**Trường đại học Khoa Học Tự Nhiên - ĐHQGHN**

**Khoa Toán - Cơ - Tin học**



**BÁO CÁO BÀI TẬP CUỐI KỲ**

**MÔN: LẬP TRÌNH NÂNG CAO**

**Đề tài:**

**“Tìm hiểu về Apache Spark Framework và ứng dụng trong xử lý dữ liệu.”**

**Giảng viên: Lê Trọng Vĩnh.**

**Sinh viên thực hiện:**

**Phạm Minh Đức 20001909**

**Lương Đình Đức 20001907**

**Đào Quốc Hội 20001924**

Hà Nội, ngày 6 tháng 12 năm 2022

**PHÂN CHIA CÔNG VIỆC:**

1. Phạm Minh Đức: Làm ứng dụng Twitter

2. Lương Đình Đức: Làm ứng dụng Twitter

3. Đào Quốc Hội: Làm báo cáo và chuẩn bị power point

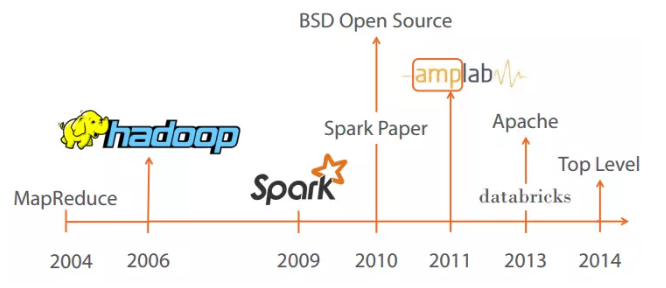
**Phần I: Apache Spark**

Ngày nay có rất nhiều hệ thống đang sử dụng Hadoop để phân tích và xử lý dữ liệu lớn. Ưu điểm lớn nhất của Hadoop là được dựa trên một mô hình lập trình song song với xử lý dữ liệu lớn là MapReduce, mô hình này cho phép khả năng tính toán có thể mở rộng, linh hoạt, khả năng chịu lỗi, chi phí rẻ. Điều này cho phép tăng tốc thời gian xử lý các dữ liệu lớn nhằm duy trì tốc độ, giảm thời gian chờ đợi khi dữ liệu ngày càng lớn.

Dù có rất nhiều điểm mạnh về khả năng tính toán song song và khả năng chịu lỗi cao nhưng Apache Hadoop có một nhược điểm là tất cả các thao tác đều phải thực hiện trên ổ đĩa cứng điều này đã làm giảm tốc độ tính toán đi gấp nhiều lần.

Để khắc phục được nhược điểm này thì Apache Spark được ra đời. Apache Spark có thể chạy nhanh hơn 10 lần so với Hadoop ở trên đĩa cứng và 100 lần khi chạy trên bộ nhớ RAM.

1. **Giới thiệu về Apache Spark**

****

Apache Spark là một hệ thống xử lý phân tán, mã nguồn mở tính toán cụm được sử dụng cho khối lượng công việc dữ liệu lớn, được phát triển sơ khởi vào năm 2009 bởi AMPLab. Sau này, Spark đã được trao cho Apache Software Foundation vào năm 2013 và được phát triển cho đến nay.

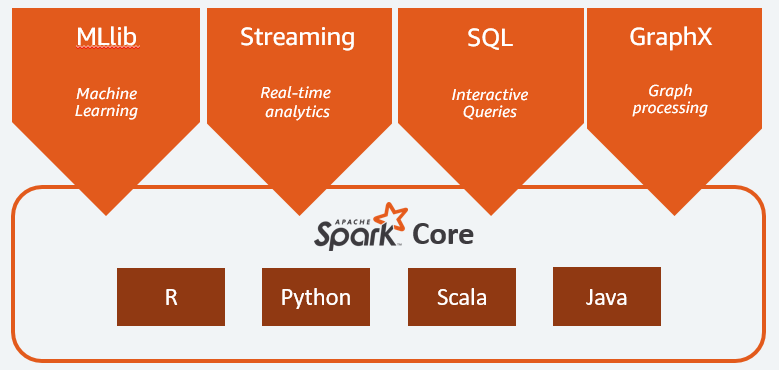
Nó sử dụng bộ nhớ đệm trong bộ nhớ và thực thi truy vấn được tối ưu hóa cho các truy vấn phân tích nhanh đối với dữ liệu ở mọi kích thước. Nó cung cấp các API phát triển bằng Java, Scala, Python và R, đồng thời hỗ trợ sử dụng lại mã trên nhiều khối lượng công việc — xử lý hàng loạt, truy vấn tương tác, phân tích thời gian thực, máy học và xử lý đồ thị. Bạn sẽ thấy nó được sử dụng bởi các tổ chức từ bất kỳ ngành nào, bao gồm FINRA, Yelp, Zillow, DataXu, Urban Institute và CrowdStrike. Apache Spark đã trở thành một trong những khung xử lý phân tán dữ liệu lớn phổ biến nhất với 365.000 thành viên meetup vào năm 2017.

Tốc độ xử lý của Spark có được do việc tính toán được thực hiện cùng lúc trên nhiều máy khác nhau. Đồng thời việc tính toán được thực hiện ở bộ nhớ trong (in-memories) hay thực hiện hoàn toàn trên RAM.

Spark cho phép xử lý dữ liệu theo thời gian thực, vừa nhận dữ liệu từ các nguồn khác nhau đồng thời thực hiện ngay việc xử lý trên dữ liệu vừa nhận được (Spark Streaming).

Spark không có hệ thống file của riêng mình, nó sử dụng hệ thống file khác như: HDFS, Cassandra, S3, … Spark hỗ trợ nhiều kiểu định dạng file khác nhau (text, csv, json…) đồng thời nó hoàn toàn không phụ thuộc vào bất cứ một hệ thống file nào.

1. **Thành phần của Spark**



**Apache Spark** gồm có 5 thành phần chính: Spark Core, Spark Streaming, Spark SQL, MLlib và GraphX, trong đó:

* **Spark Core** là nền tảng cho các thành phần còn lại và các thành phần này muốn khởi chạy được thì đều phải thông qua Spark Core do Spark Core đảm nhận vai trò thực hiện công việc tính toán và xử lý trong bộ nhớ (In-memory computing) đồng thời nó cũng tham chiếu các dữ liệu được lưu trữ tại các hệ thống lưu trữ bên ngoài.
* **Spark SQL** cung cấp một kiểu data abstraction mới (SchemaRDD) nhằm hỗ trợ cho cả kiểu dữ liệu có cấu trúc (structured data) và dữ liệu nửa cấu trúc (semi-structured data – thường là dữ liệu dữ liệu có cấu trúc nhưng không đồng nhất và cấu trúc của dữ liệu phụ thuộc vào chính nội dung của dữ liệu ấy). **Spark SQL** hỗ trợ DSL (Domain-specific language) để thực hiện các thao tác trên DataFrames bằng ngôn ngữ Scala, Java hoặc Python và nó cũng hỗ trợ cả ngôn ngữ SQL với giao diện command-line và ODBC/JDBC server.
* **Spark Streaming** được sử dụng để thực hiện việc phân tích stream bằng việc coi stream là các mini-batches và thực hiệc kỹ thuật RDD transformation đối với các dữ liệu mini-batches này. Qua đó cho phép các đoạn code được viết cho xử lý batch có thể được tận dụng lại vào trong việc xử lý stream, làm cho việc phát triển lambda architecture được dễ dàng hơn. Tuy nhiên điều này lại tạo ra độ trễ trong xử lý dữ liệu (độ trễ chính bằng mini-batch duration) và do đó nhiều chuyên gia cho rằng Spark Streaming không thực sự là công cụ xử lý streaming giống như Storm hoặc Flink.
* **MLlib** (Machine Learning Library): MLlib là một nền tảng học máy phân tán bên trên Spark do kiến trúc phân tán dựa trên bộ nhớ. Theo các so sánh benchmark Spark MLlib nhanh hơn 9 lần so với phiên bản chạy trên Hadoop (Apache Mahout).
* **GrapX**: GrapX là nền tảng xử lý đồ thị dựa trên Spark. Nó cung cấp các Api để diễn tả các tính toán trong đồ thị bằng cách sử dụng Pregel Api.

1. **Những điểm nổi bật của Spark**

Spark được tạo ra để giải quyết các hạn chế đối với MapReduce, bằng cách thực hiện xử lý trong bộ nhớ, giảm số bước trong một tác vụ và bằng cách sử dụng lại dữ liệu trên nhiều thao tác song song. Với Spark, chỉ cần một bước trong đó dữ liệu được đọc vào bộ nhớ, các hoạt động được thực hiện và kết quả được ghi lại — dẫn đến việc thực thi nhanh hơn nhiều. Spark cũng tái sử dụng dữ liệu bằng cách sử dụng bộ nhớ đệm trong bộ nhớ để tăng tốc đáng kể các thuật toán học máy 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 DataFrames, một sự trừu tượng hóa đối với Tập dữ liệu phân tán có khả năng phục hồi (RDD), là một tập hợp các đối tượng được lưu vào bộ nhớ đệm và được sử dụng lại trong nhiều hoạt động của Spark. Điều này làm giảm đáng kể độ trễ khiến Spark nhanh hơn MapReduce nhiều lần, đặc biệt là khi thực hiện học máy và phân tích tương tác.

* Xử lý dữ liệu: Spark xử lý dữ liệu theo lô và thời gian thực
* Tính tương thích: Có thể tích hợp với tất cả các nguồn dữ liệu và định dạng tệp được hỗ trợ bởi cụm Hadoop.
* Hỗ trợ ngôn ngữ: hỗ trợ Java, Scala, Python và R.
* Phân tích thời gian thực:
  + Apache Spark có thể xử lý dữ liệu thời gian thực tức là dữ liệu đến từ các luồng sự kiện thời gian thực với tốc độ hàng triệu sự kiện mỗi giây. Ví dụ: Data Twitter chẳng hạn hoặc lượt chia sẻ, đăng bài trên Facebook. Sức mạnh Spark là khả năng xử lý luồng trực tiếp hiệu quả.
  + Apache Spark có thể được sử dụng để xử lý phát hiện gian lận trong khi thực hiện các giao dịch ngân hàng. Đó là bởi vì, tất cả các khoản thanh toán trực tuyến được thực hiện trong thời gian thực và chúng ta cần ngừng giao dịch gian lận trong khi quá trình thanh toán đang diễn ra.
* Mục tiêu sử dụng:
  + Xử lý dữ liệu nhanh và tương tác
  + Xử lý đồ thị
  + Công việc lặp đi lặp lại
  + Xử lý thời gian thực
  + joining Dataset
  + Machine Learning
  + Apache Spark là Framework thực thi dữ liệu dựa trên Hadoop HDFS. Apache Spark không thay thế cho Hadoop nhưng nó là một framework ứng dụng. Apache Spark tuy ra đời sau nhưng được nhiều người biết đến hơn Apache Hadoop vì khả năng xử lý hàng loạt và thời gian thực.

1. **Các trường hợp sử dụng Apache Spark**

Spark là một hệ thống xử lý phân tán đa năng được sử dụng cho khối lượng công việc dữ liệu lớn. Nó đã được triển khai trong mọi loại trường hợp sử dụng dữ liệu lớn để phát hiện các mẫu và cung cấp thông tin chi tiết theo thời gian thực. Các trường hợp sử dụng ví dụ bao gồm:

**Dịch vụ tài chính**: Spark được sử dụng trong ngân hàng để dự đoán tỷ lệ rời bỏ của khách hàng và đề xuất các sản phẩm tài chính mới. Trong ngân hàng đầu tư, Spark được sử dụng để phân tích giá cổ phiếu để dự đoán xu hướng trong tương lai.

**Y tế:** Spark được sử dụng để xây dựng dịch vụ chăm sóc bệnh nhân toàn diện, bằng cách cung cấp dữ liệu cho nhân viên y tế tuyến đầu cho mọi tương tác của bệnh nhân. Spark cũng có thể được sử dụng để dự đoán / đề nghị điều trị cho bệnh nhân.

**Sản xuất:** Spark được sử dụng để loại bỏ thời gian chết của thiết bị kết nối internet, bằng cách khuyến nghị thời điểm thực hiện bảo trì phòng ngừa.

**Bán lẻ:** Spark được sử dụng để thu hút và giữ chân khách hàng thông qua các dịch vụ và ưu đãi được cá nhân hóa.

**Những doanh nghiệp sử dụng Apache Spark**

Hiện nay, có rất nhiều hãng lớn đã dùng Spark cho các sản phẩm của mình như Yahoo, ebay, IBM, Cisco…

