình tiền xử lý | 100% |
| **4** | Nguyễn Tân Niên | Lập trình truy vấn và phân tích  Chương 4.2: Giới thiệu môi trường  Chương 4.4: Một số truy vấn cơ bản  Format bài | 100% |
| **5** | Nguyễn Hoàng Minh Trí | Chương 1.1: Giới thiệu đề tài  Chương 2.3: Công nghệ sử dụng  Chương 3.3: Tổng quan thư viện sử dụng | 100% |

## Các đường link

* Source code demo chương 3:[**Demo Big Data.ipynb**](https://colab.research.google.com/drive/1a9bFKYy-SmnXzSTYB7anGSsarq4jyzpl?usp=sharing)
* Source code triển khai chương 4.3: [**Preprocess**](https://colab.research.google.com/drive/1w9Dl5nGrOItm06JNzvn330rocKijIzJy)
* Source code triển khai chương 4.4 và chương 5: [**Analysis**](https://colab.research.google.com/drive/1t4qTsOpvFpvvGWM1EeQFXh9gfYJDhifD)
* Bộ dữ liệu: [**Online Retail II**](https://archive.ics.uci.edu/dataset/502/online+retail+ii)
* Slides: [**Big Data slides**](https://www.canva.com/design/DAGGJuT-aUw/1eGE17ut-lKVmWWoxId3uw/edit?utm_content=DAGGJuT-aUw&utm_campaign=designshare&utm_medium=link2&utm_source=sharebutton)