

**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: IS353.021 – 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

[illegible]

## **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 đề 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

<b>LỜI CẢM ƠN</b>	<b>3</b>
<b>MỤC LỤC</b>	<b>4</b>
<b>DANH MỤC HÌNH ẢNH</b>	<b>6</b>
<b>DANH MỤC BẢNG</b>	<b>8</b>
<b>CHƯƠNG 1: GIỚI THIỆU ĐỀ TÀI</b>	<b>9</b>
1.1	9
1.2	9
<b>CHƯƠNG 2: TỔNG QUAN VỀ AWS</b>	<b>11</b>
2.1 Amazon S3	11
2.2 Amazon RDS	14
2.3 Amazon EMR	16
2.4 AWS Glue	19
2.5 SageMaker	20
2.6 Amazon QuickSight	21
2.7 Amazon CloudWatch	23
<b>CHƯƠNG 3: THU THẬP VÀ TIỀN XỬ LÝ DỮ LIỆU TỪ MẠNG XÃ HỘI</b>	<b>26</b>
3.1 Bộ dữ liệu	26
3.2 Tiền xử lý	28
<b>CHƯƠNG 4: LƯU TRỮ VÀ XỬ LÝ DỮ LIỆU</b>	<b>29</b>
4.1 Amazon S3	29
4.2 Amazon EMR	32
<b>CHƯƠNG 5: MÔ HÌNH HÓA LAN TRUYỀN THÔNG TIN</b>	<b>39</b>
5.1 Huấn luyện mô hình trên AWS SageMaker	39
5.2 Đánh giá và tối ưu mô hình	44
<b>CHƯƠNG 6: TRIỂN KHAI MÔ HÌNH VÀ DỰ ĐOÁN TRÊN AWS</b>	<b>47</b>
6.1 Triển khai mô hình với AWS Lambda	47
6.2 Tạo API với Amazon API Gateway	50
<b>CHƯƠNG 7: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN</b>	<b>54</b>
7.1 Tóm tắt kết quả	54
7.2 Hướng nghiên cứu	54

## **IS353.O21 – Mạng Xã Hội**

<b>PHÂN CÔNG CÔNG VIỆC</b>	<b>55</b>
<b>TÀI LIỆU THAM KHẢO</b>	<b>56</b>

## DANH MỤC HÌNH ẢNH

Hình 1. Logo Amazon Web Service	11
Hình 2. Cấu trúc Amazon S3	12
Hình 3. Bucket	13
Hình 4. Cấu trúc Amazon EMR	17
Hình 5. Logo Amazon Glue	19
Hình 6. Lệnh chạy và quá trình thu thập dữ liệu 500 bài tweet mới nhất	27
Hình 7. Kết quả được lưu vào file excel	28
Hình 8. File kết quả	28
Hình 9. Tạo Bucket	29
Hình 10. Đặt tên cho Bucket	29
Hình 11. Nhấn tạo Bucket	30
Hình 12. Lệnh “aws configure” để liên kết local với AWS	30
Hình 13. Đoạn script giúp ta upload file twitter_data lên bucket	31
Hình 14. Màn hình console in ra kết quả sau khi chạy đoạn script xong	31
Hình 15. file twitter_data.csv trên bucket	32
Hình 16. Khởi tạo một cluster	32
Hình 17. Đặt tên và lựa chọn “Spark” framework	33
Hình 18. Điều chỉnh các thông số	33
Hình 19. Security configuration and EC2 key pair	34
Hình 20. EC2 instance profile for Amazon EMR	34
Hình 21. Create cluster để tạo cluster	34
Hình 22. Cluster được tạo	35
Hình 23. Script upload file emr.py lên bucket	35
Hình 24. Upload thành công	36
Hình 25. Tạo job ở cluster	36
Hình 26. Chọn cluster mode	37
Hình 27. Chọn file emr.py ở bucket và tạo job	37
Hình 28. Job đang hiện là pending	38
Hình 29. tạo một Notebook Instance ở danh mục Notebook	39
Hình 30. Đặt tên và nhấn “Create notebook instance” để tạo notebook instance	39
Hình 31. Mở JupyterLab và tạo một file notebook mới	39
Hình 32. Tìm IAM role mà twitter-training này được cấp	40
Hình 33. Tạo một permission mới	40
Hình 34. Script cấp quyền cho Notebook Instance	41
Hình 35. Script cấp quyền được hiển thị	41
Hình 36. Vào mục permission để tiến hành cấp các quyền cho bucket	42
Hình 37. Script cấp quyền	42
Hình 38. Thiết lập cấu hình để kết nối với bucket S3	42
Hình 39. Lấy dữ liệu twitter_data.csv từ bucket	43
Hình 40. Chia train test	43
Hình 41. Dùng Random Forest để huấn luyện mô hình	43
Hình 42. Dùng Gradient Boosting để huấn luyện mô hình	43
Hình 43. Công thức R Square	44

## IS353.O21 – Mạng Xã Hội

Hình 44. Công thức MAE	44
Hình 45. Tìm các tham số tối ưu nhất	45
Hình 46. Mô hình đã được tối ưu hơn	46
Hình 47. Lưu và upload mô hình trên S3 bucket	47
Hình 48. Tạo function ở AWS Lambda	47
Hình 49. Thiết lập thông tin	48
Hình 50. Điền đầy đủ thông tin và tạo	48
Hình 51. Tạo một mã nguồn triển khai mô hình	49
Hình 52. Lambda function được lưu trong file zip	50
Hình 53. Màn hình sau khi upload xong	50
Hình 54. tạo một REST API	50
Hình 55. Tạo một method mới	51
Hình 56. Kết quả sau khi tạo xong	51
Hình 57. Deploy API	52
Hình 58. Script sử dụng model đã được triển khai trên Amazon	53
Hình 59. Code tiến hành dự đoán và đưa ra kết quả	53

### DANH MỤC BẢNG

Bảng 1. Bảng so sánh các loại DB instance storage	15
Bảng 2. So sánh độ đo của RandomForestRegressor và GradientBoostingRegressor	45
Bảng 3. Độ đo của RandomForestRegressor sau khi được tối ưu	45
Bảng 4. Bảng phân công công việc	55



### CHƯƠNG 1: GIỚI THIỆU ĐỀ TÀI

#### 1.1 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.

#### 1.2 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.

## **IS353.O21 – Mạng Xã Hội**

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.



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.



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.

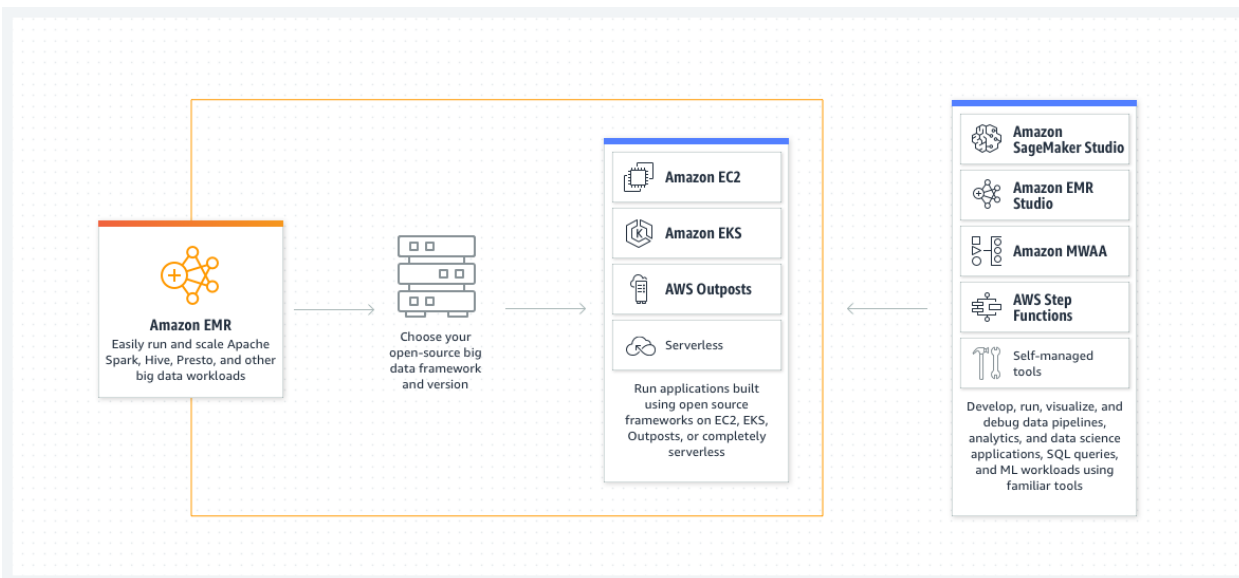
Với hỗ trợ dữ liệu bán cấu trúc trong Amazon Redshift, bạn cũng có thể nhập và lưu trữ dữ liệu bán cấu trúc trong data warehouse Amazon Redshift của mình. Amazon S3 cung cấp khả năng mở rộng, khả năng cung cấp dữ liệu, bảo mật và hiệu suất hàng đầu trong ngành. Các tổ chức thường lưu trữ dữ liệu trong 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 để 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.



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 lỗi: 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 lỗi.

## IS353.O21 – Mạng Xã Hội

- 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.

### . **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.

### **b. 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.

## IS353.O21 – Mạng Xã Hội

- 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 provision 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 provision 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:

## IS353.O21 – Mạng Xã Hội

- 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

```
E:\Python\selenium-twitter-scraper>python scraper --tweets=500
Loading .env file
Loaded .env file

Twitter Username: nguyenphuc1040
Enter Password:

Initializing Twitter Scraper...
Setup WebDriver...
Initializing FirefoxDriver...
WebDriver Setup Complete

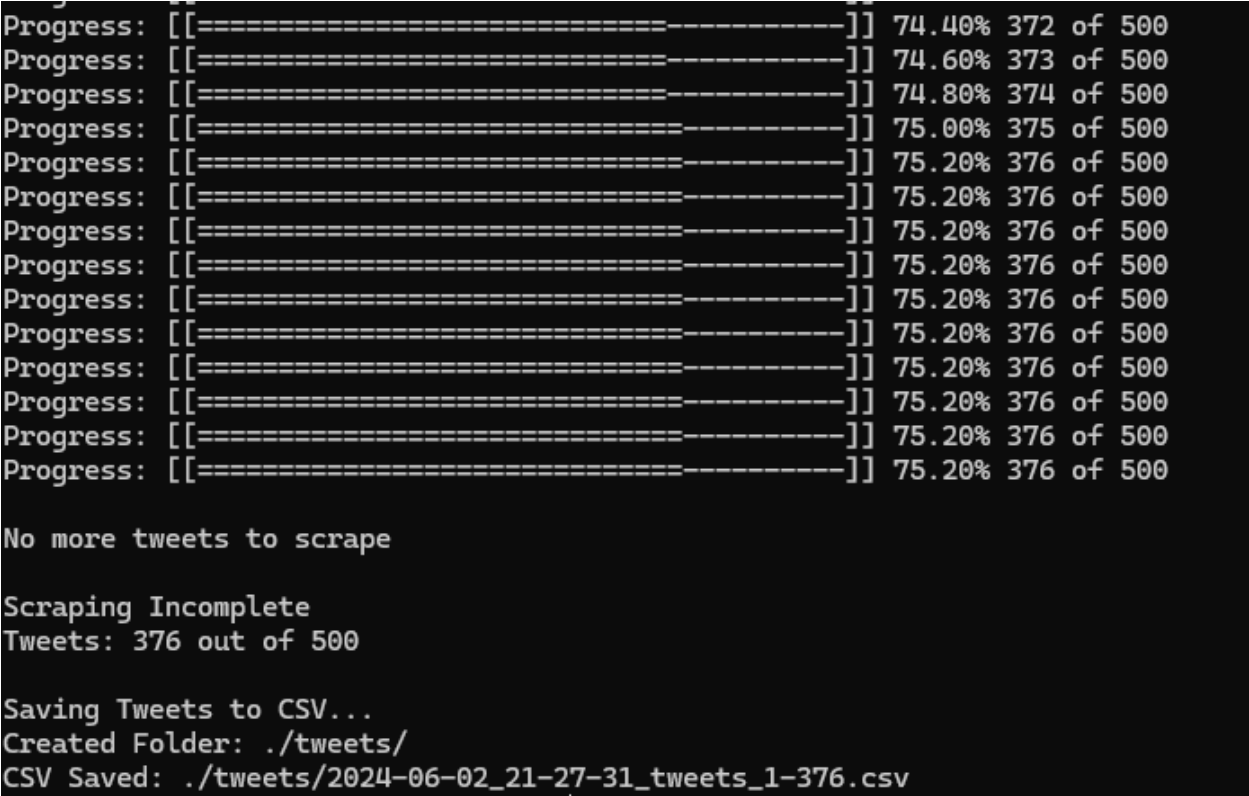
Logging in to Twitter...

Login Successful

Scraping Tweets from Home...
Progress: [[=====]] 10.00% 50 of 500
Progress: [[=====]] 10.20% 51 of 500
Progress: [[=====]] 10.40% 52 of 500
Progress: [[=====]] 10.60% 53 of 500
Progress: [[=====]] 10.80% 54 of 500
Progress: [[=====]] 11.00% 55 of 500
Progress: [[=====]] 11.20% 56 of 500
Progress: [[=====]] 11.40% 57 of 500
Progress: [[=====]] 11.60% 58 of 500
Progress: [[=====]] 11.80% 59 of 500
```

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

- Kết quả được lưu vào file excel



Hình 7. Kết quả được lưu vào file excel

• File kết quả:

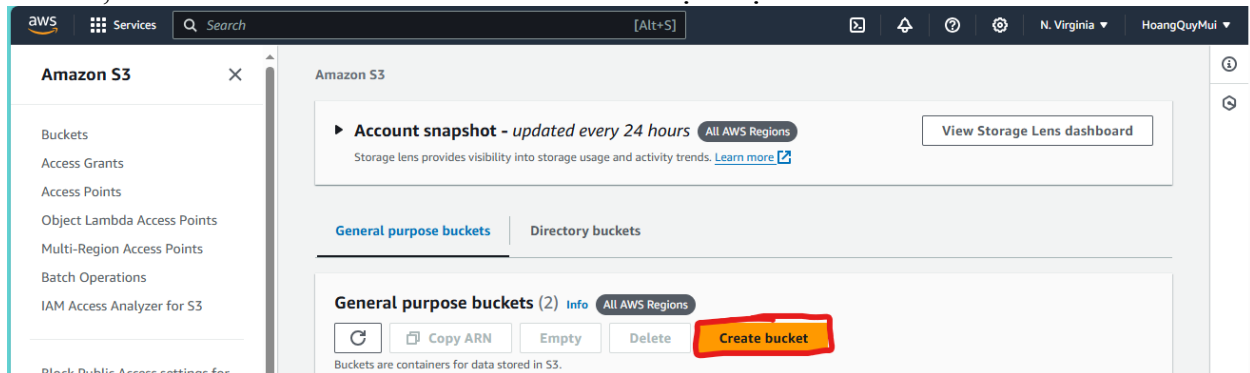
A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	
1	Name	Handle	Timestamp	Verified	Content	Comments	Retweets	Likes	Analytics	Tags	Mentions	Emojis	Profile Image	Tweet Link	Tweet ID
2	Kpop Charts	@kchartsmaster	2024-06-02T06:20:40.000Z	TRUE	Most streamed		420 3.3K	9.4K	394K	[[ ]]	[[ ]]	[[ ]]	https://pbs.twimg.com/profile_images/179151325811032271	https://x.com/kchartsmaster/status/179151325811032271	179151325811032271
3	NewJeans Loops	@newjeans_loops	2024-06-02T09:37:33.000Z	TRUE	MC Kang Haerin		113 1.6K	8.1K	165K	["#jeilw", "iLjSn", "i"]	[[ ]]	[[ ]]	https://pbs.twimg.com/profile_images/1792007489021	https://x.com/newjeans_loops/status/1792007489021	1792007489021
4	Soompi	@soompi	2024-06-02T09:39:56.000Z	TRUE	Kaespa Becom		25 2K	7.3K	174K	["Kaespa"]	[[ ]]	[[ ]]	https://pbs.twimg.com/profile_images/179065578093441084	https://x.com/soompi/status/179065578093441084	179065578093441084
5	PUBG: BATTLEGROUNDS	@PUBG	2024-06-02T09:00:08.000Z	TRUE	Next stop, the		83 1.3K	2.7K	97K	["#NewJeans", "i"]	[[ ]]	[[ ]]	https://pbs.twimg.com/profile_images/179119449931582312	https://x.com/PUBG/status/179119449931582312	179119449931582312
6	Kpop Charts	@kchartsmaster	2024-06-02T12:36:15.000Z	TRUE	Jeon Soyeon		92 1.2K	5.5K	219K	[[ ]]	["@NewJeans_twt", "@Jeans_"]	[[ ]]	https://pbs.twimg.com/profile_images/17904655141457	https://x.com/kchartsmaster/status/17904655141457	17904655141457
7	Notcoin	@thenotcoin	2024-06-02T12:12:17.000Z	TRUE	Idk did you not		478 476 4K	310K	[[ ]]	[[ ]]	[[ ]]	[[ ]]	https://pbs.twimg.com/profile_images/179028423476367652	https://x.com/thenotcoin/status/179028423476367652	179028423476367652
8	allkpop	@allkpop	2024-06-02T12:44:39.000Z	TRUE	#NewJeans ma		6 159 1K	38K	["#NewJeans"]	[[ ]]	[[ ]]	[[ ]]	https://pbs.twimg.com/profile_images/1790214654443521	https://x.com/allkpop/status/1790214654443521	1790214654443521
9	NewJeans Loops	@newjeans_loops	2024-06-02T06:18:15.000Z	TRUE	Jeon Soyeon		4 636 5.7K	156K	["#jeilw", "iLjSn", "i"]	[[ ]]	[[ ]]	[[ ]]	https://pbs.twimg.com/profile_images/179150720627757381	https://x.com/newjeans_loops/status/179150720627757381	179150720627757381
10	Ultiverse	@UltiverseDAO	2024-06-02T12:03:09.000Z	TRUE	ULTI Coin		315 277	745 385K	[[ ]]	[[ ]]	[[ ]]	[[ ]]	https://pbs.twimg.com/profile_images/179026121401628676	https://x.com/UltiverseDAO/status/179026121401628676	179026121401628676
11	NewJeans Centrs	@newjeans_centrs	2024-06-02T06:39:03.000Z	FALSE	[248692] are gi		1 229 1K	15K	["#NewJeans", "i"]	["@NewJeans_ADOR", "@jeunc"]	[[ ]]	[[ ]]	https://pbs.twimg.com/profile_images/179150555035304150	https://x.com/newjeans_centrs/status/179150555035304150	179150555035304150
12	Kpop Charts	@kchartsmaster	2024-06-02T12:32:51.000Z	TRUE	Crypto never s		24 768 4.5K	151K	["#NewJeans", "i"]	["@NewJeans_twt"]	[[ ]]	[[ ]]	https://pbs.twimg.com/profile_images/179046697450071	https://x.com/kchartsmaster/status/179046697450071	179046697450071
13	Binance	@binance	2024-06-02T00:00:15.000Z	TRUE	Crypto never s		24 722 4.5K	151K	[[ ]]	[[ ]]	[[ ]]	[[ ]]	https://pbs.twimg.com/profile_images/179055506363102111	https://x.com/binance/status/179055506363102111	179055506363102111
14	NewJeans Loops	@newjeans_loops	2024-06-02T10:33:45.000Z	TRUE	Today's out		25 514 2.5K	55K	["#jeilw", "iLjSn", "i"]	[[ ]]	[[ ]]	[[ ]]	https://pbs.twimg.com/profile_images/1791215018796335963	https://x.com/newjeans_loops/status/1791215018796335963	1791215018796335963
15	NewJeans Loops	@newjeans_loops	2024-06-02T06:17:09.000Z	TRUE	Jeon Soyeon		2 281 1.8K	74K	["#jeilw", "iLjSn", "i"]	[[ ]]	[[ ]]	[[ ]]	https://pbs.twimg.com/profile_images/179150442994217361	https://x.com/newjeans_loops/status/179150442994217361	179150442994217361
16	NEWJEANS NEW	@newjeansnew	2024-06-02T06:42:52.000Z	FALSE	The gifts given		0 231 1K	27K	[[ ]]	[[ ]]	[[ ]]	[[ ]]	https://pbs.twimg.com/profile_images/17915691267439	https://x.com/newjeansnew/status/17915691267439	17915691267439
17	NewJeans Loops	@newjeans_loops	2024-06-02T06:15:32.000Z	TRUE	Jeon Soyeon		4 688 4.3K	114K	["#jeilw", "iLjSn", "i"]	[[ ]]	[[ ]]	[[ ]]	https://pbs.twimg.com/profile_images/17915003215522242	https://x.com/newjeans_loops/status/17915003215522242	17915003215522242
18	Takanashi Kiara	@takanashikiara	2024-06-02T11:32:54.000Z	TRUE	was is loved		69 190 6.4K	110K	[[ ]]	[[ ]]	[[ ]]	[[ ]]	https://pbs.twimg.com/profile_images/179018508540154172	https://x.com/takanashikiara/status/179018508540154172	179018508540154172
19	Troll Football	@TrollFootball	2024-06-02T11:51:29.000Z	TRUE	Congratulation		778 15K	70K	3.2M	[[ ]]	[[ ]]	[[ ]]	https://pbs.twimg.com/profile_images/17902318755469337	https://x.com/TrollFootball/status/17902318755469337	17902318755469337
20	Benny Johnson	@bennyjohnson	2024-06-02T03:31:02.000Z	TRUE	UFC fighter Kev		870 11K	75K	2.9M	[[ ]]	[[ ]]	[[ ]]	https://pbs.twimg.com/profile_images/17911366959269882	https://x.com/bennyjohnson/status/17911366959269882	17911366959269882
21	Matt Wallace	@MattWallace88	2024-06-02T01:25:22.000Z	TRUE	The federal		15K 57K	3.3M	[[ ]]	[[ ]]	[[ ]]	[[ ]]	https://pbs.twimg.com/profile_images/17907701257564017	https://x.com/MattWallace88/status/17907701257564017	17907701257564017
22	Gacha Memes	@GachaMemes	2024-06-02T02:56:49.000Z	TRUE	Happy Birthday		58 2.5K	26K	505K	[[ ]]	[[ ]]	[[ ]]	https://pbs.twimg.com/profile_images/179100026662924378	https://x.com/GachaMemes/status/179100026662924378	179100026662924378
23	Tom Fitton	@TomFitton	2024-06-02T12:22:40.000Z	TRUE	BREAKING:		809 6.5K	14K	325K	[[ ]]	[[ ]]	[[ ]]	https://pbs.twimg.com/profile_images/179046136022216902	https://x.com/TomFitton/status/179046136022216902	179046136022216902
24	out of context do	@contextdogs	2024-06-02T03:00:04.000Z	TRUE			67 4.2K	60K	2.3M	[[ ]]	[[ ]]	[[ ]]	https://pbs.twimg.com/profile_images/17910004651517961	https://x.com/contextdogs/status/17910004651517961	17910004651517961
25	non aesthetic thi	@PicturesFolder	2024-06-02T18:25:09.000Z	TRUE	Do people actu		1.4K 2.8K	27K	3.3M	[[ ]]	[[ ]]	[[ ]]	https://pbs.twimg.com/profile_images/1796871262452940912	https://x.com/PicturesFolder/status/1796871262452940912	1796871262452940912
26	Nature is Amazing	@AMAZINGNAT	2024-06-02T18:33:55.000Z	TRUE	A puffer fish w		820 5.1K	56K	9M	[[ ]]	[[ ]]	[[ ]]	https://pbs.twimg.com/profile_images/179687499999942355	https://x.com/AMAZINGNAT/status/179687499999942355	179687499999942355
27	Madrid 3ra	@Madrid3ra	2024-06-02T11:11:29.000Z	TRUE	WHAT A PICT		118 15K	107K	1.8M	[[ ]]	[[ ]]	[[ ]]	https://pbs.twimg.com/profile_images/17903109173460295	https://x.com/Madrid3ra/status/17903109173460295	17903109173460295
28	Why you should	@ShouldIHaveCa	2024-06-02T02:56:04.000Z	TRUE			95 1.5K	15K	772K	[[ ]]	[[ ]]	[[ ]]	https://pbs.twimg.com/profile_images/179100342636577152	https://x.com/ShouldIHaveCa/status/179100342636577152	179100342636577152
29	Retro Anime	@retro_twt	2024-06-02T04:30:01.000Z	TRUE	Meowto		43 3.8K	51K	1.8M	[[ ]]	[[ ]]	[[ ]]	https://pbs.twimg.com/profile_images/17912343005766778	https://x.com/retro_twt/status/17912343005766778	17912343005766778
30	Madrid Zone	@theMadridZone	2024-06-02T11:13:24.000Z	TRUE	WHAT A PICTUR		71 7.9K	66K	1M	[[ ]]	[[ ]]	[[ ]]	https://pbs.twimg.com/profile_images/179013601095203608	https://x.com/theMadridZone/status/179013601095203608	179013601095203608
31	Out Of Context Fr	@InnocentFooly	2024-06-02T11:51:29.000Z	TRUE			296 9.1K	126K	3.8M	[[ ]]	[[ ]]	[[ ]]	https://pbs.twimg.com/profile_images/179023185495347458	https://x.com/InnocentFooly/status/179023185495347458	179023185495347458
32	BLCKPINK 'X' GLO	@BLACKPINKXGLO	2024-06-02T06:44:16.000Z	TRUE	PROS&W spott		28 755 4.2K	78K	["PROS&W", "Re"]	[[ ]]	[[ ]]	[[ ]]	https://pbs.twimg.com/profile_images/17915726442	https://x.com/BLACKPINKXGLO/status/17915726442	17915726442
33	Internet hall of fa	@InternetHOF	2024-06-02T18:00:27.000Z	TRUE			322 5.3K	106K	4.4M	[[ ]]	[[ ]]	[[ ]]	https://pbs.twimg.com/profile_images/17906504375458939	https://x.com/InternetHOF/status/17906504375458939	17906504375458939
34	HOURLY shippost	@hourly_shippost	2024-06-02T00:00:42.000Z	TRUE			104 6.8K	63K	1.9M	[[ ]]	[[ ]]	[[ ]]	https://pbs.twimg.com/profile_images/179055523958	https://x.com/hourly_shippost/status/179055523958	179055523958
35	NewJeans Loops	@newjeans_loops	2024-06-02T06:20:42.000Z	TRUE	Jeon Soyeon		24 281 1.1K	34K	["#jeilw", "iLjSn", "i"]	[[ ]]	[[ ]]	[[ ]]	https://pbs.twimg.com/profile_images/179181503509713435	https://x.com/newjeans_loops/status/179181503509713435	179181503509713435
36	Notcoin	@thenotcoin	2024-06-02T08:37:22.000Z	TRUE			500 1.1K	6.7K	174K	[[ ]]	[[ ]]	[[ ]]	https://pbs.twimg.com/profile_images/17919076069030369	https://x.com/thenotcoin/status/17919076069030369	17919076069030369
37	NewJeans Loops	@newjeans_loops	2024-06-02T06:42:34.000Z	TRUE	A 0 103,000		9 1.1K	2K	165K	[[ ]]	[[ ]]	[[ ]]	https://pbs.twimg.com/profile_images/179156830306339	https://x.com/newjeans_loops/status/179156830306339	179156830306339
38	Benny Johnson	@bennyjohnson	2024-06-02T03:31:18.000Z	TRUE	BREAKING:		677 3.8K	25K	1.1M	[[ ]]	[[ ]]	[[ ]]	https://pbs.twimg.com/profile_images/179113737544041	https://x.com/bennyjohnson/status/179113737544041	179113737544041

Hình 8. File kết quả

## CHƯƠNG 4: LƯU TRỮ VÀ XỬ LÝ DỮ LIỆU

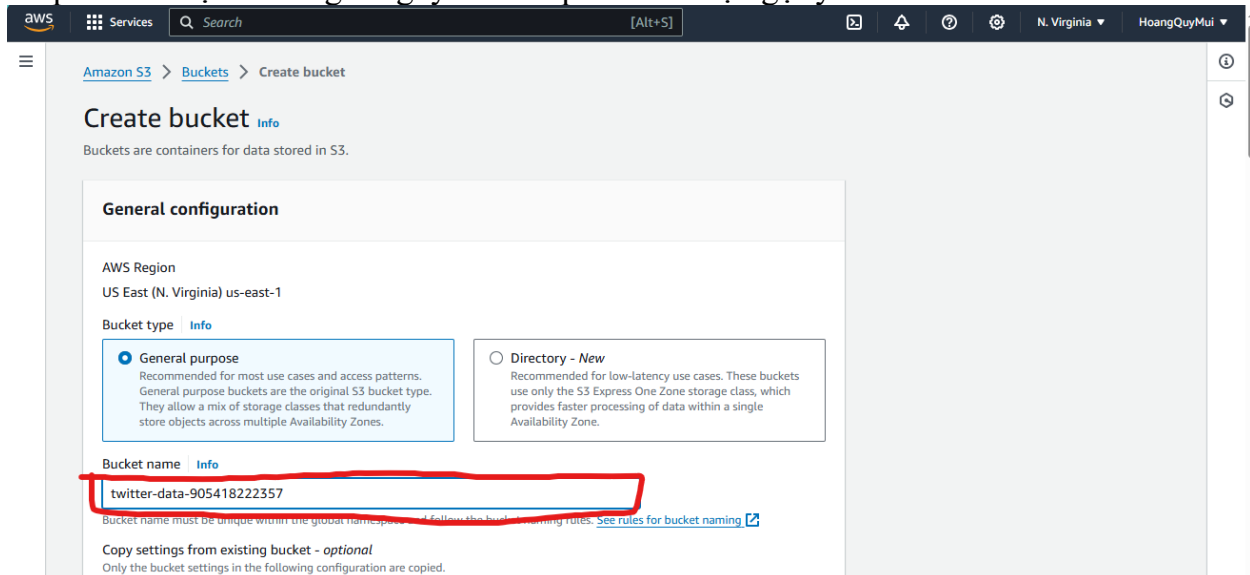
### 4.1 Amazon S3

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.



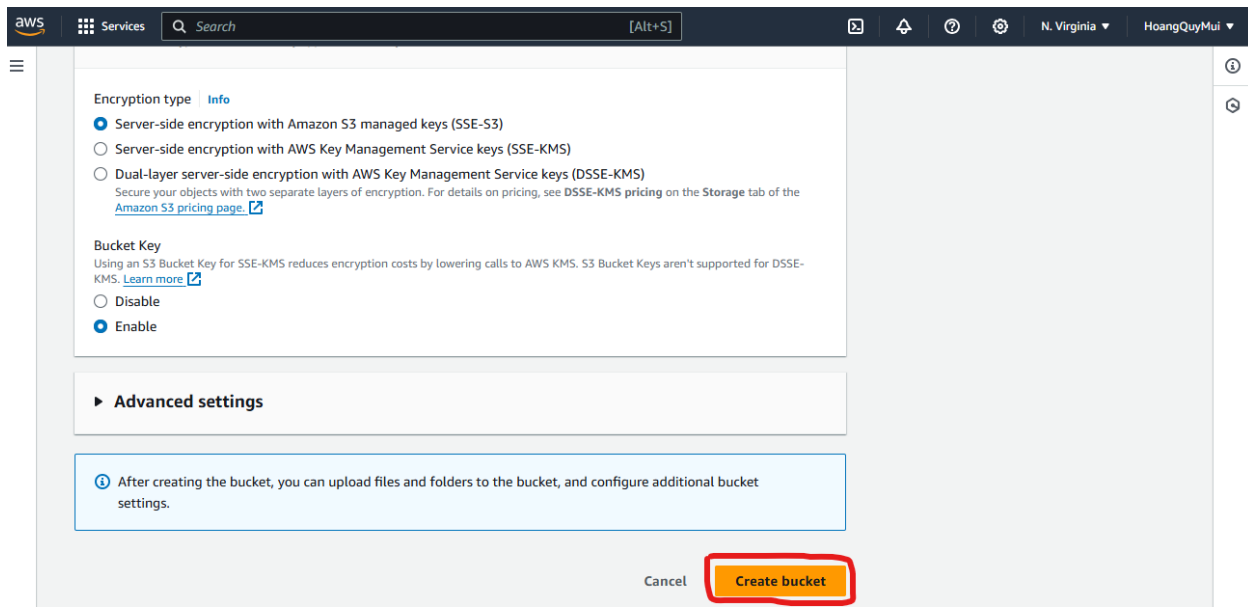
Hình 9. Tạo Bucket

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



Hình 10. Đặt tên cho Bucket

Cuối cùng là nhấn vào “Create bucket”.



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

```
C:\Users\hoang>aws configure
AWS Access Key ID [*****7QPU]:
AWS Secret Access Key [*****FK78]:
Default region name [us-east-1]:
Default output format [HoangQuyMui]:
```

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

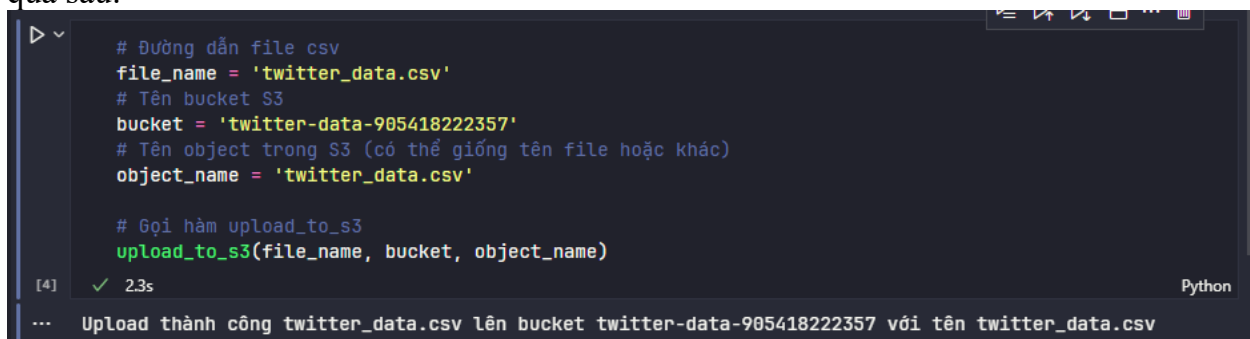
Đây là đoạn script sẽ giúp ta upload file twitter\_data lên bucket

## IS353.O21 – Mạng Xã Hội

```
1 import boto3
2 from botocore.exceptions import NoCredentialsError
3
4 def upload_to_s3(file_name, bucket, object_name=None):
5     # Nếu object_name không được cung cấp, sử dụng file_name
6     if object_name is None:
7         object_name = file_name
8
9     # Khởi tạo client S3
10    s3_client = boto3.client('s3')
11
12    try:
13        # Tải file lên S3
14        s3_client.upload_file(file_name, bucket, object_name)
15        print(f"Upload thành công {file_name} lên bucket {bucket} với tên {object_name}")
16    except FileNotFoundError:
17        print(f"File {file_name} không tồn tại.")
18    except NoCredentialsError:
19        print("Credentials không hợp lệ.")
20
21    # Đường dẫn file csv
22    file_name = 'twitter_data.csv'
23    # Tên bucket S3
24    bucket = 'demo-bucket-905418222357'
25    # Tên object trong S3 (có thể giống tên file hoặc khác)
26    object_name = 'twitter_data.csv'
27
28    # Gọi hàm upload_to_s3
29    upload_to_s3(file_name, bucket, object_name)
```

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:



```
# Đường dẫn file csv
file_name = 'twitter_data.csv'
# Tên bucket S3
bucket = 'twitter-data-905418222357'
# Tên object trong S3 (có thể giống tên file hoặc khác)
object_name = 'twitter_data.csv'

# Gọi hàm upload_to_s3
upload_to_s3(file_name, bucket, object_name)
```

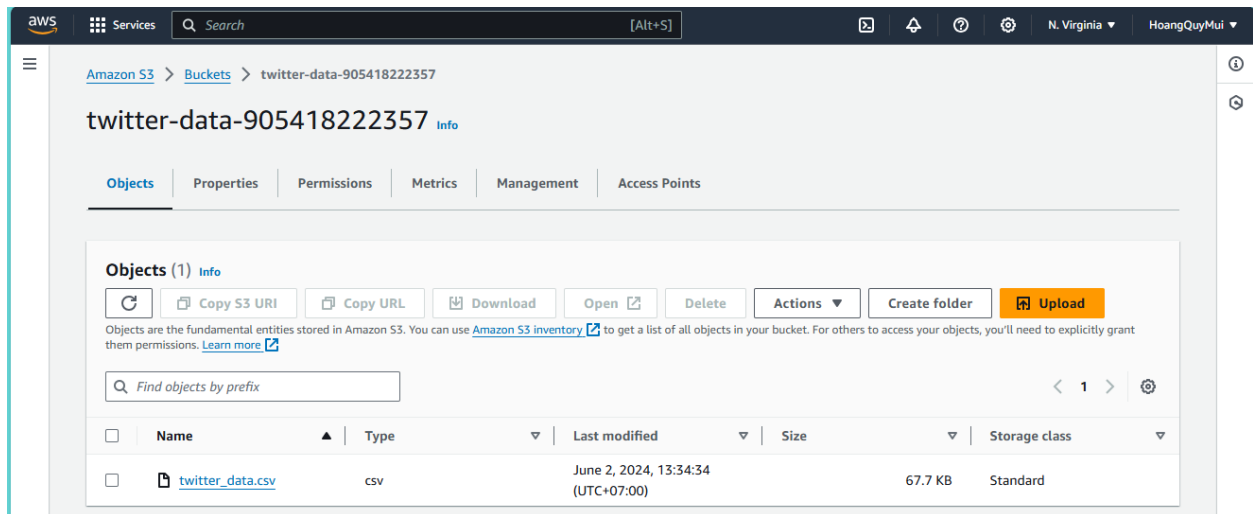
[4] ✓ 23s Python

... Upload thành công twitter\_data.csv lên bucket twitter-data-905418222357 với tên twitter\_data.csv

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`:



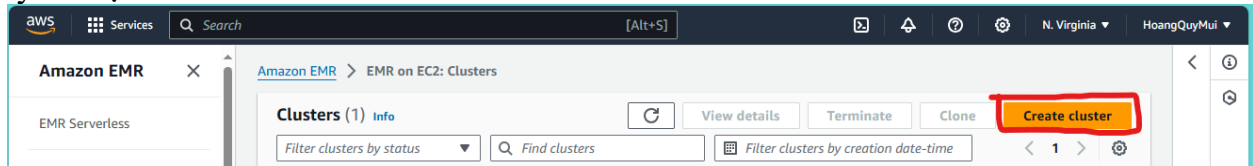


Hình 15. file `twitter_data.csv` trên bucket

### 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:



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:

**Create cluster** Info

**Name and applications - required** Info  
Name your cluster and choose the applications that you want to install to your cluster.

Name  
twitter-cluster

Amazon EMR release Info  
A release contains a set of applications which can be installed on your cluster.  
emr-7.1.0

Application bundle

Spark Interactive  
Core Hadoop  
Flink  
HBase  
Presto  
Trino  
Custom

☐ AmazonCloudWatchAgent  
☐ HCatalog 3.1.3  
☐ Hue 4.11.0  
☐ Flink 1.18.1  
☒ Hadoop 3.3.6  
☒ JupyterEnterpriseGateway 2.6.0  
☐ HBase 2.4.17  
☒ Hive 3.1.3  
☐ JupyterHub 1.5.0

**Summary** Info

**Name and applications - required**

Name  
twitter-cluster

Amazon EMR release  
emr-7.1.0

Application bundle  
Spark Interactive (Hadoop 3.3.6, Hive 3.1.3, JupyterEnterpriseGateway 2.6.0, Livy 0.8.0, Spark 3.5....)

**Cluster configuration - required**

Uniform instance groups  
Primary (m5.xlarge), Core (m5.xlarge), Task (m5.xlarge)

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

**Networking - required** Info  
Choose the network settings that determine how you and other entities communicate with your cluster.

Virtual private cloud (VPC) Info  
vpc-00f5a5c3a2b29299e Browse Create VPC

Subnet Info  
subnet-05ab8b34d71c1ce6f Browse Create subnet

▶ EC2 security groups (firewall)

Hình 18. Điều chỉnh các thông số

Sau đó là Security configuration and EC2 key pair.

**▼ Security configuration and EC2 key pair** [Info](#)  
Choose a security configuration or create a new one that you can reuse with other clusters.

**Security configuration**  
Select your cluster encryption, authentication, authorization, and instance metadata service settings.

---

**Amazon EC2 key pair for SSH to the cluster** [Info](#)

Hình 19. Security configuration and EC2 key pair

Nếu chưa có key pair, ta có thể nhấn vào phần Create 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

### EC2 instance profile for Amazon EMR

The instance profile assigns a role to every EC2 instance in a cluster. The instance profile must specify a role that can access the resources for your steps and bootstrap actions.

☐ Choose an existing instance profile

Select a default role or a custom instance profile with IAM policies attached so that your cluster can interact with your resources in Amazon S3.

☒ Create an instance profile

Let Amazon EMR create a new instance profile so that you can specify a custom set of resources for it to access in Amazon S3.

### S3 bucket access

☐ Specific S3 buckets or prefixes in your account [Info](#)

Choose the buckets or prefixes that you want this instance profile to access.

☒ All S3 buckets in this account with read and write access

Grant the instance profile access to all buckets that have read and write access enabled in your account.

Hình 20. EC2 instance profile for Amazon EMR

Cuối cùng nhấn Create cluster để tạo cluster

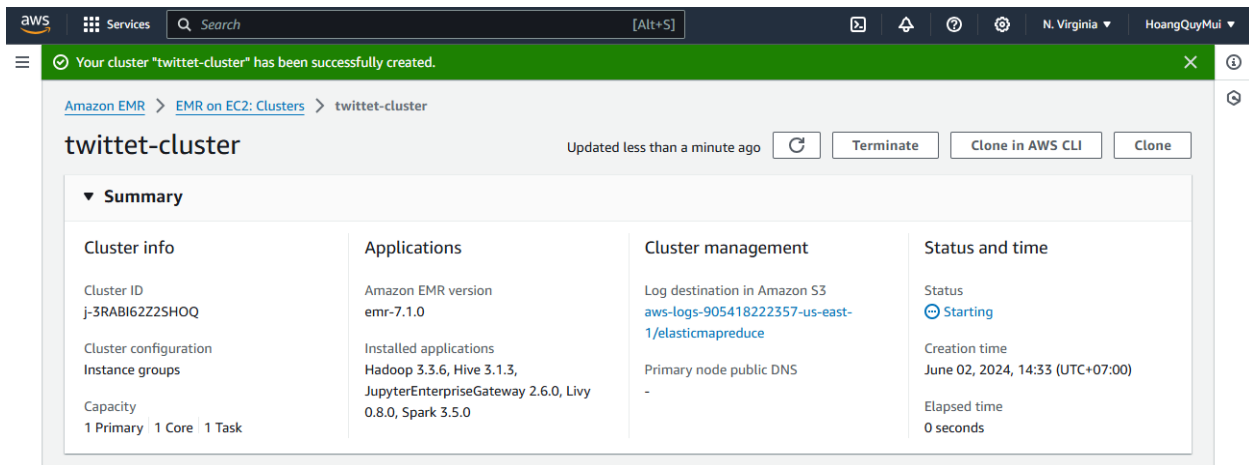
**Custom automatic scaling role - optional**  
When a custom automatic scaling rule triggers, Amazon EMR assumes this role to add and terminate EC2 instances. [Learn more](#)

Custom automatic scaling role

**Cluster scaling and provisioning - required**  
Provisioning configuration

Hình 21. Create cluster để tạo cluster

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



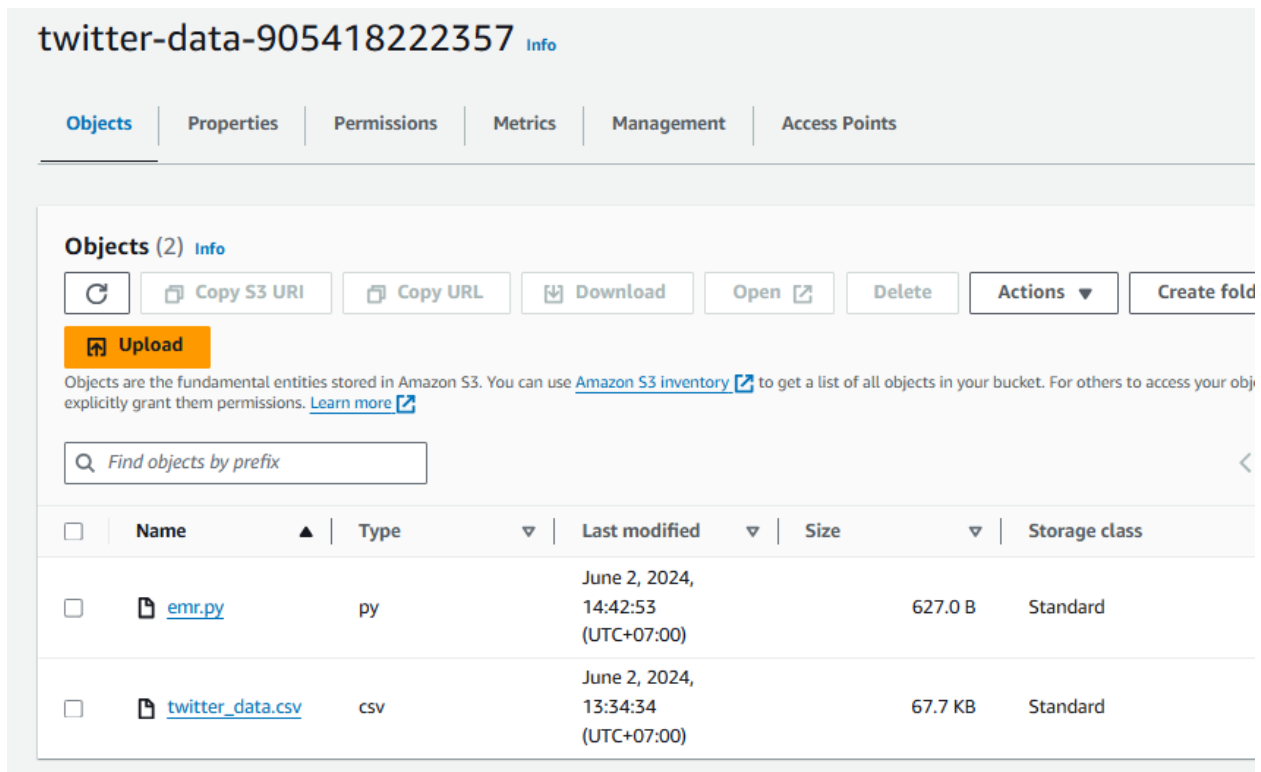
Hình 22. Cluster được tạo

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



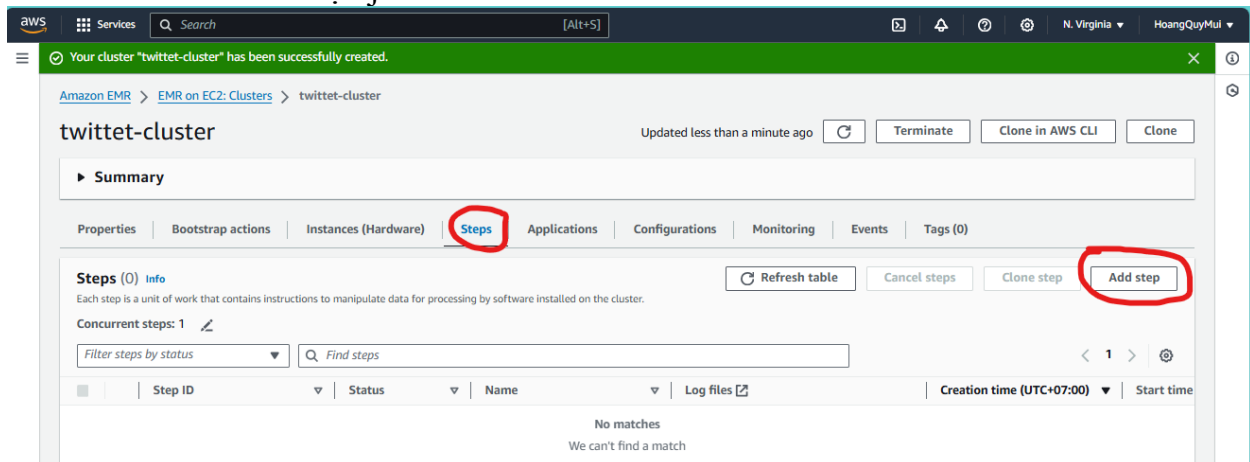
Hình 23. Script upload file emr.py lên bucket

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



Hình 24. Upload thành công

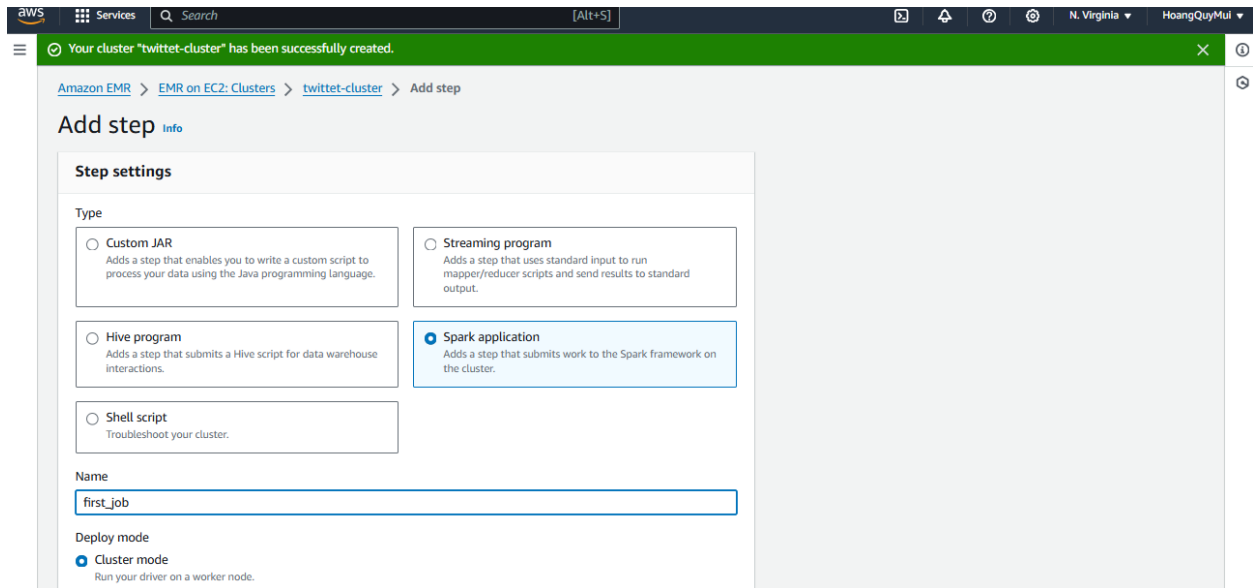
Sau đó ta sẽ tiến hành tạo job ở cluster



Hình 25. Tạo job ở cluster

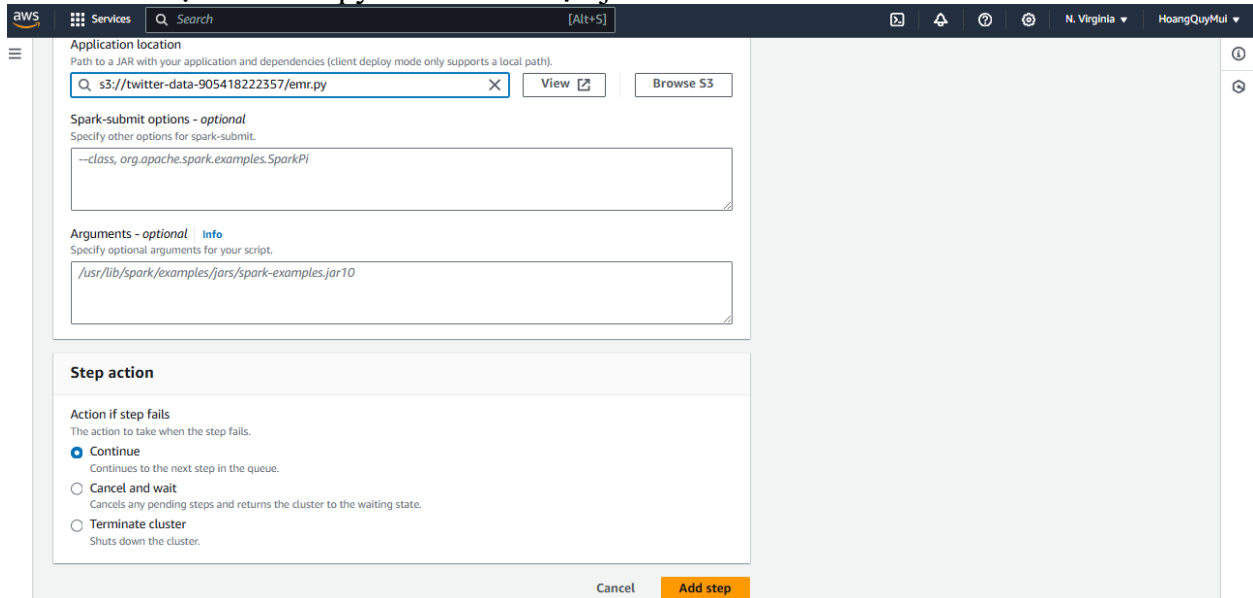
Vì nhóm muốn sử dụng spark để xử lý dữ liệu nên ta chọn Spark application và đặt tên cho job. Với Deploy mode chúng ta chọn Cluster Mode

## IS353.O21 – Mạng Xã Hội



Hình 26. Chọn cluster mode

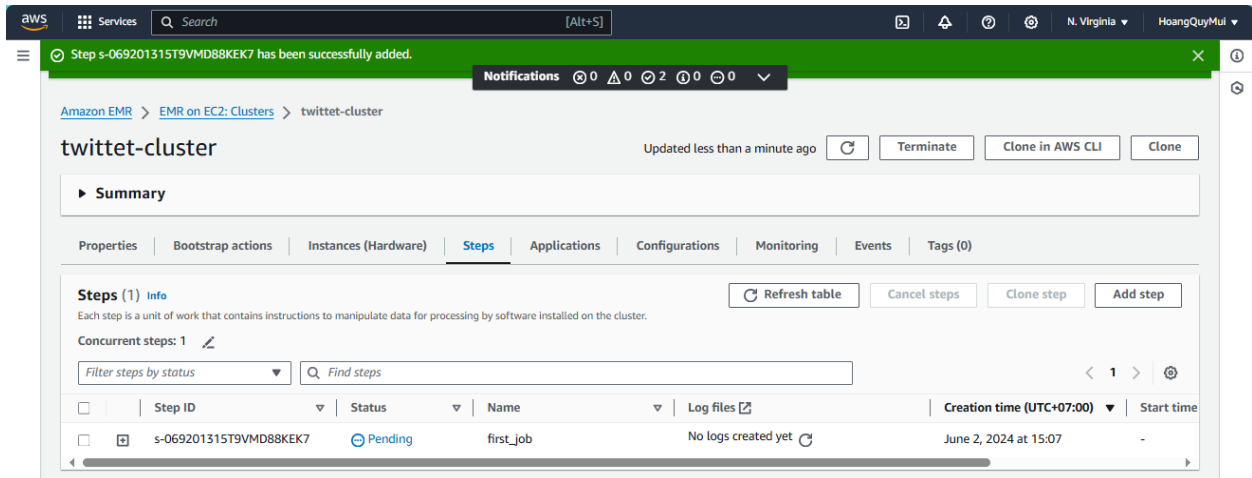
Sau đó là chọn file `emr.py` ở bucket và tạo job



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

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

## IS353.O21 – Mạng Xã Hội



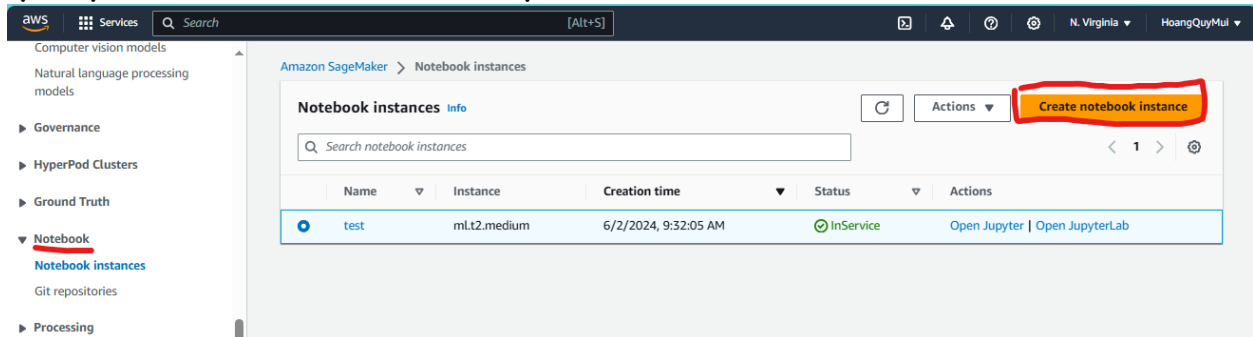
Hình 28. Job đang hiện là pending

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

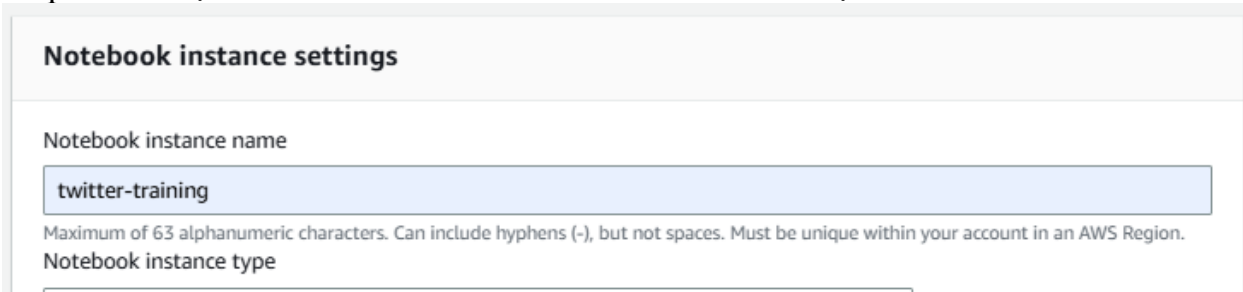
### 5.1 Huấn luyện mô hình trên AWS SageMaker

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



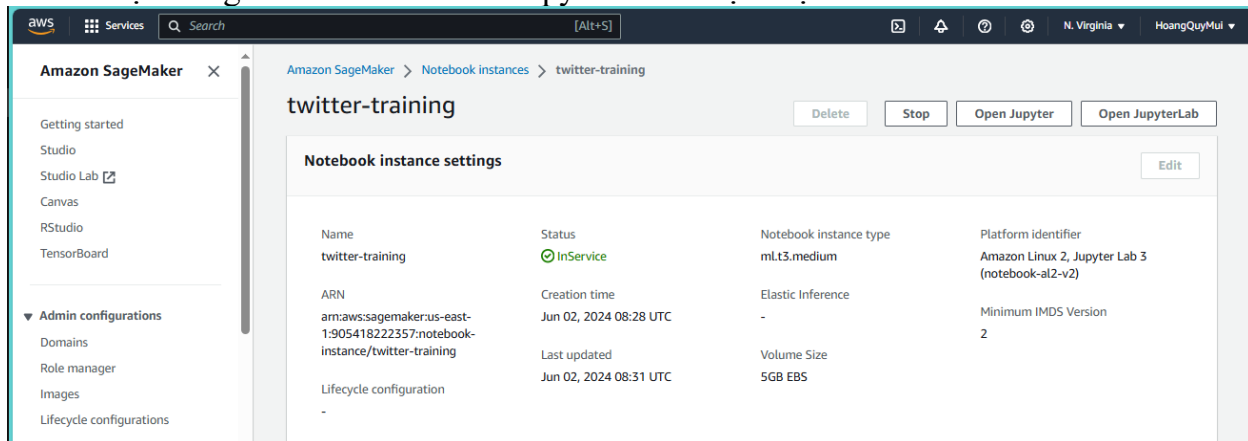
Hình 29. Tạo một Notebook Instance ở danh mục Notebook

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



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

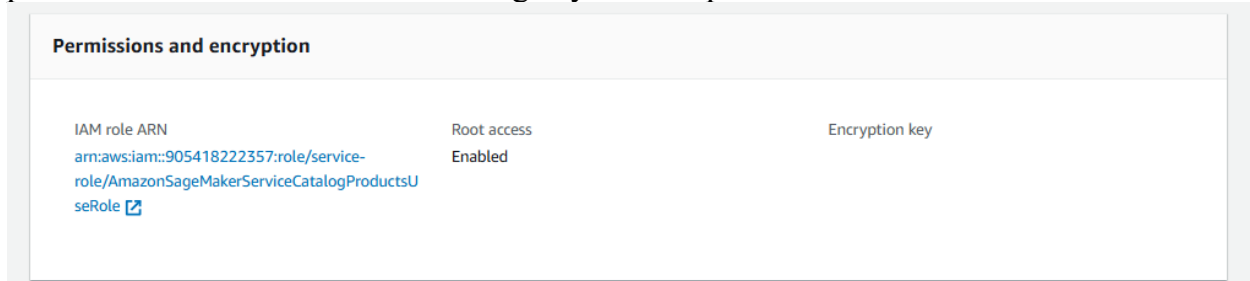


Hình 31. Mở JupyterLab và tạo một file notebook mới



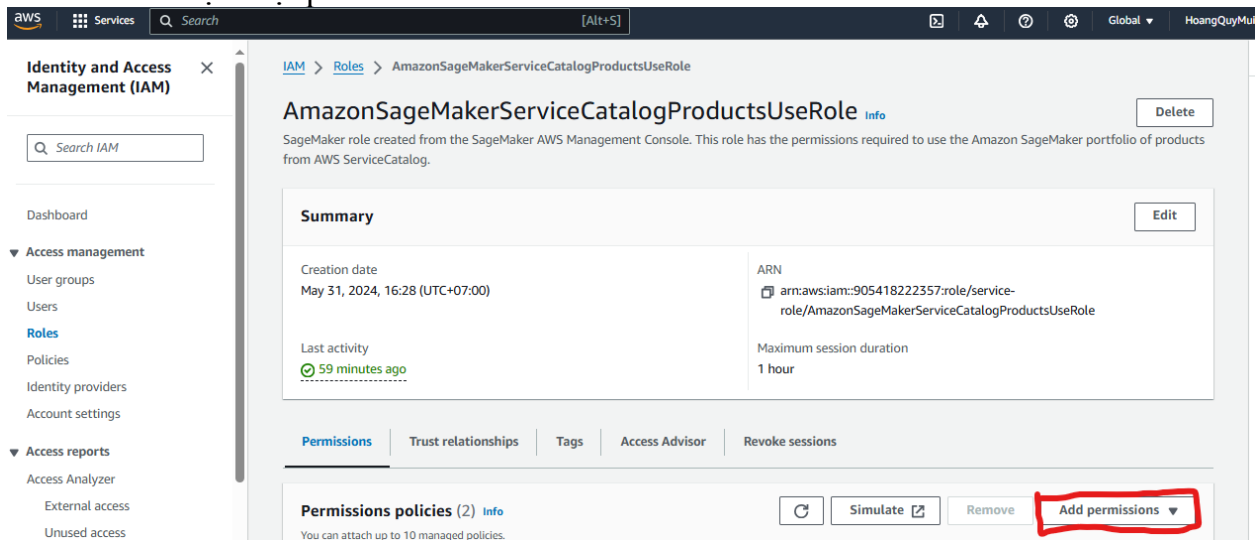
## IS353.O21 – Mạng Xã Hội

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



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

Và tiến hành tạo một permission mới

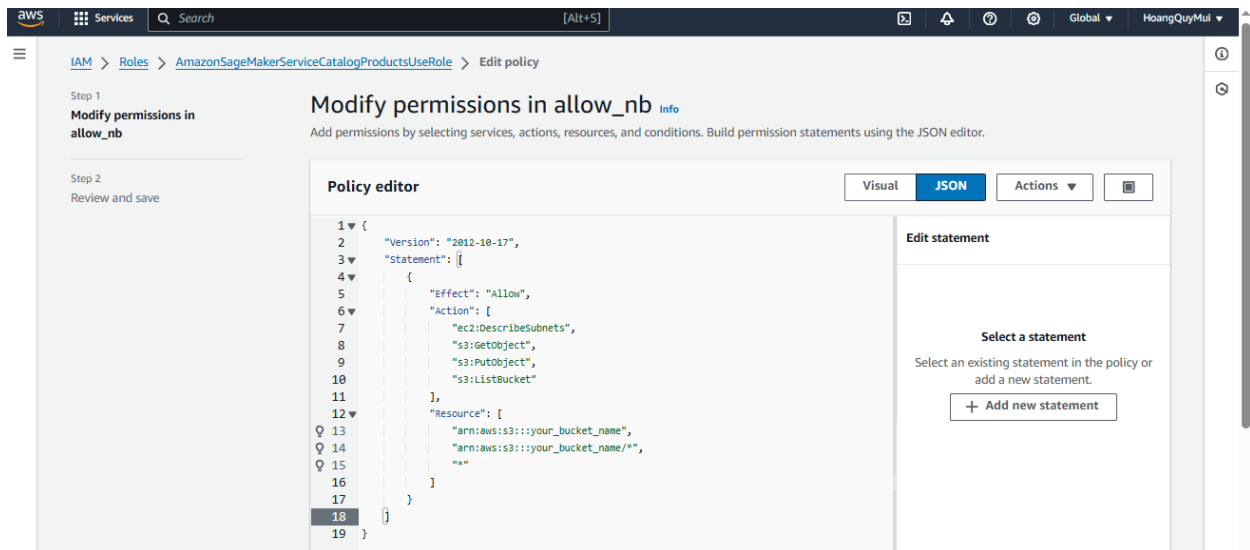


Hình 33. Tạo một permission mới

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

```
1  {
2      "Version": "2012-10-17",
3      "Statement": [
4          {
5              "Effect": "Allow",
6              "Action": [
7                  "s3:GetObject",
8                  "s3:PutObject",
9                  "s3:ListBucket"
10             ],
11             "Resource": [
12                 "arn:aws:s3:::your_bucket_name",
13                 "arn:aws:s3:::your_bucket_name/*"
14             ]
15         }
16     ]
17 }
18
```

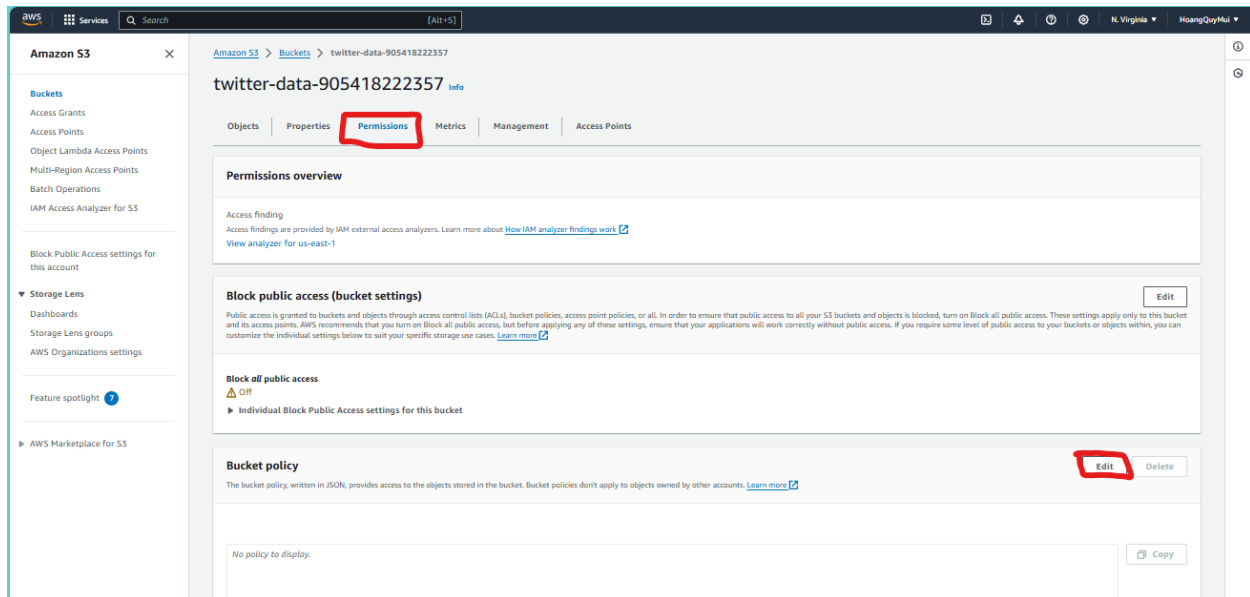
Hình 34. Script cấp quyền cho Notebook Instance



Hình 35. Script cấp quyền được hiển thị

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

## IS353.O21 – Mạng Xã Hội



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

Sử dụng đoạn script sau:

```
1 {
2   "Version": "2012-10-17",
3   "Statement": [
4     {
5       "Effect": "Allow",
6       "Principal": {
7         "AWS": "arn:aws:iam::905418222357:role/service-role/AmazonSageMakerServiceCatalogProductsUseRole"
8       },
9       "Action": [
10        "s3:GetObject",
11        "s3:PutObject"
12      ],
13       "Resource": "arn:aws:s3:::demo-bucket-905418222357/*"
14     }
15   ]
16 }
17 }
```

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

```
%%bash
aws configure
AKIA5FTZBVMKWFNGF25R
5dBRmCHHGwhrxXY9qDERhQMbHPLqURguXWrbJoB/b
us-east-1
HoangQuyMui
```

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

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

```
[2]: bucket_name = 'demo-bucket-905418222357'
    data_file = 'twitter_data.csv'

[4]: import boto3
    from botocore.exceptions import NoCredentialsError, ClientError

    s3_client = boto3.client('s3')

    s3_client.download_file(bucket_name, data_file, data_file)
    print("Download Successful")

Download Successful
```

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.

```
[23]: data = pd.read_csv('twitter_data.csv')

    # Chọn các tính năng và nhãn
    features = data[['followers_count', 'following_count', 'text_length', 'likes', 'comments']]
    labels = data['retweets']

    # Chia dữ liệu thành tập huấn luyện và tập kiểm tra
    X_train, X_test, y_train, y_test = train_test_split(features, labels, test_size=0.2, random_state=42)
```

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

```
: from sklearn.ensemble import RandomForestRegressor
    from sklearn.metrics import mean_squared_error
    from sklearn.model_selection import cross_val_score
    from sklearn.metrics import r2_score, mean_absolute_error

    # Khởi tạo và huấn luyện mô hình
    model_RF = RandomForestRegressor(n_estimators=100, random_state=42)
    model_RF.fit(X_train, y_train)
```

▼ RandomForestRegressor ⓘ ⓘ  
RandomForestRegressor(random\_state=42)

Hình 41. Dùng Random Forest để huấn luyện mô hình

```
: from sklearn.ensemble import GradientBoostingRegressor
    from sklearn.model_selection import train_test_split
    from sklearn.metrics import r2_score, mean_absolute_error
    from sklearn.model_selection import cross_val_score

    # Khởi tạo mô hình Gradient Boosting Regressor
    model_GB = GradientBoostingRegressor(n_estimators=100, learning_rate=0.1, max_depth=3, random_state=42)

    # Huấn luyện mô hình
    model_GB.fit(X_train, y_train)
```

▼ GradientBoostingRegressor ⓘ ⓘ  
GradientBoostingRegressor(random\_state=42)

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  $R^2$  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.

Công thức:

$$R^2 = 1 - \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{\sum_{i=1}^n (y_i - \bar{y})^2}$$

Trong đó:

- $y_i$  là giá trị thực tế.
- $\hat{y}_i$  là giá trị dự đoán từ mô hình.
- $\bar{y}$  là giá trị trung bình của  $y$ .

*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.

Công thức:

$$MAE = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i|$$

Trong đó:

- $y_i$  là giá trị thực tế.
- $\hat{y}_i$  là giá trị dự đoán từ mô hình.

*Hình 44. Công thức MAE*

	RandomForestRegressor	GradientBoostingRegressor
--	-----------------------	---------------------------

Mean Cross-Validation Score	0.72	0.43
R <sup>2</sup> 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.

```
from sklearn.model_selection import GridSearchCV

# Định nghĩa các hyperparameters cần tối ưu hóa
param_grid = {
    'n_estimators': [50, 100, 200],
    'max_depth': [None, 10, 20, 30],
    'min_samples_split': [2, 5, 10]
}

# Grid Search
grid_search = GridSearchCV(estimator=model_RF, param_grid=param_grid, cv=3, n_jobs=-1, verbose=2)
grid_search.fit(X_train, y_train)

# Lấy hyperparameters tốt nhất và đánh giá mô hình
best_model = grid_search.best_estimator_
best_predictions = best_model.predict(X_test)

# Cross-Validation
cv_scores = cross_val_score(best_model, X_train, y_train, cv=5)
print(f'Cross-Validation Scores: {cv_scores}')
print(f'Mean Cross-Validation Score: {cv_scores.mean()}')

# Đánh giá trên tập kiểm tra
r2 = r2_score(y_test, best_predictions)
mae = mean_absolute_error(y_test, predictions)
print(f'R2 Score: {r2}')
print(f'Mean Absolute Error: {mae}')
```

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 <sup>2</sup> Score	0.58
Mean Absolute Error	3572.92

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

## IS353.O21 – Mạng Xã Hội

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

```
[25]: best_params_ = grid_search.best_params_  
print("Best parameters found: ", best_params)  
  
Best parameters found: {'max_depth': 10, 'min_samples_split': 2, 'n_estimators': 100}
```

*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

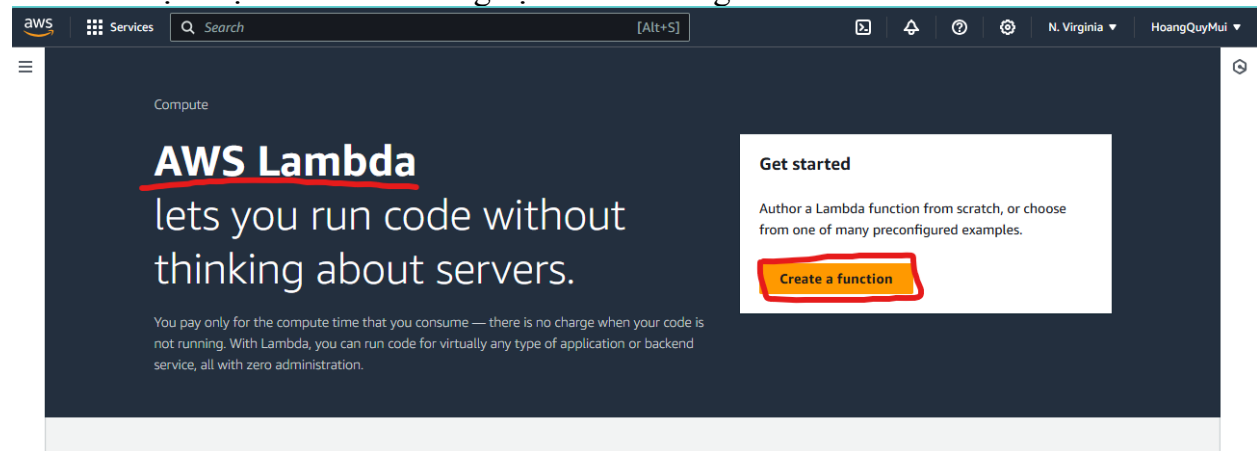
```
: import joblib

# Lưu mô hình
joblib.dump(best_model, 'predictShareModel.joblib')

# Upload mô hình lên S3
s3_client.upload_file('predictShareModel.joblib', bucket_name, 'predictShareModel.joblib')
```

Hình 47. Lưu và upload mô hình trên S3 bucket

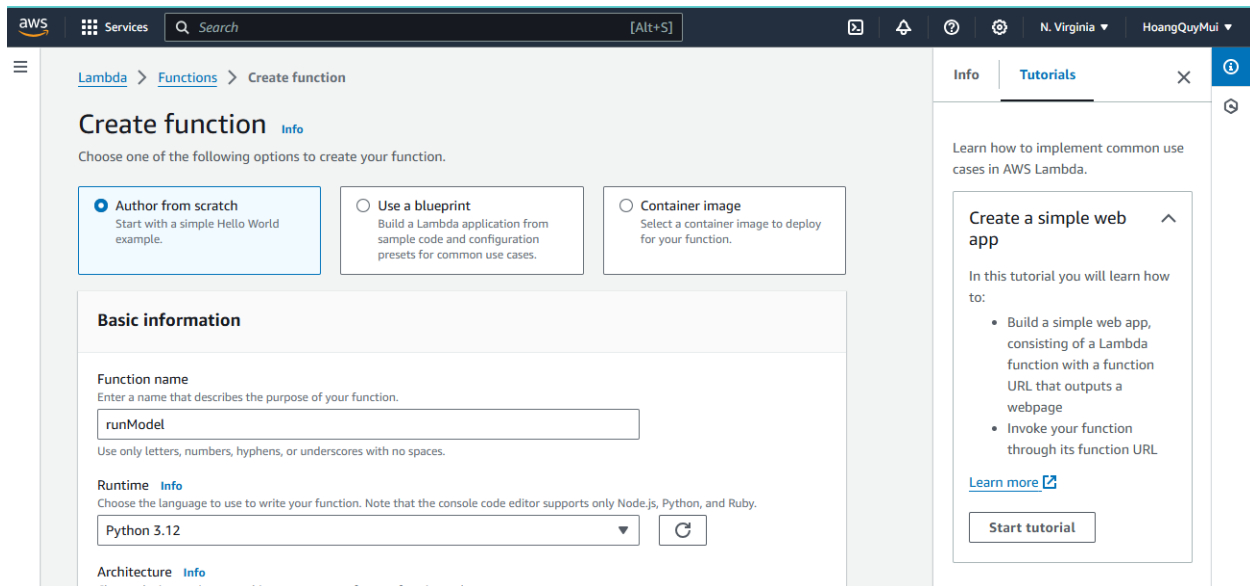
Ta sẽ cần tạo một Function ở công cụ Lambda bằng cách nhấn vào 'Create a function'



Hình 48. Tạo function ở AWS Lambda

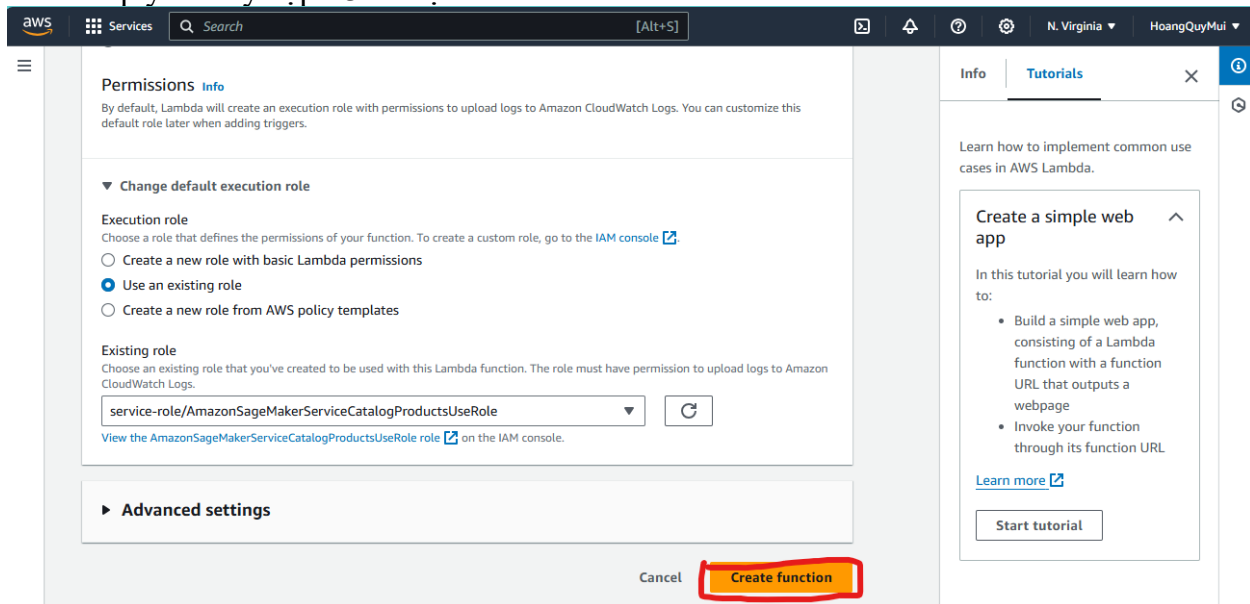
Ta sẽ thiết lập thông tin như hình để có thể chạy python





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.



Hình 50. Điền đầy 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

```
lambda_function.py 1 X
D: > zDevlearn > Notebook > SocialNetwork > AWS > my-lambda-function > lambda_function.py > ...
1  import boto3
2  import joblib
3  import json
4  import os
5
6  # Khởi tạo S3 client
7  s3 = boto3.client('s3')
8
9  # Định nghĩa biến toàn cục để lưu trữ mô hình sau khi tải từ S3
10 model = None
11
12 def download_model(bucket, key):
13     global model
14     if model is None:
15         s3.download_file(bucket, key, '/tmp/predictShareModel.joblib')
16         model = joblib.load('/tmp/predictShareModel.joblib')
17
18 def lambda_handler(event, context):
19     # Định nghĩa bucket và key cho mô hình
20     bucket = 'twitter-data-985418222357'
21     key = 'predictShareModel.joblib'
22
23     # Tải mô hình nếu chưa tải
24     download_model(bucket, key)
25
26     # Nhận dữ liệu đầu vào từ event
27     input_data = json.loads(event['body'])
28
29     # Chuẩn bị dữ liệu để dự đoán
30     features = input_data['features']
31
32     # Thực hiện dự đoán
33     prediction = model.predict([features]).tolist()
34
35     # Trả kết quả dự đoán
36     return {
37         'statusCode': 200,
38         'body': json.dumps({'prediction': prediction})
39     }
```

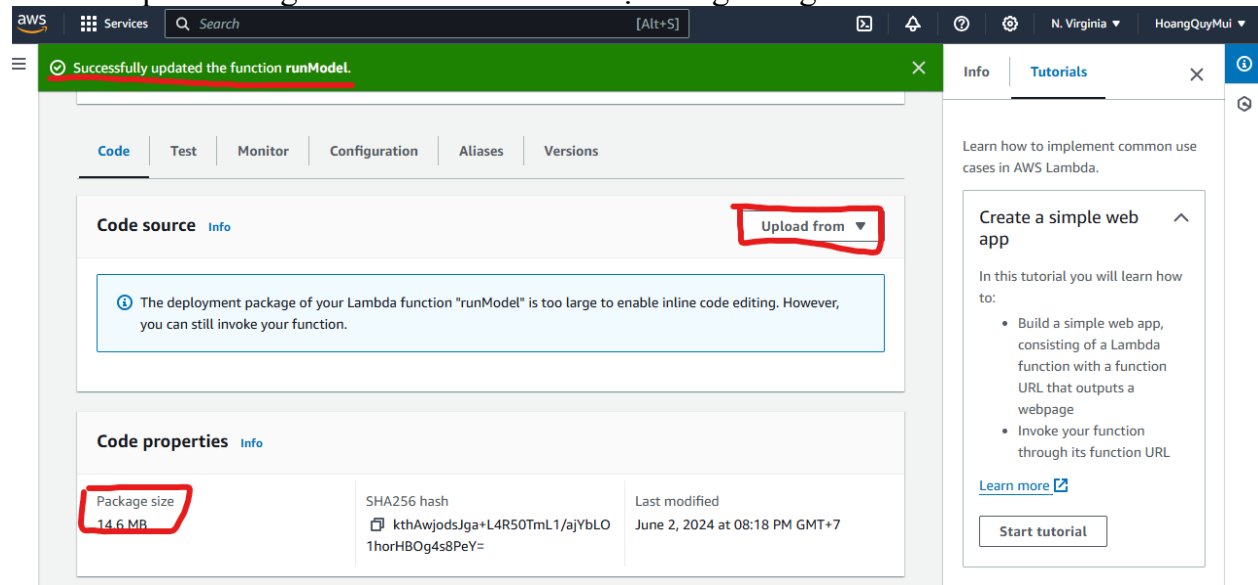
Hình 51. Tạo một mã nguồn triển khai mô hình

## IS353.O21 – Mạng Xã Hội

NAME	DATE MODIFIED	TYPE	SIZE
my-lambda-function	6/2/2024 8:40 PM	File folder	
best_model.joblib	6/2/2024 9:28 AM	JOBLIB File	1,210 KB
emr.py	6/2/2024 2:40 PM	Python Source File	1 KB
my-lambda-function.zip	6/2/2024 8:41 PM	WinRAR ZIP archive	14,937 KB
train.ipynb	6/2/2024 9:28 AM	Jupyter Source File	5 KB
twitter_data.csv	6/2/2024 9:27 AM	Microsoft Excel C...	68 KB
upload_aws.ipynb	6/2/2024 3:44 PM	Jupyter Source File	4 KB

Hình 52. Lambda function được lưu trong file zip

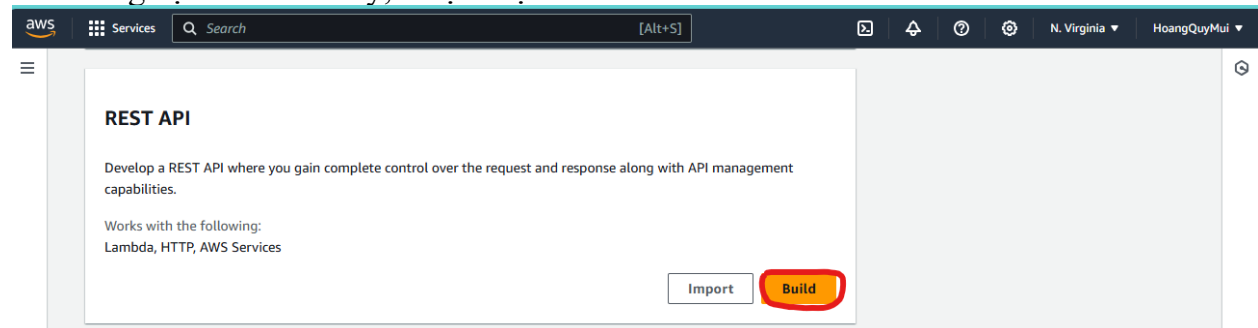
Sau khi upload xong thì màn hình sẽ hiển thị những thông tin sau



Hình 53. Màn hình sau khi upload xong

## 6.2 Tạo API với Amazon API Gateway

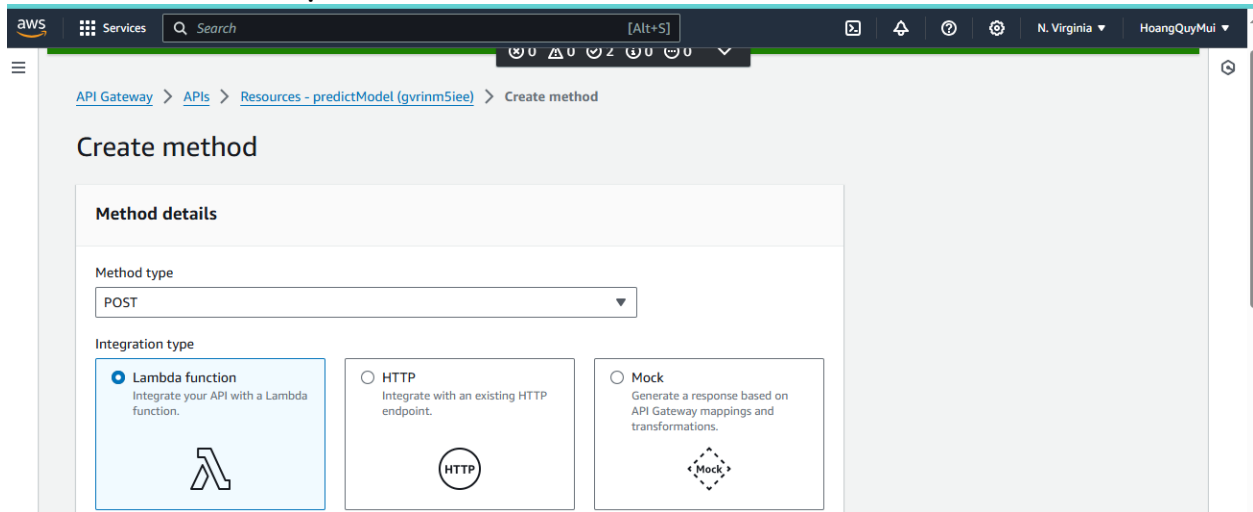
Đến công cụ API Gateway, ta tạo một REST API



Hình 54. tạo một REST API

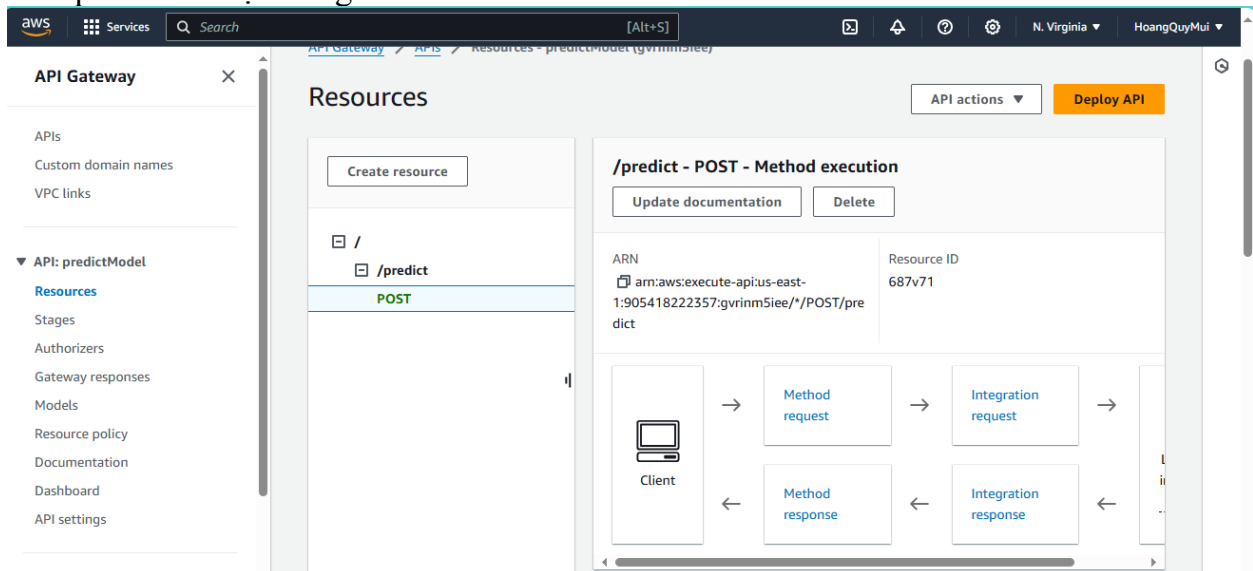
## IS353.O21 – Mạng Xã Hội

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.



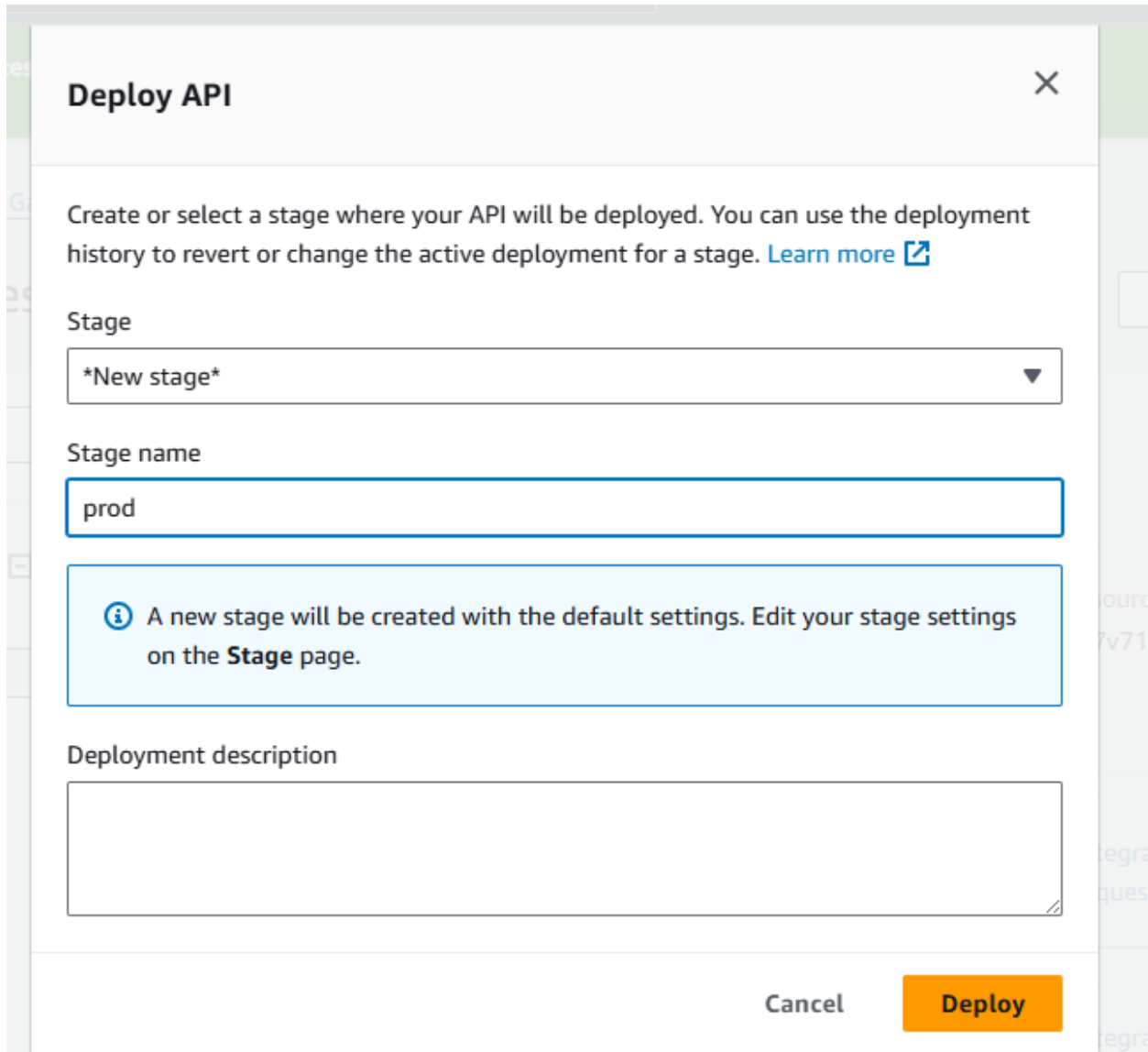
Hình 55. Tạo một method mới

Kết quả sau khi tạo xong



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.



The screenshot shows a 'Deploy API' modal window. At the top, it says 'Deploy API' with a close button. Below is a paragraph: 'Create or select a stage where your API will be deployed. You can use the deployment history to revert or change the active deployment for a stage. [Learn more](#)'. There are three main input sections: 'Stage' with a dropdown menu showing '\*New stage\*', 'Stage name' with a text box containing 'prod', and 'Deployment description' with an empty text area. A light blue information box states: 'A new stage will be created with the default settings. Edit your stage settings on the **Stage** page.' At the bottom right are 'Cancel' and 'Deploy' buttons.

Hình 57. Deploy API

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

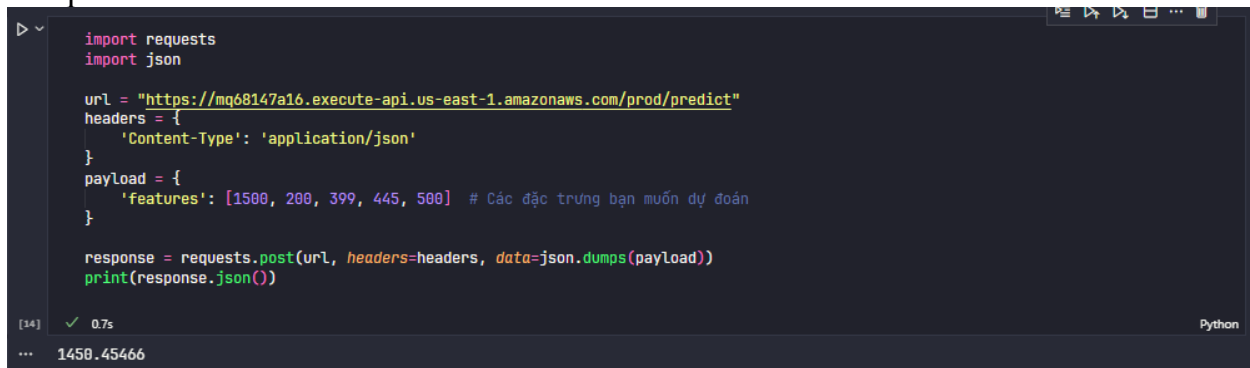
```
import requests
import json

url =
"https://mq68147a16.execute-api.us-east-1.amazonaws.com/prod/predict"
headers = {
    'Content-Type': 'application/json'
}
payload = {
    'features': [1500, 200, 399, 445, 500] # Các đặc trưng bạn muốn dự
đoán
}

response = requests.post(url, headers=headers, data=json.dumps(payload))
print(response.json())
```

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



```
import requests
import json

url = "https://mq68147a16.execute-api.us-east-1.amazonaws.com/prod/predict"
headers = {
    'Content-Type': 'application/json'
}
payload = {
    'features': [1500, 200, 399, 445, 500] # Các đặc trưng bạn muốn dự đoán
}

response = requests.post(url, headers=headers, data=json.dumps(payload))
print(response.json())
```

[14] ✓ 0.7s

... 1458.45466

Python

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

<b>Công việc</b>	<b>Tên</b>	<b><u>Mùi</u></b>	<b>Hải</b>	<b>Đăng</b>	<b>Phục</b>
Chọn đề tài, dataset		<b>X</b>	<b>X</b>	<b>X</b>	<b>X</b>
Chương 1				<b>X</b>	
Chương 2			<b>X</b>	<b>X</b>	
Chương 3					<b>X</b>
Chương 4		<b>X</b>		<b>X</b>	
Chương 5		<b>X</b>		<b>X</b>	
Chương 6		<b>X</b>		<b>X</b>	
Chương 7		<b>X</b>		<b>X</b>	
Report		<b>X</b>		<b>X</b>	
Slide		<b>X</b>	<b>X</b>	<b>X</b>	<b>X</b>
Demo		<b>X</b>			
Trình bày		<b>X</b>	<b>X</b>	<b>X</b>	<b>X</b>
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>