Tạp chí Quốc tế về Khoa học Máy tính, Kỹ thuật và Ứng dụng (IJCSEA) tập 7, số 3/4, tháng 8 năm 2017

HỆ THỐNG CƠ SỞ DỮ LIỆU HYBRID CHO VIỆC LƯU TRỮ VÀ QUẢN LÝ DỮ LIỆU LỚN

Blessing E. James and P.O.Asagba

Khoa Khoa học Máy tính, Đại học Port Harcourt, Choba, Rivers State, Nigeria

TRỪU TƯỢNG

Hệ thống cơ sở dữ liệu quan hệ là hệ thống lưu trữ tiêu chuẩn trong bốn mươi năm qua. Gần đây, những tiến bộ trong công nghệ đã dẫn đến sự gia tăng theo cấp số nhân về dung tích, tốc độ và sự đa dạng của dữ liệu ngoài những gì cơ sở dữ liệu quan hệ có thể xử lý. Các lập trình viên đang dần chuyển sang NoSQL, một cơ sở dữ liệu phi quan hệ để lưu trữ và quản lý dữ liệu. Một số tính năng cốt lõi của hệ thống cơ sở dữ liệu như ACID đã bị ảnh hưởng trong cơ sở dữ liệu NOSQL. Việc làm này đã đề xuất một hệ thống cơ sở dữ liệu lai dùng để lưu trữ và quản lý dữ liệu cực kỳ đồ sộ của các thành phần khác nhau được gọi là dữ liệu lớn, sao cho hai mô hình được tích hợp trong một hệ thống để loại bỏ các hạn chế của các hệ thống riêng lẻ. Hệ thống được triển khai trong MongoDB, một cơ sở dữ liệu NoSQL và SQL. Kết quả thu được cho thấy rằng việc có hai cơ sở dữ liệu này trong một hệ thống có thể tăng cường lưu trữ và quản lý dữ liệu lớn giúp thu hẹp khoảng cách giữa phương pháp lưu trữ quan hệ và NoSQL.

NHỮNG TỪ KHÓA

ACID, BASE, Dữ liệu lớn, NoSQL, SQL, MongoDB

1. GIỚI THIỆU

Công nghiệp hóa nhanh chóng, các thiết bị ngày càng tối ưu, việc sử dụng các thiết bị di động dần leo thang và những tiến bộ gần đây về kiến thức công nghệ như: kết nối của mọi đối tượng với Internet được gọi là Internet vạn vật (IoT), điện toán lưới hoặc điện toán đám mây, khách hàng lớn, v.v. đã dẫn đến một lĩnh vực dữ liệu đồ sộ có qui mô không thể tưởng tượng được gọi là dữ liệu lớn. Nói chung, dữ liệu lớn được coi là khối lượng dữ liệu khổng lồ (terabyte và pentabyte) bao gồm nhiều loại dữ liệu khác nhau (có cấu trúc, bán cấu trúc và không cấu trúc) và có tính sẵn dụng theo thời gian thực (vận tốc) để dữ liệu đó không thể được lưu trữ hoặc quản lý bằng các hệ thống cơ sở dữ liệu quan hệ truyền thống. Nói một cách đơn giản, dữ liệu lớn là dữ liệu có khối lượng và bản chất

(loại dữ liệu bán cấu trúc và phi cấu trúc) lớn hơn những gì hệ thống cơ sở dữ liệu thông thường có thể xử lý. Sự gia tăng khổng lồ về số lượng dữ liệu này đã mở ra cơ hội lớn cho những thành tựu khoa học quan trọng, cải thiện chiến lược kinh doanh, cũng như phương pháp chăm sóc sức khỏe, v.v.

Hệ thống cơ sở dữ liệu quan hệ đã phục vụ như hệ thống lưu trữ thực tế trong vài năm. Tuy nhiên, trong vòng bốn năm qua, đã có những giải pháp tuyệt vời trong thế giới điện toán làm giảm bớt sự phổ biến của cơ sở dữ liệu quan hệ, dẫn đến việc xem xét mô hình lưu trữ mới có tên là NoSQL. Điều này là do cơ sở dữ liệu quan hệ không bao giờ được thiết kế để lưu trữ hoặc quản lý khối lượng dữ liệu với tốc độ cao và đa dạng, dữ liệu phi cấu trúc hoặc sự phát triển nhanh. Do đó, các doanh nghiệp đang chuyển sang một phương pháp lưu trữ mới gọi là NoSQL để tìm giải pháp cho những thách thức vốn có trong dữ liệu lớn. Hầu hết NoSL chia tỷ lệ theo chiều ngang với sự gia tăng về khối lượng dữ liệu và cũng đủ linh hoạt để tổ chức thu thập dữ liệu phân tán và có cấu trúc không hoàn chỉnh. Với NoSQL, dữ liệu ở dạng âm thanh, video, email và tài liệu có thể được lưu trữ và quản lý đúng cách. Tuy nhiên, với cách tiếp cận hấp dẫn, nó không phải là không có hạn chế. Những người cải cách của NoSQL hoàn toàn hoặc có thể vô tình bỏ qua một số thành phần cơ sở dữ liệu đáng khao khát như sự toàn tác, và bảo mật để đạt được những gì cơ sở dữ liệu quan hệ không thể cung cấp.

Hệ thống cơ sở dữ liệu quan hệ đã phục vụ như hệ thống lưu trữ thực tế trong vài năm. Tuy nhiên, trong vòng bốn năm qua, đã có những giải pháp tuyệt vời trong thế giới điện toán làm giảm bớt sự phổ biến của cơ sở dữ liệu quan hệ, dẫn đến việc xem xét mô hình lưu trữ mới có tên là NoSQL. Điều này là do cơ sở dữ liệu quan hệ không bao giờ được thiết kế để lưu trữ hoặc quản lý khối lượng dữ liệu như vậy, với tốc độ cao và đa dạng, dữ liệu phi cấu trúc hoặc tăng trưởng nhanh chóng. Do đó, các doanh nghiệp đang chuyển sang một phương pháp lưu trữ mới nổi gọi là NoSQL để tìm giải pháp cho những thách thức cố hữu trong dữ liệu lớn. Hầu hết NoSL mở rộng theo chiều ngang với sự gia tăng về khối lượng dữ liệu và cũng đủ linh hoạt để tổ chức thu thập dữ liệu phân tán và có cấu trúc một phần. Với NoSQL, dữ liệu ở dạng thoại, video, email và tài liệu có thể được lưu trữ và quản lý đúng cách. Tuy nhiên, hấp dẫn như cách tiếp cận này, nó không phải là không có hạn chế. Các nhà đổi mới của NoSQL hoàn toàn hoặc có thể vô tình bỏ qua một số 544...các mục dự kiến và không phụ thuộc vào loại dữ liệu được nhập.

Trong công việc này, chúng tôi đã phát triển một hệ thống cơ sở dữ liệu lai sử dụng cơ sở dữ liệu MongoDB và MySQL, là các biến thể phổ biến của cơ sở dữ liệu NoSQL và hệ thống cơ sở dữ liệu quan hệ tương ứng. Trước khi lưu trữ dữ liệu, dữ liệu được phân loại thành dữ liệu có cấu trúc và dữ liệu phi cấu trúc tùy thuộc vào bản chất của dữ liệu. Dữ liệu phi cấu trúc được lưu trữ và quản lý trong cơ sở dữ liệu MongoDB

trong khi lưu trữ và quản lý dữ liệu có cấu trúc chặt chẽ được thực hiện bằng cơ sở dữ liệu MySQL. Hệ thống lai của chúng tôi là làm sao cho các cơ sở dữ liệu tạo nên hệ thống có thể hoạt động riêng biệt, chẳng hạn như system có thể được sử dụng như một cơ sở dữ liệu MongoDB hoàn chỉnh và riêng biệt. Thay vì từ bỏ chức năng của các hệ thống cơ sở dữ liệu quan hệ cho cơ sở dữ liệu NoSQL chúng tôi đã phát triển một cách tiếp cận khác cung cấp lợi ích của cả hai hệ thống thành một hệ thống cơ sở dữ liệu duy nhất.

2. PHƯƠNG PHÁP LƯU TRỮ CƠ SỞ DỮ LIỆU QUAN HỆ

Mô hình quản lý cơ sở dữ liệu quan hệ được xây dựng trên các nguyên tắc quan hệ trong toán học. Nhiều cơ sở dữ liệu phổ biến và miễn phí nằm trong lớp này. Có khả năng lưu trữ dữ liệu trong bảng dạng hàng và cột đồng thời giữ lại và thiết lập mối quan hệ giữa các dữ liệu là một trong những đặc điểm cơ bản của cơ sở dữ liệu quan hệ. Các quy tắc cơ bản của cơ sở dữ liệu quan hệ là:

- Dữ liệu và thông tin phải được lưu trữ trong bảng dạng hàng và cột.
- Để truy cập nội dung cột, tên bảng, cột và khóa chính phải được chỉ định.
- Các trường hợp không có và không phù hợp phải được xử lý một cách có hệ thống khác với các mục dự kiến và không phụ thuộc vào loại dữ liệu được nhập.
- Hệ thống quản lý cơ sở dữ liệu sẽ hỗ trợ một danh mục trực tuyến đang hoạt động.
- Phải có ít nhất một ngôn ngữ được hệ quản trị cơ sở dữ liệu hỗ trợ có thể được sử dụng riêng biệt hoặc chung trong các chương trình.
- Hệ quản trị cơ sở dữ liệu phải có khả năng cập nhật chế độ xem.
- Các thao tác cơ bản như chèn, cập nhật và xóa phải được hỗ trợ bởi hệ thống quản lý cơ sở dữ liệu.
- Các chỉnh sửa thực hiện trên cấu trúc logic như thêm hoặc bớt cột từ các bảng không được ảnh hưởng đến chế độ xem của người dùng.
- Các thay đổi ở mức vật lý như lưu trữ không được ảnh hưởng đến toàn bộ ứng dụng.
- Các hạn chế liên quan đến tính toàn vẹn nên được tách biệt khỏi ứng dụng.
- Trong môi trường phân bố, người dùng phải nhận thức được tác động của việc phân phối cơ sở dữ liệu.

Trong mô hình quan hệ, dữ liệu được biểu diễn dưới dạng bảng hoặc quan hệ. Một bảng như trong bảng 1 chủ yếu là một tập hợp các mục nhập dữ liệu có liên quan và nó được tạo thành từ một số cột và hàng tương ứng được gọi là các trường và bản ghi. Gần

như tất cả các cơ sở dữ liệu được xây dựng trên mô hình quan hệ đảm bảo các toàn tác: Atomicity, Nhất Quán (Consistency), Cô Lập(Isolation), Độ Bền(Durability) (viết tắt là ACID).

Bảng 1: Biểu diễn dữ liệu dạng bảng trong cơ sở dữ liệu quan hệ

ID	TÊN	VÙNG
1	James	Máy tính
2	Blessing	Toán học
3	Idara	Địa chất

3. CÁCH TIẾP CẬN LƯU TRỮ CƠ SỞ DỮ LIỆU NOSQL

Cơ sở dữ liệu NoSQL là cơ sở dữ liệu hướng đối tượng được thiết kế để giải quyết các vấn đề được tạo ra bởi sự mở rộng khối lượng và tính đa dạng của dữ liệu, đặc biệt là trong các ứng dụng dữ liệu lớn. Cơ sở dữ liệu NoSQL được coi là rất cần thiết trong trường hợp khối lượng dữ liệu vượt xa khả năng xử lý của cơ sở dữ liệu quan hệ và cả thành phần thông tin không được lưu trữ trong cơ sở dữ liệu quan hệ. Cơ sở dữ liệu NoSQL được xây dựng trên mô hình phân tán để đảm bảo các thuộc tính cơ bản có sẵn, khởi động mềm, cách thuộc tính có thể có tính nhất quán (BASE). Hình 1 cho thấy một mô hình cơ sở dữ liệu NoSQL cơ sở tài liệu. Trong mô hình này thay vì lưu trữ dữ liệu theo hàng và cột trong bảng, dữ liệu có khả năng điền vào một số cột trong bảng có thể được lưu trữ trong tài liệu được nhóm thành các bộ sưu tập. Có bốn loại cơ sở dữ liệu NoSQL cơ bản.

- Chính trị cửa hàng: Dữ liệu được lưu trữ bằng cách sử dụng hai mục được kết nối nhưng riêng biệt mục tin - một mã định danh đặc biệt được gọi là khóa và một giá trị tương ứng có thể là dữ liệu hoặc con trỏ tới vị trí của dữ liệu. Nó rất thích hợp cho các hệ thống dựa trên hệ thống khóa cơ bản. Ví dụ: Máy phát điện, Riak
- Lưu trữ dạng cột liên kết: Dữ liệu được tổ chức theo hàng và cột như trong các hệ cơ sở dữ liệu quan hệ. Ví dụ: Cassandra
- Lưu trữ dạng đồ thị liên kết: Dành cho các quy trình có thể được biểu diễn dưới dạng mối quan hệ với các yếu tố được kết nối với nhau, chẳng hạn như mạng xã hội. Ví dụ: Neo4j
- Hệ thống database hướng document: Dữ liệu được lưu trữ trong tài liệu. Thích hợp cho việc lưu trữ tài liệu ở đa định dạng. Ví dụ: MongoDB

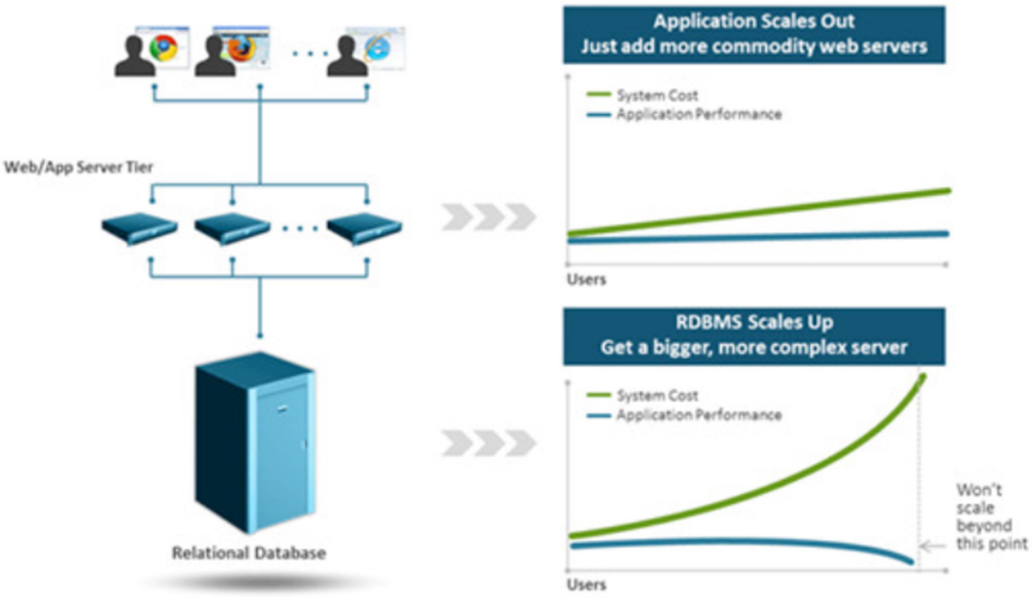
3.1 ĐIỂM MẠNH CỦA CƠ SỞ DỮ LIỆU NoSQL VÀ HẠN CHẾ CỦA CƠ SỞ DỮ LIỆU QUAN HỆ

Cơ sở dữ liệu NoSQL được phát triển để loại bỏ những hạn chế hoặc nhược điểm gặp phải trong việc sử dụng cơ sở dữ liệu quan hệ, đặc biệt là trong môi trường lưu trữ dữ liệu lớn. Như vậy, hầu hết các nhược điểm trong hệ thống lưu trữ quan hệ tạo thành thế mạnh hoặc ưu điểm của hệ thống cơ sở dữ liệu NoSQL. Điểm yếu của cơ sở dữ liệu quan hệ và điểm mạnh của cơ sở dữ liệu NoSQL chủ yếu phụ thuộc vào các tính năng được thảo luận trong các phần sau.

3.1.1 KHẢ NĂNG MỞ RỘNG

Trong các hệ thống lưu trữ quan hệ, việc mở rộng đạt được bằng cách thay thế bộ lưu trữ hoặc máy chủ hiện có bằng một máy chủ lớn hơn (đắt tiền hơn), nghĩa là tăng mã lực của phần cứng hiện có. Điều này được biết đến là sự mở rộng hoặc mở rộng quy mô. Rõ ràng là khi khối lượng dữ liệu tăng lên, có thể đến giai đoạn máy chủ giá rẻ lớn nhất có thể không đáp ứng được yêu cầu lưu trữ như trong hình 2, điều này có thể làm giảm hiệu suất của hệ thống. Và hệ thống cũng bị ảnh hưởng bởi một điểm lỗi duy nhất. Cơ sở dữ liệu NoSQL được xây dựng trên kiến trúc phân tán sao cho có thể phân vùng (sharding) cơ sở dữ liệu trên một số máy chủ. Việc mở rộng như vậy đạt được thông qua việc bổ sung các máy chủ giá rẻ được kết nối với cụm cơ sở dữ liệu được hiển thị trong hình 3. Điều này được gọi là mở rộng quy mô hoặc thu hẹp qui mô vị thế. Tỷ lệ ngang làm tăng hiệu suất của hệ thống với chi phí tối thiểu bằng cách thúc đẩy mở rộng dữ liệu nhanh chóng và loại bỏ điểm lỗi duy nhất tồn tại trong cơ sở dữ liệu quan hệ.

Hình 1: Biểu diễn dữ liệu cơ sở.



Hình 2. Thẳng đứng rộng trong cơ sở dữ liệu quan hệ.

3.1.2 TÍNH LINH HOẠT

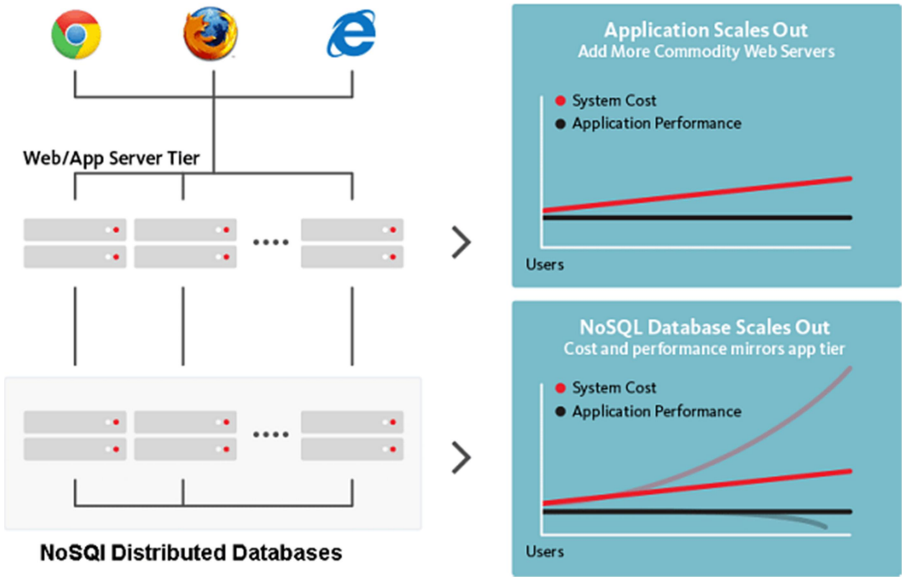
Cơ sở dữ liệu quan hệ là cơ sở dữ liệu lược đồ bất khả tri; dữ liệu không thể được lưu trữ mà không xác định lược đồ của dữ liệu đó. Như vậy, trong môi trường dữ liệu lớn, nơi cần lưu trữ dữ liệu phi cấu trúc, không thể biết trước lược đồ hoặc cấu trúc của dữ liệu. Mặt khác, NoSQL có lược đồ động sao cho lược đồ không được xác định trước. Vì vậy, cơ sở dữ liệu NoSQL có thể dùng để lưu trữ cả dữ liệu có cấu trúc và phi cấu trúc.

ĐIỂM YẾU CỦA CƠ SỞ DỮ LIỆU NOSQL

Giải quyết một số vấn đề trong cơ sở dữ liệu quan hệ đã đưa ra những điểm yếu nhất định trong cơ sở dữ liệu NoSQL. Một số điểm yếu của cơ sở dữ liệu NoSQL là:

- Giao dịch phức tạp: MongoDB không hỗ trợ giao dịch nhiều tài liệu. Với sự sẵn có của cơ sở dữ liệu NoSQL, hỗ trợ cho các giao dịch ACID trên các tài liệu thường bị loại bỏ. Loại trừ giao dịch ACID là một sự đánh đổi được NoSQL sử dụng để cung cấp giải pháp cho các vấn đề liên quan đến khả năng mở rộng.
- Tính ổn định: Một số cơ sở dữ liệu NoSQL vẫn đang trong giai đoạn tiền sản xuất và do đó không ổn định hoặc chưa đủ hoàn thiện cho một số tác vụ nhạy cảm.

- Hỗ trợ toàn cầu: Các doanh nghiệp yêu cầu dịch vụ và hỗ trợ toàn cầu từ các nhà cung cấp cơ sở dữ liệu khi một thành phần cốt lõi của hệ thống bị lỗi. NoSQL thiếu các dịch vụ như vậy cho khách hàng doanh nghiệp.



Hình 3. Mở rộng qui mô ngang trong cơ sở dữ liệu NoSQL.

4. DỮ LIỆU LỚN

Dữ liệu lớn đề cập chủ yếu đến nhóm dữ liệu đã trở nên cực kỳ đồ sộ (petabyte và terabyte) bao gồm nhiều loại dữ liệu khác nhau (có cấu trúc, bán cấu trúc, không cấu trúc) và tính khả dụng trong thời gian thực (tốc độ) sao cho không hiệu quả khi được lưu trữ hoặc xử lý với các công cụ hoặc phương tiện truyền thống như hệ thống cơ sở dữ liệu thông thường. Những công cụ truyền thống này có đã được sử dụng trong nhiều năm để kiểm soát, xử lý và phân tích các tập thông tin đồ sộ trong các công ty, các ngành công nghiệp, v.v. Dữ liệu lớn không chỉ đề cập đến việc thu thập dữ liệu đồ sộ, mà còn khối lượng cực lớn dữ liệu có cấu trúc và phi cấu trúc có tốc độ thay đổi rất cao, bắt nguồn từ các con đường khác nhau có thể bao gồm email, phương tiện truyền thông xã hội, cuộc gọi điện thoại, v.v. Dữ liệu lớn được sử dụng để mô tả một tập hợp dữ liệu phức tạp có khối lượng tăng liên tục và nhanh chóng với tốc độ mà việc sử dụng các công cụ quản lý thông thường để lưu trữ và phân tích trở nên kém hiệu quả và không phù hợp. Sự phức tạp trong khối lượng phân tán dữ liệu khổng lồ này là do thực tế dữ liệu được thu thập từ các nguồn khác nhau cũng có thể khác nhau định dạng và có thể cần phải tích hợp chúng thành một đơn vị để phân tích.

4.1 TÍNH CHẤT CỦA DỮ LIỆU LỚN

Mặc dù từ "lớn" dùng để chỉ kích thước của một thứ gì đó, nhưng trong dữ liệu lớn, lớn không chỉ giới hạn ở khối lượng dữ liệu mà còn là kết hợp các thuộc tính khác như vận tốc và tính đa dạng. Ba thuộc tính này mô tả các thuộc tính chính của dữ liệu lớn được gọi là 3 V của dữ liệu lớn:

- Khối lượng: Điều này đôi khi được coi là thuộc tính cuối cùng của dữ liệu lớn. Nó miêu tả lượng dữ liệu rất lớn và tăng dần theo thời gian từ terabyte (1012 byte) đến yotta-byte có hàng nghìn tỷ gigabyte.
- Vận tốc: Vận tốc đề cập đến tính khả dụng của dữ liệu trong thời gian thực để xử lý. Dữ liệu lớn được biểu thị đặc điểm bởi sự xuất hiện tức thời của dữ liệu khổng lồ dùng để xử lý. Nó kéo theo tỷ lệ mà tại đó dữ liệu được lan truyền trong hệ thống, Ví dụ: Tốc độ mà dữ liệu được lấy ngoài các hoạt động bên trong và bên ngoài, các nguồn như tương tác với máy móc, con người, phương tiện truyền thông xã hội, vv.
- Đa dạng: Điều này thể hiện định dạng dữ liệu đa dạng trong một tập dữ liệu. Dữ liệu lớn được tạo thành từ dữ liệu có nguồn gốc từ nhiều nguồn khác nhau như email, máy móc, mạng xã hội, kinh doanh giao dịch, thiết bị di động, v.v. Dữ liệu từ các nguồn khác nhau có các dạng riêng biệt như dạng bảng tính, hình ảnh, video, v.v. Tính đa dạng như một thuộc tính của dữ liệu lớn mô tả các dạng dữ liệu khác nhau thu được từ các nguồn khác nhau.
- Tính chính xác: Đề cập đến tính chính xác của dữ liệu. Nó liên quan đến sự thích hợp của dữ liệu (xử lý hoặc phân tích) cho nhiệm vụ hiện tại. Nó cho thấy sự cần thiết phải tránh tích lũy dữ liệu bẩn.
- Tính biến động: Giải quyết vấn đề thời gian hợp lý để lưu trữ dữ liệu trong thời gian thực. Nó điều tra tính hợp lệ của dữ liệu được lưu trữ cho phân tích hiện tại.
- Tính chất hợp lệ: Các quyết định có giá trị như dữ liệu được sử dụng trong các phân tích.

4.2 THÁCH THỨC CỦA DỮ LIỆU LỚN

Những thách thức của dữ liệu lớn cũng có thể được gọi là các bước liên quan đến quá trình xử lý dữ liệu lớn này, khối lượng dữ liệu cực lớn của các kiểu khác nhau và tốc độ sử dụng cao. Để tận dụng nhiều lợi ích của dữ liệu lớn, dữ liệu phải trải qua các quy trình sau; nói cách khác, những thách thức sau đây phải được vượt qua:

- Ăn Nhập: Quy trình thu thập và nhập dữ liệu để sử dụng hoặc lưu trữ. Dữ liệu có thể được nhập sau khi được cung cấp bởi nguồn hoặc được nhóm

thành các đợt và được nhập trong khoảng thời gian quy định. Quá trình này thường bắt đầu với việc xếp hạng các nguồn dữ liệu, xác thực từng tệp trước khi truyền dữ liệu đến đích chính xác.

- Lưu trữ: Lưu trữ rất phức tạp; nó bao gồm tìm kiếm và truy xuất dữ liệu và cũng có thể bao gồm các vấn đề bảo mật và quyền riêng tư phức tạp. Khung để lưu trữ cho dữ liệu lớn nên chứa được khối lượng lớn dữ liệu có cấu trúc, bán cấu trúc và phi cấu trúc.
- Phân tích: Dữ liệu lớn gần như vô dụng nếu không có các công cụ và quy trình phân tích hiệu quả thông qua đó thông tin hữu ích được trích xuất từ những thứ dường như là dữ liệu rác. Phân tích trong lĩnh vực dữ liệu lớn cũng bao gồm các tính toán trên dữ liệu lớn. Nó bao gồm các khung và công cụ như MapReduce và Hadoop được sử dụng để rút ra ý nghĩa từ dữ liệu lớn. Kết quả phân tích dữ liệu có thể được sử dụng để cải thiện các dịch vụ dùng để cung cấp cho khách hàng, chiến lược tiếp thị và ra quyết định chung.
- Trực quan hóa: Điều này bao gồm việc trình bày dữ liệu để dễ dàng xác định các mẫu hoặc nắm bắt của những khái niệm mới. Các công cụ và kỹ thuật trực quan hóa dữ liệu lớn cho phép biểu diễn dữ liệu ở dạng đồ thị, biểu đồ, bản đồ và thậm chí cả video giúp dễ dàng tìm kiếm thông tin liên lạc.

5. CÁC CÔNG TRÌNH LIÊN QUAN

[13] [2]Trình bày báo cáo về sự phân loại, thuộc tính và so sánh cơ sở dữ liệu NoSQL. Họ khám phá điểm mạnh và điểm yếu của các loại cơ sở dữ liệu NoSQL khác nhau. [3]Đã điều tra tính linh hoạt của cơ sở dữ liệu NoSQL dựa trên khả năng mở rộng và tốc độ đọc và hoạt động cập nhật. Nó đã chỉ ra rằng tốc độ mà thao tác đọc được thực hiện trong Hbase là cao trong khi thao tác chèn nhanh ở Cassandra và Riak chậm ở cả 2 hoạt động đọc và ghi. [17] Thực hiện so sánh chi tiết giữa MongoDB và Microsoft SQL cơ sở dữ liệu. Microsoft SQL là một hệ thống cơ sở dữ liệu quan hệ nên khả năng lưu trữ của hệ thống chỉ có thể được tăng lên bằng cách giới thiệu một máy chủ có dung lượng lớn hơn và điều này thường phát sinh chi phi thêm. Mặt khác, cơ sở dữ liệu NoSQL là hệ thống phi quan hệ có khả năng dễ dàng chia tỷ lệ ngang để chứa nhiều dữ liệu hơn. Hệ thống đã được thực hiện trong ngôn ngữ lập trình Java sử dụng môi trường phát triển tích hợp Eclipse. Được quan sát thấy rằng mặc dù MongDB và Microsoft SQL thực hiện thao tác ghi nhanh hơn đọc thuật toán, đọc và thao tác ghi trong MongoDB nhanh hơn gần 10 lần so với thao tác đọc và ghi trong cơ sở dữ liệu Microsoft SQL. Mặc dù MongoDB có khả năng đọc/ghi cao hơn nhưng vẫn tồn tại tình trạng tốc độ không phải là yêu cầu cuối

cùng hoặc duy nhất đối với cơ sở dữ liệu. MongoDB không thích hợp cho nhiệm vụ giao dịch nặng. Đánh giá hiệu suất của MongoDB, PostgreSQL và Cassandra bởi [4] tiết lộ rằng Cassandra thích hợp hơn cho các hệ thống cảm biến phân tán lớn. [6] Cho rằng nhiều nhiệm vụ kỹ thuật hơn được yêu cầu từ các lập trình viên phụ thuộc vào quan hệ cơ sở dữ liệu để lưu trữ và quản lý dữ liệu.

6. ĐỀ XUẤT HỆ THỐNG

Mục đích của đề xuất hệ thống là thiết kế một hệ thống cơ sở dữ liệu kết hợp để lưu trữ và quản lý dữ liệu lớn. Hệ thống kết hợp của chúng tôi được tạo thành từ cơ sở dữ liệu MySQL và MongoDB là những máy chủ cơ sở dữ liệu quan hệ và NoSQL (không quan hệ) phổ biến nhất. Dữ liệu được nhóm thành các loại dữ liệu có cấu trúc và phi cấu trúc, dữ liệu có cấu trúc được chuyển vào cơ sở dữ liệu MongoDB, trong khi việc lựa chọn cơ sở dữ liệu cho dữ liệu phi cấu trúc phụ thuộc vào phương thức mà dữ liệu đang hoạt động; đây có thể là MongoDB cho phương thức kết hợp và MySQL cho phương thức SQL. Các cơ sở dữ liệu được tích hợp trong đề xuất hệ thống cũng có thể hoạt động độc lập. [8] Trình bày tổng quan về cơ sở dữ liệu NoSQL hiện có sử dụng mô hình dữ liệu, mô hình truy vấn, nhân rộng và mô hình nhất quán.

6.1 THIẾT KẾ HỆ THỐNG ĐỀ XUẤT

Thiết kế hệ thống chỉ ra các thành phần tạo nên một hệ thống. Hệ thống được đề xuất bao gồm các thành phần cơ bản sau đây: cơ sở dữ liệu MySQL, cơ sở dữ liệu MongoDB. Các thành phần này là được thảo luận chi tiết hơn và thiết kế kiến trúc của hệ thống đề xuất được đưa ra trong hình 3 hiển thị các kết nối giữa các thành phần này.

- Thành phần SQL: Chứa công cụ lưu trữ xử lý việc kiểm soát dữ liệu trong cơ sở dữ liệu MySQL. Công cụ lưu trữ được tạo thành từ một tệp nhật ký giao tác và các nhóm tệp dữ liệu có thể được chia nhỏ theo thứ bậc thành bảng tệp dữ liệu, chỉ mục, phạm vi và trang là đơn vị lưu trữ nhỏ nhất trong cơ sở dữ liệu quan hệ. Tệp nhật ký giao dịch là thành phần của công cụ lưu trữ được sử dụng để thực hiện và duy trì tính toàn vẹn dữ liệu và phục hồi trong cơ sở dữ liệu. Nó ghi lại thời điểm bắt đầu và kết thúc của mỗi thao tác cũng như mọi sửa đổi được thực hiện trên dữ liệu trong cơ sở dữ liệu.
- Thành phần MongoDB: MongoDB sử dụng bản sao để đảm bảo tính dự phòng và tính nhất quán. Luồng dữ liệu từ các đích khác nhau và ở định dạng khác nhau bị hỏng và phân tán đều thành một tập hợp các thiết bị đầu cuối không tĩnh có thể mở rộng được gọi là shard. Dữ liệu mô tả dữ liệu

khác trong cụm được lưu trong máy chủ cấu hình. Mỗi các máy chủ này chứa bản sao của tất cả siêu dữ liệu nhằm mục đích dự phòng. Khi thực hiện yêu cầu của khách hàng, nó tạo thành một trong các quy trình định tuyến được sử dụng để kiểm tra máy chủ cấu hình vị trí của yêu cầu.

6.2 THUẬT TOÁN

Trong nội bộ, hệ thống kết hợp của chúng tôi sử dụng cả thuật toán B-tree và thuật toán điền theo tỷ lệ. Cơ sở dữ liệu quan hệ (SQL) chạy chủ yếu trên thuật toán điền theo tỷ lệ. Thuật toán điền theo tỷ lệ được sử dụng bởi công cụ lưu trữ SQL để ghi dữ liệu vào các tệp cơ sở dữ liệu tùy thuộc vào kích thước của không gian trống trong mỗi tệp dữ liệu thay vì ghi vào từng tệp cho đến khi đầy rồi chuyển sang thứ hai tiếp theo. Như vậy, công cụ lưu trữ của máy chủ SQL sẽ ghi thường xuyên hơn vào các tệp có nhiều dung lượng trống hơn. Vì vậy, MongoDB được xây dựng trên thuật toán B-tree. Tuy nhiên, quy trình để xử lý từng bước dữ liệu trong hệ thống kết hợp của chúng tôi được đưa ra dưới dạng:

Bước 1: Tải dữ liệu

Bước 2: Xác định lớp của dữ liệu

Bước 3: Khởi tạo giao diện kết hợp DB

Bước 4: Kiểm tra dữ liệu: Nếu

Dữ liệu là dữ liệu cấu trúc, thì

Lưu trữ trong cơ sở dữ liệu SQL

Dữ liệu là dữ liệu phi cấu trúc, thì

Lưu trữ trong cơ sở dữ liệu MongoDB

Bước 5: Cập nhật cơ sở dữ liệu giao diện kết hợp

Bước 6: Xem, Xóa, Cập nhật, Thoát

7. THIẾT LẬP THỬ NGHIỆM VÀ KẾT QUẢ

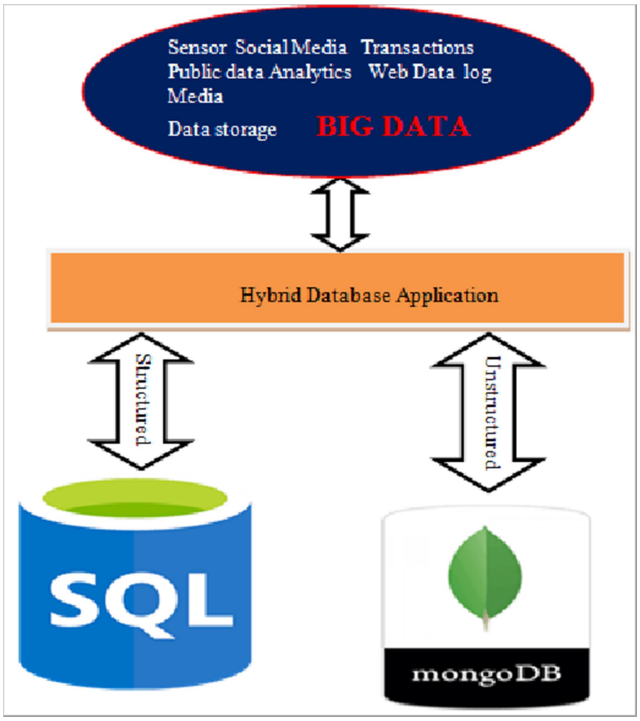
Hệ thống đề xuất được triển khai trong C# trong visual studio môi trường phát triển trực quan. MongoDB và MySQL đã được sử dụng để lưu trữ dữ liệu. Một số lớp được sử dụng trong chương trình là: Person, Symbol, PatientInfo, FileInfo, Panel, Person.

Để ứng dụng của chúng tôi hoạt động bình thường, máy chủ MongoDB trước tiên sẽ được khởi động từ dấu nhắc lệnh bằng cách điều hướng đến thư mục cài đặt và thực

hiện lệnh mongod.exe như thể hiện trong hình 4. Theo mặc định, máy chủ MongoDB sẽ bắt đầu tại cổng 27017. Để bắt đầu trình bao vỏ khách hàng, lệnh mongo.exe được thực thi trên một cửa sổ nhắc lệnh riêng biệt. Hình 5 cho thấy vỏ máy khách kết nối với Test, đây là cơ sở dữ liệu hệ thống mặc định.

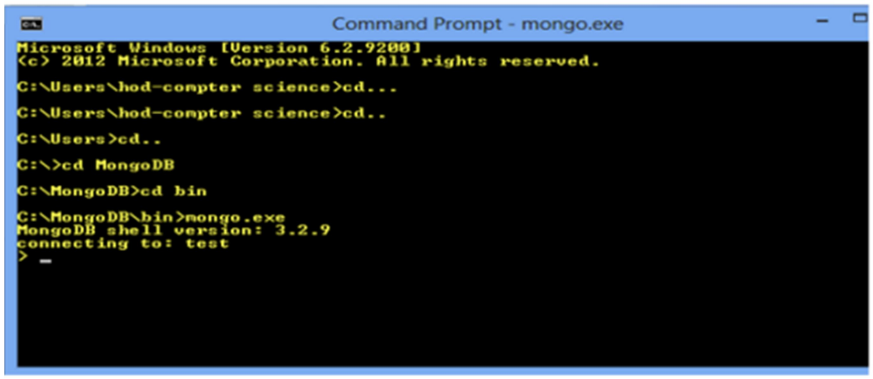
Để thực thi ứng dụng kết hợp của chúng tôi, dữ liệu đầu vào được tạo thành từ dữ liệu cấu trúc và phi cấu trúc được lưu trong đĩa cục bộ của hệ thống. Chúng tôi truy cập ứng dụng của mình bằng cách nhấp đúp vào biểu tượng ứng dụng. Tổng quan về ứng dụng của chúng tôi được hiển thị trong Hình.6 cho thấy các phương thức mà hệ thống đề xuất chúng tôi có thể hoạt động.

Khi dữ liệu lớn được tải trong hệ thống kết hợp của chúng tôi, cơ sở dữ liệu dùng để lưu trữ được xác định bởi phương thức mà ứng dụng chạy trong đó. Dữ liệu phi cấu trúc được chuyển đến cơ sở dữ liệu MongoDB (ngoại trừ khi ứng dụng ở phương thức MongoDB, trong trường hợp đó MongoDB lưu trữ cả hai dữ liệu có cấu trúc và phi cấu trúc) trong khi cơ sở dữ liệu MySQL được sử dụng để lưu trữ và quản lý cấu trúc dữ liệu.

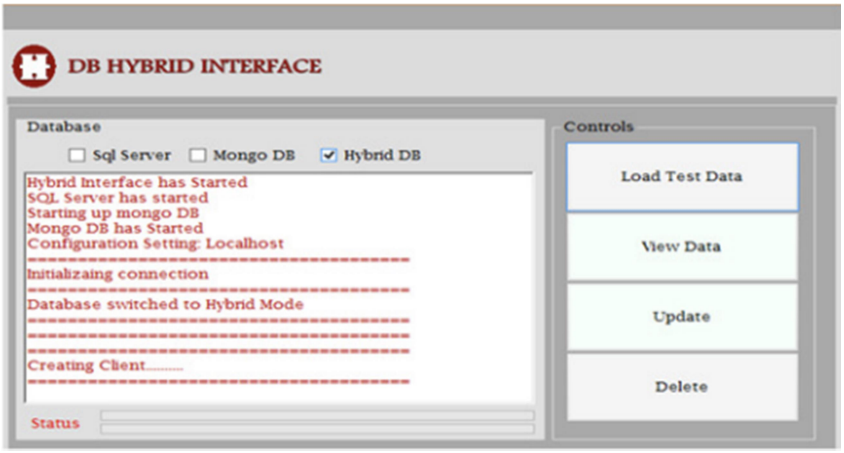


Hình 3. Thiết kế kiến trúc của hệ thống đề xuất

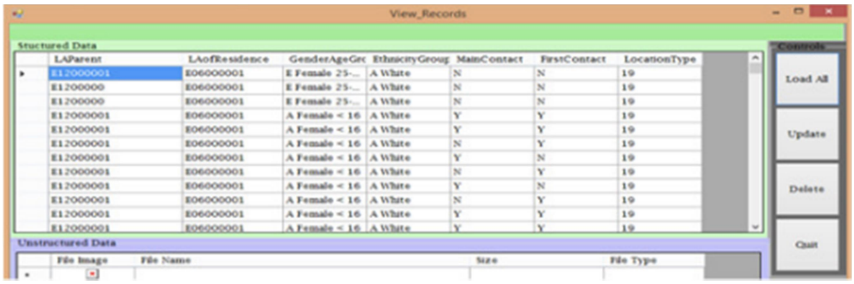
Hình 4. Khởi động MongoDB



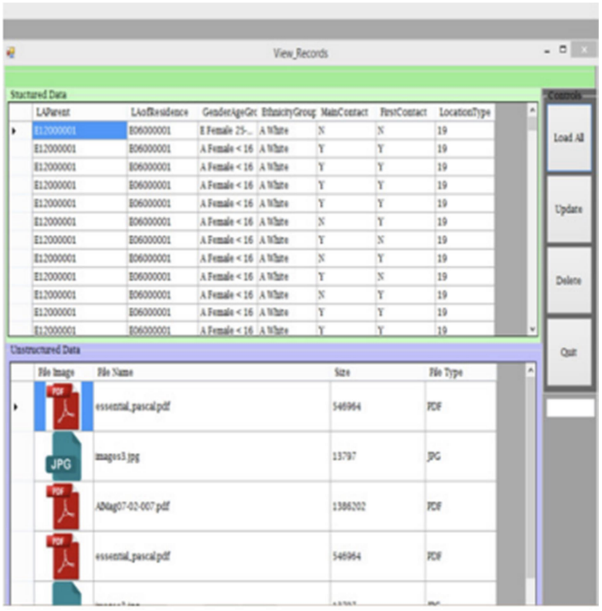
Hình 5. MongoDB kết nối với hệ thống cơ sở dữ liệu mặc định



Hình 6. Tổng quan về Giao diện DB_Hybrid



Hình 7. Xem thao tác dữ liệu trong phương thức SQL.



Hình 8: Xem hoạt động dữ liệu ở phương thức kết hợp.

8. BÀN LUẬN KẾT QUẢ

Chúng tôi đã tải dữ liệu lớn trong hệ thống cơ sở dữ liệu kết hợp của mình ở phương thức SQL, phương thức MongoDB và cả phương thức kết hợp. Cơ sở dữ liệu được sử dụng để lưu trữ và quản lý dữ liệu sẽ khác nhau tùy thuộc vào phương thức mà hệ thống chạy trong đó. Đầu ra của hệ thống cơ sở dữ liệu kết hợp đã được triển khai để lưu trữ dữ liệu lớn và quản lý được thảo luận ở đây.

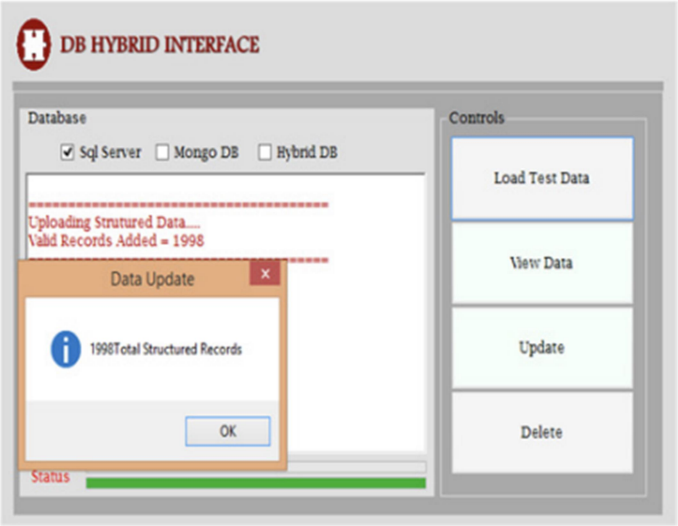
Trong phương thức SQL, dữ liệu được lưu trữ và quản lý trong cơ sở dữ liệu MySQL. Hệ thống loại bỏ dữ liệu trong dạng phi cấu trúc vì nó không thể lưu trữ được trong cơ sở dữ liệu SQL. Ảnh chụp màn việc tải lên dữ liệu được thực hiện bởi hệ thống kết hợp của chúng tôi ở chế độ SQL được đưa ra trong Hình 7, một góc nhìn của nội dung cơ sở dữ liệu được hiển thị trong Hình 8 cho thấy dữ liệu phi cấu trúc bị loại bỏ bởi ứng dụng kết hợp của chúng tôi bằng phương thức SQL. Trong phương thức MongoDB,

lưu trữ và quản lý cả dữ liệu cấu trúc và phi cấu trúc được thực hiện bằng cơ sở dữ liệu MongoDB. Điều này lưu trữ cả cấu trúc và dữ liệu phi cấu trúc trong MongoDB. Cũng trong chế độ kết hợp, dữ liệu ở dạng có cấu trúc được lưu trữ và được quản lý bằng cơ sở dữ liệu SQL trong khi MongoDB được sử dụng để lưu trữ và quản lý dữ liệu phi cấu trúc, điều này được thể hiện trong Hình 9.

Từ đầu ra trong Hình 7, Hình 8 và Hình 9, có thể thấy rằng hệ thống kết hợp của chúng tôi tích hợp các chức năng hoặc thành phần của MySQL, một cơ sở dữ liệu quan hệ phổ biến và MongoDB là một cơ sở dữ liệu phi quan hệ trong một hệ thống cơ sở dữ liệu cho phép các cơ sở dữ liệu trong hệ thống hoạt động trong sự cô lập và cả trong tích hợp.

Hệ thống đề xuất hỗ trợ các hoạt động quản lý như Cập nhật và Xóa có thể được dùng để quản lý dữ liệu trong hệ thống. Hình 10 cho thấy hoạt động cập nhật trong hệ thống cơ sở dữ liệu kết hợp của chúng tôi.

Hình 9. Hoạt động cập nhật



Hình 10. Hoạt động tải dữ liệu trong chế độ SQL.

9. KẾT LUẬN VÀ KIẾN NGHỊ

Để kết luận, chúng tôi đã đề xuất một phương pháp kết hợp cơ sở dữ liệu SQL thuộc về nhóm hệ thống cơ sở dữ liệu quan hệ và MongoDB là cơ sở dữ liệu NoSQL để lưu trữ và quản lý dữ liệu lớn. Với kết quả thu được, có thể hiểu rằng hệ thống của chúng tôi có thể được sử dụng để lưu trữ và quản lý việc loại bỏ những điểm yếu lớn trong cả hai cơ sở dữ liệu.

10. ĐÓNG GÓP CHO KIẾN THỨC

Công việc này trình bày những đóng góp sau đây cho kiến thức:

1. Phát triển hệ thống cơ sở dữ liệu kết hợp để lưu trữ và quản lý dữ liệu lớn.
2. Cách tiếp cận này cải thiện việc sử dụng cơ sở dữ liệu MongoDB để lưu trữ dữ liệu lớn.
3. Nghiên cứu củng cố khả năng của việc có tính linh hoạt và khả năng mở rộng của cơ sở dữ liệu NoSQL cũng như tính ổn định và các thành phần giao dịch của cơ sở dữ liệu quan hệ trong một hệ quản trị cơ sở dữ liệu.

TÀI LIỆU THAM KHẢO:

TÁC GIẢ:

Blessing E.James là Trợ lý sau đại học tại Khoa Khoa học Máy tính, Akwa Đại học Bang Ibom, Cô đã lấy bằng Cử nhân Công nghệ về Toán học và Khoa học Máy tính, 2010 từ Đại học Công nghệ Liên bang Owerri, Bang

Imo, Nigeria và bằng Thạc sĩ Khoa học (2007) của Đại học Port Harcourt, Nigeria. Mối quan tâm của cô ấy là Dữ liệu lớn, Hệ thống quản lý cơ sở dữ liệu, Kiến thức cơ sở dữ liệu Quản lý, Khai thác dữ liệu và Học máy.

Hoàng tử Oghenekaro ASAGBA có bằng Cử nhân Khoa học Máy tính tại Đại học Nigeria, Nsukka, vào năm 1991 với Hạng Nhì (Hons). Anh ấy đã có bằng Thạc sĩ bằng Khoa học Máy tính tại Đại học Benin vào tháng 4 năm 1998 và bằng Tiến sĩ về Khoa học Máy tính tại Đại học Port Harcourt vào tháng 3 năm 2009. Asagba là một người đam mê nghiên cứu với hơn năm mươi bài báo nghiên cứu đã xuất bản trên các tạp chí có uy tín, cả trong nước và quốc tế. Asagba sở hữu hơn 20 năm nghiên cứu phong phú hợp lý / kinh nghiệm giảng dạy ở cấp Đại học. Ông là Giáo sư / Học giả thỉnh giảng cho một số các trường đại học ở Nigeria. Lĩnh vực nghiên cứu của anh bao gồm: Máy tính và An ninh thông tin, Mạng Phân tích, Kỹ thuật phần mềm, Hệ thống quản lý cơ sở dữ liệu, Mô hình hóa và Lập trình. Anh ấy là một thành viên của Hiệp hội Máy tính Nigeria (NCS) và Hội đồng Đăng ký Chuyên nghiệp Máy tính của Nigeria (CPN).