HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG

Vương Duy Thanh

NGHIÊN CỨU ỨNG DỤNG AI XÂY DỰNG THUẬT TOÁN DỰ BÁO CÁC TÁC VỤ TRÊN ĐÁM MÂY NHẰM NÂNG CAO HIỆU QUẢ CÂN BẰNG TẢI

LUẬN VĂN THẠC SỸ KỸ THUẬT

(Theo định hướng ứng dụng)

TP. HỒ CHÍ MINH – NĂM 2022

HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG

Vương Duy Thanh

NGHIÊN CỨU ỨNG DỤNG AI XÂY DỰNG THUẬT TOÁN DỰ BÁO CÁC TÁC VỤ TRÊN ĐÁM MÂY NHẰM NÂNG CAO HIỆU QUẢ CÂN BẰNG TẢI

CHUYÊN NGÀNH:HỆ THỐNG THÔNG TIN

MÃ SỐ: 8.48.01.04

LUẬN VĂN THẠC SỸ KỸ THUẬT

(Theo định hướng ứng dụng)

NGƯỜI HƯỚNG DẪN KHOA HỌC:

PGS.TS. TRẦN CÔNG HÙNG

TP. HỒ CHÍ MINH – NĂM 2022

LỜI CAM ĐOAN

Tôi cam đoan rằng luận văn: ***"Nghiên cứu ứng dụng AI xây dựng thuật toán dự báo các tác vụ trên đám mây nhằm nâng cao hiệu quả cân bằng tải"*** là công trình nghiên cứu của chính tôi.

Tôi cam đoan các số liệu, kết quả nêu trong luận văn là trung thực và chưa từng được ai công bố trong bất kỳ công trình nào khác.

Không có sản phẩm/nghiên cứu nào của người khác được sử dụng trong luận văn này mà không được trích dẫn theo đúng quy định.

TP. Hồ Chí Minh, ngày 25 tháng 01 năm 2022

Học viên thực hiện luận văn

Vương Duy Thanh

LỜI CẢM ƠN

Trong suốt quá trình học tập và nghiên cứu thực hiện luận văn, ngoài nỗ lực của bản thân, tôi đã nhận được sự hướng dẫn nhiệt tình quý báu của quý Thầy Cô, itrọng và biết ơn sâu sắc, tôi xin gửi lời cảm ơn chân thành tới:

Ban Giám Đốc, Phòng đào tạo sau đại học và quý Thầy Cô đã tạo mọi điều kiện thuận lợi giúp tôi hoàn thành luận văn.

Tôi xin chân thành cảm ơn Thầy **PGS.TS Trần Công Hùng**, người thầy kính yêu đã hết lòng giúp đỡ, hướng dẫn, động viên, tạo điều kiện cho tôi trong suốt quá trình thực hiện và hoàn thành luận văn.

Tôi xin chân thành cảm ơn gia đình, bạn bè, đồng nghiệp trong cơ quan đã động viên, hỗ trợ tôi trong lúc khó khăn để tôi có thể học tập và hoàn thành luận văn.

Mặc dù đã có nhiều cố gắng, nỗ lực, nhưng do thời gian và kinh nghiệm nghiên cứu khoa học còn hạn chế nên không thể tránh khỏi những thiếu sót. Tôi rất mong nhận được sự góp ý của quý Thầy Cô cùng bạn bè đồng nghiệp để kiến thức của tôi ngày một hoàn thiện hơn.

Xin chân thành cảm ơn!

TP. Hồ Chí Minh, ngày 25 tháng 01 năm 2022

Học viên thực hiện luận văn

Vương Duy Thanh

DANH SÁCH HÌNH VẼ

DANH SÁCH BẢNG

DANH MỤC CÁC THUẬT NGỮ, CÁC CHỮ VIẾT TẮT

Viết tắt	Tiếng Anh	Tiếng Việt
CC	Cloud Computing	Điện toán đám mây
LB	Load Balancing	Cân bằng tải
Cloud	Cloud computing environment	Môi trường điện toán đám mây
AI	Artificial Intelligence	Trí tuệ nhân tạo
ML	Machine Learning	Học máy
DL	Deep Learning	Học sâu

MỤC LỤC

MỞ ĐẦU

1. Tính cấp thiết của đề tài

Trong thời đại ngày nay, Công nghệ thông tin và truyền thông ngày càng phát triển, đòi hỏi nhu cầu về xử lý thông tin ngày càng cao. Khi đó, nhu cầu về khả năng lưu trữ được một lượng dữ liệu to lớn là vô cùng cấp thiết. Sự phát triển không ngừng của nền kinh tế thế giới đã đẩy các doanh nghiệp, các tập đoàn lớn vào tình thế buộc phải thay đổi. Lúc này, cần có một giải pháp giúp họ lưu trữ được một khối lượng khổng lồ các dữ liệu liên quan đến công việc kinh doanh của họ.

Bên cạnh đó, cũng phải có các giải pháp nhằm thỏa mãn các yêu cầu hàng đầu của người dùng như: ***đơn giản, an toàn và dễ sử dụng*** để phục vụ cho công việc của mình.

Từ đó, khái niệm dịch vụ đã trở thành một khái niệm quen thuộc với mọi người. Tất cả đều được chuyển đổi thành dịch vụ khi người dùng không muốn tự mình phải thực hiện tất cả mọi việc. Họ muốn những gì đơn giản nhất, dễ sử dụng nhất và không phải liên tục quản lý nó khi không có nhu cầu sử dụng.

Vì vậy, giải pháp để đáp ứng tất cả các nhu cầu nói trên trong nhiều năm qua đã xuất hiện. Đó chính là Điện toán đám mây [1] (Cloud computing).

Một trung tâm dữ liệu đám mây quy mô lớn cần cung cấp độ tin cậy và tính sẵn sàng của dịch vụ cao với xác suất xảy ra lỗi thấp. Tuy nhiên, các trung tâm dữ liệu đám mây quy mô lớn hiện nay vẫn phải đối mặt với tỷ lệ hỏng hóc cao do nhiều nguyên nhân như lỗi phần cứng và phần mềm, thường dẫn đến lỗi tác vụ và công việc. Những lỗi như vậy có thể làm giảm nghiêm trọng độ tin cậy của các dịch vụ đám mây và cũng chiếm một lượng lớn tài nguyên để khôi phục dịch vụ từ các lỗi. Do đó, điều quan trọng là phải dự đoán các tác vụ tiếp theo sẽ cần phục vụ trên đám mây. Từ đó, đưa ra quyết định phân bổ tác vụ nào phù hợp nhất cho tài nguyên nào để tránh lãng phí không mong muốn.

Nhiều phương pháp dựa trên trí tuệ nhân tạo (AI) đã được đề xuất để dự báo các tác vụ nhằm nâng cao hiệu quả cân bằng tải trên đám mây. Tuy nhiên, để cải thiện

hơn nữa độ chính xác dự đoán các tác vụ của các phương pháp dựa trên máy học và học sâu trước đây, ứng dụng AI xây dựng thuật toán dự báo các tác vụ trên đám mây nhằm nâng cao hiệu quả cân bằng tải được đề xuất trong luận văn này, đề tài như sau: *"Nghiên cứu ứng dụng AI xây dựng thuật toán dự báo các tác vụ trên đám mây nhằm nâng cao hiệu quả cân bằng tải"*.

Với việc dự báo được các tác vụ và phân bổ phù hợp cho các tài nguyên, bộ cân bằng tải sẽ làm việc hiệu quả hơn. Ngoài ra, hiệu quả kinh doanh của nhà cung cấp dịch vụ đám mây cũng được cải thiện bằng cách giảm sự từ chối về số lượng công việc được gửi.

Luận văn bao gồm: Phần mở đầu, nội dung gồm bốn chương và phần kết luận.

2. Tổng quan về vấn đề nghiên cứu

Theo tài liệu [2], điện toán đám mây (Cloud Computing), còn gọi là điện toán máy chủ ảo, là mô hình điện toán sử dụng các công nghệ máy tính và phát triển dựa vào mạng Internet. Cụ thể hơn là trong mô hình điện toán đám mây, tất cả các tài nguyên, thông tin và software đều được chia sẻ và cung cấp cho các máy tính, thiết bị cùng với người dùng dưới dạng dịch vụ trên nền tảng một hạ tầng mạng công cộng (thường là mạng Internet). Người dùng cuối truy cập và sử dụng các ứng dụng đám mây thông qua các ứng dụng như trình duyệt web, ứng dụng mobile hoặc máy tính cá nhân.

Thuật ngữ Cloud Computing ra đời để khái quát lại các hướng đi của cơ sở hạ tầng thông tin vốn đã và đang diễn ra. Quan niệm này có thể được diễn giải một cách đơn giản như: các nguồn điện toán khổng lồ (phần mềm, dịch vụ) sẽ nằm tại các máy chủ ảo (đám mây) trên Internet thay vì trong máy tính gia đình hay văn phòng (trên mặt đất) để mọi người kết nối và sử dụng mỗi khi họ cần. Với các dịch vụ sẵn có trên Internet, doanh nghiệp không còn phải mua và duy trì hàng trăm, thậm chí là hàng nghìn máy tính và phần mềm. Họ chỉ cần tập trung vào kinh doanh lĩnh vực riêng của mình bởi đã có người khác lo cơ sở hạ tầng và công nghệ thông tin thay họ. Google là một trong những công ty ủng hộ điện toán đám mây tích cực nhất bởi hoạt động kinh doanh của họ dựa trên việc phân phối các cloud (virtual server). Đa số người

dùng Internet đã tiếp cận những dịch vụ đám mây phổ thông như email, album ảnh và bản đồ số.

Có 3 mô hình triển khai điện toán đám mây [2] chính là public (công cộng), private (riêng) và hybrid ("lai" giữa đám mây công cộng và riêng). Đám mây công cộng là mô hình mà các nhà cung cấp đám mây cung cấp các dịch vụ như tài nguyên, platform hay các ứng dụng lưu trữ trên đám mây và public ra bên ngoài. Các dịch vụ trên public cloud có thể miễn phí hoặc có phí. Đám mây riêng thì các dịch vụ được cung cấp nội bộ và thường là các dịch vụ kinh doanh, mục đích nhắm đến là cung cấp dịch vụ cho một nhóm người và đứng đằng sau firewall. Đám mây "lai" là môi trường đám mây mà kết hợp cung cấp các dịch vụ công cộng và riêng. Ngoài ra còn có "community cloud" là đám mây giữa các nhà cung cấp dịch vụ đám mây. Về mô hình cung cấp dịch vụ có 3 loại chính là IaaS – cung cấp hạ tầng như một service, PaaS – cung cấp Platform như một service và SaaS – cung cấp software như một service.

2.1. Lợi ích của điện toán đám mây

Giúp tiết kiệm chi phí: Vì không cần trung tâm dữ liệu tại chỗ nên không cần phải lắp đặt máy chủ, phần cứng, phần mềm...

Truy cập tức thì mọi lúc mọi nơi: Người dùng có thể truy cập vào tài khoản ngay khi đang di chuyển, thông qua bất cứ thiết bị nào, bất kỳ nơi nào trên thế giới miễn là thiết bị đó đang được kết nối với mạng Internet.

Khả năng biến đổi vô tận: Người dùng có thể tùy chọn tạo mô hình đám mây riêng, công cộng hoặc kết hợp (hybrid) hay tùy chọn để quyết định vị trí của trung tâm dữ liệu ảo.

Khả năng thích ứng: Có thể chuyển đổi từ mạng riêng sang mạng kết hợp hoặc tạm thời mở rộng dung lượng lưu trữ thì điện toán đám mây có thể làm tất cả một cách suôn sẻ, đáp ứng mọi nhu cầu người dùng.

Hợp tác bền vững, không xáo trộn: Các file được tập trung lưu trữ cố định và nhất quán, tránh được tình trạng bị mất phương hướng khi đang theo dõi dự án.

Bảo mật dữ liệu: Các nhà cung cấp dịch vụ phải luôn đảm bảo rằng hệ thống bảo vệ được cập nhật liên tục và cùng lúc với tất cả các tính năng mới thông qua việc kiểm định chặt chẽ. Tất cả các hoạt động trên đám mây sẽ được bên thứ ba giám sát và kiểm tra thường xuyên để đảm bảo chuẩn an toàn được đáp ứng.

2.2. Các mô hình dịch vụ [3]

Mô hình dịch vụ của điện toán đám mây được các nhà cung cấp dịch vụ chia thành 3 loại lớn:

2.2.1. Cơ sở hạ tầng như một dịch vụ (Infrastructure as a Service - IaaS)

IaaS là một dạng dịch vụ trả tiền theo định mức (pay-per-use) hay chỉ trả tiền cho những gì sử dụng. Dịch vụ này cho phép người sử dụng truy cập vào cơ sở hạ tầng máy tính từ xa. IaaS bao gồm các máy chủ server, storage lưu trữ và các bảo vệ an ninh nâng cao. Tất cả những yếu tố này giúp cho IaaS trở thành nguồn lực vô giá cho cả doanh nghiệp lẫn cá nhân.

2.2.2. Nền tảng như một dịch vụ (Platform as a Service - PaaS)

Mô hình hệ thống của PaaS cũng khá tương đồng với IaaS nhưng còn có thêm những công cụ phát triển doanh nghiệp thông minh (BI), middleware, các tool quản lý dữ liệu cũng như các hỗ trợ khác giúp phát triển và triển khai ứng dụng.

2.2.3. Phần mềm như một dịch vụ (Software as a Service - SaaS)

SaaS là một mô hình nổi trội trong điện toán đám mây, cho phép người dùng tận dụng các ứng dụng nền tảng đám mây thông qua Internet. Mô hình dịch vụ này mang đến khả năng truy cập tiện lợi hơn ở mọi góc độ thời gian và vị trí. Không chỉ vậy, mô hình còn giúp doanh nghiệp giảm thiểu phần lớn chi phí ban đầu nhờ loại bỏ được các nhu cầu về server hay các giải pháp backup đắt tiền.

2.3. Tổng quan về cân bằng tải

2.3.1. Cân bằng tải [1]

Cân bằng tải là một phương pháp quan trọng trong điện toán đám mây giúp các máy chủ hoạt động hiệu quả thông qua việc phân phối tài nguyên một cách đồng đều, giảm hoặc tránh trình trạng tắc nghẽn. Khi một máy chủ gặp sự cố, cân bằng tải sẽ chỉ đạo phân phối công việc của máy chủ đó cho các máy chủ còn lại. Cho đến khi

sự cố được giải quyết, công việc của máy chủ đó lại diễn ra bình thường hoặc thậm chí còn xử lý thêm một phần công việc của máy chủ khác bị sự cố. Nó làm cho hệ thống hoạt động liên tục giúp tránh mất mát chi phí do sự ngưng hoạt động. Có thể nói cân bằng tải là việc phân bổ đồng đều lưu lượng truy cập giữa các máy chủ có cùng chức năng trong cùng một hệ thống giúp cho hệ thống giảm thiểu tối đa tình trạng một máy chủ bị quá tải và hệ thống ngưng hoạt động. Đồng thời, sử dụng tốt các nguồn tài nguyên, giảm thời gian chờ và đáp ứng yêu cầu truy cập nhanh hơn.

Quá trình cân băng tải cũng xử lý hiện tượng tắc nghẽn xảy ra do mất cân bằng tải. Nó xử lý lỗi khi có một trong các thành phần của các dịch vụ bị lỗi trong khi các dịch vụ khác vẫn tiến hành giao tiếp thông tin với nhau. Trường hợp một máy chủ bị nghẽn hoặc ngừng hoạt động, bộ cân bằng tải sẽ chuyển các công việc của máy chủ đó đến các máy chủ còn lại. Nếu thêm máy chủ mới vào hệ thống thì cân bằng tải sẽ tự động chia sẻ lưu lượng của các máy chủ khác đến nó, giảm bớt tải cho các máy chủ khác, tối ưu hoạt động của máy chủ mới thêm vào. Bộ cân bằng tải có nhiều chức năng tối ưu nhưng cơ bản nhất gồm:

Chặn lưu lượng mạng đến một trang web (hoặc giao diện ứng dụng giao tiếp người dùng khác). Cân bằng tải là đối tượng đầu tiên nhận các yêu cầu trước khi chia tải nên nó phải đảm bảo máy chủ nào được xử lý chứ không phải mọi yêu cầu đến đều do 1 hoặc vài máy chủ cố định xử lý. Do đó, chức năng này là tiên quyết của bộ cân bằng tải.

Phân tải đến các máy chủ để xử lý. Đây là chức năng chính của cân bằng tải.

Kiểm soát sự liên lạc của các máy chủ với bộ cân bằng tải và giữa các máy chủ với nhau. Chức năng này làm cho việc đồng bộ tải được tối ưu và không xảy ra lỗi khi một máy chủ bị “chết”.

Cung cấp khả năng dự phòng bằng cách sử dụng nhiều hơn một kịch bản failover.

2.3.2. Cân bằng tải trên điện toán đám mây [2]

Cân bằng tải trong điện toán đám mây cung cấp giải pháp cho các vấn đề khác nhau về thiết lập và sử dụng tài nguyên trong môi trường điện toán đám mây. Cân

bằng tải phải đảm bảo đồng thời hai nhiệm vụ chủ yếu. Một là việc cung cấp tài nguyên hoặc phân bổ nguồn lực sao cho tối ưu hóa việc sử dụng tài nguyên, tối đa hóa thông lượng, giảm thiểu thời gian đáp ứng và tránh quá tải. Hai là lập lịch công việc trong môi trường phân tán. Trích lập dự phòng có hiệu quả các nguồn lực và lập kế hoạch các nguồn tài nguyên cũng như nhiệm vụ để đảm bảo:

Cấp phát nhanh tài nguyên khi có yêu cầu.

Sử dụng hiệu quả nguồn tài nguyên kể cả trong khi điều kiện tải cao

Tiết kiệm năng lượng khi điều kiện tải thấp.

Chi phí sử dụng tài nguyên thấp.

Những điều quan trọng cần xem xét trong khi phát triển thuật toán cân bằng tải đó là: ước tính tải, so sánh, tạo sự ổn định của hệ thống, hiệu suất của hệ thống, tương tác giữa các nút và tính chất của các công việc được chuyển giao. Tải này có thể được xem xét trong các thuật ngữ tải của CPU, số lượng bộ nhớ sử dụng, độ trễ hoặc tải lên mạng. Khi một khối lượng tải cho trước được đệ trình cho bất kì cụm nút, tải cho trước này có thể được thực thi hiệu quả nếu nguồn tài nguyên sẵn có được sử dụng hiệu quả. Do đó phải có một cơ chế để lựa chọn các nút có các nguồn tài nguyên. Lập lịch là một thành phần hay cơ chế chịu trách nhiệm chọn một nút hay cụm nút. Cơ chế này sẽ xem xét trạng thái cân bằng tải. Trong thực tế, cân bằng tải trên điện toán đám mây bị ảnh hưởng bởi ba yếu tố chính [3]: Môi trường muốn cân bằng tải, Bản chất của tải trọng và Các công cụ cân bằng tải có sẵn.

Môi trường muốn cân bằng tải được xác định bao gồm kiến trúc của bộ xử lý thuộc về hệ thống, loại tài nguyên sẽ được chia sẻ giữa các bộ xử lý, hình thức và loại kết nối giữa các bộ xử lý.

Bản chất của tải trọng: Các yếu tố này có thể được phân biệt thực tế bằng cách xác định bản chất của hệ thống trong tính không đồng nhất, phân bổ tài nguyên hoặc phương tiện truyền dữ liệu. Trong trường hợp tải công việc, các tác vụ nói chung có xu hướng được phân loại thành các ràng buộc I/O, giới hạn CPU hoặc các tác vụ hỗn hợp.

Các công cụ cân bằng tải đại diện cho phần chính và quan trọng của các hệ thống cân bằng tải. Nó có thể được trình bày dưới dạng các thủ tục và chương trình chịu trách nhiệm cân bằng tải. Vì lý do đó, hai công cụ chính cần thiết là: công cụ thông tin và quy trình. Các công cụ thông tin xác định vị trí của quy trình trong khi công cụ xử lý chuyển các quy trình giữa các bộ xử lý trong môi trường và cung cấp quyền truy cập vào các tài nguyên khác nhau trong hệ thống.

2.4. Một số công trình nghiên cứu liên quan đến cân bằng tải

Theo tài liệu [4] được công bố trong Hội Thảo Quốc Gia về Điện Tử, Truyền Thông và Công Nghệ Thông Tin (ECIT 2015). Tác giả đã cho ta cái nhìn tổng quan về việc phân phối tải và các thuật toán để tránh các hiện tượng quá tải. Công nghệ điện toán đám mây cho phép người dùng truy cập đến nguồn tài nguyên mạng được phân tán khắp nơi, các yêu cầu nếu tăng nhanh sẽ dễ gây hiện tượng tắc nghẽn hoặc thời gian đáp ứng chậm. Phương pháp trong bài báo nhằm tránh hiện tượng quá tải bằng cơ chế sắp xếp phần trăm mức độ sử dụng của các máy ảo theo thứ tự nhất định để phục vụ cho các yêu cầu truy cập tài nguyên.

Theo tài liệu [3] được tác giả công bố trên Tạp chí khoa học công nghệ thông tin và truyền thông, Số 4 (CS.01) 2018. Tác giả đã chỉ ra rằng cân bằng tải là thách thức lớn trong điện toán đám mây. Thông qua các nút mạng, cân bằng tải giúp phân phối tải để không có nút nào bị quá tải. Cân bằng tải còn giúp người dùng truy cập máy chủ web để hạn chế sự cố quá tải, dữ liệu tải xuống chậm, thời gian chờ hoặc thời gian đáp ứng dài. Điều đó giúp cho việc phân bổ tài nguyên tốt hơn, linh hoạt hơn và có thể mở rộng thêm để tránh tắc nghẽn.

Từ những năm 2014, trên điện toán đám mây, việc cân bằng tải được sử dụng để phân phối các tải hoạt động lớn sang các tải hoạt động ít hơn để nâng cao hiệu suất làm việc và tận dụng tối đa tài nguyên của cloud. Trong môi trường đám mây, cân bằng tải đòi hỏi phân bổ lại các tải đang hoạt động liên tục giữa tất cả các nốt:

- Cân bằng tải giúp cho đám mây đạt được việc phân bổ tài nguyên tốt nhất, hỗ trợ tính linh động và khả năng mở rộng cao để tránh bị hiện tượng cổ chai.

- Cân bằng tải là kỹ thuật phân phối trên mạng các nguồn tài nguyên bằng cách cung cấp các luồng tối ưu với thời gian đáp ứng thấp nhất. Cân bằng tải sẽ phân chia thông lượng giữa các máy chủ, từ đó dữ liệu có thể được gửi và nhận mà không bị trì hoãn.

- Trong môi trường đám mây, có rất nhiều thuật toán giảm tải lưu lượng cũng như phân phối lại lưu lượng. Đa số các thuật toán này đều có thể ứng dụng vào môi trường cloud với các trường hợp cụ thể khác nhau. Trong môi trường điện toán đám mây, các thuật toán cân bằng tải có thể được chia thành 02 nhóm chính: Nhóm thứ nhất là BMHA (thuật toán phân bổ Batch mode Heuristic hay tạm dịch là thuật toán phân bổ Heuristic theo cơ chế từng đợt). Nhóm thứ hai là thuật toán Online Mode Heuristic (tạm dịch là phân bổ Heuristic theo cơ chế trực tuyến). Các công việc trong BMHA được phối hợp với nhau khi dữ liệu được gửi tới hệ thống. Thuật toán BMHA sẽ thực hiện sau một khoảng thời gian cố định.

- Một ví dụ của nhóm thuật toán BMHA là thuật toán First Come First Served (FCFS), thuật toán Round Robin (RR), thuật toán Min Min và thuật toán Max Min. Đối với nhóm thuật toán Online Mode Heuristic, tất cả các công việc sẽ thực hiện khi dữ liệu đến hệ thống. Môi trường cloud là một hệ thống không đồng nhất và tốc độ xử lý của các bộ xử lý sẽ thay đổi khác nhau một cách nhanh chóng và dễ dàng. Nhóm thuật toán Online có vẻ như thích hợp hơn và cho kết quả tốt hơn môi trường cloud.

- Việc dự đoán và ước lượng được tải cần thiết là vô cùng quan trọng. Cần phải so sánh với tất cả các tải, tính ổn định tương đối của các hệ thống khác nhau, hiệu suất làm việc của các hệ thống mục tiêu, tương tác giữa các nốt và các công việc cần làm để truyền đi trong quá trình xây dựng một thuật toán cân bằng tải. Vấn đề quan trọng nữa đó là lựa chọn các nốt mà trong đó có nhiều loại khác nhau. Tải CPU, dung lượng bộ nhớ tổng hợp lại để tính toán ra tải chung toàn máy.

3. Mục đích nghiên cứu

- Mục tiêu chính: Nghiên cứu ứng dụng Machine Learning xây dựng thuật toán dự báo các tác vụ trên đám mây nhằm nâng cao hiệu quả cân bằng tải.
- Từ mục tiêu chính trên, luận văn sẽ dự kiến các kết quả đạt được như sau:

- Tìm hiểu tổng quan về điện toán đám mây.
- Tìm hiểu về các thuật toán trên điện toán đám mây.
- Tìm hiểu về các tác vụ trên đám mây, mô hình các tác vụ, ưu nhược điểm của các mô hình và so sánh ưu nhược điểm với các thuật toán đã được công bố.
- Tìm hiểu về nguyên nhân dẫn đến lỗi các tác vụ.
- Tìm hiểu dự báo các tác vụ và khôi phục dịch vụ từ các lỗi.
- Đề xuất thuật toán phân lớp và ứng dụng Machine Learning để cải thiện thuật toán giúp việc cân bằng tải hiệu quả hơn.
- Trên cơ sở lý thuyết đã nghiên cứu, luận văn đề xuất thuật toán dự báo tác vụ nhằm nâng cao hiệu quả cân bằng tải trên điện toán đám mây. Mô phỏng và thực nghiệm thuật toán đã đề xuất.

4. Đối tượng và phạm vi nghiên cứu

- Đối tượng nghiên cứu
  - Đối tượng nghiên cứu chính là tác vụ trên điện toán đám mây.
  - Nghiên cứu các thuật toán dự báo áp dụng vào dự báo tác vụ.
- Phạm vi nghiên cứu.
  - Xây dựng mô hình mô phỏng đám mây ở mức độ nhỏ: khoảng 10 – 20 máy ảo (Có thể sử dụng máy thật tuy nhiên để tiết kiệm chi phí nên trong đề cương này, thuật toán sẽ mô phỏng và áp dụng trên máy ảo).
  - Độ phức tạp trên mỗi máy ảo chỉ ở mức độ thấp: khoảng 1 – 4 ứng dụng trên các máy ảo đó.
  - Yêu cầu (Request) gửi về máy chủ cũng đơn giản, dung lượng dữ liệu gửi về nhỏ: Khoảng dưới 1 Mb.

5. Phương pháp nghiên cứu

- Phương pháp luận:

  Dựa trên cơ sở là các lý thuyết về điện toán đám mây, Task Prediction, cân bằng tải trên cloud.
- Phương pháp đánh giá dựa trên cơ sở toán học:

Trên cơ sở các lý thuyết về điện toán đám mây. Đề xuất ra thuật toán để dự báo tác vụ trên đám mây dựa trên các thuật toán dự báo (thống kê, AI, …). Chứng minh thuật toán và đánh giá hiệu quả của thuật toán.

- Phương pháp đánh giá bằng mô phỏng thực nghiệm

  Xây dựng mô hình mô phỏng và thực nghiệm thuật toán đã đề xuất.

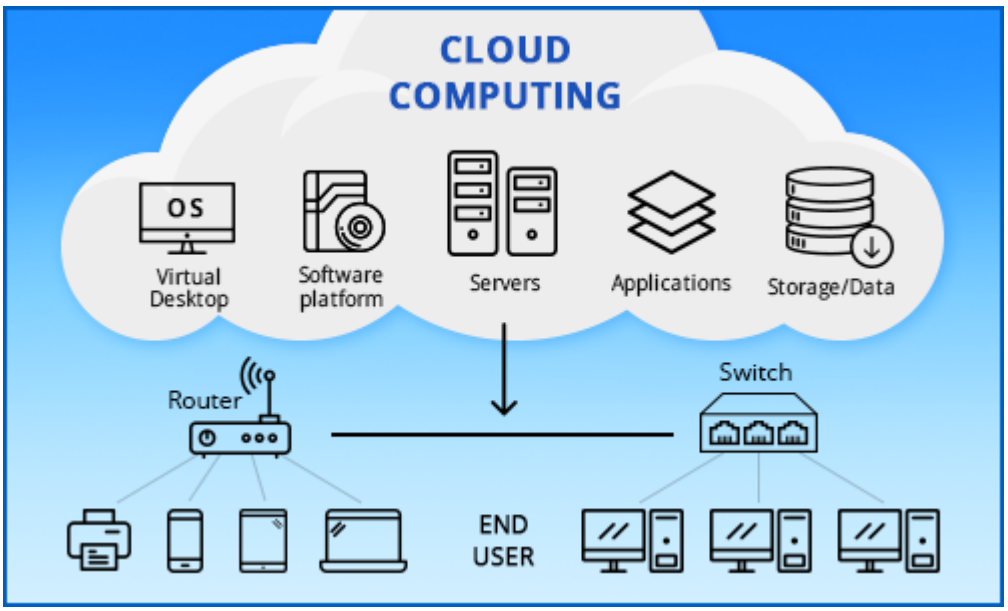
CHƯƠNG 1: TỔNG QUAN VỀ HỆ THỐNG DỰ BÁO CÁC TÁC VỤ TRÊN ĐÁM MÂY NHẰM NÂNG CAO HIỆU QUẢ CÂN BẰNG TẢI

1.1. Tổng quan về điện toán đám mây

Điện toán đám mây (cloud computing): [5], [6] hay còn gọi là điện toán máy chủ ảo nơi các tính toán được "định hướng dịch vụ" và phát triển dựa vào Internet. Cụ thể hơn, trong mô hình điện toán đám mây, tất cả các tài nguyên, thông tin, và software đều được chia sẻ và cung cấp cho các máy tính, thiết bị, người dùng dưới dạng dịch vụ trên nền tảng một hạ tầng mạng công cộng (thường là mạng Internet). Các user sử dụng dịch vụ như cơ sở dữ liệu, website, lưu trữ,… trong mô hình cloud computing không cần quan tâm đến vị trí địa lý cũng như các thông tin khác của hệ thống mạng đám mây - "điện toán đám mây trong suốt đối với người dùng". Người dùng cuối truy cập và sử dụng các ứng dụng đám mây thông qua các ứng dụng như trình duyệt web, các ứng dụng mobile hay máy tính cá nhân thông thường. Hiệu năng sử dụng phía người dùng cuối được cải thiện khi các phần mềm chuyên dụng, các cơ sở dữ liệu được lưu trữ và cài đặt trên hệ thống máy chủ ảo trong môi trường điện toán đám mây trên nền của "data center". "Data center" là thuật ngữ chỉ khu vực chứa server và các thiết bị lưu trữ, bao gồm nguồn điện và các thiết bị khác như rack, cables… luôn sẵn sàng và có độ ổn định cao. Ngoài ra còn bao gồm các tiêu chí khác như: tính module hóa cao, khả năng mở rộng dễ dàng, nguồn và làm mát, hỗ trợ hợp nhất server và lưu trữ mật độ cao.

Có 3 mô hình [7], [8] triển khai điện toán đám mây chính là public (công cộng), private (riêng) và hybrid ("lai" giữa đám mây công cộng và riêng). Đám mây công cộng là mô hình đám mây mà trên đó, các nhà cung cấp đám mây cung cấp các dịch vụ như tài nguyên, platform hay các ứng dụng lưu trữ trên đám mây và public ra bên ngoài. Các dịch vụ trên public cloud có thể miễn phí hoặc có tính phí. Đám mây riêng thì các dịch vụ được cung cấp nội bộ và thường là các dịch vụ kinh doanh, mục đích nhắm đến cung cấp dịch vụ cho một nhóm người và đứng đằng sau firewall. Đám mây "lai" là môi trường đám mây mà kết hợp cung cấp các dịch vụ công cộng và

riêng. Ngoài ra còn có "community cloud" là đám mây giữa các nhà cung cấp dịch vụ đám mây. Về mô hình cung cấp dịch vụ có 3 loại chính là IaaS – cung cấp hạ tầng như một service, PaaS – cung cấp Platform như một service và SaaS – cung cấp software như một service.



Hình 1.1: Mô hình điện toán đám mây [9]

Điện toán đám mây là xu hướng công nghệ nổi bật trên thế giới trong những năm gần đây, với những đột phá về chất lượng, quy mô triển khai và loại hình dịch vụ. Thường có nhiều nhà cung cấp nổi tiếng như Google, Amazon, Microsoft ...

Điện toán đám mây là một mô hình điện toán trong đó tất cả các giải pháp công nghệ thông tin được cung cấp dưới dạng dịch vụ qua Internet. Điều này giúp người dùng không phải đầu tư vào nhân viên, công nghệ và cơ sở hạ tầng để triển khai hệ thống. Từ đó, điện toán đám mây giúp giảm thiểu chi phí và thời gian triển khai, cho phép người dùng nền tảng điện toán đám mây tập trung nguồn lực tối đa cho công việc chuyên môn của mình. Lợi ích của điện toán đám mây mang lại không chỉ gói gọn trong phạm vi người sử dụng nền tảng điện toán đám mây mà còn từ phía các nhà cung cấp dịch vụ điện toán. Điện toán đám mây (Cloud Computing) [10] là xu hướng phát triển mạnh nhất hiện nay. Nó kế thừa các mạng lưới trước đây cũng như các khái niệm máy tính phân tán để tích hợp các tài nguyên máy tính, lưu trữ,

nền tảng cùng các dịch vụ khác theo nhu cầu một cách thuận tiện và nhanh chóng. Đồng thời, nó cũng cho phép kết thúc sử dụng dịch vụ, giải phóng tài nguyên dễ dàng và giảm thiểu các giao tiếp với nhà cung cấp. Theo đó, mô hình chính là cho phép sử dụng dịch vụ theo yêu cầu (ondemand service). Trong khi đó, nó cung cấp quyền truy cập dịch vụ (truy cập mạng rộng) qua nhiều loại mạng, từ máy tính để bàn, máy tính xách tay đến thiết bị di động. Tài nguyên máy tính động (gộp tài nguyên cho nhiều người thuê) phục vụ nhiều người, khả năng tính toán linh hoạt và đáp ứng nhanh từ nhu cầu thấp đến cao (co giãn nhanh).

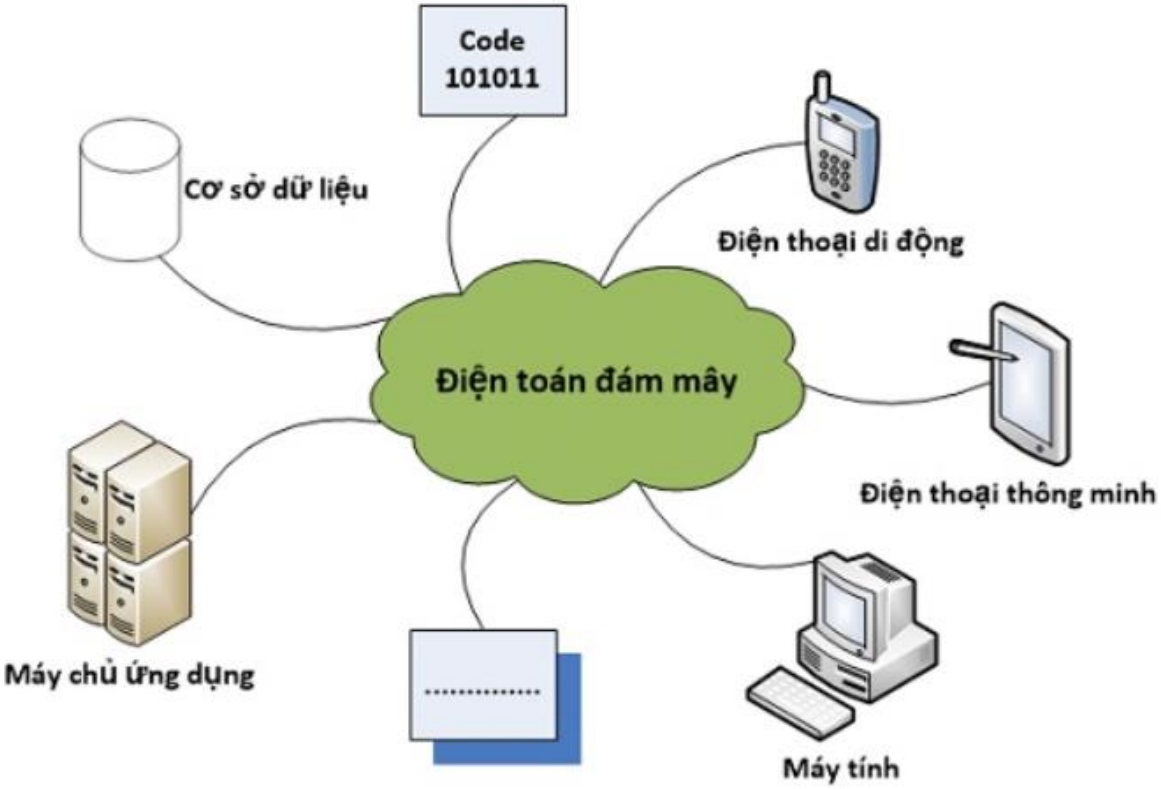
Điện toán đám mây dựa trên ảo hóa thông qua các dịch vụ mạng [11] và cung cấp cho người dùng các tài nguyên cơ bản, nền tảng ứng dụng, phần mềm và các dịch vụ khác. Với Cơ sở hạ tầng như một dịch vụ (IaaS), các nhà phát triển kết hợp phần cứng, phần mềm và thiết bị có liên quan để đáp ứng chất lượng của hợp đồng dịch vụ với người dùng, cung cấp một môi trường ứng dụng phần mềm hoàn chỉnh. Công nghệ máy ảo được sử dụng rộng rãi trong các trung tâm dữ liệu, tính toán cụm và các ứng dụng khác. Công nghệ này cho phép nhiều hệ điều hành có thể chạy trên cùng một máy tính đồng thời cung cấp các dịch vụ độc lập, đáng tin cậy và cải tiến rất nhiều về khả năng sử dụng lại các tài nguyên vật lý. Điện toán đám mây là một hướng nghiên cứu rộng, sẽ đem lại giá trị lớn về các chi phí cho các doanh nghiệp trên toàn thế giới. Điện toán đám mây sẽ giúp giải quyết được việc lưu trữ dữ liệu trên hệ thống một cách nhanh, gọn, nhẹ. Cung cấp các dịch vụ về cơ sở hạ tầng, nền tảng phần mềm và các dịch vụ theo yêu cầu người dùng thông qua Internet.

Điện toán đám mây là mô hình dịch vụ công nghệ thông tin sử dụng các mạng tiền thân trên thế giới giúp người dùng dễ dàng truy cập tài nguyên dữ liệu và lưu trữ trong các hệ thống quản lý và xử lý dữ liệu phức tạp của các công ty như Google, Facebook ... trên thực tế, truy cập vào truy cập đầu cuối để truy cập tài nguyên trên máy tính và trong hệ thống máy tính chỉ dành cho người dùng lập lịch xử lý các yêu cầu trên. Bao gồm thời gian chờ xử lý, thời gian xử lý tín hiệu đến thời gian hoàn thành tác vụ. Điện toán đám mây đang biến đổi ngành công nghệ thông tin, thay đổi cách thức sử dụng và phân phối phần mềm và phần cứng. Ngoài ra, nó còn đơn giản

hóa việc sử dụng các tài nguyên máy tính theo yêu cầu như băng thông, dung lượng lưu trữ cũng như phần mềm và ứng dụng máy tính có sẵn. Nó che giấu sự phức tạp của cơ sở hạ tầng bên dưới và cho phép người dùng cuối tập trung vào sản phẩm của họ mà không cần đầu tư phần cứng lớn. Theo hợp đồng dịch vụ đã được thiết lập giữa nhà cung cấp điện toán và khách hàng, các ràng buộc về chất lượng dịch vụ (QoS) nhất định được xác định thông qua các thỏa thuận theo mức dịch vụ (SLA). Tuân thủ với các SLA này, nhà cung cấp đảm bảo cung cấp một chất lượng nhất định cho dịch vụ đã thỏa thuận. Việc sử dụng các máy ảo cho phép sử dụng tốt hơn các tài nguyên phần cứng hiện tại trong khi vẫn duy trì QoS yêu cầu. Để tránh sự xuống cấp của hiệu suất, máy ảo được di chuyển từ quá tải đến các máy không sử dụng được. Vì vậy, các thuật toán phát hiện là cần thiết để chủ động trong việc phân loại quá tải và không quá tải. Thuật toán xác định trước kế hoạch tốt nhất để di chuyển và phân bổ các máy ảo tại thời điểm chạy. Đây là một mô hình tính toán mới được phát triển sau công nghệ điện toán phân tán, điện toán lưới, lưu trữ mạng, công nghệ cụm và tính toán song song. Do số lượng lớn các ứng dụng trên đám mây và sự không đồng nhất của các nút nguồn của máy chủ, một số máy tính bị quá tải và một số máy tính rất nhẹ do lưu lượng mạng tăng nhanh. Do đó, bạn cần một chiến lược cân bằng tải để điều chỉnh tải trên các máy chủ của mình, giảm chi phí truyền thông và cải thiện việc sử dụng tài nguyên. Tuy nhiên, sự ra đời của dữ liệu lớn và sự phát triển của điện toán đám mây đã mang đến những thay đổi mới. Ví dụ, giải quyết các vấn đề về công việc dữ liệu lớn với máy ảo điện toán đám mây. Do liên quan đến dữ liệu, việc di chuyển một số máy ảo xử lý dữ liệu làm phát sinh chi phí truyền thông cao giữa các máy chủ trong quá trình di chuyển và tính toán. Nó cũng làm giảm việc sử dụng tài nguyên hệ thống.

Điện toán đám mây là mô hình mới và phát triển đáng chú ý nhất trong lĩnh vực điện toán. Cơ chế cân bằng tải được chia thành tài nguyên và tài nguyên cung cấp, bên cạnh việc lập lịch các tác vụ giữa các hệ thống phân tán. Trong cân bằng tải truyền thống, việc cung cấp tài nguyên trong môi trường đám mây có các giai đoạn khác nhau. Việc bao gồm các thông số cân bằng tải khác nhau và bản chất của môi trường đám mây cũng có tác động đáng kể đến các hệ thống đám mây về mặt hiệu

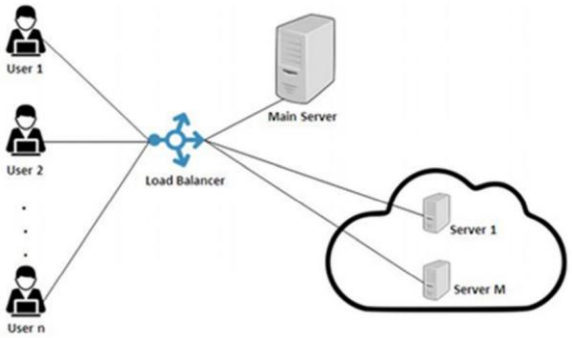
suất và đo lường. Trong thế giới ngày nay, điện toán đám mây là một cách để giữ cả phần cứng và phần mềm ở một nơi và làm cho nó khả dụng ở mọi nơi trên thế giới. Điều này làm cho phần cứng được yêu cầu linh hoạt hơn nhiều. Điều này cho phép mọi người sử dụng bao nhiêu tài nguyên họ cần và chỉ phải trả tiền cho thời gian họ sử dụng chúng. Đặc biệt, cái gọi là dịch vụ trả tiền khi sử dụng đang biến ngành công nghệ thông tin thành ngành kinh doanh điện toán đám mây. Các công ty sở hữu các cụm CPU / máy vật lý này, chẳng hạn như CPU có nhiều lỗi, được gọi là đám mây. Một cụm có một lượng không gian đĩa và bộ nhớ hữu hạn.



Hình 1.2: Cung cấp tài nguyên đám mây [16]

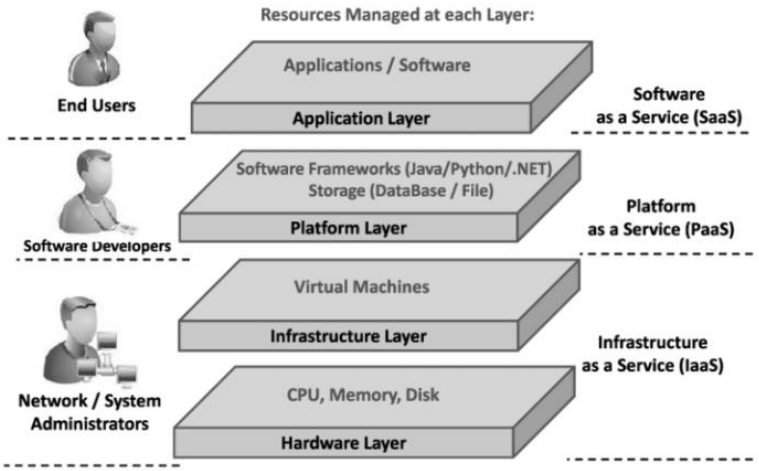
Do đó, khách hàng phải trả tiền để có được không gian đĩa và bộ nhớ từ cụm trong khoảng thời gian được phân bổ cho người dùng. Khi người dùng cần các tài nguyên như dung lượng lưu trữ, không gian đĩa và băng thông, công ty sẽ thực hiện điều này bằng cách phân bổ máy chủ cho nền tảng nhu cầu của khách hàng. Cung cấp tài nguyên đám mây là quá trình cung cấp không gian lưu trữ ảo từ các tài nguyên

bằng cách tập hợp các máy vật lý (PM) được gọi là máy ảo (VM). Bộ cân bằng tải quản lý việc ghép kênh tài nguyên khi cần thiết.



Hình 1.3: Cân bằng tải trong điện toán đám mây [17]

Các biện pháp cân bằng trước đây có hiệu quả trong việc cải thiện thời gian phản hồi và thời gian phục vụ của đám mây, nhưng không cung cấp đúng chất lượng dịch vụ. Các QoS có thể được cung cấp hiệu quả bằng cách thêm tham số của nó vào tham số cân bằng tải. Xem xét băng thông như tham số mà đối mặt với các vấn đề suy giảm và những vấn đề khác sẽ làm cho ngưỡng giá trị chính xác hơn. Do đó, QoS sẽ được coi là có hiệu quả. Vì vậy, giảm thiểu yêu cầu được cấp phát cho các máy vật lý với đúng khả năng cung cấp của các máy ảo và duy trì trạng thái ổn định trong suốt thời gian cung cấp dịch vụ.



Hình 1.4: Kiến trúc của điện toán đám mây [19]

Trong khi sử dụng tính toán tự động, tránh chi phí chung là một vấn đề lớn và giải quyết bằng cách đặt ra các nguồn lực thông qua thuật toán quy mô. Sau đó, vấn đề cuối cùng là giữ tải cân bằng ngay cả trong thời gian của giai đoạn phát triển. Điều này được thực hiện bằng cách sử dụng các thuật toán khác nhau.

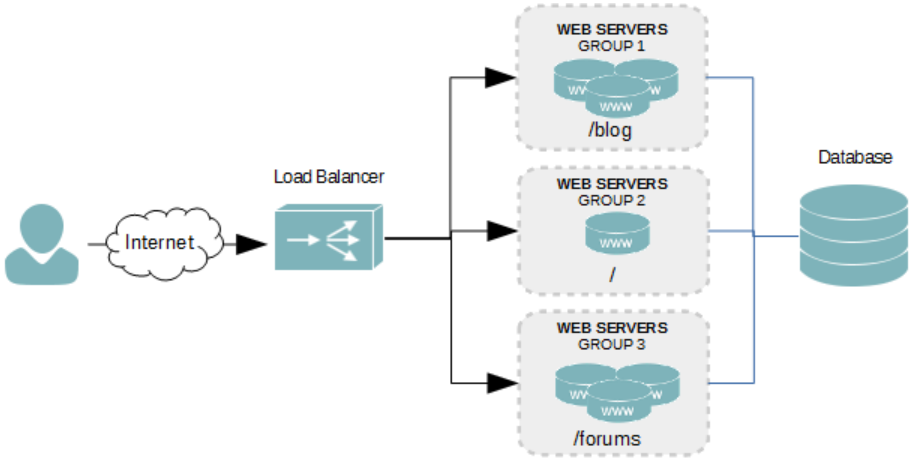
1.2. Tổng quan về cân bằng tải trong điện toán đám mây

1.2.1 Giới thiệu về cân bằng tải

Ngày nay, Ngành công nghiệp CNTT cũng như nhu cầu về tài nguyên lưu trữ và tính toán vẫn đang phát triển mỗi ngày. Một lượng lớn dữ liệu được tạo và trao đổi qua mạng, điều này đòi hỏi nhu cầu về tài nguyên máy tính ngày càng nhiều. Cloud đã giúp các doanh nghiệp tận dụng lợi ích của tài nguyên điện toán được chia sẻ trên môi trường ảo hóa. Rất nhiều doanh nghiệp đã sử dụng các dịch vụ dựa trên đám mây ở dạng này hay dạng khác. Điều này đưa chúng ta đến khái niệm cân bằng tải trong điện toán đám mây.

Với sự phát triển mạnh mẽ của Internet, các website hay các ứng dụng trực tuyến đang dần thu hút rất nhiều người truy cập và sử dụng. Khi lượng truy cập này quá lớn thường xảy ra các vấn đề là hạ tầng mạng và khả năng xử lý của Server sẽ bị tắc nghẽn cục bộ. Vì vậy Cân Bằng Tải [12] luôn là một trong những tính năng công nghệ rất quan trọng giúp các máy chủ ảo hoạt động đồng bộ và hiệu quả hơn thông qua việc phân phối đồng đều tài nguyên.

Giải pháp cân bằng tải là việc phân bố đồng đều lưu lượng truy cập giữa hai hay nhiều các máy chủ có cùng chức năng trong cùng một hệ thống. Bằng cách đó, sẽ giúp cho hệ thống giảm thiểu tối đa tình trạng một máy chủ bị quá tải và ngưng hoạt động. Hoặc khi một máy chủ gặp sự cố, Cân Bằng Tải sẽ chỉ đạo phân phối công việc của máy chủ đó cho các máy chủ còn lại, đẩy thời gian uptime của hệ thống lên cao nhất và cải thiện năng suất hoạt động tổng thể.



Hình 1.5: Mô hình Cân bằng tải trong điện toán đám mây [20]

Cân bằng tải là một trong những chủ đề quan trọng nhất trong môi trường phân tán. Vì Cloud Computing được coi là một trong những nền tảng tốt nhất giúp lưu trữ dữ liệu với chi phí tối thiểu và có thể truy cập mọi lúc qua internet, cân bằng tải cho điện toán đám mây đã trở thành một lĩnh vực nghiên cứu rất thú vị và quan trọng. Cân bằng tải nhằm mục đích thỏa mãn người dùng và sử dụng tỷ lệ tài nguyên cao bằng cách đảm bảo phân bổ hợp lý. Có rất nhiều khó khăn trong các kỹ thuật cân bằng tải như bảo mật, khả năng chịu lỗi, v.v ... vốn phổ biến trong môi trường điện toán đám mây hiện đại. Nhiều nhà nghiên cứu đã đề xuất một số kỹ thuật thuật toán để tăng cường nhằm tìm ra những phương án tốt nhất cho Cân bằng tải.

Phân tán dự đoán quá tải trong cân bằng tải thời gian gần đây đã nổi lên như một giải pháp đầy hứa hẹn. Trong đó, chuyển sang cấp độ giám sát tình trạng tắc nghẽn của mỗi con đường và phân tán dòng chảy trực tiếp đến con đường ít tắc nghẽn. Cách tiếp cận này có nhiều lợi thế thực tiễn. Là một lược đồ phân phối, nó có thể mở rộng hơn và có thể đối phó với lưu lượng truy cập nhanh hơn cách lịch trình tập trung. Là một phương pháp tiếp cận dữ liệu, nó không phụ thuộc vào ngăn xếp mạng của máy chủ lưu trữ và ngay lập tức mang lại lợi ích cho tất cả lưu lượng truy cập khi triển khai. Khả năng hiển thị tắc nghẽn cuối cùng của nó cũng làm cho nó trở nên mạnh mẽ hơn mà không cần cấu hình lại máy điều khiển. Mấu chốt của việc thiết kế một giao thức cân bằng tải tắc nghẽn là chúng ta cần phải biết thông tin về tắc nghẽn thời gian thực từ tất cả các đường đi giữa nguồn dòng chảy và điểm đến. Một cách

tiếp cận đơn giản là sử dụng thông tin định hướng đường đi cuối: một switch ToR duy trì các chỉ số tắc nghẽn đầu cuối cho tất cả các đường dẫn từ chính nó đến các thiết bị chuyển mạch ToR khác trong mạng. Các chỉ số tắc nghẽn có thể được thu thập bằng các gói dữ liệu. Thông thường, có hàng trăm đường dẫn tồn tại giữa hai ToR thiết bị chuyển mạch và công tắc ToR có thể giao tiếp với hàng trăm các thiết bị chuyển mạch ToR khác. Quan trọng hơn, không thể để thu thập thông tin tắc nghẽn thời gian thực cho tất cả các đường dẫn này, vì sẽ không có đủ dòng chảy đồng thời xảy ra đi cùng với tất cả chúng cùng một lúc. Trong giai đoạn đầu, chỉ có nguồn và thiết bị chuyển mạch ToR đích tham gia để lựa chọn tốt nhất đường dẫn từ ToR đến tầng tổng hợp. Chuyển đổi nguồn ToR sẽ gửi số liệu tắc nghẽn của nó đến đích ToR, chúng sẽ kết hợp với các chỉ số tắc nghẽn để chọn con đường tốt nhất cho lớp tổng hợp. Trong giai đoạn thứ hai, tập hợp đã chọn sẽ tiếp tục chọn công tắc lõi tốt nhất theo một cách tương tự về tình trạng tắc nghẽn của bước nhảy thứ hai và thứ ba. Con đường quyết định lựa chọn sau đó được duy trì tại ToR và tập hợp thiết bị chuyển mạch. Về cơ bản hai giai đoạn lựa chọn đường dẫn sử dụng thông tin một phần đường dẫn để tìm đường tốt nhất cho dòng chảy. Bằng cách khai thác các tính chất cấu trúc của 3 tầng, lựa chọn đường dẫn hai giai đoạn làm giảm đáng kể sự phức tạp mà không có nhiều hiệu suất. Trên thực tế, đánh giá cho thấy rằng thực hiện lựa chọn đường dẫn trên mỗi cơ sở lưu lượng trong TCP là tốt nhất và không gây ra việc sắp xếp lại gói tin cũng như không gây bất kỳ độ trễ nào.

Cân bằng tải [13] luôn là chủ đề nghiên cứu nóng của các trung tâm dữ liệu đám mây. Mục tiêu của nó là đảm bảo rằng mọi tài nguyên máy tính có thể xử lý các nhiệm vụ một cách hiệu quả và nhanh chóng. Cuối cùng, việc sử dụng nguồn lực được cải thiện. Các nhà nghiên cứu đã đề xuất một loạt cân bằng tĩnh, cân bằng động và chiến lược lập kế hoạch cân bằng tải. Ngoài ra, cũng có một số nghiên cứu sử dụng công nghệ di chuyển trực tiếp của máy ảo để đáp ứng các yêu cầu đám mây. Còn nhiệm vụ của trung tâm dữ liệu là yêu cầu hiệu suất và giới hạn tải. Các chiến lược cân bằng tải hiện được chia thành hai loại: cân bằng tải tĩnh và cân bằng tải năng động. Thuật toán lập lịch cân bằng tải tĩnh thường bao gồm Round Robin, Rounded Robin Weighted. Các thuật toán tĩnh chỉ sử dụng một số thông tin tĩnh mà không thể

phản ánh tải động. Hiện nay, hầu hết các nền tảng mã nguồn mở, kể cả IaaS, đã sử dụng các thuật toán tĩnh để tiến hành lập kế hoạch tài nguyên. Lợi thế của thuật toán lập kế hoạch cân bằng tải tĩnh là nó rất đơn giản và rất dễ sử dụng. Nhưng trong các trung tâm dữ liệu đám mây quy mô lớn có tính không đồng nhất của tài nguyên và nhu cầu người sử dụng thì không nhất quán nên hiệu quả cân bằng tải tĩnh không được lý tưởng. Cân bằng tải động (DLB) chủ yếu được sử dụng trong lĩnh vực phân phối máy tính song song. Mục tiêu chính của nó là làm thế nào để phân phối tải hợp lý hơn giữa nhiều máy chủ. Từ đó, tránh một số hiện tượng như một số các nút máy tính bị quá tải và một số nút có tải nhẹ cũng như để cải thiện toàn bộ hiệu suất của hệ thống. Chi phí truyền thông bổ sung được tạo ra trong quá trình DLB sẽ làm suy giảm hiệu năng hệ thống của cân bằng tải động. Vì vậy, làm thế nào để giảm truyền gói tin trên cao nhất giữa các nút trong quá trình DLB trở thành một vấn đề quan trọng có ảnh hưởng đến hiệu suất của DLB. Tuy nhiên, một số thuật toán ở trên không thể đáp ứng được sự lựa chọn và bản chất của cơ cấu cân bằng tải tối ưu cùng một lúc. Do đó, những cách phân phối tiếp cận thường có được sự tối ưu cục bộ của các giải pháp. Trong một số trường hợp đặc biệt, hiệu quả của việc giải quyết vấn đề phân phối tải không phải là lý tưởng. Thế nên, nó khó có thể đảm bảo cân bằng tải và sử dụng hiệu quả tài nguyên vật lý của toàn bộ cụm. Dù vậy, cân bằng tải vẫn là vấn đề và chi phí chung của đám mây trong các trung tâm dữ liệu không được xem xét. Nó chỉ tập trung vào quản lý máy ảo để tăng cường quản lý cũng như nâng cao hiệu quả hoạt động của các trung tâm dữ liệu điện toán đám mây.

Cân bằng tải [14], [15] có thể được chia thành 2 loại:

- Cân bằng tải cục bộ
- Tải toàn cầu

Cân bằng tải cục bộ được sử dụng để cân bằng dự báo tải trong một trung tâm. Nó phân phối yêu cầu từ phía máy khách sang máy chủ đáp ứng nhu cầu. Thứ hai là cân bằng tải toàn cục. Nó quản lý và kiểm soát yêu cầu từ phía khách hàng tự động đến máy chủ qua nhiều trung tâm dữ liệu. Ngoài ra, nó còn xử lý lưu lượng trên cả hai mặt gói truyền tải. Xử lý cân bằng tải toàn cầu rất phức tạp nhưng đồng thời điều

này cũng là rất hữu ích cho truyền tải gói tin trên trung tâm dữ liệu mạng. Tính khả dụng đảm bảo rằng, trong trường hợp thất bại, hệ thống vẫn tiếp tục hoạt động bình thường như mong đợi.

1.2.2 Mục đích cân bằng tải

Tăng khả năng đáp ứng và tránh tình trạng quá tải trên máy chủ đồng thời đảm bảo tính linh hoạt và mở rộng cho hệ thống.

Tăng độ tin cậy và khả năng dự phòng cho hệ thống: Sử dụng Cân bằng tải giúp tăng tính HA (High Availability) cho hệ thống đồng thời đảm bảo cho người dùng không bị gián đoạn dịch vụ khi xảy ra sự cố lỗi tại một điểm cung cấp dịch vụ.

Tăng tính bảo mật cho hệ thống: Thông thường khi người dùng gửi yêu cầu dịch vụ đến hệ thống và yêu cầu đó sẽ được xử lý trên bộ Cân bằng tải. Sau đó, thành phần Cân bằng tải mới chuyển tiếp các yêu cầu cho các máy chủ bên trong. Quá trình trả lời cho khách hàng cũng thông qua thành phần Cân bằng tải. Vì vậy mà người dùng không thể biết được chính xác các máy chủ bên trong cũng như phương pháp phân tải được sử dụng. Bằng cách này có thể ngăn chặn người dùng giao tiếp trực tiếp với các máy chủ, ẩn các thông tin và cấu trúc mạng nội bộ. Hơn nữa, có thể ngăn ngừa các cuộc tấn công trên mạng hoặc các dịch vụ không liên quan đang hoạt động trên các cổng khác.

1.3. Lợi ích, đặc điểm của điện toán đám mây [16]

- Tài nguyên Công nghệ thông tin có thể mở rộng, chẳng hạn như máy chủ, bộ nhớ và ứng dụng.
- Tiếp cận các công nghệ mới dễ dàng và nhanh chóng.
- Giúp phân phối tài nguyên hợp lý.
- Giúp tiết kiệm chi phí.
- Triển khai nhanh chóng.

1.4. Tổng quan về tác vụ

Tác vụ [17], [18] là quá trình sắp xếp các yêu cầu (request) đến theo một cách thức nhất định để lập lịch tài nguyên sẵn có sẽ được sử dụng hợp lý. Vì điện toán đám

mây là công nghệ cung cấp dịch vụ thông qua phương tiện Internet, người dùng dịch vụ phải gửi yêu cầu của họ thông qua Internet.

1.5. Vai trò của dự báo tác vụ đối với cân bằng [19] tải trên cloud

Việc dự báo các tác vụ [20], [21] có vai trò quan trọng trong việc cân bằng tải trên cloud. Với việc có thể dự báo trước các tác vụ cần sử dụng những tài nguyên nào sẽ giúp cho việc cân bằng tải hiệu quả và tránh việc phân phối các tác vụ không hợp lý.

1.6. Các thuật toán cân bằng tải

Thuật toán MaxMin: Thuật toán max-min thường được sử dụng trong môi trường phân tán bắt đầu với một tập hợp các nhiệm vụ không theo lịch trình. Sau đó, tính toán ma trận thực hiện dự kiến và thời gian hoàn thành dự kiến của từng nhiệm vụ trên các nguồn lực hiện có. Tiếp theo, chọn nhiệm vụ có thời gian hoàn thành dự kiến tối đa tổng thể và gán nó cho tài nguyên có thời gian thực hiện tổng thể tối thiểu. Cuối cùng, nhiệm vụ được lập lịch gần đây sẽ bị xóa khỏi bộ nhiệm vụ, cập nhật tất cả thời gian được tính toán, sau đó lặp lại cho đến khi bộ nhiệm vụ meta trở nên trống.

Thuật toán MinMin: thường được sử dụng để chỉ định nhiệm vụ cho các tài nguyên để có thời gian hoàn thành dự kiến tối thiểu. Nó sẽ hoạt động trong hai Giai đoạn, trong giai đoạn đầu, thời gian hoàn thành dự kiến sẽ được tính cho mỗi nhiệm vụ trong danh sách siêu nhiệm vụ. Trong giai đoạn thứ hai, nhiệm vụ có thời gian hoàn thành dự kiến tối thiểu tổng thể từ danh sách siêu nhiệm vụ được chọn và gán cho tài nguyên tương ứng. Sau đó, tác vụ này sẽ bị xóa khỏi danh sách siêu nhiệm vụ và quá trình được lặp lại cho đến khi tất cả các tác vụ trong danh sách nhiệm vụ được ánh xạ tới các tài nguyên tương ứng.

Thuật toán FCFS (First Come First Serve): Trong thuật toán FCFS, các tác vụ đến trước sẽ được xử lý trước. Các tác vụ sẽ được xếp và thêm vào cuối hàng đợi, sau đó quá trình cân bằng tải sẽ xử lý các tác vụ ở phần đầu của hàng đợi (các tác vụ đến trước).

Thuật toán RoundRobin (RR): RR là một trong những thuật toán lập lịch đơn giản ưu tiên nhất được sử dụng cho các tác vụ trong một khung công tác trong đó việc thực hiện một nhiệm vụ được dừng lại sau một khoảng thời gian cụ thể của thời gian gọi là lượng tử thời gian.

1.7. Tổng quan về AI

Trí tuệ nhân tạo (AI) [22], [23] là một ngành khoa học máy tính liên quan đến việc tạo ra các chương trình nhằm mục đích tái tạo con người nhận thức và các quá trình liên quan đến việc phân tích sự phức tạp dữ liệu. Sự ra đời của khái niệm này được liên kết phổ biến với hội nghị Dartmouth năm 1956 [24]. Tuy nhiên, công nghệ tại thời điểm này đã giới hạn việc ứng dụng AI. Gần đây, những tiến bộ đáng kể đã được thực hiện trong lĩnh vực máy tính sức mạnh vì công nghệ phần cứng và phần mềm được cải tiến. Các cá nhân và tổ chức trên một số các ngành công nghiệp đang bắt đầu nhận ra tiềm năng của AI để cải thiện các hoạt động hiện tại và nghiên cứu AI đã được được tiến hành trong nhiều lĩnh vực y tế, điện toán đám mây, xử lý ảnh,…

Bhaskar Mondal [25] đã công bố khẳng định của mình về AI trong bài nghiên cứu *"Artificial Intelligence: State of the Art"* năm 2019. Artificial (Nhân tạo – do con người tạo ra) Intelligence (Trí tuệ - sức mạnh của tư duy) là một nghiên cứu về máy móc nhằm để chúng có thể cảm nhận, ra quyết định và hành động như con người. Tóm lại, AI là một ngành nghiên cứu khoa học và kỹ thuật nhằm xây dựng những giả lập có thể nâng tầm hiểu biết bằng cách trau dồi kinh nghiệm, đọc và xử lý văn bản được viết bằng ngôn ngữ tự nhiên. Sau đó, chúng sẽ suy luận với những kiến thức đã được học (có thể thực hiện các tác vụ như: giải thích, lập kế hoạch, chẩn đoán…) và có những hành động sao cho hợp lý.

Hay theo Laura Musikansk và cộng sự [26], năm 2020 đã công bố trong bài báo *"Artificial Intelligence and Community Well-being: A Proposal for an Emerging Area of Research"* với việc phân loại AI. AI còn được phân thành ba loại chính là: AI yếu (hay AI hẹp), AI mạnh (hay AI tổng quát) và siêu AI (AI có tri giác hay AI tự nhận thức). AI yếu (hẹp) dựa trên các thuật toán được thiết kế nhằm giải quyết một vấn đề cụ thể hoặc tập hợp vấn đề trong một bối cảnh nhất định.

Với Niklas Kühl và cộng sự [23], năm 2020 trong bài nghiên cứu *"Machine Learning in Artificial Intelligence: Towards a Common Understanding"* đã có những công bố của riêng họ về những thuật ngữ. Học máy và Trí tuệ nhân tạo cũng như các thuật ngữ: Khai phá dữ liệu, Học sâu và Học thống kê đều có liên quan đến nhau. Chúng thường xuất hiện trong cùng một ngữ cảnh và đôi khi được sử dụng để thay thế cho nhau. Trong khi đó, các thuật ngữ này lại phổ biến trong các cộng đồng khác nhau, cách sử dụng và ý nghĩa của chúng cũng rất khác nhau.

1.8. Tổng quan về Machine Learning

Học máy (Machine Learning / ML) [22] là một tập hợp con của AI, cho phép máy móc có thể học từ bộ dữ liệu bất kì hoặc học từ những kinh nghiệm trước đó mà không cần phải lập trình một cách cụ thể và chi tiết. ML liên quan đến các chương trình máy tính viết lập trình của riêng chúng để hoàn thành một nhiệm vụ định trước. Quá trình này có thể được giám sát, bán giám sát hoặc không giám sát (Hình 1). Trong học tập có giám sát, máy được cung cấp dữ liệu trong đó mỗi ví dụ trong tập dữ liệu được gắn nhãn với câu trả lời. Các sau đó máy học thông qua thử và sai để dự đoán câu trả lời từ tập dữ liệu đã nhập. Học tập không giám sát liên quan đến việc phân tích dữ liệu đầu vào mà không có câu trả lời xác định. Điều này thường được sử dụng để mô hình hóa cấu trúc và phân phối dữ liệu. Cuối cùng, học tập bán giám sát là một phương pháp kết hợp liên quan đến việc kết hợp dữ liệu được gắn nhãn và không được gắn nhãn. Điều này có thể giúp giảm bớt gánh nặng của nhiệm vụ ghi nhãn. Sử dụng các thuật toán phân lớp của ML để tiến hành phân lớp các tác vụ dựa trên các đặc trưng của request để thực hiện việc cân bằng tải.

Phyllis Butow và cộng sự [27] đã công bố bài báo *"Using artificial intelligence to analyse and teach communication in healthcare"* và có những khẳng định riêng: Trong khi định nghĩa chính xác về trí tuệ nhân tạo (AI) vẫn còn gây tranh cãi, có sự nhiệt tình đáng kể liên quan đến tiềm năng của nó trong lĩnh vực này, dựa trên những thành công mới nhất của nó trong học máy và học sâu. Học máy cho phép phát triển tiểu thuyết các thuật toán có thể tự động đưa ra quyết định bằng cách dựa vào các mẫu và suy luận mà không cần bất kỳ hướng dẫn rõ ràng nào. Có hai loại thuật toán

học tập: được giám sát và không được giám sát. Các thuật toán được giám sát yêu cầu có sẵn dữ liệu được gắn nhãn để học. Hiệu suất của các thuật toán được xác định bằng cách thử nghiệm chúng trên dữ liệu "không nhìn thấy" với các nhãn đã biết. Ví dụ: Nếu thuật toán thực hiện với độ chính xác 95% trên dữ liệu không nhìn thấy, sau đó nó có thể được triển khai để thực hiện quyết định trong các tình huống thực tế, với cảnh báo rằng nó sẽ mắc sai lầm 5% thời gian. Tuy nhiên, một vấn đề đã được công nhận với AI là khả năng chuyển đổi giữa các cài đặt, tương tự như sự tương phản giữa hiệu quả và hiệu quả trong nghiên cứu can thiệp truyền thống.

Trong nhiều ứng dụng thực tế, việc không có sẵn dữ liệu được gắn nhãn hay ghi nhãn không chính xác hoặc thiên vị, là những nút thắt chính làm trì hoãn hoặc ngăn cản việc học có giám sát hiệu quả. Đối với những tình huống đó, tồn tại một tập hợp thuật toán khác có thể nhóm dữ liệu không được gắn nhãn dựa trên những điểm tương đồng, mẫu và sự khác biệt mà không có bất kỳ thông tin trước nào. Nhóm thuật toán này được gọi là học không giám sát.

Mặc dù việc ứng dụng học máy trên dữ liệu được xây dựng cẩn thận đã cho thấy nhiều hứa hẹn nhưng tiện ích của nó trong thế giới thực vẫn còn hạn chế bởi khả năng bao gồm toàn bộ sự phức tạp của giao tiếp con người.

1.9. Kết luận chương

Hiểu biết được những khái niệm tổng quan về điện toán đám mây, trí tuệ nhân tạo và học máy. Hiểu biết thuật toán điện toán đám mây để dự báo các tác vụ trên đám mây thông qua môi trường điện toán. Mục đích cân bằng tải để làm tăng hiệu năng của hệ thống.

CHƯƠNG 2: CÁC CÔNG TRÌNH LIÊN QUAN

2.1. Ở Việt Nam.

Năm 2018, Nguyễn Xuân Phi, Lê Ngọc Hiếu và Trần Công Hùng [11] đã công bố nghiên cứu "Thuật toán cân bằng tải nhằm giảm thời gian đáp ứng dựa vào ngưỡng thời gian trên điện toán đám mây". Nhóm tác giả đã nghiên cứu một thuật toán mới cho cân bằng tải trên đám mây bằng cách sử dụng mô hình dự đoán thời gian đáp ứng được đề xuất và thử nghiệm mô phỏng với một mô hình nhỏ. Thuật toán được sử dụng trong nghiên cứu này là thuật toán ARIMA, cân bằng tải dựa trên thời gian phản hồi. Thuật toán thường tiếp cận và phát triển ý tưởng về dự báo cũng như xử lý chuỗi thời gian là thuật toán ARIMA. Do đó, thuật toán đề xuất có một phương pháp khá mới trong cân bằng tải trong môi trường đám mây. Trong đó, thuật toán đã đạt được một số kết quả mô phỏng khá tích cực, cho thấy hướng phát triển tốt của thuật toán.

2.2. Trên thế giới.

Năm 2017, Muhammad Junaid và các cộng sự [19] đã công bố "Modeling an Optimized Approach for Load balancing in Cloud". Họ đã nghiên cứu đề xuất một thuật toán cân bằng tải, cụ thể là định dạng loại tệp dữ liệu (DFTF) sử dụng phiên bản sửa đổi của tối ưu hóa gói Cat (CSO) cùng với SVM. Đầu tiên, hệ thống được đề xuất phân loại dữ liệu trên đám mây từ các nguồn khác nhau thành các danh mục khác nhau. Chẳng hạn như: văn bản, hình ảnh, video và âm thanh bằng cách sử dụng một đến nhiều loại bộ phân loại SVM. Sau đó, sử dụng thuật toán cân bằng tải được sửa đổi của CSO giúp phân phối tải trên các máy ảo một cách hiệu quả. Kết quả mô phỏng so với các phương pháp hiện có cho thấy hiệu suất được cải thiện về nhiều mặt. Cụ thể là: thông lượng (7%), phản hồi thời gian (8,2%), thời gian di chuyển (13%), tiêu thụ điện năng (8,5%), thời gian tối ưu hóa (9,7%), thời gian phát sóng (6,2%), vi phạm SLA (8,9%) và thời gian thực hiện trung bình (9%). Những kết quả này vượt trội hơn một số các đường cơ sở hiện có được sử dụng trong nghiên cứu này như CBSMKC, FSALB, PSO-BOOST, IACSO-SVM, CSO-DA, và GA-ACO.

Năm 2018, Abdel-Basset, Mohamed, Mohamed, Mai và Chang, Victor [28]. đã đưa ra nghiên cứu về *"Cloud Task scheduling based on Load Balancing Ant Colony Optimization"*. Trong nghiên cứu này, nhóm tác giả đã đề xuất thuật toán LBACO để đạt được lịch trình tác vụ với cân bằng tải đồng thời cũng thực nghiệm đánh giá thuật toán LBACO trong các ứng dụng có số lượng nhiệm vụ thay đổi từ 100 đến 500. Kết quả thực nghiệm cho thấy rằng sự cân bằng LBACO cho toàn bộ hệ thống tải là hiệu quả. Tác giả đưa 2 giả thuyết cho nghiên cứu như sau. Đầu tiên, trong công việc này, tác giả giả định rằng tất cả các nhiệm vụ là độc lập lẫn nhau. Tức là, không có ràng buộc ưu tiên giữa các nhiệm vụ. Thứ hai, tác giả giả định các nhiệm vụ đó đòi hỏi nhiều về mặt tính toán nhưng điều này không thực tế cho các hệ thống đám mây.

Năm 2018, P. Ravi Kumar, P. Herbert Raj và P. Jelciana [10] đã công bố nghiên cứu *"Exploring Data Security Issues and Solutions in Cloud Computing"*. Nghiên cứu này nói về các định nghĩa cơ bản về mô hình điện toán đám mây, bảo mật dữ liệu của người dùng trên đám mây. Bài báo này làm cơ sở lý thuyết về mô hình điện toán đám mây cho đề cương.

Năm 2018, Blesson Varghese và Rajkumar Buyya [9] đã đưa ra nghiên cứu về *"Next generation cloud computing: New trends and research directions"*. Trong bài báo này, trước hết các tác giả thảo luận về việc thay đổi cơ sở hạ tầng đám mây và xem xét việc sử dụng cơ sở hạ tầng từ nhiều nhà cung cấp và lợi ích của phân quyền máy tính ra khỏi trung tâm dữ liệu. Những xu hướng này đã dẫn đến nhu cầu về nhiều loại kiến trúc điện toán mới sẽ được cung cấp bởi cơ sở hạ tầng đám mây trong tương lai. Những kiến trúc này dự kiến sẽ tác động đến các lĩnh vực, chẳng hạn như kết nối con người và thiết bị, máy tính sử dụng nhiều dữ liệu, dịch vụ không gian và hệ thống tự học. Cuối cùng, họ đã đưa ra một lộ trình về những thách thức cần phải giải quyết để nhận ra tiềm năng của các hệ thống đám mây thế hệ tiếp theo.

Năm 2020, Yiming Miao và các cộng sự [16] đã công bố nghiên cứu về *"Intelligent task prediction and computation offloading based on mobile-edge cloud computing"*. Bài báo này cung cấp một tính toán giảm tải thông minh dựa trên kiến

trúc MEC kết hợp với công nghệ trí tuệ nhân tạo (AI). Theo kích thước dữ liệu của tác vụ tính toán từ người dùng di động và hiệu suất các tính năng của các nút tính toán biên, thuật toán giúp giảm tải tính toán và di chuyển tác vụ dựa trên dự đoán nhiệm vụ đề xuất. Dự đoán nhiệm vụ tính toán dựa trên thuật toán LSTM, tính toán chiến lược giảm tải trên thiết bị di động dựa trên dự đoán nhiệm vụ và di chuyển tác vụ cho đám mây cạnh. Lược đồ lập kế hoạch được sử dụng để hỗ trợ tối ưu hóa mô hình giảm tải tính toán biên. Thí nghiệm cho thấy rằng kiến trúc và thuật toán được đề xuất của chúng tôi có thể giảm tổng độ trễ tác vụ một cách hiệu quả dữ liệu ngày càng tăng và các nhiệm vụ phụ.

Năm 2020, trong bài báo này, Jiechao Gao, Haoyu Wang và Haiying Shen [17] đã công bố nghiên cứu về *"Task Failure Prediction in Cloud Data Centers Using Deep Learning"*. Nhóm tác giả đề xuất một thuật toán dự đoán lỗi dựa trên bộ nhớ ngắn hạn hai chiều nhiều lớp (Bi-LSTM) để xác định lỗi tác vụ và công việc trên đám mây. Mục tiêu của thuật toán dự đoán lỗi Bi-LSTM là dự đoán liệu các tác vụ và các công việc bị thất bại hay đã hoàn thành. Các thử nghiệm tiếp theo cho thấy thuật toán của họ hoạt động tốt hơn các công cụ dự đoán hiện đại khác. Phương pháp này có độ chính xác lần lượt là 93% và 87% đối với lỗi nhiệm vụ và thất bại công việc.

Năm 2021, Ibrahim Mahmood Ibrahim và các cộng sự [12] đã công bố nghiên cứu *"Task Scheduling Algorithms in Cloud Computing: A Review"*. Bài báo này đưa ra ý tưởng về các thuật toán lập lịch tác vụ khác nhau trong điện toán đám mây môi trường được sử dụng bởi các nhà nghiên cứu. Cuối cùng, nhiều tác giả đã áp dụng các thông số khác nhau như thời gian hoàn thành, thông lượng và chi phí để đánh giá hệ thống. Bài báo cung cấp cái nhìn tổng quan về các nghiên cứu về xử lý tác vụ với điện toán đám mây trên thế giới.

Năm 2019, Dario Gil và các cộng sự [29] đã công bố nghiên cứu *"AI for Management: An Overview"*. Nhóm tác giả nhận định rằng: Trí tuệ nhân tạo (AI) đã đạt được tiến bộ rất nhanh trong những năm gần đây. Từ loa thông minh và chatbot trả lời câu hỏi, đến robot nhà máy và ô tô tự lái. Từ âm nhạc, tác phẩm nghệ thuật và nước hoa do AI tạo ra, đến hệ thống chơi trò chơi và tranh luận. Họ đã trải qua quá

trình chuyển đổi AI từ một lĩnh vực chủ yếu là lý thuyết trở thành một công cụ thực tế trao quyền cho rất nhiều ứng dụng mới. Một số người nói rằng “AI là Công nghệ thông tin mới”. Chúng ta có thể thấy bằng chứng trong toàn ngành máy học và các môn học AI nền tảng khác. Những ngành này có số lượng tuyển sinh kỷ lục trong khuôn viên trường đại học. Ngoài ra, các công cụ hỗ trợ AI đã giúp các bác sĩ phát hiện ra khối u ác tính, nhà tuyển dụng tìm ứng viên đủ tiêu chuẩn và ngân hàng quyết định gia hạn khoản vay cho ai. Các thuật toán đang cung cấp năng lượng cho các đề xuất sản phẩm, quảng cáo có mục tiêu hay chấm điểm bài luận. Thăng chức và giữ chân nhân viên, chấm điểm rủi ro hay ghi nhãn hình ảnh. Phát hiện gian lận, bảo vệ an ninh mạng và một loạt các ứng dụng khác.

Sự bùng nổ và áp dụng rộng rãi việc ra quyết định theo thuật toán đã thúc đẩy một lượng lớn sự quan tâm và gây ra nhiều phản ứng khác nhau (cùng với một lượng "cường điệu" đáng kể). Từ sự phấn khích về cách các khả năng của AI sẽ tăng cường khả năng ra quyết định của con người và cải thiện hiệu suất kinh doanh, đến các câu hỏi về công bằng và đạo đức. Từ nỗi sợ hãi về tình trạng sa thải việc làm và chênh lệch kinh tế, cho đến những suy đoán về mối đe dọa đối với nhân loại. Ngay cả bản thân thuật ngữ “AI” đã phát triển với ý nghĩa là những điều khác nhau đối với những người khác nhau. Nó bao gồm học máy, mạng nơ-ron và học sâu. Song, nó cũng đã trở thành một thuật ngữ chung cho nhiều chủ đề liên quan đến dữ liệu và phân tích khác (một phần của hiện tượng “AI là Công nghệ thông tin mới”).

Mục tiêu của chương này là giới thiệu ngắn gọn về AI và mô tả sự phát triển của nó từ trạng thái “hẹp” hiện tại đến một điểm mà các khả năng được nâng cao hơn hay “rộng”, thông qua trạng thái tương lai của “AI tổng quát”. Chúng tôi cũng khám phá những cân nhắc đối với các tổ chức và quản lý. Bao gồm vai trò của AI trong các nhiệm vụ hoạt động kinh doanh như lập kế hoạch chiến lược, tiếp thị, thiết kế sản phẩm và hỗ trợ khách hàng. Cuối cùng, chúng tôi nêu chi tiết các yêu cầu đối với các tổ chức trong việc xác định chiến lược AI toàn diện, được hỗ trợ bởi sự hiểu biết về giá trị của AI đối với tổ chức và tập trung vào nhu cầu. Bao gồm dữ liệu và kỹ năng, để thực hiện chiến lược AI một cách phù hợp.

Chúng tôi đã thấy những tiến bộ đáng kể đã đạt được của AI trong vài năm qua và tính đến thời điểm mà AI đang bắt đầu chuyển từ trạng thái "hẹp". Trạng thái tập trung vào một nhiệm vụ duy nhất trong một lĩnh vực duy nhất đến đỉnh của kỷ nguyên "rộng" của AI. Trong đó, các công nghệ có thể được áp dụng cho các nhiệm vụ hoặc các bộ vấn đề trên nhiều lĩnh vực. AI hứa hẹn mang lại nhiều lợi ích trong việc giúp các tổ chức thực hiện các nhiệm vụ hoạt động kinh doanh quan trọng. Ví dụ như: lập kế hoạch chiến lược, thiết kế sản phẩm, tiếp thị và hỗ trợ khách hàng. Các nhà lãnh đạo doanh nghiệp đặt mục tiêu phát triển và triển khai nhiều AI hơn trong tổ chức của họ. Bước đầu tiên quan trọng nhất là xác định kế hoạch sử dụng AI cụ thể để đáp ứng các mục tiêu kinh doanh và phát triển một chiến lược AI toàn diện. Các thành phần quan trọng của chiến lược AI bao gồm kế hoạch đạt được các khả năng AI cần thiết. Cho dù thông qua nguồn cung ứng bên ngoài hay phát triển nội bộ, phương pháp tập hợp tài năng AI, tính khả dụng và thu thập dữ liệu được dán nhãn thích hợp cần thiết để đào tạo các mô hình AI. Chúng tôi khuyến khích tất cả các nhà lãnh đạo hiểu biết và có chủ đích về những nỗ lực này để hỗ trợ triển khai thành công AI trong doanh nghiệp của họ.

2.3. Tổng kết chương

Trong chương này thông qua việc nghiên cứu tìm hiểu được một số thuật toán và những công trình liên quan đến cân bằng tải trong điện toán đám mây. Từ đó, giúp luận văn này hiểu rõ hơn về cân bằng tải và tải trên điện toán đám mây. Cuối cùng, hiểu được những ưu nhược điểm của các thuật toán và các cách xử lý cân bằng tải, tạo tiền đề và cơ sở vững chắc cho nghiên cứu của đề tài luận văn này.

CHƯƠNG 3 : ĐỀ XUẤT THUẬT TOÁN DỰ BÁO TÁC VỤ TRÊN ĐIỆN TOÁN ĐÁM MÂY NHẰM NÂNG CAO HIỆU QUẢ CÂN BẰNG TẢI

3.1. *Giới thiệu chung*

Cân bằng tải trên điện toán đám mây là một trong những công nghệ thu hút được sự chú ý của nhiều nhà khoa học cũng như các tổ chức, doanh nghiệp lớn, nhỏ và nhà cung cấp dịch vụ trong những năm gần đây. Các ứng dụng của AI và thuật toán ML được phát triển, cải tiến và ứng dụng trong các hệ thống cân bằng tải trên điện toán đám mây. Trong chương này, thuật toán đề xuất sử dụng trong quá trình cân bằng tải trên cloud được trình bày với mục tiêu dự báo các tác vụ dựa trên lịch sử thực hiện tác vụ trước đó. Mục đích là để phân bổ các request vào các máy ảo phù hợp, có khả năng đáp ứng cao.

3.2. *Mô hình nghiên cứu*

Mô hình nghiên cứu sử dụng thuật toán phân lớp AdaBoost nhằm mục đích dự báo các request tương ứng với các task dựa trên lịch sử về thời gian thực hiện cũng như xử lí các request. Lịch sử xử lí ở đây được tính toán dựa trên mức độ tiêu thụ năng lượng của task (Power Consumed), mức độ sử dụng CPU (CPU Usages), mức độ sử dụng RAM (RAM Usages) và chi phí (Costing) để thực hiện task đó trong cloud. Sau khi phân loại các job/task theo lịch sử thực hiện, bộ cân bằng tải sẽ phân bổ các request có các job/task đó vào những máy ảo/host có năng lực xử lý tốt hơn. Từ đó, phân bổ request có nhu cầu xử lý nhiều vào máy ảo/host có mức độ hoạt động thấp nhất. Với cách tiếp cận này, thuật toán đề xuất sẽ cải thiện thời gian xử lý cân bằng tải trên cloud và ứng dụng trên môi trường cloud theo thời gian thực. Trong luận văn này tạm đặt tên thuật toán là ACTPA (AdaBoost Classification of Task-Prediction Algorithm).

Về mục tiêu:

- Giảm thiểu rủi ro cho hệ thống máy chủ.
- Giảm thiểu thời gian sống cho các yêu cầu trong điện toán đám mây.

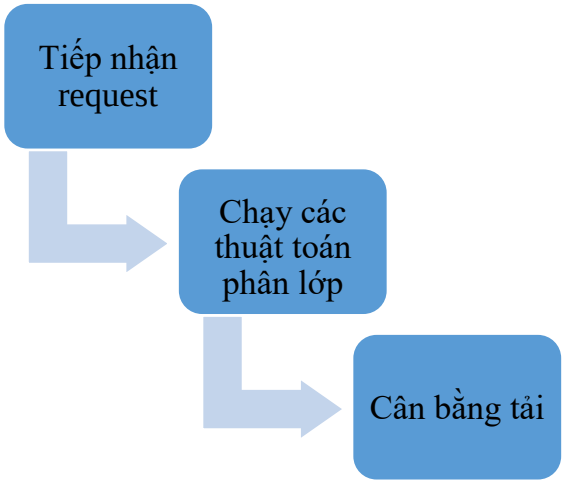
- Hạn chế tối đa cũng như ngăn chặn sự mất cân bằng tải giữa các máy ảo.
- Giúp giải quyết các yêu cầu nhanh hơn, phân loại được các tác vụ (Task) với các độ ưu tiên khác nhau tương ứng với các yêu cầu (Request). Đồng thời sử dụng hiệu quả hơn nguồn tài nguyên trên cloud để đáp ứng tốt nhất các yêu cầu của người dùng.
- Phân lớp được các yêu cầu (Coming Request) tiếp theo tương ứng với độ ưu tiên đã được phân lớp ở trên. Từ đó, có kế hoạch đưa các yêu cầu này sang những máy ảo/host có khả năng xử lý tải tương ứng.
- Sắp xếp các máy ảo/host/tài nguyên theo mức độ sử dụng từ cao đến thấp để phân bổ task cho hợp lý.

Giả định:

- Bộ cân bằng tải sẽ biết trước các dịch vụ nào đang chạy trên các máy ảo vào bất cứ thời điểm nào.
- Ở đây tập trung vào dịch vụ Web (Web Service), các máy chủ web sẽ biết trước thời gian xử lý của từng dịch vụ chạy trên web và trên từng máy ảo.
- Nếu hai máy ảo có cấu hình tương đương nhau về RAM, vi xử lý, và I/O thì thời gian thực thi của các dịch vụ sẽ không mấy là khác nhau.

Mô hình nghiên cứu:

- Trong mô hình nghiên cứu trước giờ thì bộ cân bằng tải sẽ chạy theo sơ đồ thuật toán như sau:



Hình 3.1: Mô hình dự đoán tác vụ

3.3. *Thuật toán AdaBoost*

Adaboost là một thuật toán boosting dùng để xây dựng bộ phân lớp (classifier). Trong đó, boosting là thuật toán học quần thể bằng cách xây dựng nhiều thuật toán học cùng lúc (ví dụ như cây quyết định) và kết hợp chúng lại. Mục đích là để có một cụm hoặc một nhóm các *weak learner*, sau đó kết hợp chúng lại để tạo ra một *strong learner* duy nhất. Thuật toán Adaboost có thể kết hợp với nhiều thuật toán khác để cải thiện hiệu suất. Đầu ra của một thuật toán (thường được gọi là "weak learners") được kết hợp lại thành một tổng có trọng số đại diện cho đầu ra cuối cùng của bộ phân loại tăng cường (boosted classifier).

Trong máy tính, một task được thực hiện sẽ tiêu hao nguồn năng lượng nhất định, kèm theo đó là mức độ sử dụng CPU của máy tính, mức độ sử dụng bộ nhớ tạm thời RAM,… Tất cả đều được tính toán ra chi phí thực hiện công việc đó theo thời gian hoặc MIPS. Chính vì thế, bài luận văn này dựa vào các đặc điểm đó để tính toán ra độ ưu tiên của task mà máy tính phục vụ. Task có mức tiêu hao năng lượng (Power Consumed), mức độ sử dụng CPU (CPU Usage) cũng như mức độ sử dụng RAM (RAM Usage) hay chi phí (Cost) cao hơn thì sẽ có độ ưu tiên cao hơn và ngược lại.

3.4. *Thuật toán đề xuất ACTPA*

Thuật toán đề xuất ACTPA (*AdaBoost Classification Task Prediction Algorithm)*, dựa vào yếu tố độ ưu tiên của tác vụ (Task Priority được mô tả ở trên)

tương ứng với các request, kèm theo đó là một số thuộc tính khác, ta sử dụng thuật toán AdaBoost để phân lớp các request này. Từ đó, ta biết cách phân bố tài nguyên cho các request này một cách tối ưu nhất. Song song đó, các tài nguyên (máy ảo/ host) được sắp xếp theo mức độ sử dụng tăng dần. Kết hợp với đánh giá số lần sai và sai số, ta cải thiện thuật toán bằng cách áp dụng máy học vào. Tuy nhiên, việc áp dụng này sẽ ít diễn ra vì có sai số cho phép.

Luận văn này xin đề xuất thuật toán gồm 3 nhóm Module chính:

(1) Module phân lớp các request bằng thuật toán Random Forest (RF):

Trong Module này, thuật toán RF sẽ dựa vào các thuộc tính của request mà tính toán ra thời gian xử lý, từ đó phân lớp request này. Các thuộc tính bao gồm: Size, Response Length, Max Length,…

Nhóm Thời Gian xử lý = MK_{New} = DT (X_1, X_2,…, X_n)

Trong đó, X_i là các thuộc tính của Request khi gửi lên cloud.

Ở đây có thể chia thành nhiều nhóm (từ 4 ~10 nhóm) hoặc nhiều hơn dựa vào độ biến thiên của Request.

(2) Module phân lớp tác vụ dựa trên thời gian dự báo:

Trong Module này sẽ sử dụng thuật toán phân lớp AdaBoost để phân lớp request đang xét, dựa vào tính chất của độ ưu tiên các tác vụ. Việc phân lớp này sẽ thông qua việc xây dựng mô hình phân lớp AdaBoost của các request đã được xử lý trong quá khứ và đánh nhãn từ 1 đến 5, tương ứng với 5 mức độ ưu tiên. Mức 1 là độ ưu tiên thấp nhất, mức 5 là độ ưu tiên cao nhất. Dựa vào mô hình này, ta phân chia được 44 lớp từ các request đang cần xử lý và xác định được label (từ 1 đến 5). Sau đó, ta chọn ra máy ảo có thứ tự tương ứng 1 đến 5. Thứ tự này được sắp xếp dựa trên mức độ hoạt động thấp hay ít tải của máy. Tức, mức 1 là máy tải nhiều nhất và mức 5 là máy tải ít nhất nhưng tính sẵn sàng cao nhất.

VMselect = AdaBoost(Po, CPU, RAM);

Trong đó:

VMselect là máy ảo được chọn ra

AdaBoost là hàm phân lớp từ bộ thư viện WEKA

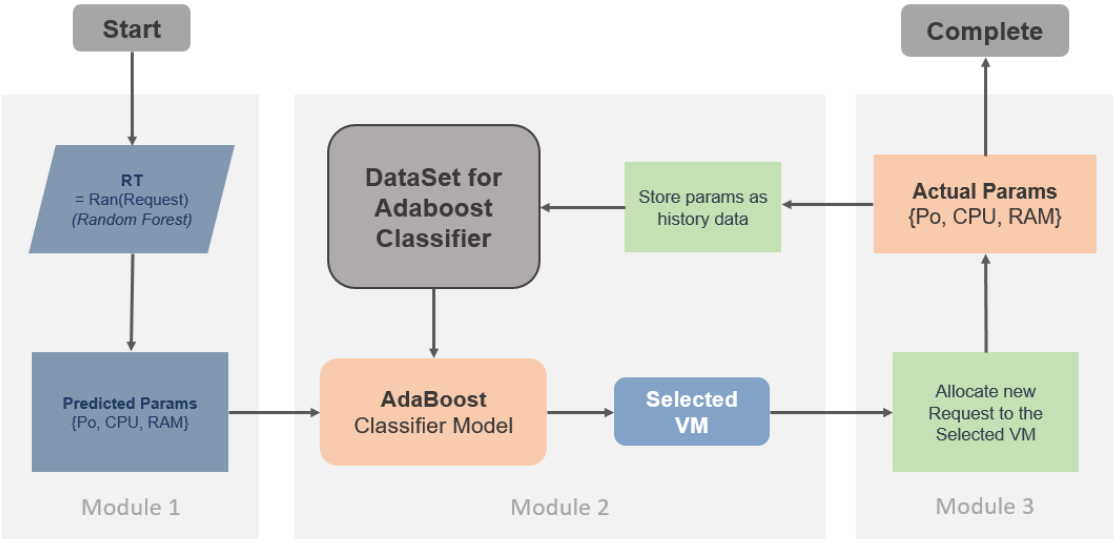
Po là Power dự đoán tính toán từ Module 1

CPU là mức sử dụng CPU dự đoán tính toán từ Module 1

RAM là mức sử dụng RAM dự đoán tính toán từ Module 1

(3) Module phân bổ các dịch vụ (chọn máy ảo)

Module này có nhiệm vụ phân bổ các yêu cầu đến các máy ảo thông qua dự báo tác vụ và máy ảo phù hợp. Nếu một yêu cầu được gửi đến thì sẽ được phân loại bởi Module 1 và các VM đang xét, kể cả VM không tải cũng được phân cụm theo Module 2. Ở đây, Module 3 có nhiệm vụ phân bổ Request đang xét vào máy ảo đã tìm thấy từ Module 2. Từ đó xử lý và trả về kết quả cho request đồng thời lưu vào lịch sử bộ nhớ các request gần nhất đã xử lý, làm dữ liệu đầu vào cho quá trình xây dựng mô hình AdaBoost ở Module 2.



Hình 3.2: Sơ đồ thuật toán đề xuất ACTPA

Thuật toán ACTPA

```
1.    For each Request in CloudRequests
2.          isLocated = false;
3.          RT = RandomForest(T1,T2….. Tn); //Module 1
4.          Request.Class = AdaBoost(RT); // AdaBoost là mô hình phân lớp tác vụ
5.          For each VM in VMList
6.                If isFitSituation(Request.Class, VM)
7.                   AllocateRequestToVM(VM, Request); // Module 3
8.                   isLocated = true;
9.                End If
10.         End For
11.   If(!isLocated)
12.         VM = VMList.getSelectedVM(); // Module 2
13.         AllocateRequestToVM(VM, Request);
14.   End If
15.   End For
```

3.5. *Kết luận chương*

Chương này giới thiệu vì sao tác giả lại chọn việc dự báo thời gian tải tối đa và tải tối thiểu cũng như giá trị của tải đó thông qua thời gian xử lý để phục vụ công việc cân bằng tải. Với mục tiêu duy trì trạng thái an toàn và hoạt động liên tục của cloud, nghiên cứu hướng đến mục đích tối ưu hóa nguồn tài nguyên của cloud và giúp cho cân bằng tải hoạt động tốt nhất. Thuật toán đề xuất sẽ giải quyết được cân bằng tải dựa trên cải thiện thời gian thực thi. Qua đó, số lượng thất bại trong công việc triển khai sẽ ít hơn, số điểm nút chết cũng giảm mạnh hơn các thuật toán cân bằng tải hiện tại.

CHƯƠNG 4. MÔ PHỎNG THUẬT TOÁN VÀ ĐÁNH GIÁ KẾT QUẢ

4.1. Giới thiệu chung

Để có thể hiểu rõ hơn về thuật toán đề xuất, chương này sẽ trình bày về cài đặt mô phỏng thuật toán ACTPA (AdaBoost Classification of Task Prediction Algorithm). Quá trình mô phỏng cân bằng tải sẽ sử dụng thuật toán đề xuất ACTPA để phân loại và dự báo các tác vụ dựa trên thời gian xử lí. Sau đó, điều phối các tác vụ đến các máy ảo phù hợp. Các tác vụ có thời gian xử lí càng cao sẽ được phân bổ vào các máy ảo có mức độ sử dụng thấp, tức máy ảo có tính sẵn sàng cao và ngược lại. Với cách tiếp cận này, thuật toán đề xuất ACTPA sẽ tối ưu hóa thời gian xử lý cân bằng tải trên cloud và ứng dụng trên môi trường cloud theo thời gian thực. Sau khi tiến hành các bước thực nghiệm và thu được các kết quả, ta sẽ phân tích cũng như so sánh tính hiệu quả của thuật toán đề ra với các thuật toán cân bằng tải nổi tiếng khác. Các thuật toán được so sánh lần lượt là Round Robin, MaxMin, MinMin và FCFS.

4.2. Môi trường mô phỏng thực nghiệm

Dựa vào dữ liệu của các request, ta sử dụng thuật toán SVM để phân loại chúng bằng cách tính toán ra thời gian xử lý. Cũng từ đó, ta biết cách phân bố tài nguyên cho cái request vào các máy ảo đã phân cụm. Kết hợp với đánh giá số lần sai và sai số, ta sẽ cải thiện thuật toán bằng cách áp dụng máy học vào. Dù vậy, việc áp dụng này sẽ ít diễn ra vì có sai số cho phép.

Giả lập môi trường cloud sử dụng bộ thư viện CloudSim và lập trình trên ngôn ngữ JAVA. Môi trường giả lập cloud từ 5 đến 15 máy ảo chính là môi trường request ngẫu nhiên đến các dịch vụ trên cloud. Các dịch vụ trên cloud bao gồm: cung cấp máy ảo, cung cấp và đáp ứng người dùng muốn thử nghiệm của CloudSim.

Cài đặt thuật toán SVM, AdaBoost trên môi trường mô phỏng và kiểm nghiệm kết quả.

Các tham số của mô hình mạng mô phỏng:

Thực nghiệm mô phỏng thuật toán đề xuất được cài đặt trên ngôn ngữ JAVA, sử dụng APACHE NETBEAN IDE để chạy thử và hiển thị kết quả dưới dạng console. Môi trường giả lập với bộ thư viện mã nguồn mở CloudSim 4.0 (được cung cấp bởi http://www.cloudbus.org/) và bộ thư viện Weka.

Môi trường mô phỏng giả lập gồm các thông số sau:

- 01 Datacenter với thông số như sau:

Bảng 4.1: Thông số cấu hình Datacenter

Thông tin Datacenter	***Thông tin Host trong Datacenter***
- Số lượng máy (host) trong datacenter: 5 - Không sử dụng Storage (các ổ SAN) - Kiến trúc(arch): x86 - Hệ điều hành (OS): Linux - Xử lý (VMM): Xen - TimeZone: +7 GMT - Cost: 3.0 - Cost per Memory: 0.05 - Cost per Storage: 0.1 - Cost per Bandwidth: 0.1	Mỗi host trong Datacenter có cấu hình như sau: - CPU có 4 nhân, mỗi nhân có tốc độ xử lý là 1000 (mips) - Ram: 16384 (MB) - Storage: 1000000 - Bandwidth: 10000

- Các máy ảo có cấu hình giống nhau khi được khởi tạo:

Bảng 4.2: Cấu hình máy ảo

Kích thước (size)	Ram	Mips	Bandwidth	Số lượng cpu (pes no.)	VMM
10000 MB	512 MB	250	1000	1	Xen

- Các Request (các request chạy trên web, WebRequest) được đại diện bởi Cloudlet trong CloudSim và kích thước của các Cloudlet được khởi tạo một các ngẫu nhiên bằng hàm random của JAVA. Số lượng Cloudlet lần lượt là 30 □ 1000.

Bảng 4.3: Cấu hình thông số các Request

Chiều dài (Length)	Kích thước file (File Size)	Kích thước file xuất ra (Output Size)	Số CPU xử lý (PEs)
3000 ~ 1700	5000 ~ 45000	450 ~ 750	1

- Thuật toán đề xuất ACTPA được xây dựng bằng cách tạo ra lớp **ACTPASchedulingAlgorithm**, lớp này đã được kế thừa từ đối tượng có sẵn là **DataAwareSchedulingAlgorithmExample**. Đồng thời, thuật toán cũng cập nhật thêm một số phương thức và thuộc tính liên quan tới **predictRequestSVM**, sau đó điều chỉnh các hàm dựng sẵn để phù hợp với thuật toán đề xuất:

@Override
*public void **run**() // Module 3*

*public CondorVM **getFittingVm**(double label)*
// Module 2

*public String **predictRequestPowerConsume**(Cloudlet req)*

public String ***predictRequestCpuUsage****(Cloudlet req)*

public String ***predictRequestRamUsage*** *(Cloudlet req)*

// Module 1

Tiêu chí đánh giá:

Thực nghiệm mô phỏng cloud với các tham số như trên và chạy thuật toán cân bằng tải của CloudSim có sẵn. Sau đó, chạy thuật toán đề xuất mới cài đặt với cùng dữ liệu đầu vào và so sánh kết quả đầu ra, đặc biệt là thông số thời gian xử lý.

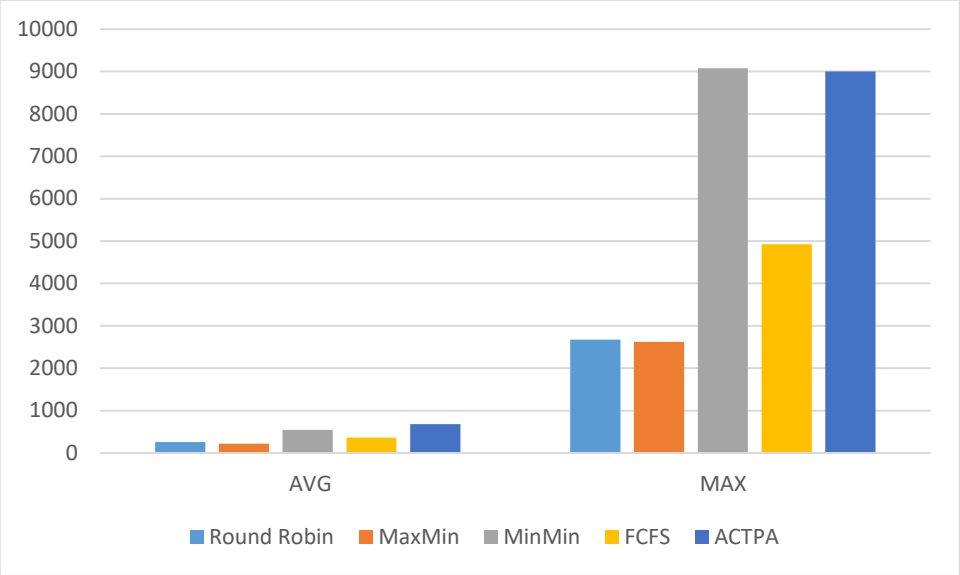
Thời gian xử lý của các máy ảo cũng như thời gian xử lý của cloud với sai số càng thấp thì hiệu quả của thuật toán càng tốt.

4.3. Thực nghiệm và kết quả mô phỏng

Kết quả chạy thực nghiệm mô phỏng trên CloudSim với 5 máy ảo được dựng sẵn để đáp ứng các yêu cầu. Các yêu cầu này được khởi tạo với chiều dài và kích thước ngẫu nhiên cùng số lượng Request từ 30, 60, 100 đến 1000. Kết quả thu được của thuật toán đề xuất được đem đi so sánh với các thuật toán Round-Robin, MaxMin, MinMin và FCFS có thời gian thực hiện là:

Bảng 4.4: Kết quả thực nghiệm mô phỏng với 30 Request

Thuật toán	Thời gian thực hiện (ms)		
	AVG	MAX	MIN
Round Robin	259.37	2677.21	0.26
MaxMin	217.71	2621.24	0.12
MinMin	548.94	9080.72	1.03
FCFS	364.19	4925.67	1.3
ACTPA	679.49	9001.37	0.38



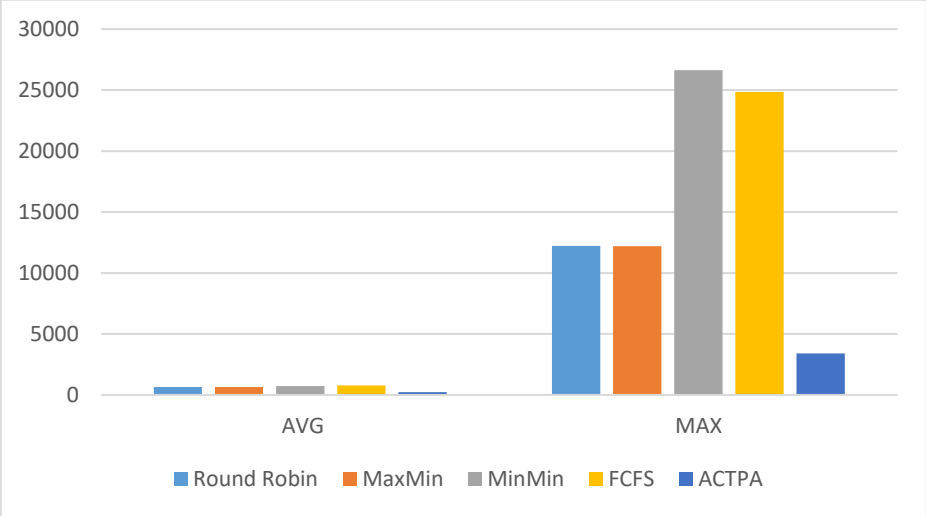
Hình 4.1: Biểu đồ so sánh thời gian thực hiện của 5 thuật toán với 30 Request

Quá trình thực nghiệm với 30 request trở lại, thuật toán MaxMin đã chiếm ưu thế với thời gian xử lí thấp nhất ở cả hai mốc thời gian trung bình và thời gian cao nhất. Ngoài ra, thuật toán Round Robin cũng có thời gian xử lí tác vụ rất tốt, chỉ kém thuật toán MaxMin 41.66 ms ở mức trung bình và 55.97 ms ở mức cao nhất. Thuật toán đề xuất trong phạm vi dưới 30 request vẫn chưa phát huy được sự tối ưu trong việc xử lí các tác vụ so với các thuật toán còn lại.

Kết quả chạy thực nghiệm mô phỏng trên CloudSim với 5 máy ảo được dựng sẵn để đáp ứng các yêu cầu. Chúng được khởi tạo cùng chiều dài và kích thước ngẫu nhiên, với số lượng Request là 60:

Bảng 4.5: Kết quả thực nghiệm mô phỏng với 60 request

Thuật toán	Thời gian thực hiện (ms)		
	AVG	MAX	MIN
Round Robin	650.77	12226.29	0.13
MaxMin	648.04	12194.58	0.18
MinMin	731.74	26626.42	0.11
FCFS	770.81	24847.42	0.11
ACTPA	222.83	3411.38	0.11

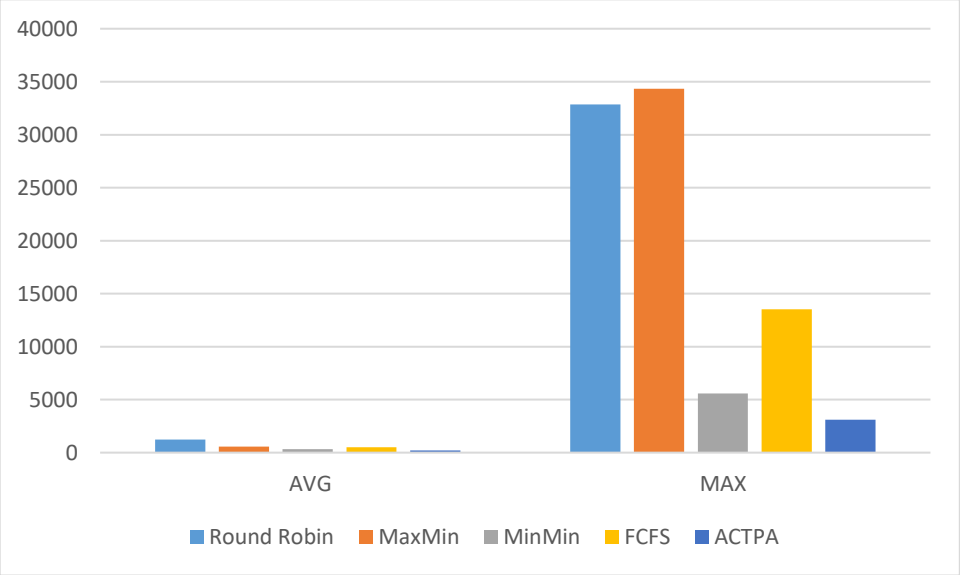


Hình 4.2: Biểu đồ so sánh thời gian thực hiện của 5 thuật toán với 60 Request

Kết quả thu được trong phạm vi dưới 60 request dần chứng minh được khả năng tối ưu hóa thời gian xử lí tác vụ của thuật toán đề xuất ở cả hai mốc thời gian. Từ biểu đồ ta có thể thấy được thời gian xử lý tác vụ của thuật toán đề xuất ACTPA đã vượt xa các thuật toán cân bằng tải còn lại và vẫn đang có chiều hướng phát triển tốt.

Bảng 4.6: Kết quả thực nghiệm mô phỏng với 100 request

Thuật toán	Thời gian thực hiện (ms)		
	AVG	MAX	MIN
Round Robin	1221.92	32851.68	0.15
MaxMin	578.37	34341.74	0.12
MinMin	339.68	5578.17	0.18
FCFS	503.30	13517.15	0.17
ACTPA	196.54	3104.3	0.11

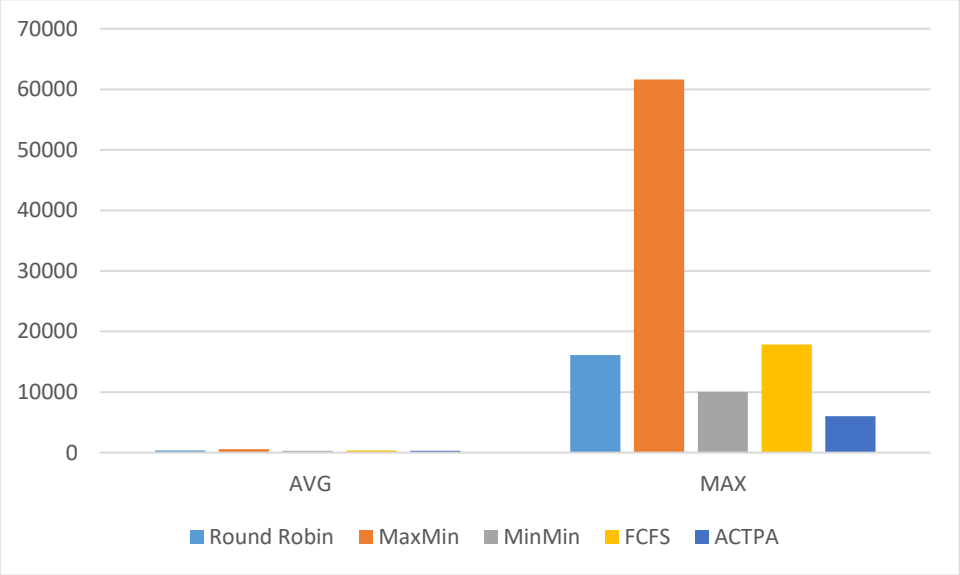


Hình 4.3: Biểu đồ so sánh thời gian thực hiện của 5 thuật toán với 100 Request

Từ request thứ 100 trở đi, thuật toán ACTPA vượt trội hơn hẳn so với MaxMin và Round Robin. Ở phạm vi này, ngoài thuật toán đề xuất ACTPA, thuật toán MinMin cũng cho thấy kết quả khá tốt với thời gian xử lý tác vụ tối đa khoảng 5578ms, chỉ kém thuật toán đề xuất khoảng 2474ms.

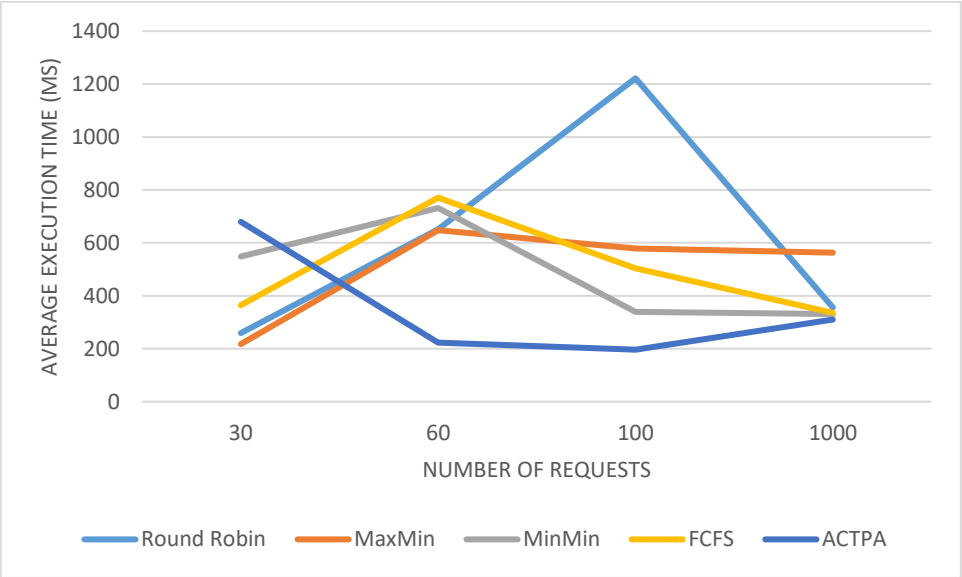
Bảng 4.7: Kết quả thực nghiệm mô phỏng với 1000 request

Thuật toán	Thời gian thực hiện (ms)		
	AVG	MAX	MIN
Round Robin	356.87	16134.21	0.11
MaxMin	563.26	61632.4	0.11
MinMin	331.16	10018.56	0.1
FCFS	335.86	17833.29	0.11
ACTPA	310.09	6038.66	0.17

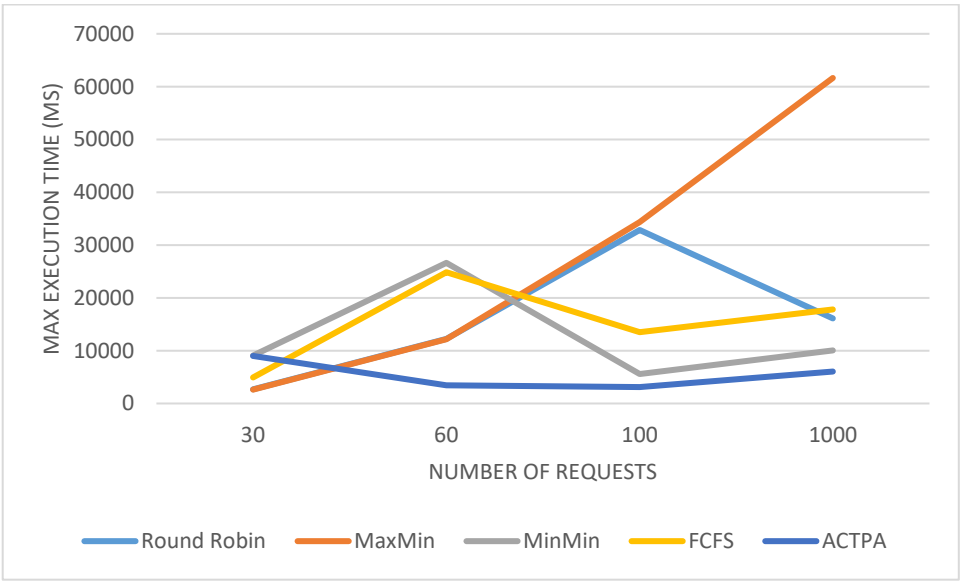


Hình 4.4: Biểu đồ so sánh thời gian thực hiện của 5 thuật toán với 1000 Request

Kết quả thực nghiệm 1000 request cho thấy thuật toán đề xuất vượt trội hơn hẳn các thuật toán cân bằng tải khác ở cả hai mốc thời gian là AVG và MAX.



Hình 4.5: Thời gian thực hiện trung bình của 5 thuật toán với 1000 Request



Hình 4.6: Thời gian thực hiện trung bình của 5 thuật toán với 1000 Request

Thông qua 02 biểu đồ tổng hợp so sánh thời gian xử lý của các thuật toán với điều kiện như nhau, ta có thể thấy sự phân bổ khá ổn định của thuật toán đề xuất ACTPA. Cụ thể, thời gian xử lý của các máy ảo khả quan hơn so với thời gian xử lý của các thuật toán khác trên cloud (cả trường hợp ít và nhiều request).

Thực nghiệm mô phỏng này chỉ là mô phỏng nhóm các máy ảo và chưa tính đến việc mở rộng tập các máy ảo (VM pool) để giảm tải trong trường hợp cần thiết. Vì ta

chỉ giả định nhóm các máy ảo này xử lý tối đa bao nhiêu request, nếu vượt quá ta mới mở rộng pool. Tuy nhiên, việc thí nghiệm mô phỏng với lượng request lớn trên 1000 request đòi hỏi máy tính mạnh hơn và có bộ xử lý tốt hơn. Chính vì vậy, đây cũng là hạn chế của thí nghiệm mô phỏng này.

4.4. Tổng kết chương

Chương 4 của luận văn trình bày mô hình thực nghiệm mô phỏng, các thông số cũng như kịch bản đưa ra là dựa vào quá trình request của các browser trên môi trường cloud. Từ đó, ghi nhận các thông số về thời gian đáp ứng dự báo của các máy ảo và của cloud. Việc chạy thực nghiệm mô phỏng với thông số 5 máy ảo, chịu tải từ 30 đến 1000 request đã cho thấy kết quả tương đối tốt. Đồng thời, việc phân bổ các request đến các máy ảo xử lý cũng khá đồng đều và có tính khả thi cao.

KẾT LUẬN VÀ KIẾN NGHỊ

Luận văn "**Nghiên cứu ứng dụng AI xây dụng thuật toán dự báo các tác vụ trên đám mây nhằm nâng cao hiệu quả cân bằng tải**" đã cơ bản hoàn thành được các mục tiêu đề ra. Thuật toán đề xuất dựa trên ý tưởng từ nhiều công trình nghiên cứu trong và ngoài nước. Từ đó, tiến hành tổng hợp, phân tích và đánh giá. Nhờ sự hỗ trợ mạnh mẽ từ bộ thư viện WEKA và công cụ mô phỏng điện toán đám mây Cloudsim, các máy ảo và thuật toán được cài đặt một cách dễ dàng, nhanh chóng thông qua môi trường APACHE NETBEAN với ngôn ngữ lập trình Java. Quá trình mô phỏng sử dụng thuật toán phân lớp Adaboost nhằm phân loại các máy ảo và thuật toán Random Forest cũng như phân loại các các tác vụ dựa trên thời gian xử lí. Từ đó, dự báo các tác vụ và phân bổ vào các máy ảo tương ứng. Các kết quả thu được từ quá trình thực nghiệm thuật toán đề xuất được phân tích và so sánh với các thuật toán cân bằng tải trên đám mây khác như Round Robin, MaxMin, MinMin, FCFS. Dựa vào kết quả so sánh dựa trên ba mốc thời gian là AVG, MAX và MIN, nhìn chung kết quả cho thấy rằng thuật toán đề xuất có tính hiệu quả cao và khả năng vượt trội hơn các thuật toán khác về việc dự báo các tác vụ trên điện toán đám mây.

Một thuật toán mới về cân bằng tải trên môi trường cloud bằng phương pháp phân loại các request theo thời gian xử lý đã được đề xuất và thực nghiệm mô phỏng với mô hình nhỏ. Dựa trên ý tưởng và các công trình nghiên cứu trước, đưa ra một giải thuật mới ứng dụng khai phá dữ liệu là thuật toán AdaBoost để cân bằng tải dựa vào thời gian xử lý. Trong đó, việc tính toán ra thời gian xử lí càng chính xác thì hiệu quả thuật toán càng cao. Tuy nhiên, việc tính toán càng chính xác thì càng đòi hỏi tốn nhiều bộ nhớ và phải xử lí nhiều. Bên cạnh đó, người dùng trên môi trường cloud có các request vô cùng đa dạng và phong phú nên thời gian xử lý cũng biến đổi khôn lường trên cloud. Thuật toán được đề xuất trong luận văn này tiếp cận một cách khái quát cũng như phát huy ý tưởng của phân lớp theo Regression, toán học và thời gian xử lý, điển hình là thuật toán AdaBoost. Do đó, thuật toán đề ra đã có một hướng tiếp cận khá mới trong cân bằng tải ở môi trường đám mây đồng thời đạt được một số kết quả thực nghiệm mô phỏng khá tích cực, cho thấy hướng phát triển tốt của thuật toán.

Hướng phát triển của thuật toán đề xuất là việc đo lường và hiệu chỉnh chính xác hơn bằng cách kết hợp AdaBoost với học máy, học không giám sát hoặc có giám sát bằng cách đưa ra các khoảng thời gian cao điểm hoặc thấp điểm của cloud. Để phát triển thuật toán tốt hơn và sâu hơn, cần thực nghiệm mô phỏng trên máy tính có cấu hình mạnh hơn và quy mô lớn hơn.

Bên cạnh đó, việc cài đặt thuật toán trên cloud thực tế sẽ giúp ta nghiên cứu chuyên sâu và cụ thể hơn. Bởi, môi trường cloud thực tế sẽ phát sinh nhiều vấn đề liên quan đến thời gian xử lý. Từ đó, ta sẽ biết hiệu chỉnh thuật toán một cách hợp lý hơn và hiệu quả hơn.

TÀI LIỆU THAM KHẢO

[1] Kaur, Rajwinder; Luthra, Pawan;, "Load Balancing in Cloud Computing," *Recent Trends in Information, Telecommunication and Computing, Association of Computer Electronics and Electrical Engineers,* pp. 374-381, 2014.

[2] H. V. Giap, "VIBLO," VIBLO, 20 3 2017. [Trực tuyến]. Available: https://viblo.asia/p/tong-quan-ve-dien-toan-dam-may-1VgZv36MlAw. . [Đã truy cập 1 5 2021].

[3] N. X. Phi, L. N. Hiếu and T. C. Hùng, "Thuật toán cân bằng tải nhằm giảm thời gian đáp ứng dựa vào ngưỡng thời gian trên điện toán đám mây," *Tạp chí khoa học công nghệ thông tin và truyền thông,* 2018.

[4] Phi, Nguyễn Xuân; Hùng, Trần Công;, "Giải Thuật Phòng Tránh Tình Trạng Quá Tải Trong Điện Toán Đám Mây," *Proceedings of The 2015 National Conference on Electronics, Communications and Information Technology ECIT 2015*, pp. 66-70, 2015.

[5] Jayashri, C.; Abitha, P.; Subburaj, S.; S. Y. Devi, Suthir S; S, Janakiraman;, "Big data transfers through dynamic and load balanced flow on cloud networks," *2017 Third International Conference on Advances in Electrical, Electronics, Information, Communication and Bio-Informatics (AEEICB), Chennai,* pp. 342-346, 2017.

[6] P. Wang, H. Xu, Z. Niu, D. Han and Y. Xiong;, "Expeditus: Congestion-Aware Load Balancing in Clos Data Center Networks," *in IEEE/ACM Transactions on Networking,* tập 25, số 5, pp. 3175-3188, 2017.

[7] GIBET TANI, H. ;C. EL AMRANI, "Smarter Round Robin Scheduling Algorithm for Cloud Computing and Big Data," *Journal of Data Mining and Digital Humanities, 2018. Special Issue on Scientific and Technological Strategic Intelligence,* p. 2018, 2016.

[8] Q. L. J. Zhang và J. Chen, "An Advanced Load Balancing Strategy for Cloud Environment," *," 2016 17th International Conference on Parallel and Distributed Computing, Applications and Technologies (PDCAT), Guangzhou,* pp. 240-243, 2016.

[9] Varghese, Blesson; Buyya, Rajkumar;, "Next generation cloud computing: New trends and research directions," *Future Generation Computer Systems,* p. 849–861, 2018.

[10] P. R. Kumar, P. H. Raj và P. Jelciana, “Exploring Data Security Issues and Solutions in Cloud Computing,” *Procedia Computer Science 125*, p. 691–697, 2018.

[11] Khiet, Bui Thanh; Que, Nguyen Thi Nguyet; Hung, Ho Dac; Vu, Pham Tran; Hung, Tran Cong;, “A Fair VM Allocation for Cloud Computing based on Game Theory,” trong *Proceedings of the 10th National Conference on Fundamental and Applied Information Technology Research (FAIR'10)*, Da Nang, Viet Nam, 2017.

[12] Wen, Y. F.; Chang, C. L.;, “Load balancing job assignment for cluster-based cloud computing,” *Sixth International Conference on Ubiquitous and Future Networks (ICUFN), Shanghai,* pp. 199-204, 2014.

[13] Sommer, Matthias; Klink, Michael; Tomforde, Sven; Hähner, Jörg;, “Predictive Load Balancing in Cloud Computing Environments Based on Ensemble Forecasting,” *2016 IEEE International Conference on Autonomic Computing (ICAC2016),Wurzburg, Germany,* pp. 300 - 307, 2016.

[14] A. K. Upadhaya, C. Jha và S. Pandey, “Suboptimal Mechanism For Load Balancing In CloudEnvironment,” *2017 International Conference on Algorithms, Methodology, Models and Applications in Emerging Technologies (ICAMMAET), Chennai, India,* 2017.

[15] Shao, G.; Chen, J.;, “A Load Balancing Strategy Based on Data Correlation in Cloud Computing,” *2016 IEEE/ACM 9th International Conference on Utility and Cloud Computing (UCC), Shanghai,* pp. 364-368, 2016.

[16] A. Ghoneim, M. Al-Rakhami, M. S. Hossain, Y. Miao, G. Wu và M. Li, “Intelligent task prediction and computation offloading based on mobile-edge cloud computing,” *Future Generation Computer Systems,* pp. 925-931, 2020.

[17] J. Gao, H. Wang và H. Shen, “Task Failure Prediction in Cloud Data Centers Using Deep Learning,” *Transactions on Services Computing,* 2020.

[18] A. Arunarani; D. Manjula ;V. Sugumaran, “Task scheduling techniques in cloud computing: A literature survey,” *Future Generation Computer Systems 91, p. Future Generation Computer Systems 91,* 2019.

[19] Junaid, Muhammad; Sohail, Adnan; Rais, Rao Naveed Bin; Ahmed, Adeel; Khalid, Osman; Khan, Imran Ali; Hussain, Syed Sajid; Ejaz, Naveed;, “Modeling an Optimized Approach for Load balancing in Cloud,” *IEEE Access,* tập 8, pp. 173208-173226, 2020.

[20] J. Zha, K. Y. Y. D. X. Wei, L. Hu và G. Xu, "A Heuristic Clustering-Based Task Deployment Approach for Load Balancing Using Bayes Theorem in Cloud Environment," *in IEEE Transactions on Parallel and Distributed Systems,* tập 27, số 2, pp. 305-316, 2016.

[21] I. M. Ibrahim, S. R. M. Zeebaree, M. A. M.Sadeeq, A. H. Radie và H. M. Shukur, "Task Scheduling Algorithms in Cloud Computing: A Review," *Turkish Journal of Computer and Mathematics Education,* tập 12, pp. 1041-1053, 2021.

[22] Huu, Tiep Vu, "Machine learning cơ bản," Machine learning cơ bản, 26 12 2016. [Trực tuyến]. Available: https://machinelearningcoban.com/2016/12/26/introduce/. [Đã truy cập 21 4 2021].

[23] Kühl, N.; Goutier, M.; Hirt, R.; Satzger, G.;, "Machine Learning in Artificial Intelligence: Towards a Common Understanding," trong *Hawaii International Conference on System Sciences (HICSS-52),* Hawaii, 2019.

[24] J. C. S. Daniel T. Hogarty, K. Pha, M. Attia, M. Hossny, S. Nahavandi, P. Lenane, F. J. Moloney và A. Yazdabadi, "Artificial Intelligence in Dermatology—Where We Are and the Way to the Future: A Review," *American Journal of Clinical Dermatology,* tập 21, pp. 41-47, 2020.

[25] B. Mondal, "Artificial Intelligence: State of the Art," *Recent Trends and Advances in Artificial Intelligence and Internet of Things,* pp. 389-425, 2020.

[26] Musikanski, L.; Rakova, B.; Bradbury, J.; Phillips, R.; Manson, M.;, "Artificial Intelligence and Community Well-being: A Proposal for an Emerging Area of Research," *International Journal of Community Well-Being,* 2020.

[27] Butow, P.; Hoque, E.;, "Using artificial intelligence to analyse and teach communication in healthcare," *The Breast,* tập 50, pp. 49-55, 2020.

[28] Kun Li;Gaochao Xu;Guangyu Zhao;Yushuang Dong;Dan Wang, "Cloud Task scheduling based on Load Balancing Ant Colony Optimization," *Sixth Annual ChinaGrid Conference,* 2011.

[29] J. Canals và F. Heukamp, "AI for Management: An Overview," *The Future of Management in an AI World,* pp. 3-19, 2019.

[30] Umadevi, K. S.; Chaturvedi, P.;, "Predictive load balancing algorithm for cloud computing," *2017 International conference on Microelectronic Devices, Circuits and Systems (ICMDCS),* pp. 1-5, 2017.

BẢNG CAM ĐOAN

Tôi cam đoan đã thực hiện việc kiểm tra mức độ tương đồng nội dung luận văn qua phần mềm DoIT một cách trung thực và đạt kết quả mức độ tương đồng 15% toàn bộ nội dung luận văn. Bản luận văn kiểm tra qua phần mềm là bản cứng đã nộp để bảo vệ trước hội đồng. Nếu sai tôi xin chịu hình thức kỷ luật theo quy định hiện hành của Học viện.

Tp.HCM, ngày 25 tháng 01 năm 2022

HỌC VIÊN CAO HỌC

Vương Duy Thanh

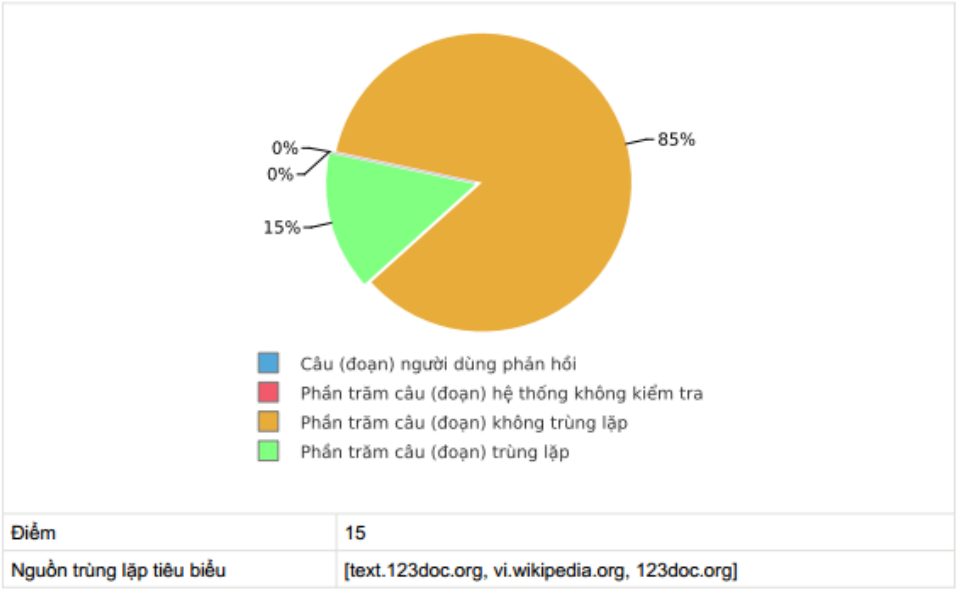
Hệ thống hỗ trợ nâng cao chất lượng tài liệu

KẾT QUẢ KIỂM TRA TRÙNG LẶP TÀI LIỆU

THÔNG TIN TÀI LIỆU

Tác giả	Vương Duy Thanh
Tên tài liệu	LuanVan_VuongDuyThanh_V5
Thời gian kiểm tra	08-02-2022, 07:52:49
Thời gian tạo báo cáo	08-02-2022, 07:54:44

KẾT QUẢ KIỂM TRA TRÙNG LẶP



Điểm	15
Nguồn trùng lặp tiêu biểu	[text.123doc.org, vi.wikipedia.org, 123doc.org]

(*) Kết quả trùng lặp phụ thuộc vào dữ liệu hệ thống tại thời điểm kiểm tra

Học viên **Người hướng dẫn khoa học**

Vương Duy Thanh **PGS.TS Trần Công Hùng**

BỘ THÔNG TIN VÀ TRUYỀN THÔNG
HỌC VIỆN CÔNG NGHỆ
BƯU CHÍNH VIỄN THÔNG

CỘNG HOÀ XÃ HỘI CHỦ NGHĨA VIỆT NAM
Độc lập - Tự do - Hạnh phúc

BÁO CÁO GIẢI TRÌNH
SỬA CHỮA, HOÀN THIỆN LUẬN VĂN THẠC SĨ

Họ và tên học viên: **Vương Duy Thanh**
Chuyên ngành: **Hệ Thống Thông Tin**
Khóa : **2020- 2022**

Tên đề tài: **Nghiên cứu ứng dụng AI xây dựng thuật toán dự báo các tác vụ trên đám mây nhằm nâng cao hiệu quả cân bằng tải**

Người hướng dẫn khoa học: **PGS.TS. Trần Công Hùng**

Ngày bảo vệ: **15/01/2022**

Các nội dung học viên đã sửa chữa, bổ sung trong luận văn theo ý kiến đóng góp của Hội đồng chấm luận văn:

TT	Ý kiến hội đồng	Sửa chữa của học viên
1	Hiệu chỉnh lỗi trình bày, hiệu chỉnh văn phong khoa học, hiệu chỉnh lưu đồ giải thuật	Học viên đã hiệu chỉnh lỗi trình bày, văn phong khoa học ở chương 4 và hiệu chỉnh lại lưu đồ giải thuật.
2	Bổ sung giải thích các kết quả đạt được 1000 request	Học viên đã bổ sung giải thích kết quả đạt được 1000 request.
3	Bổ sung giải thích các thuật toán so sánh	Học viên đã bổ sung giải thích các thuật toán so sánh ở chương 1, phần 1.6, trang 22

Tp.HCM, ngày 25 tháng 01 năm 2022

Ký xác nhận của

CHỦ TỊCH HỘI ĐỒNG CHẤM LUẬN VĂN	THƯ KÝ HỘI ĐỒNG	NGƯỜI HƯỚNG DẪN KHOA HỌC	HỌC VIÊN
PGS.TS. Đinh Đức Anh Vũ	TS. Trần Trung Duy	PGS.TS. Trần Công Hùng	Vương Duy Thanh

BỘ THÔNG TIN VÀ TRUYỀN THÔNG
HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG

CỘNG HOÀ XÃ HỘI CHỦ NGHĨA VIỆT NAM
Độc lập - Tự do - Hạnh phúc

BIÊN BẢN
HỌP HỘI ĐỒNG CHẤM LUẬN VĂN THẠC SĨ

Căn cứ quyết định số Quyết định số 1255-131/ QĐ-HV ngày 16 tháng 12 năm 2021 của Giám đốc Học viện Công nghệ Bưu chính Viễn thông về việc thành lập Hội đồng chấm luận văn thạc sĩ đợt tháng 01 năm 2022. Hội đồng đã họp vào hồi 8h00-8h45, ngày 15 tháng 01 năm 2022 tại Học viện Công nghệ Bưu chính Viễn thông để chấm luận văn thạc sĩ cho:

Học viên: **Vương Duy Thanh**

Tên luận văn: **Nghiên cứu ứng dụng AI xây dựng thuật toán dự báo các tác vụ trên đám mây nhằm nâng cao hiệu quả cân bằng tải**

Chuyên ngành: **Hệ thống Thông tin** Mã số: 8.48.01.04

Các thành viên của Hội đồng chấm luận văn có mặt: .05.../ 05

TT	HỌ VÀ TÊN	TRÁCH NHIỆM TRONG HĐ	GHI CHÚ
1	**PGS.TS. Đinh Đức Anh Vũ**	Chủ tịch	
2	**TS. Trần Trung Duy**	Thư ký	
3	**PGS.TS. Võ Thị Lưu Phương**	Phản biện 1	
4	**PGS.TS. Nguyễn Đình Thuân**	Phản biện 2	
5	**PGS.TS. Lê Hoàng Thái**	Uỷ viên	

Các nội dung thực hiện:

1. Chủ tịch Hội đồng điều khiển buổi họp. Công bố quyết định của Giám đốc Học Viện Công nghệ Bưu chính Viễn thông về việc thành lập Hội đồng chấm luận văn thạc sĩ
2. Người hướng dẫn khoa học hoặc thư ký đọc lý lịch khoa học và các điều kiện bảo vệ luận văn của học viên. (có bản lý lịch khoa học và kết quả các môn học cao học của học viên kèm theo).
3. Học viên trình bày tóm tắt luận văn.
4. Phản biện 1 đọc nhận xét (có văn bản kèm theo)
5. Phản biện 2 đọc nhận xét (có văn bản kèm theo)
6. Các câu hỏi của thành viên Hội đồng:

Câu hỏi 1: Trong Hình 4.1, giải thích tại sao thuật toán đề xuất có hiệu năng cao hơn? Hình 4.5 giải thích xu hướng hiệu năng?
Câu hỏi 2: Trang 35, Học viên xem lại lưu đồ thuật toán
Câu hỏi 3: Học viên cần giải thích rõ hơn khái niệm AI được sử dụng trong luận văn có ý nghĩa thực tiễn.

7. Trả lời của học viên:

Trả lời 1: ACTPA là thuật toán đề xuất với 03 metrum chính
Trả lời 2: Học viên điều chỉnh lại thuật toán

Trả lời 3: Học viên bổ sung thuật ngữ AI trong luận văn

8. Thư ký đọc nhận xét về quá trình thực hiện luận văn của học viên *(có văn bản kèm theo).*

9. Hội đồng họp riêng:

- Bầu Ban kiểm phiếu:
  1. Trưởng Ban kiểm phiếu: PGS.TS. Võ Thị Lưu Phương
  2. Uỷ viên Ban kiểm phiếu: PGS.TS. Nguyễn Đình Thuân
  3. Uỷ viên Ban kiểm phiếu: TS. Trần Trung Duy
- Hội đồng chấm luận văn bằng bỏ phiếu kín.
- Ban kiểm phiếu làm việc:
- Trưởng Ban kiểm phiếu báo cáo kết quả kiểm phiếu *(có Biên bản họp Ban kiểm phiếu kèm theo)*
- **Điểm trung bình của luận văn:** 7.86

Kết luận:

1. Các nội dung cần chỉnh sửa, hoàn thiện sau bảo vệ luận văn:

Học viên cần hiệu chỉnh luận văn theo góp ý của Hội đồng:
- Hiệu chỉnh lời trình bày
- Hiệu chỉnh văn phong khoa học
- Hiệu chỉnh các lưu đồ giải thuật
- Bổ sung giải thích các kết quả đạt được mong
- Bổ sung giải thích các thuật toán

2. Đề nghị Học viện công nhận (hoặc không) và cấp bằng (hoặc không) thạc sĩ cho học viên: **Vương Duy Thanh**

3. Luận văn có thể phát triển thành đề tài nghiên cứu cho NCS.... không.....

Buổi làm việc kết thúc vào... 8h.45... cùng ngày.

Chủ Tịch	Thư ký	Phản biện 1
PGS.TS. Đinh Đức Anh Vũ	TS. Trần Trung Duy	PGS.TS. Võ Thị Lưu Phương
Phản biện 2		**Ủy viên**
PGS.TS. Nguyễn Đình Thuân		PGS.TS. Lê Hoàng Thái

BỘ THÔNG TIN VÀ TRUYỀN THÔNG
HỌC VIỆN CÔNG NGHỆ
BƯU CHÍNH VIỄN THÔNG

CỘNG HOÀ XÃ HỘI CHỦ NGHĨA VIỆT NAM
Độc lập - Tự do - Hạnh phúc

BẢN NHẬN XÉT LUẬN VĂN TỐT NGHIỆP THẠC SĨ

(*Dùng cho người phản biện*)

Tên đề tài luận văn: Nghiên cứu ứng dụng AI xây dựng thuật toán dự báo các tác vụ trên đám mây nhằm nâng cao hiệu quả cân bằng tải.

Chuyên ngành: Hệ thống thông tin

Mã số: 8.48.01.04

Tên học viên: Vương Duy Thanh

Họ và tên người nhận xét: Võ Thị Lưu Phương

Học hàm, học vị: PGS.TS

Chuyên ngành: Công nghệ thông tin

Cơ quan công tác: Trường ĐH Quốc Tế - ĐHQG HCM

NỘI DUNG NHẬN XÉT

I/ Cơ sở khoa học và thực tiễn, tính cấp thiết của đề tài:

Đề tài sử dụng thuật toán Adaboost để dự đoán và phân lớp request từ người dùng, từ đó phân bổ máy chủ ảo hợp lý trong điện toán đám mây.

Chủ đề nghiên cứu có tính khoa học và thực tiễn.

II/ Về nội dung, chất lượng của luận văn, các kết quả đã đạt được (so với đề cương đã được duyệt):

Về cơ bản nội dung thực hiện giống như đề cương đã được duyệt.

Về mặt trình bày luận văn cần một số cải tiến sau:

- Trang 31-32 chỉ có gạch đầu dòng. Nên diễn dịch thành đoạn văn.

- Các hình trong phần giới thiệu về cloud hiện giống với luận văn của Huỳnh Phi Long. Học viên nên vẽ lại hoặc trình bày cách khác để tránh giống nhau, đúng cấp độ luận văn thạc sỹ.

- Các thuật toán MaxMin, MinMin, Round Robin dùng trong so sánh ở chương 4 cần được giải thích.

- Các hình vẽ trong chương 4 cần thêm đơn vị cho các trục. Có thể dùng error bar cho các hình.

III/ Những vấn đề cần giải thích thêm:

Một số vấn đề cần giải thích thêm:

1) Trong thuật toán đề xuất (trang 36), cân bằng tải thể hiện như thế nào?

2) Tại sao trong hình 4.1, thời gian xử lý của ACTPA cao hơn? Cần giải thích nguyên nhân. Tương tự như vậy, 1000 requests thì các thuật toán có thời gian xử lý gần nhau (hình 4.5). Vui lòng có giải thích.

IV/ Kết luận:

Học viên chỉnh sửa luận văn theo những nhận xét trên trước khi bảo vệ trước Hội đồng chấm luận văn thạc sĩ.

Tôi đồng ý để học viên Vương Duy Thanh được bảo vệ luận văn trước Hội đồng chấm luận văn thạc sĩ.

Ngày 07 tháng 01 năm 2022
NGƯỜI NHẬN XÉT
(Ký và ghi rõ họ tên)

Võ Thị Lưu Phương

BỘ THÔNG TIN VÀ TRUYỀN THÔNG
HỌC VIỆN CÔNG NGHỆ
BƯU CHÍNH VIỄN THÔNG

CỘNG HOÀ XÃ HỘI CHỦ NGHĨA VIỆT NAM
Độc lập - Tự do - Hạnh phúc

BẢN NHẬN XÉT LUẬN VĂN TỐT NGHIỆP THẠC SĨ

(*Dùng cho người phản biện*)

Tên đề tài luận văn: **Nghiên cứu ứng dụng AI xây dựng thuật toán dự báo các tác vụ trên đám mây nhằm nâng cao hiệu quả cân bằng tải.**

Chuyên ngành: Hệ thống thông tin

Mã số: 60 48 01 04

Tên học viên: **Vương Duy Thanh**

Họ và tên người nhận xét: Nguyễn Đình Thuân

Học hàm, học vị: PGS. TS

Chuyên ngành: Công nghệ thông tin

Cơ quan công tác: Trường Đại học Công nghệ thông tin- ĐHQGTP.HCM

NỘI DUNG NHẬN XÉT

I/ Cơ sở khoa học và thực tiễn, tính cấp thiết của đề tài:

Nghiên cứu ứng dụng AI và xây dựng thuật toán dự báo các tác vụ trên đám mây nhằm nâng cao hiệu quả cân bằng tải là một trong những vấn đề được quan tâm trong thời gian gần đây.

II/ Về nội dung, chất lượng của luận văn, các kết quả đã đạt được (so với đề cương đã được duyệt):

Luận văn đã thực hiện:

- Phân tích nhu cầu ứng dụng các thuật toán học máy (machine learning) trong dự báo các tác vụ đám mây nhằm nâng cao hiệu quả cân bằng tải.
- Xây dựng ứng dụng cài đặt 5 thuật toán Round Robin, MaxMix, MinMin, FCFS, và thuật toán đề xuất ACTPA.
- So sánh các thuật toán trên môi trường đám mây giả lập sử dụng thư viện CloudSim với 5 máy ảo và số lượng request là 30, 60, 100, 1000.

III/ Những vấn đề cần giải thích thêm:

- Cần xem lại tính chính xác của Sơ đồ thuật toán đề xuất ACTPA (trang 35)
- Cần giải thích rõ hơn từ "AI" trong luận văn.

IV/ Kết luận:

Luận văn đáp ứng các yêu cầu của luận văn thạc sĩ chuyên ngành Hệ thống thông tin.

Tôi đồng ý để học viên Nguyễn Thanh Nhân được bảo vệ luận văn trước Hội đồng chấm luận văn thạc sĩ.

Ngày 10 tháng 01 năm 2022
NGƯỜI NHẬN XÉT

PGS. TS. Nguyễn Đình Thuân