Phân Tích Cảm Xúc Người Dùng Về ChatGPT Dựa Trên Dữ Liệu Reddit Theo Thời Gian Thực

Nguyễn Thiện Trí[1,2,3], Đoàn Bảo Long[1,2,4], and Nguyễn Viết Tiến[1,2,5], Đỗ Trọng Hợp[1,2]

[1] Trường Đại học Công Nghệ Thông Tin
[2] Đại học Quốc Gia Thành phố Hồ Chí Minh
{21522707[3],21520332[4],20520805[5]}@gm.uit.edu.vn

Abstract. Trong thời đại công nghệ số, Trí tuệ Nhân tạo (AI) đang trở thành một lĩnh vực quan trọng với tiềm năng và ứng dụng rộng lớn. ChatGPT, một mô hình ngôn ngữ dựa trên công nghệ GPT-3.5 của OpenAI, đang trở nên cực kì phổ biến vì được huấn luyện trên một lượng lớn dữ liệu và có khả năng tạo ra các phản hồi tự nhiên và mạch lạc. Trong báo cáo này, nhóm giới thiệu một hệ thống phân tích cảm xúc về ChatGPT dựa trên dữ liệu Reddit theo thời gian thực. Hệ thống được xây dựng phục vụ cho mục đích học tập, nghiên cứu. Kiến trúc của hệ thống gồm có một số thành phần chính sau: Praw (một Package Python hỗ trợ để lấy dữ liệu từ Reddit thông qua Reddit API) dùng để lấy các Submissions (bài đăng) liên quan đến ChatGPT và lưu vào Apache Kafka; Spark Streaming đọc dữ liệu từ Kafka, đưa vào Spark MLlib và Spark DataFrame xử lý và phân tích; Kết quả phân tích, dùng các mô hình học máy bao gồm: Naive Bayes, Support Vector Machine và Logistic Regression, được lưu vào Kafka; Dữ liệu được phân tích sẽ qua Spark Streaming và Spark SQL xử lý, truy vấn và trích xuất dữ liệu báo cáo và thống kê, sau đó ghi dữ liệu vào hệ thống tập tin với định dạng CSV; Streamlit đọc dữ liệu từ hệ thống tập tin, dùng thư viện Plotly xử lý và hiển thị thông tin gồm các bảng, biểu đồ trực quan lên giao diện web.

Keywords: Machine Learning · Reddit · ChatGPT · Apache Kafka · Apache Spark

1 Giới thiệu

Với sự phát triển của Trí tuệ nhân tạo (AI) cũng là sự phát triển của các ChatBots, chúng ngày càng trở nên thông minh và có thể tương tác với con người thông qua văn bản một cách rất thuần thục và mạch lạc. Một ví dụ cụ thể là ChatGPT, một công cụ xử lý ngôn ngữ được tạo bởi OpenAI và ra mắt vào ngày 30 tháng 11 năm 2022. Chỉ trong vòng 5 ngày sau khi ra mắt, ChatGPT đã thu hút hơn một triệu người dùng. Hiện tại, ChatGPT có hơn 100 triệu người dùng hoạt động khiến nó trở thành một trong nhưng ChatBot được sử dụng rộng rãi nhất trên thế giới.

Trong bài viết này, nhóm giới thiệu một hệ thống phân tích cảm xúc của cộng đồng về ChatGPT theo thời gian thực dựa trên dữ liệu từ Reddit và sử dụng các nền tảng công nghệ từ Kafka và Apache Spark. Mô hình phân tích cảm xúc bao gồm tiêu đề của các submissions có liên quan tới ChatGPT trên Reddit và việc sử dụng các công cụ máy học được hỗ trợ bởi Apache Spark để phân tích tiêu đề của bài đăng đó là tích cực hay tiêu cực. Từ đó, dựa vào các biểu đồ thống kê, ta có thể biết được tình hình cảm xúc của cộng đồng về ChatGPT theo thời gian thực.

2 Các nghiên cứu liên quan

Trong thời đại công nghệ ngày nay, sự phát triển của các kỹ thuật xử lý ngôn ngữ tự nhiên, xử lý dữ liệu lớn và các thuật toán học máy hiện đại, đã tạo ra nhiều ứng dụng thông minh, áp dụng vào nhiều lĩnh vực trong thế giới thực. Đặc biệt, phân tích cảm xúc đã trở thành một chủ đề rất được quan tâm cho lĩnh vực nghiên cứu, kinh doanh và một số lĩnh vực khác. Điều này đề cập đến cảm xúc hoặc suy nghĩ của con người về một số vấn đề nhất định trong đời sống thực tế. Hơn nữa, nó cũng được coi là một ứng dụng trực tiếp để khai thác ý kiến. Một số ứng dụng cụ thể liên quan đến lĩnh vực này gồm có: Phân tích cảm xúc về ChatGPT dùng các thuật toán học máy [1] [2]; Phân tích cảm xúc dùng nền tảng công nghệ Apache Spark [3], nằm trong lĩnh vực mà nhóm đề xuất trong bài viết này.

3 Kiến trúc hệ thống

Hệ thống phân tích cảm xúc theo thời gian thực gồm hai thành phần chính: Xây dựng một mô hình máy học phân tích cảm xúc offline (thành phần offline) và Cấu trúc Pipeline phân loại theo thời gian thực bằng mô hình được xây dựng sẵn (thành phần online).

Quy trình thực hiện : ban đầu cần xây dựng sẵn mô hình máy học cho việc phân loại cảm xúc và lưu lại mô hình (thành phần offline). Sau đó, xây dựng hệ thống streaming để lấy dữ liệu liên tục và tải mô hình đã lưu ở trên để phân loại văn bản. Cuối cùng, xây dựng chương trình để trực quan hóa dữ liệu qua giao diện web(thành phần online).

3.1 Xây dựng mô hình máy học

Các mô hình máy học được huấn luyện trên Google Colab với Apache Spark 3.4.1, Python 3.10, java 11 cùng một số thư viện khác.

Thu thập và tiền xử lý dữ liệu : Bộ dữ liệu được sử dụng để huấn luyện cho các mô hình machine learning được tham khảo tại trang web sau: `https://www.kaggle.com/datasets/charunisa/chatgpt-sentiment-analysis`. Bộ dữ

liệu bao gồm 2 thuộc tính: đoạn text đánh giá của người dùng về chatGPT và label tương ứng. Số observations có trong bộ dữ liệu là 217622 dòng. Chi tiết số observations trong mỗi nhãn được thể hiện qua biểu đồ sau:

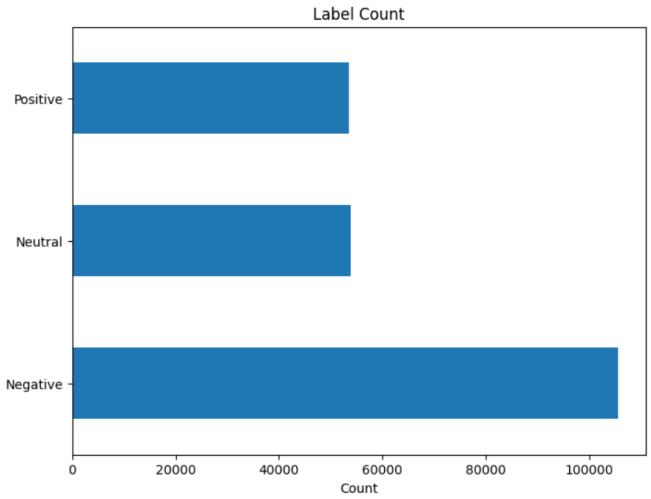

Fig. 1. Phân phối nhãn của bộ dữ liệu

Từ biểu đồ trên ta có thể nhận thấy sự chênh lệnh về số lượng observations ở mỗi label, trong đó số câu mang sentiment Negative chiếm đa số trong bộ dữ liệu, điều này có thể dẫn đến việc các mô hình phân loại dự đoán lệch về phía nhãn tiêu cực. Tuy nhiên nhóm nhận thấy việc cân bằng lại dữ liệu là không cần thiết bởi vì các câu có sentiment Negative sẽ mang lại nhiều thông tin có giá trị hơn so với các sentiment còn lại, việc phân tích các câu mang cảm xúc tiêu cực có thể giúp các nhà phát triển xác định được những vấn đề thiếu sót hoặc chưa phù hợp trong cách hoạt động của ChatGPT, từ đó đưa ra các chiến lược cải thiện sản phẩm một cách tốt hơn.

Nhóm cũng đã thực hiện kiểm tra lại ngẫu nhiên nhãn của 1000 câu trong bộ dữ liệu nhằm đảm bảo tính đúng đắn của bộ dữ liệu, độ đồng thuận tính được theo chỉ số Cohen's Kappa là 0.89, cho thấy bộ dữ liệu đã được gán nhãn khá tốt và có thể được sử dụng cho tác vụ phân tích cảm xúc. Bộ dữ liệu được tách ra với 80% dùng để huấn luyện và 20% dùng để thử nghiệm, đồng thời trải qua một số bước tiền xử lý văn bản sau :

- Replacing URL: thay thế các URL có trong đoạn text thành "URL", ví dụ: "ChatGPT: Optimizing Language Models for Dialogue `https://t.co/K9rKRygYyn` @OpenAI" -> "ChatGPT: Optimizing Language Models for Dialogue URL @OpenAI"
- LowerCasing: lower-case tất cả các kí tự có trong đoạn text.
- Removing non-alphabet: loại bỏ các kí tự đặc biệt có trong câu.

– Removing stop-words: loại bỏ các stop-words có trong câu.

Tiền xử lý dữ liệu văn bản giúp giảm kích thước của bộ dữ liệu, từ đó giảm thời gian huấn luyện cho các mô hình máy học, đồng thời giúp đảm bảo tính nhất quán trong dữ liệu và cải thiện khả năng trích xuất thông tin sau này. Các bước tiền xử lý văn bản trên vẫn sẽ được áp dụng cho dữ liệu mới được lấy về khi khởi chạy Kafka và Spark streaming.

Feature Extractor và Machine Learning Model : Các feature extractor được sử dụng là HashingTF, IDF và NGram.

– HashingTF (Term Frequency) là một phép toán trong xử lý văn bản, được sử dụng để biểu diễn từ vựng của văn bản dưới dạng vector số học. Nó chuyển đổi văn bản thành một vector đặc trưng bằng cách ánh xạ các từ vào các chỉ mục ngẫu nhiên trong một không gian vector cố định. Cách tiếp cận này sử dụng hàm băm (hashing) để xử lý số lượng lớn từ vựng và giảm chiều dữ liệu.
– IDF (Inverse Document Frequency): IDF là một phép đo được sử dụng để đánh giá sự quan trọng của một từ trong văn bản. Nó tính toán nghịch đảo tần suất xuất hiện của từ trong tập dữ liệu. Các từ xuất hiện ít thường xuyên nhưng xuất hiện trong một số văn bản đặc biệt sẽ có IDF cao, cho thấy tính đặc biệt của từ đó đối với văn bản đó.
– NGram là một kỹ thuật trong NLP để xác định chuỗi các từ liên tiếp trong văn bản. Một NGram có thể là unigram (một từ), bigram (hai từ liên tiếp), trigram (ba từ liên tiếp), và n-gram (n từ liên tiếp). Xây dựng các NGram giúp mô hình hiểu được ngữ cảnh và quan hệ giữa các từ trong văn bản. Trong bài nghiên cứu này nhóm sử dụng 2 loại NGram là unigram và bigram

Các mô hình máy học được sử dụng là Logistic Regression, Support Vector Machine và Naive Bayes:

– Logistic Regression là một trong những thuật toán học máy truyền thống thường được sử dụng cho các tác vụ phân loại, bao gồm phân tích cảm xúc, nó cung cấp kết quả có thể diễn giải bằng cách gán trọng số cho các đặc trưng, cho biết ảnh hưởng của chúng đối với việc dự đoán cảm xúc. Khả năng diễn giải này có thể hữu ích trong việc tìm hiểu các đặc trưng quan trọng tương ứng với mỗi sentiment trong đoạn văn bản.
– Support Vector Machine (SVM) cũng là một thuật toán học máy mạnh mẽ được sử dụng trong nhiều ứng dụng khác nhau, bao gồm cả phân tích cảm xúc. SVM được sử dụng để xây dựng các mô hình phân loại dựa trên việc tìm một đường phân chia tối ưu giữa các lớp dữ liệu.
– Thuật toán Naive Bayes dựa trên nguyên lý của định lý Bayes để ước lượng xác suất một văn bản thuộc về một lớp cảm xúc cụ thể. Tính toán xác suất xảy ra của từng từ trong từ điển dựa trên tập huấn luyện và nhãn cảm xúc tương ứng. Sau đó sử dụng các xác suất tính toán được, thuật toán tính toán xác suất của một đoạn văn bản thuộc về từng loại cảm xúc và chọn ra nhãn có xác xuất cao nhất làm giá trị dự đoán cho mô hình.

Để bộ phân lớp Logistic Regression và Support Vector Machine có thể xử lý tốt dữ liệu với nhiều hơn 2 nhãn, nhóm sử dụng tham số *family* = 'multinomial' cho LogisticRegression và chiến lược *OneVsRest* cho LinearSVC (Support Vector Machine). One-vs-rest (còn có cách gọi One-vs-All) là một phương pháp chia bài toán phân loại đa lớp thành các bài toán phân loại nhị phân, trong đó, mỗi bài toán phân loại nhị phân sẽ phân loại 1 class nhất định với tất cả class còn lại. Nhãn của một điểm dữ liệu sẽ là nhãn có xác suất lớn nhất được tính toán từ các mô hình máy học.

Ở đây nhóm sử dụng độ đo Average F1-score để đánh giá cho các bộ phân lớp, kết quả của các mô hình máy học được trình bày qua bảng và biểu đồ dưới đây:

	Ngram=1		Ngram=2	
	Macro	Weighted	Macro	Weighted
LinearSVC	79%	81%	62%	66%
Logistic Regression	76%	79%	60%	64%
Naïve Bayes	61%	66%	60%	64%

Fig. 2. F1-score

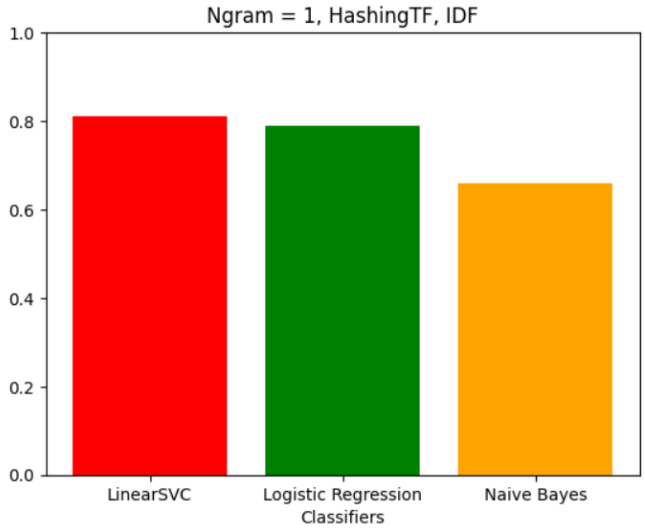

Fig. 3. Weighted Avg F1-score

Mô hình cho kết quả cao nhất là bộ phân lớp LinearSVC với kết quả đạt được khoảng 81% khi được kết hợp cùng bộ trích xuất đặc trưng Unigram, HashingTF và IDF. Mô hình này sẽ được lưu lại thành file, sau đó sẽ được sử dụng để dự đoán cảm xúc cho các bài viết liên quan đến chatGPT trong quá trình thực hiện streaming dữ liệu từ Reddit.

3.2 Cấu trúc Pipeline

Để có thể phân loại được cảm xúc của các bài đăng có liên quan tới GPT từ Reddit theo thời gian thực thì Pipeline của chúng ta có cấu trúc như sau: Reddit Streaming để lấy các submissions và đưa vào Kafka, sau đó Spark Streaming sẽ đọc dữ liệu từ Kafka để thực hiện công việc xử lí, dự đoán cảm xúc của submissions. Tất cả quá trình đều diễn ra theo thời gian thực và mỗi quá trình được mô tả như sau:

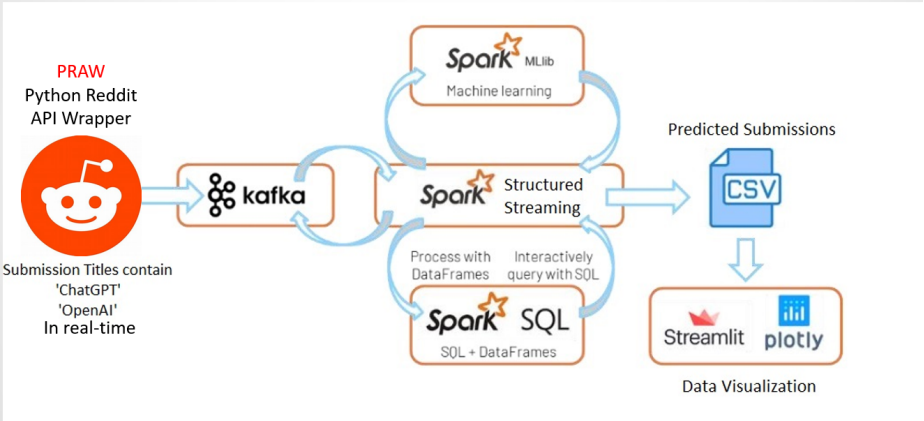

Fig. 4. Kiến trúc hệ thống online

Thu thập Submissions thông qua Reddit Streaming Ta sẽ lấy tiêu đề các Submission có liên quan tới ChatGPT bằng cách thu thập những tiêu đề có chứa một trong các từ 'ChatGPT', 'OpenAI', 'GPT'. Để thu thập các dữ liệu nói trên theo thời gian thực ta sử dụng tính năng Stream Generator được cung cấp bởi Python Package là Praw và Reddit API. Ta sẽ tiến hành tạo ra ba Submission-Title Producer ứng với ba từ khóa trên. Các Producer sẽ đẩy các Submission-Titles thu thập được vào Kafka.

Xử lí và phân loại cảm xúc theo thời gian thực Tiêu đề của các Submissions có liên quan tới ChatGPT sau khi được đẩy vào Kafka thì Spark Streaming sẽ tiến hành đọc dữ liệu từ Kafka. Tiếp theo, ta tiến hành tiền xử lí dữ liệu bằng

Spark SQL + Dataframe và đưa vào mô hình phân loại cảm xúc (xây dựng trên nền SparkMlib) được chọn ra trước đó. Sau khi mô hình thực thi phân loại nhãn cho văn bản, Spark Streaming sẽ ghi dữ liệu đã được phân loại vào một file CSV.

Trực quan hóa dữ liệu Dữ liệu trong file csv sẽ được minh họa bằng giao diện web giúp cho việc theo dõi hệ thống dễ dàng hơn. Trong đồ án này, nhóm sử dụng 2 thư viện là streamlit và plotly để trực quan hóa dữ liệu. Streamlit là thư viện dùng để tạo giao diện đồ họa web, còn plotly dùng để vẽ biểu đồ. Hệ thống sẽ cập nhật liên tục mỗi khi có dữ liệu mới được thêm vào file csv. Dữ liệu được minh họa tổng quát bằng biểu đồ đường, biểu thị số lượng nhãn của từng lớp là Positive, Neutral và Negative trên toàn bộ dữ liệu. *Fig*.5

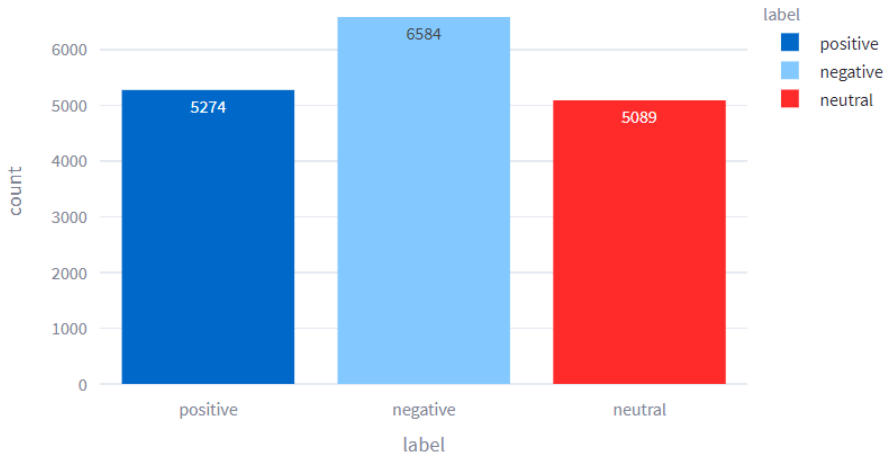

Fig. 5. Biểu đồ cột tổng quát

Tuy nhiên, để có được một góc nhìn khách quan hơn về cảm xúc của cộng đồng về ChatGPT, dữ liệu còn được minh họa bởi một biểu đồ đường, biểu thị số lượng các Submissions Positive, Negative và Neutral theo ngày. *Fig*.6

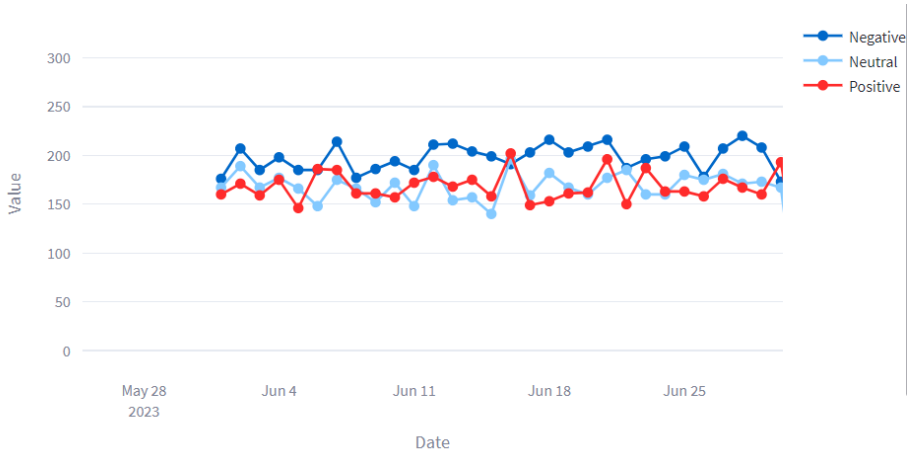

Fig. 6. Biểu đồ đường theo ngày

4 Hướng phát triển

Hệ thống tuy đã khá hoàn thiện nhưng vẫn cần cải thiện và phát triển một số yếu tố.

- Ngoài việc phân tích cảm xúc của cộng đồng người dùng về ChatGPT, chúng ta cũng có thể sử dụng hệ thống vừa xây dựng để tiếp tục phát triển bài toán cho các sản phẩm công nghệ tiên tiến khác hiện nay như Google Bard AI, Midjourney, DALL-E,...
- Hệ thống hiện tại đang hoạt động tốt với việc liên tục cập nhật dữ liệu bằng cách overwrite lại toàn bộ dữ liệu. Tuy nhiên, với lượng dữ liệu liên tục tăng theo thời gian thì để duy trì được hệ thống cần phải thay đổi lại cách cập nhật dữ liệu.
- Dữ liệu đầu vào đang được sử dụng là các Submission Titles, không phản ánh được toàn bộ nội dung của một Submission. Dựa vào Apache Kafka và PRAW, ta có thể lấy thêm các bình luận hoặc nội dung với Submission tương ứng, từ đó có thể dữ liệu cho việc phân loại cảm xúc.
- Mô hình Web hiện tại chỉ có hai biểu đồ là thành phần chính. Để hệ thống có thể giúp ích hơn cho người sử dụng, ta có thể phát triển Web thành một Interactive Web với nhiều tính năng như xử lí cơ sở dữ liệu SQL trực tuyến, thay đổi loại biểu đồ được sử dụng,...

5 Kết luận chung

Nhóm đã trình bày về cách xây dựng một hệ thống phân loại cảm xúc dựa vào các Submission Title theo thời gian thực để theo dõi cảm xúc của cộng đồng về ChatGPT. Hệ thống được triển khai dựa vào sự hỗ trợ của các công nghệ

PRAW $PythonRedditAPIWrapper$, Apache Kafka, Apache Spark. Nhóm cũng đã thực hiện huấn luyện một số mô hình máy học bằng thư viện Spark Machine Learning, từ đó chọn ra mô hình phân loại cho kết quả tốt nhất để phân loại cảm xúc là bộ phân lớp LinearSVC với các bộ trích xuất đặc trưng là Unigram, HashingTF, IDF. Dữ liệu sau đó sẽ được mô hình hóa bằng biểu đồ và web với 2 công cụ là Streamlit và Plotly. Hệ thống cho chúng ta một cái nhìn trực quan về cảm xúc của cộng đồng về ChatGPT theo thời gian bằng sự chênh lệch giữa các nhãn Negative và Positive. Đóng góp chính của bài nghiên cứu là cấu trúc pipeline cho phép tự động phân loại cảm xúc theo thời gian thực, từ đó tạo tiền đề cho các đề tài nghiên cứu mới.

References

1. N. Chintalapudi, G. Battineni, and F. Amenta, “Sentimental Analysis of COVID-19 Tweets Using Deep Learning Models,” Infect. Dis. Rep., vol. 13, no. 2, Art. no. 2, Jun. 2021, doi: 10.3390/idr13020032.
2. X. Zhang, H. Saleh, E. M. G. Younis, R. Sahal, and A. A. Ali, “Predicting Coronavirus Pandemic in Real-Time Using Machine Learning and Big Data Streaming System,” Complexity, vol. 2020, p. e6688912, Dec. 2020, doi: 10.1155/2020/6688912.
3. H. Elzayady, K. M. Badran, and G. I. Salama, “Sentiment Analysis on Twitter Data using Apache Spark Framework,” in 2018 13th International Conference on Computer Engineering and Systems (ICCES), Dec. 2018, pp. 171–176. doi: 10.1109/ICCES.2018.8639195.