Phân tích webserver log sử dụng hệ thống xử lý dữ liệu lớn HDFS, Spark on YARN Cluster

Nguyen Chi Thang[1,1*]

[1] University of Information Technology,
Vietnam National University, Ho Chi Minh City, Viet Nam

*Corresponding author(s). E-mail(s):
19522205@gm.uit.edu.vn;

Abstract: Trong thời đại số hóa hiện nay, internet và ứng dụng web đóng vai trò quan trọng trong việc thu thập và trao đổi thông tin. Các máy chủ web là những thành phần chủ chốt của hạ tầng này, thu thập và ghi lại thông tin về mọi hoạt động trên trang web. Nhưng lượng dữ liệu log sinh ra từ máy chủ web rất lớn và phức tạp, gây ra một thách thức lớn trong việc phân tích và tìm hiểu thông tin hữu ích từ những dữ liệu này. Đề tài "Phân tích webserver log sử dụng hệ thống xử lý dữ liệu lớn HDFS, Spark on YARN Cluster" nhằm nghiên cứu và xây dựng một hệ thống xử lý dữ liệu hiệu quả cho việc phân tích dữ liệu log máy chủ web. Điều này sẽ giúp tối ưu hóa hoạt động và hiệu suất của các trang web, đồng thời cung cấp thông tin hữu ích cho các chuyên gia bảo mật và nhà quản lý hệ thống.

Keywords: Webserver log, HDFS, Spark Cluster

1 Giới thiệu

Đề tài này tập trung vào việc phân tích các tệp log của máy chủ web bằng sử dụng một hệ thống xử lý dữ liệu lớn sử dụng Hadoop Distributed File System (HDFS), Apache Spark trên YARN Cluster. Bằng cách kết hợp sức mạnh của HDFS để lưu trữ lượng dữ liệu lớn và Spark để xử lý hiệu quả các phân tích, đề tài nhằm cải thiện khả năng xử lý dữ liệu và khám phá thông tin hữu ích từ tệp log máy chủ web.

Trong bài này, ở phần tiếp theo nhóm giới thiệu về dataset được sử dụng và hướng xử lý dữ liệu, phần ba về các công nghệ được sử dụng và kiến trúc của hệ thống, phần bốn là kết quả thực nghiệm và cuối cùng là phần kết luận.

2 Dataset

2.1 Tổng quan về bộ dữ liệu

Webserver log là tệp tin chứa thông tin về bất kỳ sự kiện nào xảy ra và sự kiện đó được ghi lại. Từ tệp tin log đó, có nhiều thông tin, cụ thể như chi tiết về địa chỉ IP khách truy cập trang web, đường dẫn cụ thể, đường dẫn referer, phương thức, kích thước trang web trả về, trình thu thập dữ liệu truy cập trang web, thông tin chi tiết về doanh nghiệp,...
Bộ dữ liệu mà nhóm đã sử dụng là **"Online Shopping Store - Web Server Logs"**, bộ dữ liệu này được thu thập từ một trang thương mại điển tử của Iran (zanbil.ir), có nhiều mặt hàng được bày bán như đồ điện gia dụng, trang thiết bị cần thiết,..
Kích thước của bộ dữ liệu là 3.3GB, khoảng hơn 10 triệu dòng log, định dạng file text, mỗi một dòng đại diện cho một record của một lần request gửi về phía server.

Dưới đây là ví dụ cho một số dòng record

54.36.149.41 - - [22/Jan/2019:03:56:14 +0330] "GET /filter/27|13%20%D9%85%DA%AF%D8%A7%D9%BE%DB%8C%DA%A9%D8%B3%D9%84,27|%DA%A9%D9%85%D8%AA%D8%B1%20%D8%A7%D8%B2%205%20%D9%85%DA%AF%D8%A7%D9%BE%DB%8C%DA%A9%D8%B3%D9%84,p53 HTTP/1.1" 200 30577 "-" "Mozilla/5.0 (compatible; AhrefsBot/6.1; +http://ahrefs.com/robot/)" "-"

31.56.96.51 - - [22/Jan/2019:03:56:16 +0330] "GET /image/60844/productModel/200x200 HTTP/1.1" 200 5667 "https://www.zanbil.ir/m/filter/b113" "Mozilla/5.0 (Linux; Android 6.0; ALE-L21 Build/HuaweiALE-L21) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/66.0.3359.158 Mobile Safari/537.36" "-"

2.2 Xử lý dữ liệu

Khi xử lý dữ liệu, nhóm sẽ chỉ giữ lại các log theo mẫu và loại bỏ những log "lạ" để lưu trữ riêng phục vụ việc nghiên cứu sau này. Để phân tách dữ liệu theo mẫu, nhóm sẽ sử dụng các biểu thức chính quy (Regex) để tách ra những mẫu liên quan.

Tiến trình xử lý dữ liệu lớn sẽ được triển khai bằng công nghệ phân tích dữ liệu lớn. Mỗi lượt xử lý, nhóm sẽ chia dữ liệu thành các nhóm chứa 250.000 log và sau đó lưu trữ dưới dạng định dạng Parquet, là một định dạng lưu trữ hiệu suất cao phổ biến trong phân tích dữ liệu lớn.

Quá trình xử lý dữ liệu sẽ được thực hiện bằng các bước sau:

Đầu tiên, nhóm sẽ tiến hành tiền xử lý để loại bỏ các log không phù hợp theo mẫu. Các log không hợp lệ hoặc không chứa thông tin quan trọng sẽ bị loại bỏ khỏi tập dữ liệu.

Tiếp theo, sử dụng Regex để phân tách dữ liệu: nhóm sẽ sử dụng các biểu thức chính quy (Regex) để phân tách dữ liệu thành các mẫu liên quan. Biểu thức chính quy sẽ giúp nhóm trích xuất các thông tin quan trọng từ các log, chẳng hạn như thời gian, địa chỉ IP, sự kiện, và các thông tin khác liên quan.

Chia dữ liệu thành các nhóm nhỏ: Sau khi dữ liệu đã được phân tách thành các mẫu, nhóm sẽ chia dữ liệu thành các nhóm nhỏ chứa khoảng 250.000 log mỗi nhóm. Việc chia nhỏ dữ liệu này giúp tối ưu việc xử lý và lưu trữ, đồng thời giảm thiểu tác động đến tài nguyên hệ thống.

Lưu trữ dữ liệu dưới dạng Parquet: Tiếp theo, mỗi nhóm dữ liệu đã phân tách sẽ được lưu trữ dưới dạng định dạng Parquet. Định dạng Parquet được lựa chọn vì tính hiệu quả và khả năng nén dữ liệu tốt, giúp tiết kiệm không gian lưu trữ và tăng tốc độ truy xuất dữ liệu. Sau đó các file này được đẩy lên hệ thống HDFS để thực hiện lưu trữ dữ liệu lớn.

Chuẩn bị cho việc nghiên cứu: Khi tất cả các nhóm dữ liệu đã được xử lý và lưu trữ, nhóm sẽ sẵn sàng cho việc nghiên cứu dữ liệu lớn. Nhóm phân tích dữ liệu lớn qua công nghệ Spark, đẩy nhanh quá trình xử lý dữ liệu.

Quá trình này sẽ đảm bảo rằng nhóm chỉ lưu trữ các log quan trọng và theo mẫu, giúp tiết kiệm không gian lưu trữ và tăng hiệu suất khi tiến hành nghiên cứu và phân tích dữ liệu lớn sau này. Công nghệ phân tích dữ liệu lớn sẽ đảm bảo việc xử lý dữ liệu được thực hiện một cách nhanh chóng và hiệu quả, đồng thời giúp tối ưu hóa tài nguyên máy tính.

Dưới đây là một số dòng record sau khi xử lý:

	client	datetime	method	request	status	size
0	37.152.163.59	2019-01-22 12:38:27+03:30	GET	/image/29314?name=%D8%AF%DB%8C%D8%A8%D8%A7-7.j...	200	1105
1	37.152.163.59	2019-01-22 12:38:27+03:30	GET	/static/images/zanbil-kharid.png	200	358
2	85.9.73.119	2019-01-22 12:38:27+03:30	GET	/static/images/next.png	200	3045

3 HDFS, Spark Cluster và Kiến trúc hệ thống

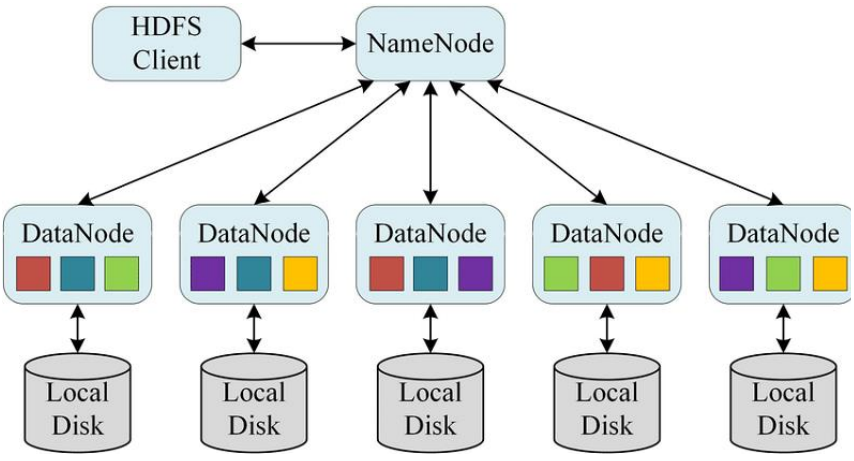
3.1 Hadoop Distributed File System (HDFS)

HDFS (Hadoop Distributed File System) là một hệ thống tệp phân tán được phát triển bởi Apache Hadoop để lưu trữ và quản lý dữ liệu lớn. Chức năng chính của HDFS là cung cấp một nền tảng lưu trữ dữ liệu phân tán có khả năng chịu lỗi, linh hoạt và hiệu suất cao.

HDFS hoạt động bằng cách chia nhỏ dữ liệu thành các phân đoạn và lưu trữ chúng trên nhiều nút trong một cụm máy tính. Cấu trúc phân tán này giúp tăng tính sẵn có của dữ liệu và đồng thời giảm tải cho mỗi nút lưu trữ, tăng khả năng xử lý và hiệu suất.

HDFS cũng hỗ trợ sao lưu đồng bộ, đảm bảo tính toàn vẹn và sẵn sàng của dữ liệu. Nó cung cấp khả năng tự động phục hồi khi có lỗi xảy ra trên một số nút lưu trữ, giữ cho dữ liệu được an toàn và không bị mất.

Đối với việc xử lý dữ liệu lớn và các tác vụ phân tích, HDFS chịu trách nhiệm cho việc lưu trữ dữ liệu đáng tin cậy và đáp ứng nhu cầu của hệ thống xử lý phân tán. Nhờ vào tính chịu lỗi, khả năng mở rộng và hiệu suất cao, HDFS đã trở thành một thành phần quan trọng trong hệ sinh thái của Apache Hadoop và đóng góp lớn vào việc xử lý và quản lý dữ liệu lớn trong các ứng dụng hiện đại.



3.2 Apache Spark, YARN Cluster

Apache Spark và YARN (Yet Another Resource Negotiator) Cluster là hai thành phần quan trọng trong hệ sinh thái Apache Hadoop, được sử dụng rộng rãi trong việc xử lý dữ liệu lớn và tính toán phân tán.

Apache Spark:

Apache Spark là một hệ thống xử lý dữ liệu mã nguồn mở được thiết kế để xử lý và phân tích dữ liệu lớn một cách hiệu quả. Chức năng chính của Apache Spark là cung cấp môi trường tính toán phân tán với khả năng xử lý dữ liệu nhanh chóng và hiệu quả trên các cụm máy tính.

Spark sử dụng mô hình xử lý dữ liệu trong bộ nhớ (in-memory processing) để giảm thời gian truy cập dữ liệu từ đĩa và tăng tốc độ xử lý. Nó hỗ trợ nhiều ngôn ngữ lập trình như Scala, Java, Python và R, giúp cho các nhà phát triển có thể thực hiện các phân tích phức tạp và tính toán song song một cách dễ dàng.

Spark cũng đi kèm với các thư viện phân tích dữ liệu mạnh mẽ như Spark SQL, Spark Streaming, Spark MLlib và Spark GraphX, giúp cho việc xử lý và phân tích dữ liệu trở nên dễ dàng và linh hoạt hơn. Spark cũng có khả năng tích hợp tốt với HDFS, giúp cho việc truy xuất và lưu trữ dữ liệu được thực hiện hiệu quả.

YARN Cluster:

YARN (Yet Another Resource Negotiator) là một trình quản lý tài nguyên phân tán trong hệ sinh thái Apache Hadoop. Chức năng chính của YARN là quản lý và phân phối tài nguyên hệ thống (CPU, bộ nhớ, lưu trữ) cho các ứng dụng chạy trên một cụm máy tính.

YARN Cluster cho phép chia sẻ tài nguyên giữa các ứng dụng khác nhau, đồng thời tối ưu hóa việc sử dụng tài nguyên trong hệ thống. Nó giúp xử lý tài nguyên và công việc một cách hiệu quả, đảm bảo tính nhất quán và chia sẻ công bằng tài nguyên giữa các ứng dụng khác nhau.

YARN cũng giúp quản lý việc triển khai và khởi chạy các ứng dụng trên cụm máy tính một cách dễ dàng. Nó chịu trách nhiệm cho việc quản lý và giám sát các công việc trong hệ thống, đồng thời cung cấp khả năng tự động phục hồi khi có lỗi xảy ra.

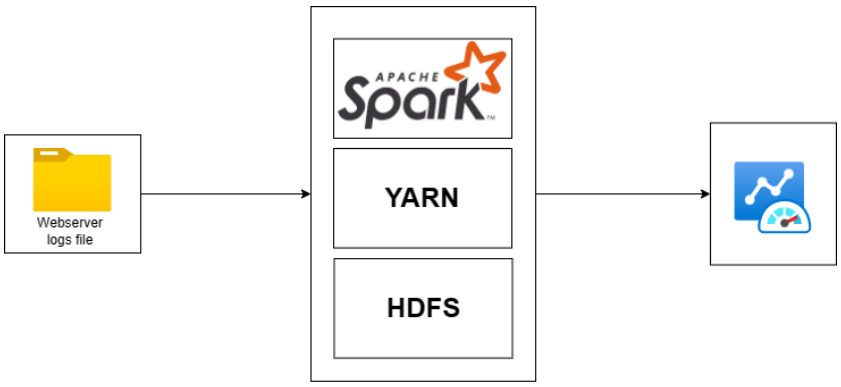
Kết hợp giữa Apache Spark và YARN Cluster cho phép thực hiện các công việc xử lý dữ liệu lớn và tính toán phân tán một cách mạnh mẽ, hiệu quả và linh hoạt. Spark sử dụng YARN để quản lý tài nguyên và triển khai các ứng dụng, giúp cho việc xử lý và phân tích dữ liệu lớn trở nên dễ dàng và hiệu quả.

3.3 Kiến trúc hệ thống

Trong bài viết này, chúng ta sẽ thực hiện cài đặt Hadoop và Spark trên hai máy ảo chạy Ubuntu 18.04.6 trên phần mềm VMware. Mỗi máy ảo được cấu hình với 2 cores, 4GB RAM và 20GB bộ nhớ ngoài. Trong cụm máy ảo này, chúng ta sẽ có một máy ảo đóng vai trò làm datanode (worker) và máy ảo còn lại thực hiện hai nhiệm vụ, làm namenode (master) và làm datanode (worker).

Các phiên bản mà chúng ta sử dụng cho Hadoop và Spark là 3.2.4, đảm bảo tính ổn định và tương thích trong việc triển khai cụm xử lý dữ liệu lớn.

Tiến hành cài đặt, chúng ta sẽ cấu hình Hadoop để tạo môi trường hệ thống tệp phân tán (HDFS) và triển khai Spark để tận dụng tính toán phân tán.



Nhóm sử dụng trực tiếp Jupyter Notebook để chạy code PySpark, phân tích và hiển thị trực quan dữ liệu. Cụ thể, trong Jupyter Notebook, nhóm có thể viết và chạy code PySpark để xử lý dữ liệu lớn từ HDFS và sử dụng các thư viện trực quan hóa dữ liệu như Matplotlib, Seaborn hoặc Plotly để hiển thị dữ liệu dưới dạng biểu đồ và biểu đồ mô tả dữ liệu và xu hướng trong dữ liệu. Điều này giúp cho việc phân tích dữ liệu và trực quan hóa kết quả trở nên dễ dàng và hiệu quả, đồng thời cung cấp môi trường tương tác để thực hiện các tác vụ xử lý dữ liệu lớn trong ứng dụng thực tế.

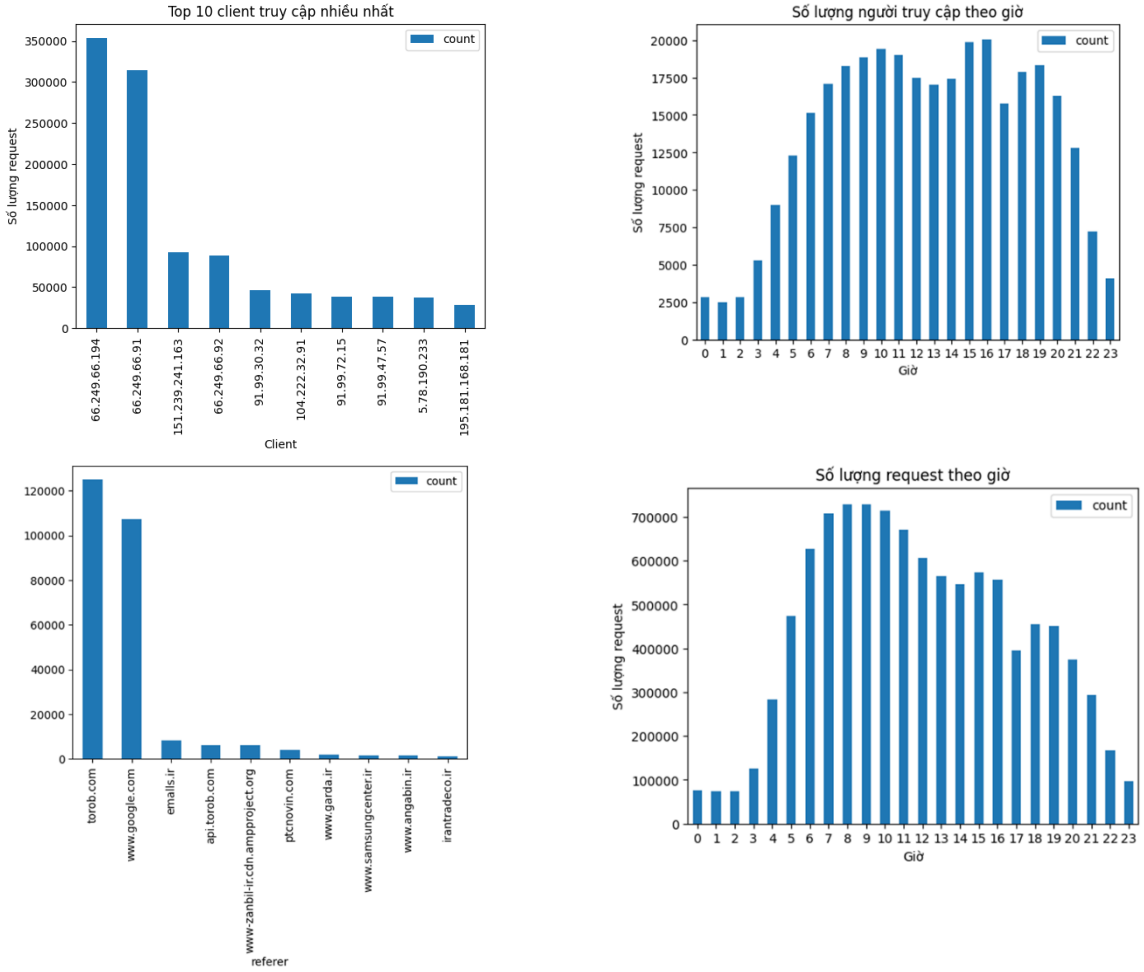
4 Thực nghiệm và một số phân tích

Dưới đây liệt kê một số kết quả thu về sau khi phân tích bộ dữ liệu. Tổng thời gian phân tích hết 8 phút 56s, bao gồm cả thời gian khởi tạo Spark Session.

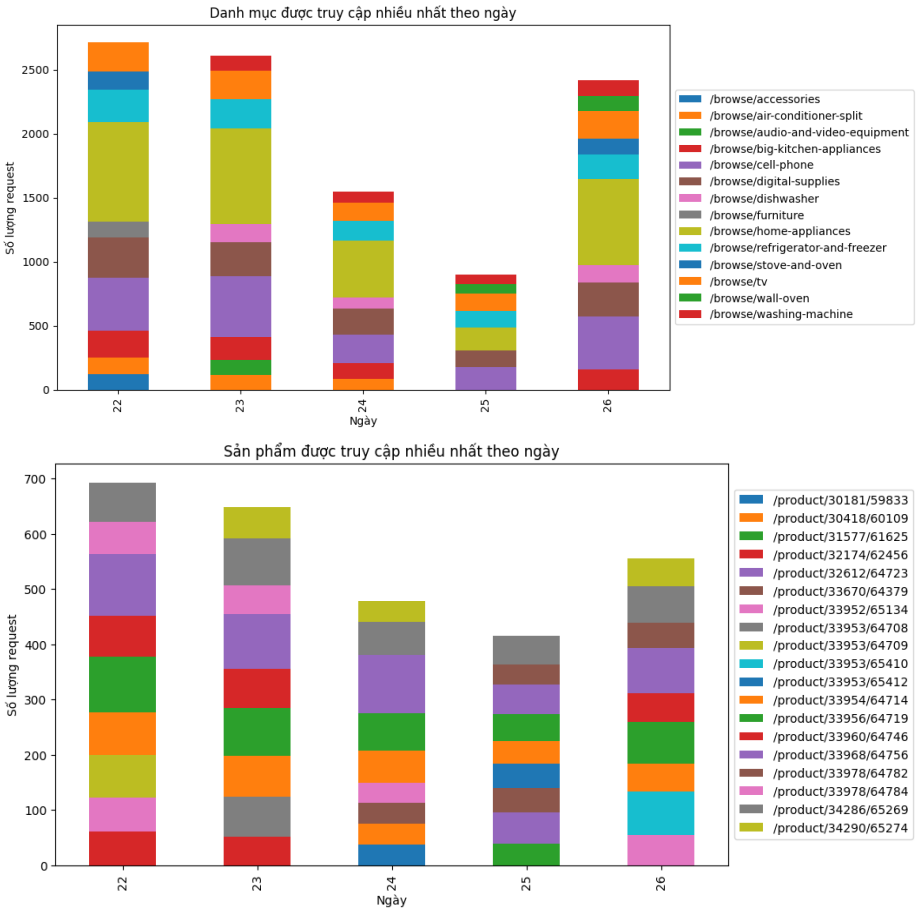
Một số thống kê tổng quát:

Tổng số lượng request	10.364.865
Số lượng người truy cập	258.445
Lượng request trung bình mỗi ngày	2.072.973
Lượng request trung bình mỗi giờ	431.869

Một số biểu đồ trực quan.



Theo chiều kim đồng hồ, lần lượt là “Top 10 người (theo địa chỉ IP) request nhiều nhất”, “Số lượng người (theo địa chỉ IP) request theo thời gian”, “Số lượng request theo giờ” và cuối cùng là “Top các đường dẫn referer”.



Từ trên xuống dưới lần lượt là “Các danh mục được truy cập nhiều nhất theo từng ngày” và “Sản phẩm được truy cập nhiều nhất theo từng ngày.

5 Kết luận và hướng phát triển

Qua quá trình triển khai hệ thống, chúng ta đã xây dựng một môi trường xử lý dữ liệu lớn hiệu quả, có khả năng mở rộng và linh hoạt. Việc sử dụng HDFS cho phép chúng ta lưu trữ và quản lý dữ liệu log một cách phân tán và chịu lỗi, đồng thời giảm tải cho mỗi nút lưu trữ và tăng tốc độ truy xuất dữ liệu. Điều này giúp tiết kiệm không gian lưu trữ và tối ưu hóa việc lưu trữ dữ liệu lớn từ webserver. Kết hợp giữa HDFS và Spark trên YARN Cluster đã mang lại những lợi ích to lớn trong việc phân tích webserver log. Quá trình phân tích dữ liệu đã diễn ra hiệu quả, đồng thời cung cấp kết quả trực

quan và đáng tin cậy. Từ đó, chúng ta có thể nắm bắt thông tin quan trọng từ dữ liệu log, phát hiện xu hướng và vấn đề tiềm ẩn, từ đó đưa ra các quyết định và cải tiến để tối ưu hóa hoạt động của webserver.

Trong tương lai, chúng ta có thể tiếp tục phát triển và mở rộng hệ thống phân tích dữ liệu lớn này. Và tiến hành nghiên cứu thêm để triển khai xây dựng việc huấn luyện các mô hình Machine learning, Deep Learning sử dụng hệ thống phân tán

References

1. Online Shopping Store - Web Server Logs
2. Setup Hadoop Cluster
3. Setup Spark YARN Cluster