**ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH**

**TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN**

**KHOA HỆ THỐNG THÔNG TIN**





**ĐỒ ÁN CUỐI KỲ**

**ỨNG DỤNG AMAZON WEB SERVICES ĐỂ HUẤN LUYỆN VÀ DỰ ĐOÁN SỰ LAN TRUYỀN THÔNG TIN TRÊN MẠNG XÃ HỘI**

GVHD: **ThS. Thái Bảo Trân**

Lớp: **IS353.O21 – Mạng Xã Hội**

Nhóm thực hiện:

**Hoàng Quý Mùi 21521147**

**Dương Ngọc Hải 20521275**

**Nguyễn Hải Đăng 20521158**

**Nguyễn Hữu Khắc Phục 19520851**

***TP.HCM, Tháng 05 năm 2024***

**NHẬN XÉT CỦA GIÁO VIÊN**

# LỜI CẢM ƠN

Trước tiên, chúng em xin bày tỏ lòng biết ơn sâu sắc và chân thành nhất đến Ban Giám hiệu Trường Đại học Công Nghệ Thông Tin - Đại học Quốc gia Thành phố Hồ Chí Minh và Khoa Hệ Thống Thông Tin. Chúng em vô cùng biết ơn vì đã tạo điều kiện hỗ trợ và giúp đỡ chúng em trong suốt quá trình học tập và thực hiện đồ án môn học này.

Tiếp theo, chúng em xin gửi lời cảm ơn chân thành tới giảng viên của môn học, cô Thái Bảo Trân. Chúng em rất biết ơn cô đã truyền đạt những kiến thức chuyên sâu, tận tình hướng dẫn, quan tâm và động viên chúng em trong suốt quá trình thực hiện đề tài. Những tài liệu và kinh nghiệm mà cô chia sẻ đã là hành trang vô cùng quý giá cho chúng em.

Cuối cùng, chúng em xin gửi lời cảm ơn chân thành đến tất cả các bạn trong nhóm. Cảm ơn các bạn đã cùng nhau chia sẻ công việc, hoàn thành tốt trách nhiệm của cá nhân dưới sự hướng dẫn của cô và sự phân công của nhóm trưởng. Các bạn là những nhân tố quan trọng không thể thiếu, là chìa khóa để hoàn thành đề tài.

Mặc dù đã cố gắng hoàn thành đề tài với tất cả nỗ lực, nhưng chúng em vẫn mong nhận được sự thông cảm và những đóng góp, nhận xét quý báu từ cô. Những lời góp ý từ cô sẽ là hành trang vô cùng quý giá để chúng em vận dụng cho những môn học khác trong tương lai.

Chúng em xin chân thành cảm ơn!

Thành phố Hồ Chí Minh, tháng năm

Nhóm sinh viên thực hiện

**MỤC LỤC**

[**LỜI CẢM ƠN 3**](#_heading=h.30j0zll)

[**MỤC LỤC 4**](#_heading=h.1fob9te)

[**DANH MỤC HÌNH ẢNH 6**](#_heading=h.3znysh7)

[**DANH MỤC BẢNG 8**](#_heading=h.tyjcwt)

[**CHƯƠNG 1: GIỚI THIỆU ĐỀ TÀI 9**](#_heading=h.2et92p0)

[1.1](#_heading=h.3dy6vkm) 9

[1.2](#_heading=h.1t3h5sf) 9

[**CHƯƠNG 2: TỔNG QUAN VỀ AWS 11**](#_heading=h.4d34og8)

[2.1 Amazon S3 11](#_heading=h.17dp8vu)

[2.2 Amazon RDS 14](#_heading=h.lnxbz9)

[2.3 Amazon EMR 16](#_heading=h.1ksv4uv)

[2.4 AWS Glue 19](#_heading=h.2jxsxqh)

[2.5 SageMaker 20](#_heading=h.3j2qqm3)

[2.6 Amazon QuickSigh 21](#_heading=h.1y810tw)

[2.7 Amazon CloudWatch 23](#_heading=h.4i7ojhp)

[**CHƯƠNG 3: THU THẬP VÀ TIỀN XỬ LÝ DỮ LIỆU TỪ MẠNG XÃ HỘI 26**](#_heading=h.2xcytpi)

[3.1 Bộ dữ liệu 26](#_heading=h.1ci93xb)

[3.2 Tiền xử lý 28](#_heading=h.3as4poj)

[**CHƯƠNG 4: LƯU TRỮ VÀ XỬ LÝ DỮ LIỆU 29**](#_heading=h.49x2ik5)

[4.1 Amazon S3 29](#_heading=h.2p2csry)

[4.2 Amazon EMR 32](#_heading=h.2grqrue)

[**CHƯƠNG 5: MÔ HÌNH HÓA LAN TRUYỀN THÔNG TIN 39**](#_heading=h.111kx3o)

[5.1 Huấn luyện mô hình trên AWS SageMaker 39](#_heading=h.3l18frh)

[5.2 Đánh giá và tối ưu mô hình 44](#_heading=h.kgcv8k)

[**CHƯƠNG 6: TRIỂN KHAI MÔ HÌNH VÀ DỰ ĐOÁN TRÊN AWS 47**](#_heading=h.1x0gk37)

[6.1 Triển khai mô hình với AWS Lambda 47](#_heading=h.4h042r0)

[6.2 Tạo API với Amazon API Gateway 50](#_heading=h.48pi1tg)

[**CHƯƠNG 7: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN 54**](#_heading=h.1gf8i83)

[7.1 Tóm tắt kết quả 54](#_heading=h.40ew0vw)

[7.2 Hướng nghiên cứu 54](#_heading=h.2fk6b3p)

[**PHÂN CÔNG CÔNG VIỆC 55**](#_heading=h.upglbi)

[**TÀI LIỆU THAM KHẢO 56**](#_heading=h.1pxezwc)

# DANH MỤC HÌNH ẢNH

[Hình 1. Logo Amazon Web Service 11](#_heading=h.2s8eyo1)

[Hình 2. Cấu trúc Amazon S3 12](#_heading=h.3rdcrjn)

[Hình 3. Bucket 13](#_heading=h.26in1rg)

[Hình 4. Cấu trúc Amazon EMR 17](#_heading=h.44sinio)

[Hình 5. Logo Amazon Glue 19](#_heading=h.z337ya)

[Hình 6. Lệnh chạy và quá trình thu thập dữ liệu 500 bài tweet mới nhất 27](#_heading=h.3whwml4)

[Hình 7. Kết quả được lưu vào file excel 28](#_heading=h.2bn6wsx)

[Hình 8. FIle kết quả 28](#_heading=h.qsh70q)

[Hình 9. Tạo Bucket 29](#_heading=h.147n2zr)

[Hình 10. Đặt tên cho Bucket 29](#_heading=h.3o7alnk)

[Hình 11. Nhấn tạo Bucket 30](#_heading=h.23ckvvd)

[Hình 12. Lệnh “aws configure” để liên kết local với AWS 30](#_heading=h.ihv636)

[Hình 13. Đoạn script giúp ta upload file twitter\_data lên bucket 31](#_heading=h.32hioqz)

[Hình 14. Màn hình console in ra kết quả sau khi chạy đoạn script xong 31](#_heading=h.1hmsyys)

[Hình 15. file twitter\_data.csv trên bucket 32](#_heading=h.41mghml)

[Hình 16. Khởi tạo một cluster 32](#_heading=h.vx1227)

[Hình 17. Đặt tên và lựa chọn “Spark” framework 33](#_heading=h.3fwokq0)

[Hình 18. Điều chỉnh các thông số 33](#_heading=h.1v1yuxt)

[Hình 19. Security configuration and EC2 key pair 34](#_heading=h.4f1mdlm)

[Hình 20. EC2 instance profile for Amazon EMR 34](#_heading=h.2u6wntf)

[Hình 21. Create cluster để tạo cluster 34](#_heading=h.19c6y18)

[Hình 22. Cluster được tạo 35](#_heading=h.3tbugp1)

[Hình 23. Script upload file emr.py lên bucket 35](#_heading=h.28h4qwu)

[Hình 24. Upload thành công 36](#_heading=h.nmf14n)

[Hình 25. Tạo job ở cluster 36](#_heading=h.37m2jsg)

[Hình 26. Chọn cluster mode 37](#_heading=h.1mrcu09)

[Hình 27. Chọn file emr.py ở bucket và tạo job 37](#_heading=h.46r0co2)

[Hình 28. Job đang hiện là pending 38](#_heading=h.2lwamvv)

[Hình 29. tạo một Notebook Instance ở danh mục Notebook 39](#_heading=h.206ipza)

[Hình 30. Đặt tên và nhấn “Create notebook instance” để tạo notebook instance 39](#_heading=h.4k668n3)

[Hình 31. Mở JupyterLab và tạo một file notebook mới 39](#_heading=h.2zbgiuw)

[Hình 32. Tìm IAM role mà twitter-training này được cấp 40](#_heading=h.1egqt2p)

[Hình 33. Tạo một permission mới 40](#_heading=h.3ygebqi)

[Hình 34. Script cấp quyền cho Notebook Instance 41](#_heading=h.2dlolyb)

[Hình 35. Script cấp quyền được hiển thị 41](#_heading=h.sqyw64)

[Hình 36. Vào mục permission để tiến hành cấp các quyền cho bucket 42](#_heading=h.3cqmetx)

[Hình 37. Script cấp quyền 42](#_heading=h.1rvwp1q)

[Hình 38. Thiết lập cấu hình để kết nối với bucket S3 42](#_heading=h.4bvk7pj)

[Hình 39. Lấy dữ liệu twitter\_data.csv từ bucket 43](#_heading=h.2r0uhxc)

[Hình 40. Chia train test 43](#_heading=h.1664s55)

[Hình 41. Dùng Random Forest để huấn luyện mô hình 43](#_heading=h.3q5sasy)

[Hình 42. Dùng Gradient Boosting để huấn luyện mô hình 43](#_heading=h.25b2l0r)

[Hình 43. Công thức R Square 44](#_heading=h.34g0dwd)

[Hình 44. Công thức MAE 44](#_heading=h.1jlao46)

[Hình 45. Tìm các tham số tối ưu nhất 45](#_heading=h.2iq8gzs)

[Hình 46. Mô hình đã được tối ưu hơn 46](#_heading=h.3hv69ve)

[Hình 47. Lưu và upload mô hình trên S3 bucket 47](#_heading=h.2w5ecyt)

[Hình 48. Tạo function ở AWS Lambda 47](#_heading=h.1baon6m)

[Hình 49. Thiết lập thông tin 48](#_heading=h.3vac5uf)

[Hình 50. Điền dầy đủ thông tin và tạo 48](#_heading=h.2afmg28)

[Hình 51. Tạo một mã nguồn triển khai mô hình 49](#_heading=h.pkwqa1)

[Hình 52. Lambda function được lưu trong file zip 50](#_heading=h.39kk8xu)

[Hình 53. Màn hình sau khi upload xong 50](#_heading=h.1opuj5n)

[Hình 54. tạo một REST API 50](#_heading=h.2nusc19)

[Hình 55. Tạo một method mới 51](#_heading=h.1302m92)

[Hình 56. Kết quả sau khi tạo xong 51](#_heading=h.3mzq4wv)

[Hình 57. Deploy API 52](#_heading=h.2250f4o)

[Hình 58. Script sử dụng model đã được triển khai trên Amazon 53](#_heading=h.haapch)

[Hình 59. Code tiến hành dự đoán và đưa ra kết quả 53](#_heading=h.319y80a)

# DANH MỤC BẢNG

[Bảng 1. Bảng so sánh các loại DB instance storage 15](#_heading=h.35nkun2)

[Bảng 2. So sánh độ đo của RandomForestRegressor và GradientBoostingRegressor 45](#_heading=h.43ky6rz)

[Bảng 3. Độ đo của RandomForestRegressor sau khi được tối ưu 45](#_heading=h.xvir7l)

[Bảng 4. Bảng phân công công việc 55](#_heading=h.3ep43zb)

# CHƯƠNG 1: GIỚI THIỆU ĐỀ TÀI

## Ý nghĩa của việc mô hình hóa lan truyền thông tin trên mạng xã hội

Việc mô hình hóa lan truyền thông tin trên mạng xã hội là vô cùng quan trọng trong thời đại kỹ thuật số ngày nay. Trên các nền tảng mạng xã hội như Facebook, Twitter, YouTube, etc., thông tin và nội dung có thể lan truyền với tốc độ chóng mặt, ảnh hưởng đến rất nhiều người dùng.

Hiểu được cách thức và động lực của quá trình lan truyền thông tin trên các mạng xã hội này có thể giúp chúng ta:

* Phát hiện và ngăn chặn hiệu quả các thông tin sai lệch, tin giả, và nội dung có hại.
* Tối ưu hóa việc truyền tải thông tin, nội dung và chiến dịch truyền thông của các tổ chức, doanh nghiệp.
* Phân tích và dự đoán xu hướng, hành vi người dùng trên các nền tảng mạng xã hội.
* Hiểu sâu hơn về các động lực xã hội, tâm lý học và các khía cạnh nhân văn khác liên quan đến sự lan truyền thông tin.

Bằng cách mô hình hóa quá trình lan truyền thông tin trên mạng xã hội, chúng ta có thể khai thác được nhiều bài học quý giá, góp phần tạo dựng một không gian mạng lành mạnh, an toàn và có trách nhiệm hơn. Đây là một lĩnh vực đang được quan tâm và nghiên cứu sâu rộng trong thời gian gần đây.

## Dự đoán lượt chia sẻ có tác dụng gì?

Dự đoán lượt chia sẻ (share) đóng vai trò quan trọng trong việc xác định mức độ lan truyền của nội dung trên các nền tảng mạng xã hội. Thông qua phân tích các mẫu chia sẻ, doanh nghiệp có thể hiểu rõ hơn về việc người dùng tương tác với các loại nội dung khác nhau và chia sẻ chúng với cộng đồng của họ.

Các số liệu dự đoán về lượt chia sẻ giúp doanh nghiệp xác định được những chủ đề, định dạng nội dung nào thu hút sự quan tâm và tương tác của người dùng nhiều nhất. Từ đó, họ có thể điều chỉnh chiến lược nội dung để tạo ra những bài viết, hình ảnh, video có khả năng lan truyền rộng rãi trên mạng xã hội.

Hơn nữa, dự đoán lượt chia sẻ còn giúp doanh nghiệp đánh giá được hiệu quả của các hoạt động tiếp thị trên mạng xã hội. Bằng cách theo dõi các mẫu chia sẻ, họ có thể xác định được những nội dung nào thu hút được sự quan tâm và chia sẻ nhiều nhất, từ đó điều chỉnh chiến lược tiếp thị cho phù hợp.

Tóm lại, dự đoán lượt chia sẻ đóng vai trò then chốt trong việc xác định mức độ lan truyền của nội dung, giúp doanh nghiệp tối ưu hóa các hoạt động tiếp thị trên mạng xã hội và thu hút sự tương tác từ người dùng..

# CHƯƠNG 2: TỔNG QUAN VỀ AWS

Amazon Web Services (AWS) là nền tảng điện toán đám mây hàng đầu trên thế giới. Được thành lập bởi Amazon vào năm 2006, AWS cung cấp nhiều dịch vụ đám mây đa dạng và phong phú, bao gồm điện toán, lưu trữ, cơ sở dữ liệu, machine learning, phân tích,... Với mạng lưới trung tâm dữ liệu toàn cầu, AWS đảm bảo tính sẵn sàng, khả năng mở rộng và độ tin cậy cao. Điều khiến AWS trở nên khác biệt là mô hình định giá trả theo mức sử dụng, cho phép người dùng chỉ trả tiền cho những tài nguyên họ sử dụng, loại bỏ nhu cầu đầu tư trả trước. AWS hỗ trợ các tổ chức đổi mới, mở rộng quy mô nhanh chóng và duy trì tính linh hoạt trong thế giới kỹ thuật số ngày càng phát triển, khiến giải pháp này trở thành lựa chọn phù hợp cho các công ty khởi nghiệp, doanh nghiệp và nhà phát triển đang tìm cách tận dụng lợi ích của điện toán đám mây.



*Hình 1. Logo Amazon Web Service*

Hiện nay AWS (Amazon Web Services) là một nền tảng điện toán đám mây hàng đầu trên thế giới, cung cấp một loạt các dịch vụ và công cụ mạnh mẽ để xử lý dữ liệu mạng xã hội. Với sự phát triển nhanh chóng của các mạng xã hội và dữ liệu lớn mà chúng tạo ra, AWS đã trở thành một công cụ quan trọng cho việc thu thập, lưu trữ, xử lý và phân tích dữ liệu từ các nền tảng mạng xã hội hàng đầu như Facebook, Twitter, Instagram và nhiều khía cạnh khác của Internet. Nhờ vào khả năng tính toán và lưu trữ mạnh mẽ cùng với các công cụ phân tích, AWS giúp các doanh nghiệp tận dụng dữ liệu mạng xã hội để hiểu rõ hơn về khách hàng, tạo ra chiến lược tiếp thị tốt hơn và đưa ra quyết định kinh doanh thông minh hơn.

## 2.1 Amazon S3

**Amazon Simple Storage Service (Amazon S3)** là một dịch vụ lưu trữ đối tượng cung cấp khả năng thay đổi quy mô, mức độ sẵn sàng của dữ liệu, độ bảo mật và hiệu suất hàng đầu trong ngành. Khách hàng thuộc mọi quy mô và ngành nghề có thể lưu trữ và bảo vệ dữ liệu thuộc mọi kích thước cho hầu hết tất cả các trường hợp sử dụng, chẳng hạn như hồ dữ liệu, ứng dụng hoạt động trên đám mây và ứng dụng di động. Với các lớp lưu trữ tiết kiệm chi phí và tính năng quản lý dễ sử dụng, bạn có thể tối ưu hóa chi phí, tổ chức dữ liệu và cấu hình các biện pháp kiểm soát quyền truy cập được tinh chỉnh để đáp ứng yêu cầu cụ thể về kinh doanh, tổ chức và tuân thủ.

A screenshot of a computer

Description automatically generated

*Hình 2. Cấu trúc Amazon S3*

Amazon S3 lưu trữ dữ liệu dưới dạng objects trong buckets. Một object là một tệp hay bất kỳ siêu dữ liệu (metadata) nào trong tệp đó. Một bucket là một vùng chứa cho các đối tượng. Để lưu trữ dữ liệu của bạn trong Amazon S3, bạn phải tạo một bucket và chỉ định tên vùng chứa cũng như AWS Region. Sau đó, bạn tải dữ liệu lên bucket đó dưới dạng object trong Amazon S3. Mỗi object có một key, là mã định danh duy nhất cho đối tượng trong nhóm.

**Objects** là các thực thể cơ bản được lưu trữ trong Amazon S3. Nó bao gồm dữ liệu đối tượng (object data) và siêu dữ liệu (metadata). Siêu dữ liệu là một tập hợp các cặp name-value mô tả đối tượng. Các cặp này bao gồm một số siêu dữ liệu mặc định, chẳng hạn như ngày sửa đổi lần cuối và siêu dữ liệu HTTP tiêu chuẩn,... Một object được xác định duy nhất trong một bucket bằng key (name) và version ID.

**Key** (hay key name) là mã định danh duy nhất cho một object trong bucket. Mỗi object trong một bucket có chính xác một key. Mỗi object trong Amazon S3 có thể được xử lý duy nhất thông qua sự kết hợp của điểm cuối web service, bucket name, key và một version.

**Buckets** là nơi chứa các đối tượng được lưu trữ trong Amazon S3. Bạn có thể lưu trữ bất kỳ đối tượng nào trong một nhóm và có thể có tối đa 100 nhóm trong tài khoản của mình. Khi tạo bucket, bạn nhập tên bộ chứa và chọn AWS Region nơi bộ chứa sẽ cư trú. Sau khi bạn tạo một bucket, bạn không thể thay đổi tên hoặc Region (Vùng) của nó. Như thế, một bucket giống như một vùng chứa cho các object lưu trữ trên Amazon S3. Mỗi object sẽ được chứa trong một bucket.

A diagram of different types of objects

Description automatically generated

*Hình 3. Bucket*

* Sắp xếp, quản lý Amazon S3 namespace ở mức cao nhất.
* Xác định tài khoản chịu trách nhiệm về phí lưu trữ và truyền dữ liệu.
* Cung cấp các tùy chọn kiểm soát truy cập, chẳng hạn như chính sách bộ chứa, danh sách kiểm soát truy cập (ACL) và Điểm truy cập S3 mà bạn có thể sử dụng để quản lý quyền truy cập vào tài nguyên Amazon S3 của mình.
* Đóng vai trò là đơn vị tổng hợp cho báo cáo sử dụng.

Redshift và Amazon S3 cung cấp storage layer thống nhất, được tích hợp nguyên bản của kiến ​​trúc tham chiếu Lakehouse. Thông thường, Amazon Redshift lưu trữ dữ liệu đáng tin cậy, phù hợp và được quản lý cao, được cấu trúc thành các schema chiều tiêu chuẩn, trong khi Amazon S3 cung cấp bộ lưu trữ data lake quy mô exabyte cho dữ liệu có cấu trúc, bán cấu trúc và phi cấu trúc.

Với hỗ trợ dữ liệu bán cấu trúc trong Amazon Redshift, bạn cũng có thể nhập và lưu trữ dữ liệu bán cấu trúc trong data warehouse Amazon Redshift của mình. Amazon S3 cung cấp khả năng mở rộng, khả năng cung cấp dữ liệu, bảo mật và hiệu suất hàng đầu trong ngành. Các tổ chức thường lưu trữ dữ liệu trong Amazon S3 bằng các định dạng tệp mở. Các định dạng tệp mở cho phép phân tích cùng một dữ liệu Amazon S3 bằng cách sử dụng nhiều thành phần lớp xử lý và tiêu thụ vì thế Storage Layer cho phép lưu trữ dữ liệu cấu trúc (ví dụ: cơ sở dữ liệu quan hệ) và dữ liệu bán cấu trúc (ví dụ: dữ liệu JSON, Avro) trong cùng một hệ thống. Điều này giúp tổ chức và truy cập dữ liệu một cách hiệu quả và linh hoạt.

Lớp danh mục chung lưu trữ các schema của tập dữ liệu có cấu trúc hoặc bán cấu trúc trong Amazon S3. Các thành phần sử dụng tập dữ liệu S3 thường áp dụng schema này cho tập dữ liệu khi chúng đọc nó (hay còn gọi là schema-on-read).

Trong S3 Data Lake, cả dữ liệu có cấu trúc và không có cấu trúc đều được lưu trữ dưới dạng các đối tượng S3. Các đối tượng S3 trong data lake được tổ chức thành các nhóm hoặc tiền tố đại diện cho các vùng landing, raw, trusted, và curated. Đối với các pipeline lưu trữ dữ liệu trong data lake S3, dữ liệu được nhập từ nguồn vào vùng đích như hiện tại.

## 2.2 Amazon EMR

AWS EMR là dịch vụ cho phép bạn chạy các framework dữ liệu lớn như Apache Spark và Apache Hadoop trên AWS. EMR được sử dụng trong processing layer bằng cách cung cấp nền tảng cụm dễ quản lý giúp đơn giản hóa hoạt động phân tích, thực hiện các thao tác ETL trên lượng lớn dữ liệu trong thời gian gần thực, đưa dữ liệu lớn vào và ra khỏi data stores thuộc AWS khác như S3.

A screenshot of a computer

Description automatically generated

*Hình 4. Cấu trúc Amazon EMR*

Thành phần trung tâm của Amazon EMR là cụm. Cụm là tập hợp các phiên bản Amazon Elastic Computing Cloud (Amazon EC2). Mỗi phiên bản trong cụm được gọi là một nút. Mỗi nút có một vai trò trong cụm. Amazon EMR cũng cài đặt các thành phần phần mềm khác nhau trên từng loại nút, trao cho mỗi nút một vai trò trong ứng dụng phân tán như Apache Hadoop.

Các loại nút trong Amazon EMR như sau:

* Nút chính: Nút quản lý cụm bằng cách chạy các thành phần phần mềm để điều phối việc phân phối dữ liệu và nhiệm vụ giữa các nút khác để xử lý. Nút chính theo dõi trạng thái của các tác vụ và theo dõi tình trạng của cụm. Mỗi cụm có một nút chính và có thể tạo một cụm nút đơn chỉ với nút chính.
* Nút lõi: Nút có các thành phần phần mềm chạy tác vụ và lưu trữ dữ liệu trong Hệ thống tệp phân tán Hadoop (HDFS) trên cụm của bạn. Các cụm nhiều nút có ít nhất một nút lõi.
* Nút tác vụ: Nút có các thành phần phần mềm chỉ chạy các tác vụ và không lưu trữ dữ liệu trong HDFS. Các nút nhiệm vụ là tùy chọn.

Khả năng mở rộng của các cụm mang lại khả năng mở rộng linh hoạt, cho phép doanh nghiệp điều chỉnh kích thước của các cụm để đáp ứng nhu cầu xử lý dữ liệu thay đổi.

1. **Big data processing:**

AWS EMR là giải pháp xử lý dữ liệu mạnh mẽ, cho phép bạn xây dựng ứng dụng bằng các khung phổ biến như Apache Spark và Hadoop. Nó cung cấp một hệ thống xử lý dữ liệu lớn hoàn chỉnh với các yêu cầu xử lý được quản lý tốt và có thể mở rộng.

AWS Glue là dịch vụ ETL không có máy chủ được quản lý toàn phần giúp chuyển đổi và truyền dữ liệu giữa các kho dữ liệu khác nhau dễ dàng hơn. Cung cấp kho lưu trữ siêu dữ liệu tập trung, giúp dễ dàng khám phá, chuẩn bị và kết hợp dữ liệu từ nhiều nguồn khác nhau.

Khi được sử dụng cùng nhau, AWS Glue và AWS EMR có thể tạo ra pipeline xử lý dữ liệu lớn mạnh mẽ. AWS Glue có thể được sử dụng để khám phá, chuẩn bị và kết hợp dữ liệu để phân tích, trong khi AWS EMR có thể được sử dụng để xử lý dữ liệu này bằng các khung xử lý dữ liệu mạnh mẽ như Apache Spark. Sự kết hợp này cho phép bạn tận dụng điểm mạnh của cả hai dịch vụ, tạo ra quy trình xử lý dữ liệu có khả năng mở rộng và hiệu quả cao.

1. **ETL sát thời gian thực:**

ETL là một quá trình bao gồm trích xuất dữ liệu từ các nguồn khác nhau, chuyển đổi dữ liệu sang định dạng mong muốn và tải dữ liệu đó vào hệ thống đích để phân tích hoặc sử dụng thêm. ETL gần thời gian thực có nghĩa là dữ liệu được xử lý và phân phối với độ trễ tối thiểu, thường trong vòng vài giây hoặc vài phút. Để xây dựng một ETL pipeline ta sử dụng các dịch vụ Kinesis Data Analytics, EMR, Glue:

Thu thập dữ liệu bằng AWS Kinesis:Các luồng AWS Kinesis có thể thu thập và xử lý lượng lớn dữ liệu trong thời gian thực. Với Thư viện khách hàng Amazon Kinesis (KCL), bạn có thể xây dựng các ứng dụng xử lý truyền dữ liệu theo thời gian thực để hỗ trợ bảng thông tin, tạo cảnh báo và triển khai tính năng định giá và quảng cáo linh hoạt. Bạn cũng có thể phát dữ liệu từ Kinesis Data Streams tới các dịch vụ AWS khác như Amazon S3, Amazon Redshift, Amazon EMR và AWS Lambda.

Xử lý dữ liệu theo thời gian thực với AWS EMR:

* Các cụm Amazon EMR có thể đọc và xử lý trực tiếp các luồng Amazon Kinesis bằng cách sử dụng các công cụ quen thuộc trong hệ Hadoop như Hive, Pig, MapReduce, API phát trực tuyến Hadoop và Cascading. Bạn cũng có thể kết hợp dữ liệu thời gian thực từ Amazon Kinesis với dữ liệu hiện có trên Amazon S3, Amazon DynamoDB và HDFS trong một cụm đang chạy.
* Việc tích hợp giữa Amazon EMR và Amazon Kinesis có thể giúp các tình huống xử lý dữ liệu theo thời gian thực trở nên dễ dàng hơn nhiều. Ví dụ: bạn có thể phân tích nhật ký web phát trực tuyến, viết truy vấn kết hợp dữ liệu luồng nhấp chuột từ Amazon Kinesis với thông tin chiến dịch quảng cáo được lưu trữ trong bảng DynamoDB hoặc tải định kỳ dữ liệu từ stream Amazon Kinesis vào HDFS để có các truy vấn phân tích, tương tác nhanh chóng
* Chuyển đổi dữ liệu bằng AWS Glue: AWS Glue tương thích với Cơ quan đăng ký lược đồ AWS Glue, một tính năng phi máy chủ của AWS Glue cho phép bạn xác thực và kiểm soát sự phát triển của dữ liệu truyền phát bằng cách sử dụng các schema Apache Avro đã đăng ký. AWS Glue có thể được sử dụng để khám phá, lập danh mục và chuyển đổi dữ liệu từ nhiều nguồn khác nhau. Sau đó, dữ liệu đã chuyển đổi có thể được xử lý và phân tích bằng Amazon EMR.

## 2.3 SageMaker

Amazon SageMaker là một dịch vụ được quản lý đầy đủ, kết hợp một bộ công cụ rộng lớn để kích hoạt học máy (ML) hiệu suất cao, chi phí thấp cho bất kỳ trường hợp sử dụng nào. Với SageMaker, bạn có thể xây dựng, huấn luyện và triển khai các mô hình ML quy mô lớn sử dụng các công cụ như sổ ghi chép, trình gỡ lỗi, trình phân tích hiệu suất, đường ống, MLOps và hơn thế nữa - tất cả trong một môi trường phát triển tích hợp (IDE). SageMaker hỗ trợ các yêu cầu quản trị với việc kiểm soát truy cập đơn giản hóa và minh bạch đối với các dự án ML của bạn. Ngoài ra, bạn có thể xây dựng các FM của riêng mình, những mô hình lớn được huấn luyện trên các bộ dữ liệu lớn, với các công cụ được xây dựng đặc biệt để điều chỉnh, thử nghiệm, đào tạo lại và triển khai các FM. SageMaker cung cấp quyền truy cập vào hàng trăm mô hình được huấn luyện trước, bao gồm cả các FM có sẵn công cộng, mà bạn có thể triển khai chỉ với vài cú nhấp chuột.

Lợi ích của SageMaker:

* Công cụ Machine Learning: SageMaker cung cấp một loạt công cụ để xây dựng, huấn luyện và triển khai mô hình ML ở quy mô, bao gồm sổ tay, trình gỡ lỗi, trình phân tích hiệu suất, đường ống, MLOps và nhiều hơn nữa trong một môi trường phát triển tích hợp (IDE).
* Cơ Sở Hạ Tầng: SageMaker cung cấp cơ sở hạ tầng hiệu suất cao, chi phí thấp và được quản lý hoàn toàn, cho phép bạn xây dựng mô hình ML của riêng mình, bao gồm cả mô hình lớn (FMs) để hỗ trợ ứng dụng AI sinh học.
* Quy Trình Làm Việc ML: SageMaker tự động hóa và chuẩn hóa các thực hành MLOps và quản trị trên toàn tổ chức, hỗ trợ minh bạch và khả năng kiểm toán.
* Human-in-the-Loop: SageMaker tận dụng sức mạnh của phản hồi từ con người trong suốt vòng đời ML để cải thiện độ chính xác và liên quan của mô hình lớn (FMs) với khả năng can thiệp của con người.

## 2.4 AWS Lambda

AWS Lambda là một dịch vụ máy tính không máy chủ (serverless) do Amazon Web Services (AWS) cung cấp. Nó cho phép bạn chạy code mà không cần phải quản lý hoặc orovision các máy chủ. Với Lambda, bạn chỉ cần tập trung vào việc viết và triển khai code, và Lambda sẽ tự động cung cấp và mở rộng các tài nguyên máy tính cần thiết để chạy code của bạn.

Một số đặc điểm chính của AWS Lambda:

* Không máy chủ: Với Lambda, bạn không cần phải quản lý hoặc orovision các máy chủ. Lambda sẽ tự động cung cấp và mở rộng tài nguyên cần thiết để chạy code của bạn.
* Tự động mở rộng: Lambda sẽ tự động mở rộng việc chạy code dựa trên lượng yêu cầu. Nó có thể mở rộng từ một yêu cầu duy nhất đến hàng nghìn yêu cầu mà không cần can thiệp của người dùng.
* Tính sẵn sàng cao: Lambda cung cấp tính sẵn sàng cao và khả năng chịu lỗi, tự động khắc phục sự cố và phân phối lại các tài nguyên khi cần thiết.
* Tính toán theo yêu cầu: Bạn chỉ phải trả tiền cho lượng tính toán mà bạn sử dụng, không phải trả tiền cho các máy chủ đang chạy.
* Hỗ trợ nhiều ngôn ngữ lập trình: Lambda hỗ trợ nhiều ngôn ngữ lập trình phổ biến như Node.js, Python, Java, C#, Go và Ruby.
* Tích hợp với các dịch vụ AWS khác: Lambda có thể được tích hợp với nhiều dịch vụ AWS khác như Amazon S3, Amazon DynamoDB, Amazon API Gateway, v.v. để tạo ra các ứng dụng hoàn chỉnh.

Người dùng có thể sử dụng Lambda để xử lý dữ liệu theo yêu cầu, phản hồi các sự kiện trong thời gian thực, xây dựng API, tự động hóa quy trình kinh doanh và nhiều tác vụ khác mà không cần phải quản lý cơ sở hạ tầng.

Tóm lại, AWS Lambda cung cấp một nền tảng tính toán linh hoạt và không máy chủ, cho phép người dùng tập trung vào việc viết code hơn là quản lý cơ sở hạ tầng. Nó là một công cụ rất hữu ích trong việc xây dựng các ứng dụng và dịch vụ mới trên nền tảng AWS.

## 2.5 Amazon API Gateway

Amazon API Gateway là một dịch vụ quản lý và vận hành API do Amazon Web Services (AWS) cung cấp. Nó cho phép bạn tạo, triển khai, bảo trì, theo dõi và bảo vệ các API cho bất kỳ ứng dụng nào, bao gồm các ứng dụng không dùng máy chủ (serverless).

Một số tính năng chính của Amazon API Gateway:

* Quản lý API toàn diện: API Gateway cho phép bạn tạo, triển khai, bảo vệ, giám sát và điều chỉnh các API một cách dễ dàng. Nó hỗ trợ các phương thức HTTP, WebSocket và REST.
* Không máy chủ: API Gateway hoạt động tốt với các dịch vụ không dùng máy chủ như AWS Lambda, cho phép bạn tạo và chạy ứng dụng mà không cần phải quản lý cơ sở hạ tầng.
* Bảo mật và kiểm soát truy cập: API Gateway cung cấp các tính năng bảo mật và kiểm soát truy cập mạnh mẽ như xác thực, ủy quyền, giới hạn tốc độ, etc.
* Tích hợp với các dịch vụ AWS khác: API Gateway có thể tích hợp với nhiều dịch vụ khác của AWS như Lambda, DynamoDB, EC2, v.v. để tạo ra các ứng dụng phức tạp.
* Giám sát và phân tích: API Gateway cung cấp các tính năng giám sát và phân tích mạnh mẽ, cho phép theo dõi và phân tích lưu lượng API.
* Tự động mở rộng: API Gateway tự động mở rộng để xử lý lưu lượng tăng, đảm bảo hiệu suất tối ưu.
* Điều khiển lưu lượng: API Gateway cung cấp các tính năng để kiểm soát và điều chỉnh lưu lượng API, như giới hạn tốc độ, cân bằng tải, v.v.

Người dùng có thể sử dụng API Gateway để tạo ra các API mới hoặc sử dụng lại các API hiện có, sau đó triển khai chúng một cách an toàn và có thể mở rộng. Nó đặc biệt hữu ích khi xây dựng các ứng dụng không dùng máy chủ, microservices, và các ứng dụng di động.

Tóm lại, Amazon API Gateway là một dịch vụ quản lý API mạnh mẽ, cho phép bạn tạo, triển khai và bảo vệ các API một cách dễ dàng, đồng thời tích hợp chúng với các dịch vụ AWS khác.

# CHƯƠNG 3: THU THẬP VÀ TIỀN XỬ LÝ DỮ LIỆU TỪ MẠNG XÃ HỘI

## 3.1 Bộ dữ liệu

* Nguồn thu thập: Twitter
* Phương pháp thu thập: Web Scraping với Selenium
* Nội dung thu thập: 1 tweet gồm có

'name': tên user

'handle': tên username

'timestamp': ngày đăng bài

'verified': tài khoản đăng được xác minh hay chưa

'content': nội dung bài tweet

'comments': số lượng comment

'retweets': số lượt chia sẻ

'likes': số lượng like

‘analytics’: số lượt xem tweet

‘tags’: hashtag có trong bài đăng

'mentions': những người được đề cập

'emojis': các cảm xúc được thả

‘profile image’: ảnh đại diện chủ sở hữu

'tweet link': link bài tweet

'tweet id’: id bài tweet

* Lệnh chạy và quá trình thu thập dữ liệu 500 bài tweet mới nhất

A screenshot of a computer

Description automatically generated

*Hình 6. Lệnh chạy và quá trình thu thập dữ liệu 500 bài tweet mới nhất*

* Kết quả được lưu vào file excel

A screenshot of a computer

Description automatically generated

*Hình 7. Kết quả được lưu vào file excel*

* File kết quả:

A screenshot of a computer screen

Description automatically generated

*Hình 8. FIle kết quả*

# CHƯƠNG 4: LƯU TRỮ VÀ XỬ LÝ DỮ LIỆU

## 4.1 Amazon S3

Để lưu trữ file data nhóm đã crawl về, đầu tiên ta cần tạo một bucket bằng cách sử dụng công cụ Amazon S3. Để có thể truy cập tới công cụ S3, ta cần mở Amazon Console Home lên và tìm kiếm “S3” ở thanh tìm kiếm là có thể thấy công cụ S3 xuất hiện ở đầu tiên.

Sau đó, ta nhấn vào “Create bucket” để bắt đầu tạo một bucket mới.

A screenshot of a computer

Description automatically generated

*Hình 9. Tạo Bucket*

Tiếp theo là đặt tên và giữ nguyên các option đã được gợi ý cho bucket.

A screenshot of a computer

Description automatically generated

*Hình 10. Đặt tên cho Bucket*

Cuối cùng là nhấn vào “Create bucket”.

A screenshot of a computer

Description automatically generated

*Hình 11. Nhấn tạo Bucket*

Khi hoàn thành tạo bucket, thì lúc này ta có thể upload file data bằng 2 cách. Một là trực tiếp đưa file lên bằng chức năng sẵn có của Amazon. Hai là sử dụng script để upload gián tiếp. Ở đây, nhóm sẽ dùng một đoạn script để upload nhưng trước đó ta phải sử dụng lệnh “aws configure” để liên kết local với AWS

A black screen with white text

Description automatically generated

*Hình 12. Lệnh “aws configure” để liên kết local với AWS*

Đây là đoạn script sẽ giúp ta upload file twitter\_data lên bucket

A screen shot of a computer program

Description automatically generated

*Hình 13. Đoạn script giúp ta upload file twitter\_data lên bucket*

Sau khi chạy đoạn script xong, nếu upload thành công thì màn hình console sẽ in ra kết quả sau:  
A computer screen shot of a computer code

Description automatically generated

*Hình 14. Màn hình console in ra kết quả sau khi chạy đoạn script xong*

Và trên bucket ta sẽ thấy file twitter\_data.csv:

A screenshot of a computer

Description automatically generated

*Hình 15. file twitter\_data.csv trên bucket*

## 4.2 Amazon EMR

Sau khi đã hoàn thành tạo bucket có chứa data, bước tiếp theo ta cần làm là xử lý dữ liệu như xử lý các giá trị bị khuyết, gom nhóm các cột hay làm sạch dữ liệu thô. Ta có thể thực hiện trên Amazon EMR.

Đầu tiên ta cần tìm kiếm “EMR” ở thanh tìm kiếm để  khởi tạo một cluster để tiến hành xử lý dữ liệu:

A screenshot of a computer

Description automatically generated

*Hình 16. Khởi tạo một cluster*

Sau đó là đặt tên và lựa chọn “Spark” framework để ta có thể sử dụng spark trong việc xử lý dữ liệu:

A screenshot of a computer

Description automatically generated

*Hình 17. Đặt tên và lựa chọn “Spark” framework*

Tiếp theo là điều chỉnh các thông số khác như là Networking, ta có thể chọn theo default mà Amazon gợi ý

A screenshot of a computer

Description automatically generated

*Hình 18. Điều chỉnh các thông số*

Sau đó là Security configuration and EC2 key pair.

A screenshot of a computer

Description automatically generated

*Hình 19. Security configuration and EC2 key pair*

Nếu chưa có key pair, ta có thể nhấn vào phần Creaet key pair ở bên cạnh để tạo một EC2 key pair

Tiếp theo là phần EC2 instance profile for Amazon EMR, ta chọn Create an instance profile, và chọn All S3 buckets in this account with read and write access

A screenshot of a computer

Description automatically generated

*Hình 20. EC2 instance profile for Amazon EMR*

Cuối cùng nhấn Create cluster để tạo cluster

A close up of a computer screen

Description automatically generated

*Hình 21. Create cluster để tạo cluster*

Sau khi tạo xong ta đã có một cluster để xử lý dữ liệu:

A screenshot of a computer

Description automatically generated

*Hình 22. Cluster được tạo*

Để tạo một job thực hiện xử lý dữ liệu cho cluster đầu tiên ta phải upload file job lên bucket. Ở đây nhóm có một file job xử lý in ra số lượng bản ghi có tên là emr.py, ta sẽ tiến hành upload file emr.py lên bucket bằng đoạn script ở trên

A screen shot of a computer

Description automatically generated

*Hình 23. Script upload file emr.py lên bucket*

Upload thành công thì bucket cũng đã có file emr.py

A screenshot of a computer

Description automatically generated

*Hình 24. Upload thành công*

Sau đó ta sẽ tiến hành tạo job ở cluster

A screenshot of a computer

Description automatically generated

*Hình 25. Tạo job ở cluster*

Vì nhóm muốn sử dụng spark để xử lý dữ liệu nên ta chọn Spark application và đặt tên cho job. Với Deploy mode chúng ta chọn Cluster Mode

A screenshot of a computer

Description automatically generated

*Hình 26. Chọn cluster mode*

Sau đó là chọn file emr.py ở bucket và tạo job

A screenshot of a computer

Description automatically generated

*Hình 27. Chọn file emr.py ở bucket và tạo job*

Sau khi tạo job xong thì job sẽ đang hiện là pending

A screenshot of a computer

Description automatically generated

*Hình 28. Job đang hiện là pending*

Nếu như muốn thêm các step khác như là thêm xóa sửa dữ liệu ta có thể thêm Step để sau khi chạy job đầu tiên xong thì sẽ tiếp tục job tiếp theo.

# CHƯƠNG 5: MÔ HÌNH HÓA LAN TRUYỀN THÔNG TIN

## 5.1 Huấn luyện mô hình trên AWS SageMaker

Sau khi chạy xong các step ở cluster, tiếp theo ta sẽ sử dụng AWS SageMaker để tiến hành huấn luyện mô hình dự đoán lượt chia sẻ. Đầu tiên, ta vào công cụ SageMaker và tạo một Notebook Instance ở danh mục Notebook

A screenshot of a computer

Description automatically generated

*Hình 29. Tạo một Notebook Instance ở danh mục Notebook*

Tiếp theo là đặt tên và nhấn “Create notebook instance” để tạo notebook instance

A screenshot of a computer

Description automatically generated

*Hình 30. Đặt tên và nhấn “Create notebook instance” để tạo notebook instance*

Sau khi tạo xong ta sẽ tiến hành mở JupyterLab và tạo một file notebook mới

A screenshot of a computer

Description automatically generated

*Hình 31. Mở JupyterLab và tạo một file notebook mới*

Trước đó, ta cần cấp quyền cho twitter-training để có thể truy cập với bucket S3, ta sẽ phải tìm IAM role mà twitter-training này được cấp

A screenshot of a computer

Description automatically generated

*Hình 32. Tìm IAM role mà twitter-training này được cấp*

Và tiến hành tạo một permission mới

A screenshot of a computer

Description automatically generated

*Hình 33. Tạo một permission mới*

Dùng đoạn script sau để cấp các quyền cho Notebook Instance và lưu lại

A screen shot of a computer program

Description automatically generated

*Hình 34. Script cấp quyền cho Notebook Instance*

A screenshot of a computer

Description automatically generated

*Hình 35. Script cấp quyền được hiển thị*

Đối với bucket, ta cần vào mục permission để tiến hành cấp các quyền cần thiết

A screenshot of a computer

Description automatically generated

*Hình 36. Vào mục permission để tiến hành cấp các quyền cho bucket*

Sử dụng đoạn script sau:

A screenshot of a computer code

Description automatically generated

*Hình 37. Script cấp quyền*

Sau khi đã cấp quyền đầy đủ, ta tiến hành bật Jupyter Lab và tạo một notebook để tiến hành huấn luyện mô hình.

Đầu tiên ta tiến hành thiết lập cấu hình để kết nối với bucket S3

A white rectangular frame with blue border

Description automatically generated

*Hình 38. Thiết lập cấu hình để kết nối với bucket S3*

Tiếp theo là lấy dữ liệu twitter\_data.csv từ bucket

A screenshot of a computer code

Description automatically generated

*Hình 39. Lấy dữ liệu twitter\_data.csv từ bucket*

Sau đó dựa vào quá trình xử lý dữ liệu, ta chọn ra các đặc trưng cần thiết cho huấn luyện và chia tập huấn luyện và kiểm thử. Ở đây nhóm sẽ chia bộ dữ liệu ra làm 80% dùng để huấn luyện và 20% dùng để kiểm thử, bộ dữ liệu được chia hoàn toàn là ngẫu nhiên.

A close-up of a computer screen

Description automatically generated

*Hình 40. Chia train test*

Ở đây, nhóm sẽ sử dụng 2 mô hình máy học là Random Forest và Gradient Boosting để huấn luyện và chọn ra mô hình phù hợp nhất

A computer code on a white background

Description automatically generated

*Hình 41. Dùng Random Forest để huấn luyện mô hình*

A screenshot of a computer program

Description automatically generated

*Hình 42. Dùng Gradient Boosting để huấn luyện mô hình*

## 5.2 Đánh giá và tối ưu mô hình

Để đánh giá mô hình, nhóm sử dụng các thông số đánh giá sau:

* Cross-Validation (CV) là kỹ thuật dùng để đánh giá hiệu suất của mô hình bằng cách chia dữ liệu thành nhiều tập con (folds). Mô hình được huấn luyện trên các tập con và được đánh giá trên tập con còn lại. Điều này giúp đảm bảo rằng mô hình không bị overfitting hoặc underfitting.
* Mean Cross-Validation Score là trung bình của các 𝑅 2 R 2 score từ các lần cross-validation. Nó cung cấp một chỉ số tổng quát về hiệu suất của mô hình trên toàn bộ dữ liệu.
* R^2 Score (Coefficient of Determination) là một chỉ số đánh giá độ phù hợp của mô hình. Nó biểu thị tỷ lệ phương sai của biến phụ thuộc được giải thích bởi các biến độc lập trong mô hình.

A white background with black text and black text

Description automatically generated

*Hình 43. Công thức R Square*

* Mean Absolute Error (MAE) đo lường trung bình giá trị tuyệt đối của các lỗi giữa các giá trị dự đoán và giá trị thực tế. Nó cung cấp một chỉ số về độ chính xác của mô hình dự đoán.

A math equations and numbers

Description automatically generated with medium confidence

*Hình 44. Công thức MAE*

|  |  |  |
| --- | --- | --- |
|  | RandomForestRegressor | GradientBoostingRegressor |
| Mean Cross-Validation Score | 0.72 | 0.43 |
| R^2 Score | 0.55 | 0.40 |
| Mean Absolute Error | 3572.92 | 3572.92 |

*Bảng 2. So sánh độ đo của RandomForestRegressor và GradientBoostingRegressor*

So sánh 2 kết quả ta thấy RandomForest tối ưu hơn nên ta chọn RandomForest để huấn luyện mô hình. Tiếp theo, ta sẽ tiến hành tối ưu hóa mô hình bằng GridSearchCV(), ở đây ta sẽ tìm các tham số tối ưu nhất để có được một mô hình tối ưu.

A screenshot of a computer program

Description automatically generated

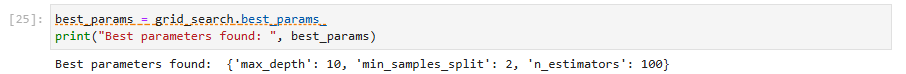
*Hình 45. Tìm các tham số tối ưu nhất*

Kết quả đánh giá mô hình sau khi được tối ưu:

|  |  |
| --- | --- |
|  | RandomForestRegressor |
| Mean Cross-Validation Score | 0.75 |
| R^2 Score | 0.58 |
| Mean Absolute Error | 3572.92 |

*Bảng 3. Độ đo của RandomForestRegressor sau khi được tối ưu*

Ta có thể thấy mô hình đã được tối ưu hơn với các tham số mới mà GridSearchSV() tìm cho ta



*Hình 46. Mô hình đã được tối ưu hơn*

# CHƯƠNG 6: TRIỂN KHAI MÔ HÌNH VÀ DỰ ĐOÁN TRÊN AWS

## 6.1 Triển khai mô hình với AWS Lambda

Sau khi đã có một mô hình dự đoán, bây giờ ta có thể sử dụng các công cụ của Amazon để triển khai mô hình hoặc giám sát và cải thiện. Trước hết là ta sẽ triển khai mô hình với AWS Lambda và đảm bảo đã lưu mô hình trên S3 bucket

A white card with colorful text

Description automatically generated

*Hình 47. Lưu và upload mô hình trên S3 bucket*

Ta sẽ cần tạo một Function ở công cụ Lambda bằng cách nhấn vào 'Create a function'

A screenshot of a computer

Description automatically generated

*Hình 48. Tạo function ở AWS Lambda*

Ta sẽ thiết lập thông tin như hình để có thể chạy python

A screenshot of a computer

Description automatically generated

*Hình 49. Thiết lập thông tin*

Tiếp theo là đặt tên cho function của bạn, chọn runtime là Python 3.x, và thiết lập IAM role có quyền truy cập S3 rồi tạo.

A screenshot of a computer

Description automatically generated

*Hình 50. Điền dầy đủ thông tin và tạo*

Sau khi đã tạo xong Lambda function, ta cần tạo một mã nguồn triển khai mô hình như sau và đóng gói .zip với các thư viện cần thiết để upload lên Function

A screenshot of a computer program

Description automatically generated

*Hình 51. Tạo một mã nguồn triển khai mô hình*

A screenshot of a computer

Description automatically generated

*Hình 52. Lambda function được lưu trong file zip*

Sau khi upload xong thì màn hình sẽ hiển thị những thông tin sau

A screenshot of a computer

Description automatically generated

*Hình 53. Màn hình sau khi upload xong*

## 6.2 Tạo API với Amazon API Gateway

Đến công cụ API Gateway, ta tạo một REST API

A screenshot of a computer

Description automatically generated

*Hình 54. tạo một REST API*

Bước tiếp theo là tạo một resource mới là /predict)và tạo một method POST cho resource này. Khi tạo method chú ý chọn Chọn "Lambda Function" cho loại Integration và chọn Lambda function đã tạo trước đó.

A screenshot of a computer

Description automatically generated

*Hình 55. Tạo một method mới*

Kết quả sau khi tạo xong

A screenshot of a computer

Description automatically generated

*Hình 56. Kết quả sau khi tạo xong*

Cuối cùng là triển khai API bằng cách chọn "Deploy API"và tạo một stage mới là prod.

A screenshot of a stage setting

Description automatically generated

*Hình 57. Deploy API*

Cuối cùng, ta sẽ sử dụng script sau để sử dụng model đã được triển khai trên Amazon

A screen shot of a computer

Description automatically generated

*Hình 58. Script sử dụng model đã được triển khai trên Amazon*

Đoạn code này sẽ cho vào những input cần thiết để mô hình tiến hành dự đoán và đưa ra kết quả

A screenshot of a computer program

Description automatically generated

*Hình 59. Code tiến hành dự đoán và đưa ra kết quả*

# CHƯƠNG 7: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

## 7.1 Tóm tắt kết quả

Nhóm đã xây dựng và triển khai thành công một mô hình dự đoán lượt chia sẻ nhằm đánh giá khả năng lan truyền của một bài tweet. Quá trình này bao gồm các bước thu thập dữ liệu, xử lý và phân tích dữ liệu, xây dựng mô hình machine learning, và triển khai mô hình trên nền tảng AWS.

Các Bước Thực Hiện:

* Thu Thập và Lưu Trữ Dữ Liệu: dữ liệu được thu thập từ Twitter và lưu trữ trên Amazon S3.
* Xử Lý và Phân Tích Dữ Liệu: sử dụng Amazon EMR để xử lý dữ liệu lớn và Amazon Athena để truy vấn dữ liệu.
* Xây Dựng Mô Hình Machine Learning: sử dụng Amazon SageMaker để huấn luyện và tối ưu hóa mô hình dự đoán lượt chia sẻ.
* Triển Khai Mô Hình: triển khai mô hình thông qua AWS Lambda và tạo API với Amazon API Gateway để cung cấp dịch vụ dự đoán theo thời gian thực.

Kết Quả Đạt Được:

* Đầu tiên là mô hình đã được triển khai thành công và có thể dự đoán khả năng lan truyền mạnh hay yếu của một bài tweet dựa trên các đặc trưng như số lượng người theo dõi, số lượt thích, số lượt retweet, và các đặc trưng khác.
* Nhóm đã hiểu  các công cụ và dịch vụ AWS như Amazon S3, AWS Lambda, Amazon API Gateway, Amazon SageMaker, Amazon EMR, Amazon Athena, Amazon CloudWatch, và AWS Step Functions. Những công cụ này giúp tối ưu hóa quá trình xây dựng, triển khai, và giám sát mô hình machine learning.

## 7.2 Hướng nghiên cứu

Dựa trên kết quả hiện tại, nhóm đề xuất một số hướng nghiên cứu tiếp theo để cải thiện và mở rộng khả năng của mô hình:

* Tích hợp thêm dữ liệu ngữ cảnh: Bao gồm thông tin về xu hướng hiện tại trên Twitter, phân tích sentiment của nội dung bài tweet, và mối quan hệ giữa các người dùng để cải thiện độ chính xác của dự đoán.
* Sử dụng mô hình học sâu (Deep Learning): Áp dụng các mô hình LSTM hoặc Transformer để hiểu rõ hơn về ngữ cảnh và nội dung bài tweet.
* Tối ưu hóa và tự động hóa: Tối ưu hóa quá trình huấn luyện và triển khai mô hình bằng cách sử dụng AWS Step Functions để tự động hóa luồng công việc.
* Giám sát và cải thiện liên tục: Sử dụng dữ liệu thực tế để liên tục giám sát và cải thiện mô hình, đảm bảo rằng mô hình duy trì được hiệu suất cao trong môi trường thay đổi liên tục của mạng xã hội.

Nhóm  hy vọng rằng mô hình và các công cụ học được sẽ đóng góp vào việc dự đoán và hiểu rõ hơn về sự lan truyền thông tin trên mạng xã hội, từ đó giúp các tổ chức và cá nhân tối ưu hóa chiến lược truyền thông của mình.

# PHÂN CÔNG CÔNG VIỆC

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tên**  **Công việc** | **Mùi** | **Hải** | **Đăng** | **Phục** |
| Chọn đề tài, dataset | **X** | **X** | **X** | **X** |
| Chương 1 |  |  | **X** |  |
| Chương 2 |  | **X** | **X** |  |
| Chương 3 |  |  |  | **X** |
| Chương 4 | **X** |  | **X** |  |
| Chương 5 | **X** |  | **X** |  |
| Chương 6 | **X** |  | **X** |  |
| Chương 7 | **X** |  | **X** |  |
| Report | **X** |  | **X** |  |
| Slide | **X** | **X** | **X** | **X** |
| Demo | **X** |  |  |  |
| Trình bày | **X** | **X** | **X** | **X** |
| Đánh giá | 100% | 100% | 100% | 100% |

*Bảng 4. Bảng phân công công việc*

# TÀI LIỆU THAM KHẢO

AWS. (2024). *AWS Management Console.* Retrieved from AWS Management Console: https://aws.amazon.com/console/

Jichang Zhao, J. W. (2010, 7 6). *Weak ties: Subtle role of information diffusion in online social networks.* Retrieved 5 2024, from PHYSICAL REVIEW E: https://tinyurl.com/479xt5dp

Joshi, H. S. (2022, 5 19). *Detect social media fake news using graph machine learning with Amazon Neptune ML.* Retrieved 5 30, 2024, from AWS Machine Learning Blog: https://tinyurl.com/34c32eup

Services, A. W. (2024). *Welcome to AWS Documentation.* Retrieved from AWS: https://tinyurl.com/488d5x4w