

ĐẠI HỌC QUỐC GIA TP HCM
TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN
KHOA CÔNG NGHỆ THÔNG TIN



CSC17107 - Ứng dụng phân tích dữ liệu thông minh

Báo cáo Đồ án Cuối kỳ

Đề tài:

**Phân tích dữ liệu TikTok và Xây dựng công cụ
hỗ trợ viết kịch bản cho video TikTok**

Sinh viên thực hiện:

21127731 - Nguyễn Trọng Tín

21127038 - Võ Phú Hân

21127351 - Hồ Đình Duy Lực

21127739 - Vũ Minh Phát

21127742 - Nguyễn Minh Hiếu

19127216 - Đặng Hoàn Mỹ

Giảng viên hướng dẫn:

Nguyễn Tiến Huy

Nguyễn Trần Duy Minh

Ngày 10 tháng 5 năm 2025

Thông tin nhóm

Lớp: Ứng dụng phân tích dữ liệu thông minh - 21KHDL

Sinh viên thực hiện: Nhóm 01 - Data Explorers

STT	MSSV	Họ và tên	Mức độ đóng góp
1	21127731	Nguyễn Trọng Tín	100%
2	21127038	Võ Phú Hân	100%
3	21127351	Hồ Đình Duy Lực	100%
4	21127739	Vũ Minh Phát	100%
5	21127742	Nguyễn Minh Hiếu	$\leq 50\%$
6	19127216	Đặng Hoàn Mỹ	100%

Bảng phân công công việc:

STT	Công việc	Thành viên phụ trách
1	Thu thập dữ liệu	Võ Phú Hân
2	Khám phá và tiền xử lý dữ liệu	Vũ Minh Phát
3	Rút trích đặc trưng	Vũ Minh Phát, Võ Phú Hân
4	Xây dựng công cụ hỗ trợ viết kịch bản cho video TikTok	Võ Phú Hân
5	Xây dựng các dashboard phân tích user	Hồ Đình Duy Lực
6	Xây dựng các dashboard phân tích video	Đặng Hoàn Mỹ
7	Xây dựng các dashboard phân tích xu hướng	Nguyễn Trọng Tín
8	Sử dụng AI để tự động tạo nhận xét từ biểu đồ và tạo báo cáo từ dữ liệu	Vũ Minh Phát, Đặng Hoàn Mỹ
9	Xây dựng hai công cụ hỗ trợ: công cụ hỗ trợ nghiên cứu chủ đề và công cụ gợi ý cách quay video ẩm thực	Vũ Minh Phát
10	Đề xuất các câu hỏi nghiên cứu cần giải đáp và thực hiện kiểm định giả thuyết thống kê để chứng minh/bác bỏ các giả định	Vũ Minh Phát
11	Viết báo cáo	Nguyễn Trọng Tín, Võ Phú Hân, Hồ Đình Duy Lực, Vũ Minh Phát, Đặng Hoàn Mỹ
12	Quay video hướng dẫn sử dụng bộ ba công cụ	Võ Phú Hân, Vũ Minh Phát

Mục lục

Thông tin nhóm	i
1 CHƯƠNG 1: GIỚI THIỆU CHUNG	1
1.1 Tổng quan về chủ đề	1
1.2 Động lực nghiên cứu	1
1.3 Phạm vi nghiên cứu	2
2 CHƯƠNG 2: TỔNG QUAN VỀ CÁC CÔNG NGHỆ ĐƯỢC SỬ DỤNG TRONG ĐỒ ÁN	4
2.1 Thu thập dữ liệu từ TikTok	4
2.2 Khám phá và tiền xử lý dữ liệu	4
2.2.1 Sử dụng bộ thư viện Python phổ biến	4
2.2.2 Sử dụng Jupyter Notebook	5
2.2.3 Lưu trữ dữ liệu đã tiền xử lý dưới định dạng Parquet	5
2.3 Trích xuất đặc trưng	6
2.3.1 Tạo đường dẫn đến video TikTok	7
2.3.2 Tải xuống audio bằng FFmpeg và YT-DLP	7
2.3.3 Sử dụng Gemini API để trích xuất transcript và xác định nội dung	7
2.3.4 Sử dụng Gemini API để phân loại video	7
2.4 Xây dựng dashboard và webapp	8
2.4.1 Tạo biểu đồ tương tác với Plotly và nhận xét từ Gemini API	8
2.4.2 Xây dựng webapp với Streamlit	8
2.4.3 Triển khai trên Streamlit Community Cloud	9
2.5 Xây dựng ứng dụng Streamlit đa trang (Multipage App)	9
2.6 Cache dữ liệu để giảm thời gian tải trang	10
2.7 Quy trình sử dụng Gemini API để tạo nhận xét cho các biểu đồ	10
2.7.1 Xác định rõ nhiệm vụ trọng tâm	11
2.7.2 Cung cấp “Hướng dẫn chi tiết” cho AI (Prompt Engineering)	11
2.7.3 Cung cấp “Đầu vào” cho AI	11
2.7.4 Ví dụ minh họa về câu prompt	11
3 CHƯƠNG 3: THU THẬP DỮ LIỆU	14

4	CHƯƠNG 4: TIỀN XỬ LÝ DỮ LIỆU VÀ RÚT TRÍCH ĐẶC TRƯNG	16
4.1	Giới thiệu chung	16
4.2	Khám phá và Tiền xử lý dữ liệu (Data Exploration and Preprocessing - DEP)	16
4.2.1	Tìm hiểu tập dữ liệu thô	16
4.2.2	Phân tích và xử lý dữ liệu trùng lặp (Duplicate Analysis)	17
4.2.3	Phân tích và xử lý giá trị bị thiếu (Missing Value Analysis)	18
4.2.4	Phân tích và chuẩn hóa kiểu dữ liệu	18
4.2.5	Phân tích phân bố giá trị	19
4.2.6	Loại bỏ các cột không cần thiết/trùng lặp	19
4.2.7	Xử lý dữ liệu thiếu trong các cột còn lại	20
4.2.8	Lưu trữ dữ liệu đã xử lý	20
4.3	Rút trích đặc trưng (Feature Engineering - FE)	20
4.3.1	Trích xuất hashtag từ mô tả video (<code>desc</code>)	21
4.3.2	Trích xuất đặc trưng thời gian từ <code>createTime</code>	21
4.3.3	Chọn top video hàng tuần để phân tích nội dung	22
4.3.4	Trích xuất nội dung audio thành văn bản (Audio Transcription)	23
4.3.5	Trích xuất thông tin món ăn và địa điểm	23
4.3.6	Trích xuất đặc trưng về thể loại của video	24
5	CHƯƠNG 5: CÔNG CỤ HỖ TRỢ VIẾT KỊCH BẢN CHO VIDEO TIKTOK	25
5.1	Trích xuất đặc trưng nội dung	25
5.2	Lọc video tương đồng theo mô tả người dùng	26
5.3	Sinh kịch bản thô từ video mẫu	27
5.4	Định dạng lại kịch bản có cấu trúc	27
5.5	Triển khai hệ thống trên giao diện web	29
6	CHƯƠNG 6: DASHBOARD PHÂN TÍCH USER	30
6.1	Phân phối của dữ liệu	30
6.1.1	Phân Phối Số Lượt Thích	30
6.1.2	Phân Phối Số Người Theo Dõi	31
6.1.3	Phân Phối Số Lượng Video	31
6.2	Phân Tích Tương Quan	32
6.3	Phân Tích Mức Độ Tương Tác Theo Nhóm Người Theo Dõi	34
6.3.1	Phân Chia Nhóm Người Theo Dõi	34

6.3.2	Bảng So Sánh Mức Độ Tương Tác	35
6.4	Ảnh Hưởng của Số Lượng Video Đến Tương Tác	35
6.5	Phân tích ý nghĩa thống kê về tần suất đăng tải video mỗi tuần, số lượng hashtag trung bình trên mỗi video và thời lượng video trung bình giữa các nhóm người dùng	39
6.5.1	Về Tần Suất Đăng Tải Video Mỗi Tuần	39
6.5.2	Về Số Lượng Hashtag	41
6.5.3	Về Thời Lượng Video Trung Bình	43
6.6	Tổng quan Kỹ thuật	45
6.6.1	Nguồn Dữ liệu	46
6.6.2	Hằng số	47
6.6.3	Hàm Tiện ích Chung	48
6.6.4	Hàm Tiện ích và Trực quan hóa theo Chức năng Bảng Điều khiển	48
6.6.5	Bố cục Chung của các Bảng Điều khiển	50
6.6.6	Lưu trữ Đệm (Caching)	51
6.6.7	Tích hợp Trí tuệ Nhân tạo (AI Integration)	52
6.6.8	Kiểm định Thống kê	52
6.6.9	Hướng dẫn tương tác trên trang web	52
7	CHƯƠNG 7: DASHBOARD PHÂN TÍCH VIDEO	54
7.1	Giới thiệu	54
7.2	Kiến trúc Hệ thống và Công nghệ Sử dụng	54
7.3	Luồng Dữ liệu và Tiền xử lý	55
7.4	Các Trang Phân tích	55
7.4.1	Phân tích Hiệu suất Video	55
7.4.2	Phân tích Hashtag Tổng quan	57
7.4.3	Phân tích Hashtag Đơn lẻ	60
7.5	Tích hợp Trí tuệ Nhân tạo (AI)	62
8	CHƯƠNG 8: DASHBOARD PHÂN TÍCH XU HƯỚNG	64
8.1	Giới thiệu và Tổng quan dữ liệu	64
8.2	Phân tích Xu hướng theo thời gian	64
8.3	Phân tích Phân bố địa lý	65
8.3.1	Phân bố Tỉnh/Thành phố được đề cập	65
8.3.2	Phân bố Tỉnh/Thành phố thuộc từng vùng miền	66

8.3.3	Phân bố Quận/Huyện thuộc Tỉnh/Thành phố	67
8.4	Phân tích Món ăn theo danh mục và Xu hướng	68
8.4.1	Phân tích món ăn theo danh mục	68
8.4.2	Phân tích món ăn Nổi bật qua từng tuần	69
9	CHƯƠNG 9: MỘT VÀI CÔNG CỤ BỔ TRỢ CHO VIỆC PHÁT TRIỂN KÊNH TIKTOK	71
9.1	Giới thiệu chung	71
9.2	Công cụ hỗ trợ nghiên cứu chủ đề	71
9.2.1	Tổng quan	71
9.2.2	Mục tiêu và Lợi ích	72
9.2.3	Kiến trúc và Công nghệ sử dụng	72
9.2.4	Thiết kế giao diện người dùng và Luồng tương tác	73
9.2.5	Kỹ thuật Prompting	76
9.2.6	Tương tác với Gemini API và Xử lý kết quả	77
9.2.7	Minh họa cách sử dụng công cụ	77
9.3	Công cụ gợi ý cách quay video âm thực	78
9.3.1	Tổng quan	78
9.3.2	Mục tiêu và Lợi ích	79
9.3.3	Kiến trúc và Công nghệ sử dụng	79
9.3.4	Thiết kế giao diện người dùng và Luồng tương tác	80
9.3.5	Kỹ thuật Prompting chuyên biệt theo thể loại video	82
9.3.6	Tương tác với Gemini API và Xử lý kết quả	83
9.3.7	Minh họa cách sử dụng công cụ	84
10	CHƯƠNG 10: BÀN LUẬN VÀ KẾT LUẬN	85
10.1	Bàn luận	85
10.1.1	Tóm tắt các kết quả đạt được	85
10.1.2	Các phát hiện chính từ phân tích dữ liệu TikTok	86
10.1.3	Ý nghĩa và Đóng góp	87
10.2	Kết luận và Hướng phát triển tương lai	87
10.2.1	Kết luận	87
10.2.2	Hướng phát triển tương lai	88

Tài liệu tham khảo

90

1 CHƯƠNG 1: GIỚI THIỆU CHUNG

1.1 Tổng quan về chủ đề

Trong bối cảnh bùng nổ của các nền tảng mạng xã hội video ngắn, **TikTok** đã khẳng định vị thế là một trong những ứng dụng phổ biến nhất toàn cầu. Đặc trưng bởi các video có thời lượng ngắn, TikTok đặt ra một thách thức không nhỏ cho các nhà sáng tạo nội dung: “*Làm thế nào để thu hút sự chú ý của người xem ngay từ những giây đầu tiên và duy trì sự quan tâm của họ trong suốt thời lượng video?*”. Việc tạo ra nội dung **hấp dẫn, độc đáo và có khả năng lan tỏa (viral)** trở thành yếu tố then chốt quyết định sự thành công trên nền tảng này.

Tuy nhiên, quá trình lên ý tưởng và xây dựng kịch bản cho video TikTok thường đòi hỏi sự sáng tạo liên tục và khả năng nắm bắt xu hướng nhanh nhạy. Điều này tạo ra nhu cầu về các phương pháp và công cụ hỗ trợ hiệu quả, giúp tối ưu hóa quy trình sáng tạo nội dung. Nhận thức được tiềm năng của việc ứng dụng khoa học dữ liệu vào lĩnh vực này, đồ án “**Phân tích dữ liệu TikTok và Xây dựng công cụ hỗ trợ viết kịch bản cho video TikTok**” được thực hiện. Mục tiêu chính của đồ án là khám phá các yếu tố cấu thành nên sự thành công của video TikTok thông qua phân tích dữ liệu và từ đó, phát triển một công cụ có khả năng hỗ trợ người dùng trong việc xây dựng kịch bản video hiệu quả hơn.

1.2 Động lực nghiên cứu

Sự cạnh tranh trên nền tảng TikTok ngày càng gia tăng, đòi hỏi các nhà sáng tạo phải liên tục cải tiến nội dung và phương thức sản xuất. Việc xây dựng một kịch bản thu hút, giữ chân người xem và đạt được mục tiêu truyền thông không phải là điều dễ dàng, thường phụ thuộc nhiều vào kinh nghiệm cá nhân và cảm nhận chủ quan.

Hiểu được thách thức này, nhóm nhận thấy rằng việc áp dụng một quy trình khoa học dữ liệu bài bản - bao gồm thu thập, xử lý và phân tích dữ liệu từ chính các video trên TikTok - có thể mang lại những giá trị thiết thực. Cụ thể, nghiên cứu của nhóm được thúc đẩy bởi các mục tiêu sau:

1. **Khai thác thông tin hữu ích:** Thông qua phân tích dữ liệu, đồ án hướng tới việc rút ra **những thông tin hữu ích** về các cấu trúc kịch bản, mô-típ nội dung, cách sử dụng hình ảnh, âm thanh, thời lượng tối ưu và các yếu tố khác thường xuất hiện trong các video thành công.

2. **Xác định yếu tố thành công:** Đi sâu vào phân tích để **tìm hiểu những yếu tố** cụ thể (ví dụ: cách mở đầu, cao trào, lời kêu gọi hành động, yếu tố bất ngờ, âm nhạc thịnh hành, v.v.) đóng góp vào mức độ tương tác và khả năng lan tỏa của video.
3. **Phát triển công cụ hỗ trợ:** Dựa trên những hiểu biết thu được từ dữ liệu, mục tiêu cuối cùng là xây dựng một công cụ có khả năng gợi ý, đề xuất cấu trúc, thậm chí hỗ trợ **xây dựng kịch bản hoàn chỉnh**, giúp các nhà sáng tạo nội dung tiết kiệm thời gian, công sức và nâng cao chất lượng sản phẩm video của mình.

Động lực chính của nhóm là mong muốn đóng góp một giải pháp dựa trên dữ liệu, giúp các nhà sáng tạo nội dung trên TikTok, đặc biệt là những người mới bắt đầu hoặc gặp khó khăn trong việc lên ý tưởng, có thể tạo ra những kịch bản hiệu quả và hấp dẫn hơn.

1.3 Phạm vi nghiên cứu

Nội dung trên nền tảng TikTok vô cùng phong phú và đa dạng, bao trùm nhiều lĩnh vực khác nhau. Để đảm bảo tính khả thi và chiều sâu cho nghiên cứu trong khuôn khổ một đồ án môn học, việc giới hạn phạm vi nghiên cứu là cần thiết.

Do đó, nhóm đã quyết định tập trung phân tích và xây dựng công cụ hỗ trợ viết kịch bản cho một lĩnh vực cụ thể: **ẩm thực (food)**. Đây là một trong những chủ đề phổ biến, có lượng người xem và nhà sáng tạo nội dung đông đảo trên TikTok, đồng thời có những đặc trưng riêng về mặt nội dung và hình thức thể hiện.

Phạm vi nghiên cứu của đồ án sẽ bao gồm các hoạt động chính sau:

- **Thu thập dữ liệu:** Tập trung vào các video TikTok thuộc chủ đề ẩm thực, thu thập các thông tin liên quan như lượt xem, lượt thích, bình luận, chia sẻ, thời lượng, mô tả, hashtag, âm thanh sử dụng, và các yếu tố khác có thể định lượng được.
- **Phân tích dữ liệu:** Sử dụng các phương pháp thống kê, khai phá dữ liệu và học máy để phân tích tập dữ liệu đã thu thập, nhằm xác định các đặc điểm, xu hướng và yếu tố ảnh hưởng đến sự thành công của video ẩm thực trên TikTok.
- **Xây dựng công cụ:** Phát triển một công cụ dưới dạng ứng dụng web có chức năng chính là hỗ trợ người dùng viết kịch bản cho video TikTok về ẩm thực, dựa trên các kết quả phân tích đã thực hiện.

Việc giới hạn nghiên cứu trong lĩnh vực ẩm thực cho phép nhóm đi sâu tìm hiểu các đặc thù của ngành nội dung này, từ đó đưa ra những phân tích chính xác hơn và xây dựng một công cụ hỗ trợ phù hợp, đáp ứng tốt hơn nhu cầu của các nhà sáng tạo nội dung trong lĩnh vực này.

2 CHƯƠNG 2: TỔNG QUAN VỀ CÁC CÔNG NGHỆ ĐƯỢC SỬ DỤNG TRONG ĐỒ ÁN

2.1 Thu thập dữ liệu từ TikTok

Trong giai đoạn đầu tiên của đồ án, nhóm sẽ tiến hành thu thập dữ liệu từ nền tảng TikTok. Công cụ chính được sử dụng cho mục đích này là thư viện mã nguồn mở **TikTok-API** [1], một dự án được phát triển và duy trì bởi cộng đồng lập trình viên, có thể truy cập tại đường dẫn chính thức trên GitHub: <https://github.com/davidteather/TikTok-API>.

Cần lưu ý rằng, **TikTok-API** không phải là API chính thức do TikTok cung cấp. Thay vào đó, thư viện này được tạo ra nhằm mục đích hỗ trợ các hoạt động học tập và nghiên cứu, cho phép người dùng truy xuất dữ liệu công khai từ TikTok, ví dụ như danh sách video thịnh hành, thông tin người dùng hay các hashtag phổ biến.

Về mặt kỹ thuật, **TikTok-API** hoạt động bằng cách gọi các endpoint ẩn của TikTok và trả về dữ liệu dưới định dạng JSON. Ứng dụng thư viện này, nhóm sẽ tự động thu thập các thông tin chi tiết về video (bao gồm tiêu đề, số lượt thích, số lượt chia sẻ và các hashtag liên quan, v.v.), đồng thời lấy dữ liệu của những người dùng có liên quan đến các video đó.

2.2 Khám phá và tiền xử lý dữ liệu

2.2.1 Sử dụng bộ thư viện Python phổ biến

Sau khi dữ liệu thô từ TikTok được thu thập, giai đoạn tiếp theo là khám phá và tiền xử lý dữ liệu để chuẩn bị cho các bước phân tích sâu hơn. Trong đồ án này, nhóm đã sử dụng một bộ thư viện rất phổ biến trong hệ sinh thái Python cho khoa học dữ liệu, bao gồm **Pandas** [2], **NumPy** [3], **Matplotlib** [4], và **Seaborn** [5].

Thư viện chính được sử dụng là **Pandas**, cung cấp cấu trúc DataFrame để thao tác dữ liệu linh hoạt. Chức năng của Pandas bao gồm đọc file (CSV, JSON, Excel), lọc, gộp, tính toán thống kê và xử lý giá trị thiếu. Trong đồ án, nhóm áp dụng Pandas để: đọc dữ liệu đầu vào (file CSV được tổng hợp từ các file JSON), kiểm tra và xử lý giá trị rỗng/thiếu (sử dụng các hàm `dropna()`, `fillna()`, `drop_duplicates()`), chuyển đổi kiểu dữ liệu (datetime, không phải số, v.v.) và tóm tắt thông tin ban đầu (như tính mean, median, đếm số dòng theo nhóm). Nhóm cũng dùng Pandas để khám phá dữ liệu ban đầu, ví dụ hiển thị một vài dòng dữ liệu mẫu, biểu đồ phân bố đơn giản bằng hàm `describe()` để hiểu vùng dữ liệu, phát hiện điểm bất thường

(*outlier*) hoặc dữ liệu bị lỗi. Quá trình này giúp dữ liệu sạch hơn và phù hợp cho việc trích xuất đặc trưng sau đó.

Bên cạnh Pandas, nhóm còn sử dụng:

- **NumPy:** Cung cấp nền tảng cho các tính toán số học hiệu quả, thường được dùng kết hợp với Pandas.
- **Matplotlib** và **Seaborn:** Các thư viện mạnh mẽ để trực quan hóa dữ liệu, giúp nhận diện các mẫu, xu hướng và hiểu rõ hơn về cấu trúc dữ liệu thu thập được. Seaborn được xây dựng trên Matplotlib và cung cấp khả năng tạo biểu đồ thống kê đẹp mắt, phức tạp hơn.

2.2.2 Sử dụng Jupyter Notebook

Các hoạt động khám phá và tiền xử lý dữ liệu trong đồ án này chủ yếu được thực hiện trên các file **Jupyter Notebook** [6]. Jupyter Notebook là một môi trường điện toán tương tác mạnh mẽ, cho phép người dùng viết và thực thi code theo từng ô riêng biệt, đồng thời hiển thị kết quả (bao gồm văn bản, bảng biểu và biểu đồ) ngay bên dưới.

Cơ chế hoạt động này của Jupyter Notebook đặc biệt phù hợp cho việc khám phá dữ liệu. Nó cho phép nhóm thực hiện các bước tiền xử lý và phân tích một cách lặp đi lặp lại (*iterative*), xem xét kết quả ngay lập tức sau mỗi thao tác và dễ dàng điều chỉnh nếu cần thiết. Điều này rất quan trọng khi làm việc với các tập dữ liệu mới hoặc phức tạp, hỗ trợ quá trình hiểu rõ dữ liệu một cách liên tục.

Nhờ tính linh hoạt và khả năng tích hợp cao với các thư viện khoa học dữ liệu phổ biến như Pandas, NumPy, Matplotlib, và Seaborn, Jupyter Notebook đã trở thành công cụ tiêu chuẩn trong lĩnh vực này. Ngoài ra, việc kết hợp code, giải thích bằng văn bản và hình ảnh trực quan trong cùng một tài liệu còn giúp ghi lại chi tiết quy trình phân tích, tăng cường khả năng tái hiện và chia sẻ kết quả nghiên cứu của đồ án.

2.2.3 Lưu trữ dữ liệu đã tiền xử lý dưới định dạng Parquet

Sau khi dữ liệu TikTok được khám phá và tiền xử lý bằng các thư viện Python, bước tiếp theo là lưu trữ dữ liệu đã làm sạch và chuyển đổi này một cách hiệu quả để sẵn sàng cho các giai đoạn phân tích sau. Để tối ưu hóa hiệu suất và hiệu quả lưu trữ, đồ án đã lựa chọn định dạng **Parquet** [7] thay vì các định dạng truyền thống như CSV.

Khác biệt cốt lõi mang lại lợi thế cho Parquet là cơ chế lưu trữ dữ liệu theo cột (*column-oriented*), thay vì theo hàng (*row-oriented*) như CSV. Thiết kế này cho phép Parquet nén dữ

liệu hiệu quả hơn, đặc biệt khi các giá trị trong cùng một cột có nhiều điểm tương đồng. Đồng thời, nó giúp tăng tốc đáng kể các truy vấn chỉ cần đọc một tập hợp con các cột, giảm thiểu lượng dữ liệu cần đọc từ đĩa (I/O).

Ngoài ra, Parquet còn có khả năng **giữ nguyên và bao gồm thông tin về kiểu dữ liệu** (*schema*) của mỗi cột. Điều này rất quan trọng để đảm bảo tính chính xác và nhất quán của dữ liệu khi được tải lại và sử dụng trong các bước phân tích tiếp theo.

Bảng 1: So sánh định dạng Parquet và CSV

Tính năng	Parquet	CSV
Hiệu quả lưu trữ	Nén dữ liệu theo cột hiệu quả, đặc biệt với dữ liệu trùng lặp	Nén kém hiệu quả, thường dẫn đến kích thước file lớn hơn
Hiệu suất truy vấn	Truy vấn nhanh hơn khi chỉ cần đọc một số cột cụ thể, giảm I/O	Cần đọc toàn bộ file ngay cả khi chỉ cần một vài cột
Hỗ trợ schema	Bao gồm thông tin về kiểu dữ liệu và tên cột trong file	Không có schema tích hợp, dễ gây lỗi về kiểu dữ liệu
Hỗ trợ kiểu dữ liệu phức tạp	Xử lý được các cấu trúc dữ liệu lồng nhau	Chỉ lưu trữ dữ liệu dạng phẳng
Khả năng nén	Tích hợp nhiều thuật toán nén hiệu quả theo cột	Khả năng nén hạn chế, thường cần nén file riêng biệt
Khả năng đọc/ghi	Đọc nhanh hơn cho các truy vấn chọn lọc cột, ghi có thể chậm hơn	Đọc/ghi đơn giản, có thể nhanh hơn cho các thao tác trên toàn bộ hàng
Tính phổ biến	Ngày càng phổ biến trong các hệ thống xử lý dữ liệu lớn	Định dạng phổ biến, dễ dàng được hỗ trợ bởi nhiều công cụ và hệ thống

Việc lựa chọn Parquet đặc biệt phù hợp với dữ liệu TikTok, vốn có thể có dung lượng rất lớn. Nhờ khả năng nén và truy vấn hiệu quả theo cột, Parquet giúp **giảm chi phí lưu trữ** và tăng tốc độ phân tích dữ liệu khổng lồ này so với CSV, mang lại hiệu quả xử lý cao hơn cho đồ án.

2.3 Trích xuất đặc trưng

Giai đoạn trích xuất đặc trưng tập trung vào việc rút trích các thông tin và dữ liệu hữu ích từ dữ liệu TikTok đã thu thập và tiền xử lý, làm cơ sở cho các bước phân tích và tạo nội dung sau này.

Trong mục này, nhóm chỉ giới thiệu tổng quan về các công nghệ và công cụ chính được sử dụng để thực hiện các tác vụ trích xuất này. Chi tiết kỹ thuật cụ thể về cách triển khai từng bước xử lý sẽ được trình bày đầy đủ và chi tiết trong các chương tiếp theo của báo cáo.

2.3.1 Tạo đường dẫn đến video TikTok

Để dễ dàng tham chiếu đến nội dung gốc, nhóm đã tạo đường dẫn trực tiếp đến mỗi video TikTok bằng cách kết hợp ID duy nhất của video và thông tin người đăng tải.

2.3.2 Tải xuống audio bằng FFmpeg và YT-DLP

Một bước quan trọng trong trích xuất đặc trưng là thu thập thông tin âm thanh từ video. Do dung lượng file video gốc rất lớn, nhóm đã chọn phương pháp chỉ tải xuống file audio tương ứng để tiết kiệm đáng kể tài nguyên lưu trữ [8]. Tác vụ này được thực hiện nhờ sự kết hợp của hai công cụ mạnh mẽ: **YT-DLP** [9] và **FFmpeg** [10].

YT-DLP là một công cụ dòng lệnh mã nguồn mở phổ biến, được xem là một fork cải tiến của dự án youtube-dl đã ngừng hoạt động. YT-DLP được sử dụng rộng rãi để **kết nối** và **tải xuống** video và audio từ nhiều trang web khác nhau (bao gồm cả TikTok và YouTube) với chất lượng tốt nhất.

FFmpeg là một framework đa phương tiện mã nguồn mở rất mạnh mẽ, hỗ trợ xử lý audio sau khi tải về, đảm bảo việc **trích xuất và chuyển đổi định dạng âm thanh** được thực hiện hiệu quả.

2.3.3 Sử dụng Gemini API để trích xuất transcript và xác định nội dung

Sau khi đã tải xuống file audio của video TikTok, bước tiếp theo trong quá trình trích xuất đặc trưng là chuyển đổi nội dung âm thanh này thành văn bản (transcript). Để thực hiện tác vụ chuyển đổi từ audio sang text này, nhóm đã sử dụng **Gemini API**. Được phát triển bởi Google, Gemini API cung cấp khả năng phân tích audio mạnh mẽ, cho phép tạo ra bản ghi với độ chính xác cao nhờ năng lực **nhận dạng giọng nói** và **hiểu ngôn ngữ tự nhiên** tiên tiến.

Sau khi có được transcript từ audio, nhóm kết hợp thông tin này với mô tả gốc của video TikTok. **Gemini API** tiếp tục được ứng dụng để phân tích nguồn văn bản kết hợp này nhằm xác định các thông tin chi tiết về nội dung, cụ thể là các **món ăn** và **địa điểm** được đề cập trong video. **Khả năng hiểu ngữ cảnh và mối quan hệ giữa các từ** của API giúp nhận diện các thực thể này hiệu quả hơn so với việc chỉ dựa vào phân tích từ khóa đơn thuần.

2.3.4 Sử dụng Gemini API để phân loại video

Cuối cùng, **Gemini API** còn được sử dụng cho một tác vụ trích xuất đặc trưng quan trọng khác: rút trích các đặc trưng liên quan đến thể loại video. API chủ yếu phân tích các yếu tố về

nội dung (dưới dạng văn bản) của video để đưa ra phân loại phù hợp.

Thông tin về thể loại video là dữ liệu cấu trúc quan trọng, được sử dụng trực tiếp trong giai đoạn xây dựng kịch bản. Việc phân loại này giúp công cụ có thể đưa ra các gợi ý phù hợp với từng loại nội dung cụ thể, từ đó nâng cao khả năng tạo ra các video hấp dẫn và thu hút người xem.

2.4 Xây dựng dashboard và webapp

2.4.1 Tạo biểu đồ tương tác với Plotly và nhận xét từ Gemini API

Để trực quan hóa dữ liệu đã được phân tích và trích xuất, nhóm đã sử dụng thư viện **Plotly** [11]. Plotly là một thư viện Python mạnh mẽ, cho phép tạo ra các **biểu đồ tương tác** có chất lượng cao. Thư viện này hỗ trợ các loại biểu đồ đa dạng và cung cấp các tính năng tương tác linh hoạt như phóng to, thu nhỏ hay xem thông tin chi tiết khi di chuột, giúp người dùng dễ dàng khám phá dữ liệu.

Không chỉ dừng lại ở việc hiển thị hình ảnh, nhóm còn tích hợp **Gemini API** để tự động hóa việc cung cấp nhận xét và giải thích cho các biểu đồ do Plotly tạo ra. Bằng cách phân tích các biểu đồ và bảng biểu thống kê (đã được chuyển đổi sang dạng văn bản hoặc chuỗi byte), Gemini API có khả năng tạo ra các đoạn văn bản nhận xét, mô tả các xu hướng, mối quan hệ hoặc các điểm nổi bật trong dữ liệu được trực quan hóa.

Sự kết hợp giữa khả năng trực quan hóa tương tác của Plotly và khả năng tự động phân tích, tạo nhận xét của Gemini API tạo nên một giải pháp mạnh mẽ. Nó không chỉ giúp người dùng xem dữ liệu mà còn nhanh chóng có được những hiểu biết sâu sắc hơn về nội dung được biểu diễn.

2.4.2 Xây dựng webapp với Streamlit

Để cung cấp một giao diện người dùng (UI) thân thiện và dễ sử dụng cho các dashboard và công cụ hỗ trợ viết kịch bản, nhóm đã lựa chọn framework **Streamlit** [12] để xây dựng webapp. Streamlit là một framework Python mã nguồn mở, cho phép các nhà khoa học dữ liệu và kỹ sư nhanh chóng biến các script Python thành các ứng dụng web tương tác mà không đòi hỏi kinh nghiệm về phát triển front-end như HTML, CSS hay JavaScript.

Ưu điểm nổi bật của Streamlit là sự **đơn giản** và **tốc độ triển khai**. Framework này rất dễ học, dễ sử dụng và tương thích tốt với hầu hết các thư viện Python phổ biến trong lĩnh vực khoa học dữ liệu, bao gồm Pandas, NumPy, Matplotlib, Seaborn, và đặc biệt là việc hiển thị các

biểu đồ tương tác từ Plotly. Streamlit cung cấp các API trực quan để tạo ra các widget tương tác như nút bấm, thanh trượt, hộp chọn, giúp người dùng dễ dàng tương tác với dữ liệu và các chức năng của ứng dụng.

Nhờ những đặc điểm này, Streamlit là một lựa chọn lý tưởng để nhanh chóng xây dựng giao diện web cho các dashboard và công cụ hỗ trợ viết kịch bản [13]. Nó cho phép người dùng truy cập, tương tác với kết quả phân tích dữ liệu và sử dụng các chức năng của công cụ một cách dễ dàng thông qua trình duyệt web.

2.4.3 Triển khai trên Streamlit Community Cloud

Để người dùng có thể truy cập và sử dụng webapp mà không cần phải cài đặt và chạy ứng dụng trên máy tính cá nhân (local), nhóm đã lựa chọn triển khai sản phẩm hoàn chỉnh trên **Streamlit Community Cloud** [14].

Streamlit Community Cloud là một nền tảng miễn phí được cung cấp bởi Streamlit, cho phép người dùng triển khai, quản lý, và chia sẻ các ứng dụng Streamlit một cách dễ dàng. Quá trình triển khai thường rất đơn giản, chủ yếu bằng cách **kết nối tài khoản GitHub** của người dùng với nền tảng Streamlit Cloud và **chọn repository chứa code của ứng dụng**. Sau khi triển khai, ứng dụng sẽ có một URL duy nhất mà người dùng có thể truy cập thông qua trình duyệt web.

Một ưu điểm nữa là Streamlit Community Cloud có khả năng **tự động cập nhật ứng dụng** mỗi khi có thay đổi được đẩy lên repository GitHub, giúp việc duy trì và phát triển ứng dụng trở nên thuận tiện hơn.

Việc triển khai trên nền tảng đám mây này giúp dễ dàng chia sẻ và tiếp cận ứng dụng với nhiều người dùng mà không cần lo lắng về các vấn đề liên quan đến cơ sở hạ tầng hoặc cấu hình phức tạp. Người dùng có thể truy cập ứng dụng từ bất kỳ đâu có kết nối internet, chỉ cần một trình duyệt web, mà không cần phải cài đặt bất kỳ phần mềm nào trên thiết bị của mình.

2.5 Xây dựng ứng dụng Streamlit đa trang (Multipage App)

Để tạo ra một trải nghiệm người dùng mạch lạc và tiện lợi, nhóm đã tích hợp nhiều trang web (thực chất là các trang khác nhau của webapp Streamlit) thành một website duy nhất có khả năng điều hướng mượt mà giữa các trang. Chức năng này được thực hiện bằng cách sử dụng module **Page** và **navigation** được cung cấp trong thư viện **Streamlit**. Module **st.navigation** đóng vai trò trung tâm trong việc định nghĩa các trang có sẵn trong ứng dụng multipage. Nó hoạt động như một bộ định tuyến, trả về trang hiện tại mà người dùng đã chọn. Để định nghĩa

một trang cụ thể, module **st.Page** được sử dụng để khởi tạo một đối tượng `StreamlitPage`. Trang này có thể được tạo từ một file Python riêng biệt hoặc từ một hàm được định nghĩa trong file chính của ứng dụng.

Streamlit cũng cho phép tạo ra các mục (section) trong menu điều hướng bằng cách truyền một dictionary vào hàm **st.navigation**. Trong dictionary này, mỗi key sẽ là nhãn của section, và value sẽ là một list chứa các đối tượng `StreamlitPage` thuộc section đó. Việc sử dụng **st.navigation** và **st.Page** là phương pháp được khuyến nghị để xây dựng các ứng dụng Streamlit có nhiều trang, giúp tổ chức nội dung một cách logic và cung cấp một hệ thống điều hướng trực quan cho người dùng. Thay vì phải tự mình xây dựng hệ thống điều hướng phức tạp, các module tích hợp sẵn của Streamlit giúp đơn giản hóa quá trình này, cho phép nhà phát triển tập trung vào nội dung và chức năng chính của từng trang trong ứng dụng.

2.6 Cache dữ liệu để giảm thời gian tải trang

Để tối ưu hóa hiệu suất của webapp và mang lại trải nghiệm người dùng mượt mà hơn, đặc biệt khi ứng dụng cần xử lý và hiển thị lượng dữ liệu lớn, nhóm đã sử dụng cơ chế cache dữ liệu được cung cấp bởi Streamlit thông qua decorator **@st.cache_data**.

Cơ chế này cho phép Streamlit lưu trữ kết quả trả về của các hàm (ví dụ: đọc dữ liệu từ file, kết quả truy vấn dữ liệu, các phép biến đổi phức tạp). Khi một hàm được *đánh dấu* bằng **@st.cache_data** được gọi lại với cùng các tham số đầu vào và code, thay vì thực thi lại toàn bộ hàm, Streamlit sẽ nhanh chóng trả về kết quả đã được lưu trữ trong bộ nhớ cache. Điều này giúp tránh lãng phí thời gian và tài nguyên tính toán vào việc thực hiện lại các quy trình xử lý hoặc tải dữ liệu đã có. Streamlit cũng cung cấp các tùy chọn quản lý bộ nhớ cache, chẳng hạn như thiết lập thời gian tồn tại tối đa cho dữ liệu.

Việc sử dụng caching hiệu quả giúp giảm đáng kể thời gian tải trang, đặc biệt là đối với các dữ liệu hoặc kết quả đã được tính toán trước đó, từ đó cải thiện tốc độ phản hồi và nâng cao trải nghiệm cho người dùng. Đây là một kỹ thuật thiết yếu để xây dựng các ứng dụng web hiệu suất cao, nhất là khi làm việc với lượng lớn dữ liệu hoặc các phép tính phức tạp.

2.7 Quy trình sử dụng Gemini API để tạo nhận xét cho các biểu đồ

Để tận dụng khả năng của AI trong việc phân tích và diễn giải dữ liệu trực quan, nhóm đã xây dựng một quy trình sử dụng **Gemini API** để tự động đưa ra nhận xét cho các biểu đồ được tạo bằng thư viện Plotly. Quy trình này bao gồm một số thành phần chính cần được xác

định và cung cấp cho API.

2.7.1 Xác định rõ nhiệm vụ trọng tâm

Bước đầu tiên và quan trọng nhất là phải xác định rõ ràng **nhiệm vụ phân tích cụ thể** mà ta muốn Gemini API thực hiện trên biểu đồ và xây dựng thành một đoạn prompt. Ví dụ, prompt có thể là: “Hãy đưa ra nhận xét ngắn gọn về biểu đồ thể hiện lượt xem trung bình theo ngày”. Đoạn prompt này mô tả rõ ràng mong muốn của người dùng là tạo phần thuyết minh cho biểu đồ đó. Việc xác định rõ nhiệm vụ giúp tập trung quá trình phân tích và đảm bảo rằng các nhận xét được tạo ra sẽ liên quan trực tiếp đến mục tiêu này.

2.7.2 Cung cấp “Hướng dẫn chi tiết” cho AI (Prompt Engineering)

Sau khi đã xác định được nhiệm vụ, bước tiếp theo là cung cấp cho Gemini API những “hướng dẫn chi tiết” về **nội dung và khía cạnh cụ thể** mà nó cần tập trung phân tích. Điều này thường được thực hiện thông qua việc xây dựng một câu prompt (lời nhắc) chi tiết. Trong prompt này, cần chỉ rõ những gì AI cần **tìm kiếm trong biểu đồ**, các **mối quan hệ** hoặc **xu hướng** nào cần được làm nổi bật, và thậm chí cả **cấu trúc đầu ra** mong muốn (ví dụ: một đoạn văn ngắn gọn, một danh sách các điểm chính). Việc cung cấp hướng dẫn chi tiết giúp AI hiểu rõ hơn về mục tiêu phân tích và tạo ra các nhận xét súc tích, chính xác và đúng trọng tâm.

2.7.3 Cung cấp “Đầu vào” cho AI

Thành phần quan trọng nhất để Gemini API có thể tạo ra nhận xét là “đầu vào”. Đầu vào này bao gồm **toàn bộ dữ liệu cần thiết** để AI thực hiện phân tích, cũng như bất kỳ **quy tắc tham khảo** hoặc **thông tin bổ sung** nào có thể giúp AI có cơ sở đầy đủ và chính xác để đưa ra nhận xét.

Tùy thuộc vào loại dữ liệu đầu vào, cách cung cấp có thể khác nhau. Đối với hình ảnh, chẳng hạn như một biểu đồ được tạo bằng Plotly, nó có thể được chuyển đổi thành một chuỗi byte và cung cấp cho API. Đối với các bảng thống kê, chúng có thể được định dạng thành chuỗi LaTeX hoặc Markdown để dễ dàng truyền tải thông tin cấu trúc. Các giá trị scalar (giá trị đơn) có thể được truyền trực tiếp dưới dạng số hoặc chuỗi.

2.7.4 Ví dụ minh họa về câu prompt

Dưới đây là một ví dụ minh họa về cách nhóm đã xây dựng câu prompt để yêu cầu Gemini API tạo nhận xét cho biểu đồ phân tích mối tương quan giữa "Số người theo dõi", "Số lượt

thích", và "Số lượng video" của các TikToker.

```

1 correlation_analysis_prompt = f"""
2 Hãy phân tích mối tương quan giữa 'Số người theo dõi', 'Số lượt thích'
3 và 'Số lượng video' của các TikToker dựa trên dữ liệu được cung cấp.
4 Viết một đoạn phân tích súc tích (khoảng 250-350 từ) tập trung vào:
5     1. Mức độ tương quan (mạnh, trung bình, yếu) giữa các cặp biến
6     2. Hướng tương quan (dương/âm) và ý nghĩa thực tế của nó
7     3. Các điểm bất thường hoặc xu hướng đáng chú ý từ biểu đồ phân tán
8     4. Các hàm ý cho người sáng tạo nội dung TikTok
9
10 Dữ liệu phân tích:
11     1. Biểu đồ phân tán thể hiện mối quan hệ giữa ba chỉ số.
12        Biểu đồ này sẽ được đính kèm dưới dạng byte:
13        {scatter_fig.to_image()}
14
15     2. Bảng ma trận tương quan giữa các chỉ số (hệ số Pearson).
16        Dưới đây là bảng thống kê thể hiện các thông tin này dưới dạng LaTeX:
17        {get_correlation_matrix(select_columns(df, METRICS)).to_latex()}
18
19     3. Thông tin bổ sung:
20         - Hệ số tương quan từ 0.7-1.0: tương quan mạnh
21         - Hệ số tương quan từ 0.3-0.7: tương quan trung bình
22         - Hệ số tương quan từ 0.0-0.3: tương quan yếu
23
24 Cấu trúc phân tích nên bao gồm:
25     - Tổng quan về mức độ tương quan chung giữa các biến
26     - Phân tích chi tiết từng cặp tương quan quan trọng
27     - Kết luận và gợi ý thực tiễn cho người sáng tạo nội dung
28 """

```

Câu prompt này đã được xây dựng một cách chi tiết, bao gồm các hướng dẫn cụ thể về các **khía cạnh cần phân tích**, chẳng hạn như mức độ và hướng của mối tương quan giữa các cặp biến, các điểm bất thường hoặc xu hướng đáng chú ý trên biểu đồ, và các hàm ý thực tế cho người sáng tạo nội dung TikTok.

Prompt cũng chỉ rõ **định dạng của dữ liệu đầu vào**, bao gồm việc biểu đồ phân tán sẽ được cung cấp dưới dạng byte, và bảng ma trận tương quan sẽ được cung cấp dưới dạng LaTeX. Ngoài ra, prompt còn cung cấp thông tin bổ sung về **cách diễn giải hệ số tương quan** (mạnh, trung bình, yếu) và **chỉ định cấu trúc mong muốn** cho đoạn phân tích, bao gồm tổng quan, phân tích chi tiết từng cặp tương quan quan trọng, và kết luận cùng các gợi ý thực tiễn.

Việc thiết kế một câu prompt cẩn thận và chi tiết như vậy là rất quan trọng để tận dụng tối đa khả năng của các mô hình ngôn ngữ lớn như Gemini API trong việc phân tích và tạo ra các nhận xét có giá trị từ dữ liệu trực quan.

3 CHƯƠNG 3: THU THẬP DỮ LIỆU

Dữ liệu được thu thập theo hai giai đoạn chính nhằm đảm bảo tính đại diện, chất lượng và phục vụ tốt cho quá trình phân tích nội dung video TikTok trong lĩnh vực ẩm thực.

Giai đoạn 1: Lọc theo hashtag và hiệu suất tài khoản

- 1. Tìm kiếm và lựa chọn hashtag:** Bắt đầu từ việc thu thập thủ công các **hashtag nổi bật về ẩm thực** trên nền tảng TikTok Việt Nam, ví dụ như #reviewanngon, #ancungtiktok, #diadiemanuong, v.v.. Tiếp theo, sử dụng các hashtag này để **crawl video theo từng hashtag**, từ đó lấy được tập hợp các video ẩm thực đầu tiên.
- 2. Thu thập video theo người dùng:** Từ danh sách video thu thập được, xác định **các tài khoản người dùng có ít nhất 2 video** thuộc các hashtag trên và tiến hành crawl video được đăng trong năm 2024 đến thời điểm thu thập từ các tài khoản này.
- 3. Mở rộng tập hashtag và video:** Dựa trên thống kê sử dụng hashtag trong tập dữ liệu trên, **trích xuất 50 hashtag ẩm thực phổ biến nhất** nhưng không trùng với danh sách ban đầu. Sau đó, thực hiện bước **crawl video mới** theo các hashtag này để mở rộng tập dữ liệu.
- 4. Lọc người dùng theo hiệu suất kênh:** Tập video từ các bước trên được hợp nhất, sau đó lọc các tài khoản dựa trên các tiêu chí định lượng:
 - **Tổng lượt xem** $> 100,000$ (Q1)
 - **Lượt xem trung bình** $> 40,000$ (Q1)
 - **Tỉ lệ tương tác** > 0.03 (Q2), với công thức:

$$\text{Tỉ lệ tương tác} = \frac{\#Like + \#Share + \#Comment + \#Collect}{\#View}$$

Giai đoạn 2: Lọc theo tính đặc trưng của nội dung

Các tài khoản được lọc thêm để đảm bảo họ là các **nhà sáng tạo ở Việt Nam làm về nội dung thuộc chủ đề Ẩm thực** và **vẫn còn hoạt động trong năm 2025**. Cụ thể các tài khoản đáp ứng các tiêu chí sau sẽ bị **loại bỏ** khỏi tập dữ liệu:

- **Không có video đăng trong năm 2025** – loại trừ các tài khoản không còn hoạt động.
- **Ít hơn 75% video liên quan đến ẩm thực** – ưu tiên tài khoản đa dạng nhưng có dấu ấn trong lĩnh vực ẩm thực.
- **Ít hơn 90% video bằng tiếng Việt** – đảm bảo tính nội địa nhưng vẫn cho phép độ mở ngôn ngữ.

Công cụ thu thập dữ liệu

Trong toàn bộ quá trình crawl dữ liệu, nhóm sử dụng thư viện mã nguồn mở **TikTokApi**, giúp mô phỏng hành vi người dùng trên trình duyệt để truy xuất dữ liệu công khai từ nền tảng TikTok. TikTokApi cho phép thu thập video theo hashtag hoặc user, đồng thời hỗ trợ lấy thông tin chi tiết về từng video và tài khoản người dùng một cách tự động và hiệu quả.

Cách lưu trữ dữ liệu thu thập

Với **mỗi tài khoản** người dùng được chọn, hệ thống lưu hai tập `.json`:

- **Tập user:** chứa thông tin hồ sơ và thống kê của tài khoản như `uniqueId`, `nickname`, `followerCount`, `heartCount`, `videoCount`, v.v..
- **Tập video:** chứa danh sách các video của user, định dạng `list of dictionary`, bao gồm các trường như `id`, `desc`, `stats`, `video`, `author`, v.v..

Chi tiết về cấu trúc dữ liệu và các bước xử lý tiếp theo được trình bày tại Chương [4](#).

4 CHƯƠNG 4: TIỀN XỬ LÝ DỮ LIỆU VÀ RÚT TRÍCH ĐẶC TRƯNG

4.1 Giới thiệu chung

Chương này trình bày chi tiết quy trình khám phá, tiền xử lý tập dữ liệu thô thu thập được từ TikTok và các bước rút trích đặc trưng cần thiết. Mục tiêu của quá trình này là chuẩn bị một tập dữ liệu sạch, có cấu trúc tốt và giàu thông tin, sẵn sàng cho các giai đoạn phân tích sâu hơn và làm cơ sở cho việc xây dựng công cụ hỗ trợ viết kịch bản video TikTok.

Dữ liệu thô thu thập được bao gồm hai tập chính: **dữ liệu về video TikTok** (thông tin chi tiết của từng video như lượt xem, thích, chia sẻ, bình luận, v.v.) và **dữ liệu về người dùng TikTok** (thông tin về tài khoản người đăng như số lượng người theo dõi, video đã đăng, v.v.).

Trong chương này, nhóm sẽ tập trung trình bày chi tiết các bước khám phá, tiền xử lý và rút trích đặc trưng đối với tập dữ liệu video TikTok. Lý do là vì quy trình xử lý dữ liệu trên tập người dùng cũng bao gồm nhiều bước tương đồng với dữ liệu video. Do đó, chi tiết về quá trình tiền xử lý dữ liệu người dùng sẽ không được trình bày cụ thể tại đây mà có thể được tìm hiểu kỹ hơn trong file Jupyter Notebook trong link GitHub của nhóm.

Quy trình xử lý chung được áp dụng (đặc biệt trên dữ liệu video) bao gồm các bước chính như tìm hiểu cấu trúc và đặc điểm dữ liệu thô, xử lý các giá trị bị thiếu và trùng lặp, chuẩn hóa kiểu dữ liệu, loại bỏ các cột không cần thiết, và cuối cùng là tạo ra các đặc trưng mới từ dữ liệu văn bản và thời gian để làm giàu tập dữ liệu.

4.2 Khám phá và Tiền xử lý dữ liệu (Data Exploration and Preprocessing - DEP)

Quá trình tiền xử lý dữ liệu được thực hiện chủ yếu dựa trên phân tích trong file `02_DEP_01-Preprocess_video_data.ipynb`.

4.2.1 Tìm hiểu tập dữ liệu thô

Đọc dữ liệu: Tập dữ liệu thô (`final_raw_videos.csv`) được đọc vào DataFrame của Pandas. Ngay từ bước này, kiểu dữ liệu của một số cột đặc biệt (ví dụ: các cột ID, cột chứa giá trị dạng phân loại) đã được xác định trước là `object` (chuỗi) thay vì `int` (số nguyên) để tránh việc

Pandas tự động nhận diện sai và phù hợp hơn với bản chất dữ liệu (không thực hiện phép toán số học trên các cột này).

Kích thước dữ liệu: Tập dữ liệu thô ban đầu bao gồm **71260 hàng** và **174 cột**.

Ý nghĩa dữ liệu: Mỗi hàng trong tập dữ liệu đại diện cho thông tin thống kê và siêu dữ liệu (metadata) của một video TikTok thuộc chủ đề ẩm thực. Các cột bao gồm nhiều thông tin đa dạng như:

- **Thông tin về tác giả:** tên, ID, số liệu thống kê như lượt theo dõi, lượt thích, v.v..
- **Thông tin về video:** ID, mô tả, thời gian tạo, lượt xem, lượt thích, bình luận, v.v..
- **Thông tin kỹ thuật:** thời lượng, độ phân giải, codec, v.v..
- **Thông tin về âm nhạc sử dụng:** tên bài hát, tác giả, v.v..
- **Các tính năng tương tác đặc biệt của TikTok** như *duet*, *stitch*, v.v..
- Và còn nhiều thông tin khác.

4.2.2 Phân tích và xử lý dữ liệu trùng lặp (Duplicate Analysis)

Xác định video: Việc xác định các video trùng lặp dựa trên các cột ID (chứa mã định danh duy nhất) của video trong tập dữ liệu. Các cột này bao gồm: `id`, `video.id`, và `video.videoID`.

Xử lý các video bị thiếu ID: Phân tích cho thấy có **264 hàng** bị thiếu giá trị ở cả 3 cột ID này (chiếm 0.37%). Do không thể xác định định danh duy nhất cho các video này, chúng đã bị loại bỏ khỏi tập dữ liệu. Tập dữ liệu sau khi loại bỏ còn **70996 hàng**.

Kiểm tra trùng lặp: Sử dụng cột `video.id` làm khóa chính để kiểm tra các hàng trùng lặp (giữ lại bản ghi đầu tiên nếu có trùng). Kết quả cho thấy **không có hàng nào bị trùng lặp** (tỷ lệ trùng lặp 0.00%).

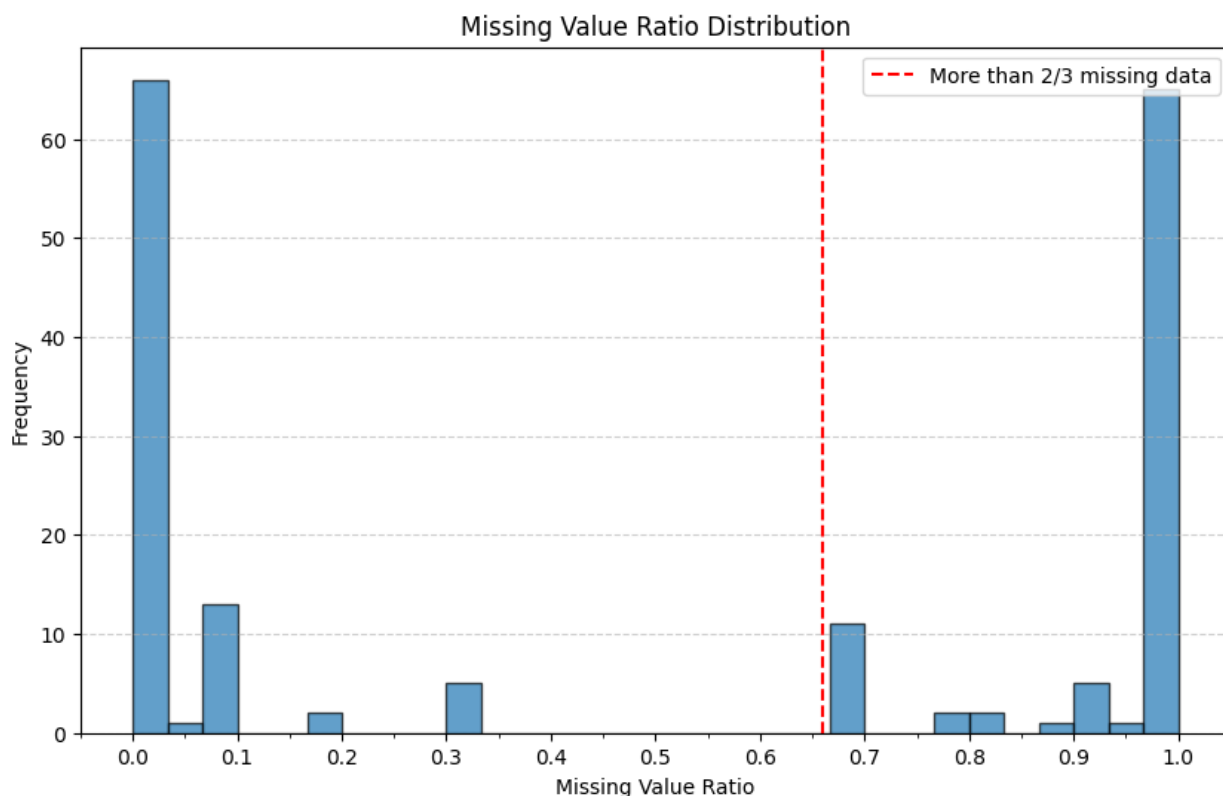
Bảng 2: Thống kê dữ liệu ban đầu

Thông số	Giá trị
Số hàng ban đầu	71.260
Số cột ban đầu	174
Số hàng thiếu ID	264 (0,37%)
Số hàng sau loại bỏ	70.996
Tỷ lệ trùng lặp	0,00%

4.2.3 Phân tích và xử lý giá trị bị thiếu (Missing Value Analysis)

Phân tích tỷ lệ thiếu: Biểu đồ phân bố tỷ lệ thiếu giá trị (Hình 1) cho thấy một số lượng lớn các cột (khoảng 50% tổng số cột) có tỷ lệ thiếu giá trị rất cao, **vượt quá 66.67%** (thiếu hơn 2/3 dữ liệu).

Loại bỏ cột thiếu nhiều: Các cột có tỷ lệ thiếu trên 66.67% được xem là không đủ chất lượng để phân tích và đã bị loại bỏ. Tổng cộng **87 cột** đã bị loại bỏ khỏi tập dữ liệu, giảm số lượng cột từ 174 xuống còn 87.



Hình 1: Biểu đồ phân bố tỷ lệ thiếu giá trị của các cột

4.2.4 Phân tích và chuẩn hóa kiểu dữ liệu

Kiểm tra kiểu dữ liệu: Sau khi loại bỏ các cột thiếu nhiều, kiểu dữ liệu của các cột còn lại được kiểm tra lại.

Chuẩn hóa cột thời gian: Hai cột `collectTime` (thời gian thu thập dữ liệu) và `createTime` (thời gian video được đăng tải) ban đầu có kiểu `object` và chứa giá trị dạng timestamp. Chúng đã được chuyển đổi sang kiểu `datetime` bằng hàm `pd.to_datetime` với đơn vị là giây (`unit='s'`). Sau đó, múi giờ được chuẩn hóa về **GMT+7 (Asia/Ho_Chi_Minh)** để phù hợp với bối cảnh.

cảnh dữ liệu tại Việt Nam.

```

1 # Convert the 'collectTime' and 'createTime' columns
2 # from POSIX to datetime64[ns] data type
3 video_df["collectTime"] = pd.to_datetime(video_df["collectTime"], unit="s")
4 video_df["createTime"] = pd.to_datetime(video_df["createTime"], unit="s")
5
6 # Change the timezone of the 'collectTime' and 'createTime' columns
7 # from UTC to Asia/Ho_Chi_Minh
8 video_df['createTime'] = video_df['createTime'].dt.tz_localize(
9     'UTC').dt.tz_convert("Asia/Ho_Chi_Minh")
10 video_df['collectTime'] = video_df['collectTime'].dt.tz_localize(
11     'UTC').dt.tz_convert("Asia/Ho_Chi_Minh")

```

4.2.5 Phân tích phân bố giá trị

Cột dạng số (Numerical): Phân tích thống kê mô tả (min, Q1, median, Q3, max) cho các cột số. Các nhận xét chính:

- `authorStats.heart` và `authorStats.heartCount` có phân bố giống hệt nhau, cho thấy sự trùng lặp thông tin.
- Các cột `stats.*` và `statsV2.*` chứa thông tin tương tự (lượt xem, thích, bình luận, v.v.).
- Các cột `authorStats.friendCount` và `statsV2.repostCount` chỉ chứa một giá trị duy nhất (0.0), không mang lại nhiều thông tin biến thiên.

Cột không phải dạng số (Non-numerical): Phân tích số lượng giá trị duy nhất và tỷ lệ xuất hiện của từng giá trị. Nhận xét chính:

- Nhiều cột chỉ chứa **1 giá trị duy nhất** (ví dụ: `author.commentSetting` chỉ có giá trị '0.0', `author.ftc` chỉ có 'False', v.v.). Các cột này không hữu ích cho việc phân tích sự khác biệt giữa các video.

4.2.6 Loại bỏ các cột không cần thiết/trùng lặp

Dựa theo các phân tích trước đó, ta tiếp tục loại bỏ các cột không có nhiều ý nghĩa trong quá trình phân tích. Các cột được loại bỏ bao gồm:

- **Cột ID trùng lặp:** Loại bỏ `id` và `video.videoID`, giữ lại `video.id` làm định danh chính cho video.

- **Cột thống kê cũ:** Loại bỏ các cột bắt đầu bằng `stats.*` (ví dụ: `stats.collect-Count`) vì trùng lặp với `statsV2.*`.
- **Cột có một giá trị duy nhất:** Loại bỏ 25 cột như `author.commentSetting`, `collected`, v.v. vì không cung cấp nhiều thông tin để phân tích.
- **Cột liên quan đến ngôn ngữ:** Loại bỏ các cột về ngôn ngữ video (ví dụ: `video.claimInfo.originalLanguageInfo.*`), vì dữ liệu tập trung vào khán giả Việt Nam.
- **Cột ‘heart’ trùng lặp:** Loại bỏ `authorStats.heart` do trùng thông tin với `authorStats.heartCount`.

Tổng cộng, **37 cột được loại bỏ**, giảm số lượng cột từ 87 xuống còn 50. Quá trình này được thực hiện bằng cách sử dụng các hàm phân tích của **Pandas**, như kiểm tra số giá trị duy nhất và so sánh phân phối giữa các cột.

4.2.7 Xử lý dữ liệu thiếu trong các cột còn lại

Đối với 50 cột còn lại, nhóm tiến hành xử lý dữ liệu thiếu như sau:

- **Cột số:** Dữ liệu thiếu được điền bằng **giá trị trung vị** của cột, đảm bảo không làm lệch phân phối dữ liệu.
- **Cột không phải số:** Dữ liệu thiếu được điền bằng **giá trị đặc biệt** “others”, tránh gây nhiễu trong phân tích.

Quá trình này đảm bảo không còn giá trị thiếu trong tập dữ liệu, với thông tin được xác nhận bằng phương thức `info()` của **Pandas**.

4.2.8 Lưu trữ dữ liệu đã xử lý

Kết thúc quá trình tiền xử lý, tập dữ liệu đã được làm sạch, chuẩn hóa, không còn giá trị thiếu và có kích thước **70996 hàng x 50 cột**. Kiểu dữ liệu của các cột đã phù hợp. Tập dữ liệu này được lưu dưới định dạng Parquet vào file `preprocessed_videos.parquet` và sẵn sàng cho bước rút trích đặc trưng.

4.3 Rút trích đặc trưng (Feature Engineering - FE)

Sau khi làm sạch dữ liệu, nhóm tiến hành rút trích các đặc trưng quan trọng để hỗ trợ phân tích nội dung video và xây dựng công cụ viết kịch bản. Các đặc trưng bao gồm hashtag, nội dung video hàng đầu, transcript audio, thông tin món ăn, địa điểm và phân loại video.

4.3.1 Trích xuất hashtag từ mô tả video (desc)

Quy trình: Từ cột `desc` chứa mô tả video, ta sử dụng biểu thức chính quy (thư viện `re` trong Python) để tìm và trích xuất các hashtag (chuỗi bắt đầu bằng ký tự `#`). Các hashtag được chuẩn hóa bằng cách chuyển thành chữ thường và loại bỏ các dấu câu ở cuối (nếu có), đồng thời chuyển thành dạng không dấu (sử dụng thư viện `unidecode`).

Đặc trưng mới:

- `hashtags`: Một cột mới chứa danh sách (list) các hashtag đã được chuẩn hóa cho mỗi video.
- `hashtag_count`: Một cột mới chứa số lượng hashtag có trong mỗi video.

Phân tích: Bảng 3 cho thấy có đến **99,51%** video (70.650/70.996) được đăng tải chứa ít nhất một hashtag, với trung bình **6-7 hashtag mỗi video**. Bảng 4 cho thấy các hashtag phổ biến nhất bao gồm `ancungtiktok`, `learnontiktok`, `reviewanngon`, `xuhuong`, `mukbang`, phù hợp với chủ đề ẩm thực và xu hướng chung trên TikTok.

Bảng 3: Kết quả thống kê về hashtag

Total number of videos: 70996
Videos with hashtags: 70650
Percentage with hashtags: 99.51%
Average hashtags per video: 6.85

Bảng 4: Danh sách các hashtag phổ biến

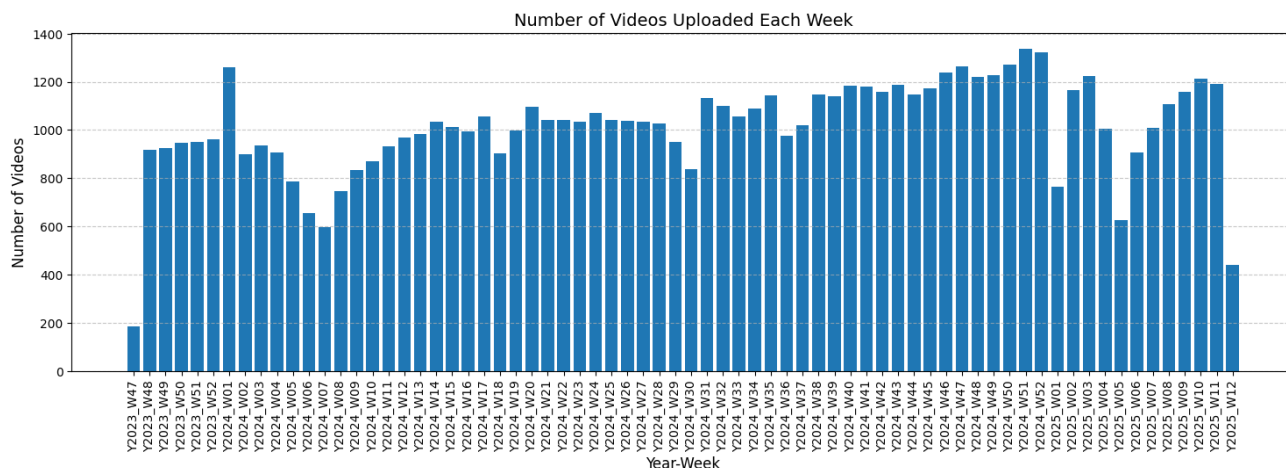
Hashtag	Số lần xuất hiện
<code>ancungtiktok</code>	50231
<code>learnontiktok</code>	30866
<code>reviewanngon</code>	22232
<code>xuhuong</code>	20930
<code>mukbang</code>	9876

4.3.2 Trích xuất đặc trưng thời gian từ `createTime`

Quy trình: Từ cột `createTime` (đã có kiểu `datetime`), ta trích xuất thông tin về **năm** và **số thứ tự của tuần trong năm**.

Đặc trưng mới:

- `createTime_year`: Năm video được đăng.
- `createTime_week`: Số thứ tự tuần trong năm video được đăng (theo chuẩn ISO).
- `year_week`: Một cột kết hợp năm và tuần (ví dụ: "Y2024_W08") để dễ dàng nhóm dữ liệu theo tuần.



Hình 2: Biểu đồ phân bố số lượng video được đăng tải theo các tuần trong năm

4.3.3 Chọn top video hàng tuần để phân tích nội dung

Mục tiêu: Để tập trung phân tích sâu vào nội dung (transcript, món ăn, địa điểm) của các video tiềm năng nhất, nhóm quyết định chỉ xử lý một tập con các video nổi bật.

Tiêu chí lựa chọn: Nhóm đề xuất một **độ đo đánh giá** cho mỗi video dựa trên các chỉ số thống kê có sẵn trong tập dữ liệu, với **trọng tâm là số lượt xem** (`statsV2.playCount`). Cụ thể, độ đo này là **tổng có trọng số của lượt xem và các chỉ số tương tác khác** (như lượt thích, bình luận, chia sẻ) **sau khi được chuẩn hóa** để đảm bảo tính công bằng giữa các chỉ số. Mức độ đóng góp của mỗi chỉ số thống kê vào độ đo đánh giá như sau:

- `statsV2.playCount` (lượt xem): 40%.
- `statsV2.diggCount` (lượt thích): 25%.
- `statsV2.shareCount` (lượt chia sẻ): 15%.
- `statsV2.commentCount` (lượt bình luận): 10%.
- Tỷ lệ tương tác: 10%.

Quy trình: Tập dữ liệu bao gồm dữ liệu từ 70 tuần (xem Hình 2). Trong mỗi tuần, ta chọn ra **20% số video có điểm số đánh giá cao nhất**. Tuy nhiên, để đảm bảo đủ dữ liệu cho phân tích, nếu 20% số video ít hơn 100, thì ta sẽ lấy **tối thiểu 100 video** có điểm số cao nhất trong tuần đó.

Kết quả: Tạo ra một tập dữ liệu con chứa khoảng **14252 video** nổi bật nhất theo từng tuần, sẵn sàng cho các bước trích xuất nội dung tốn kém hơn về mặt tính toán (như gọi API).

4.3.4 Trích xuất nội dung audio thành văn bản (Audio Transcription)

Quy trình này được thực hiện trong file `03_FE_01_01-Transcribe_audio_colab.ipynb` và bao gồm các bước:

1. **Tái tạo URL:** Xây dựng lại URL của video TikTok từ `author.uniqueId` và `video.id` theo định dạng: `https://www.tiktok.com/@author_id/video/video_id`.
2. **Tải audio:** Sử dụng thư viện `yt-dlp` kết hợp với `FFmpeg` để tải về **chỉ phần âm thanh** từ URL video TikTok. Audio được chuyển đổi sang định dạng WAV và lưu vào thư mục `AUDIO_FOLDER`.
3. **Chuyển đổi audio thành văn bản:** Sử dụng Gemini API (mô hình `gemini-2.0-flash`) để chuyển audio thành transcript. Dữ liệu âm thanh (file `.wav`) được đọc dưới dạng bytes và gửi kèm một **prompt** yêu cầu:
 - Chuyển đổi **giọng nói tiếng Việt** trong audio thành văn bản (transcript).
 - Rút ra 3 ý chính (takeaways) từ nội dung.
 - Kiểm tra sự tồn tại của câu kêu gọi hành động (Call to Action - CTA).
 - Kiểm tra sự tồn tại của yếu tố gây tò mò (Curiosity Gap).
 - Trả về kết quả dưới dạng JSON với các trường: `transcript`, `takeaways`, `has_call_to_action`, `has_curiosity_gap`. Nếu không có giọng nói, trả về `None`.
4. **Lưu kết quả:** Kết quả JSON trả về từ API được lưu vào file `{video_id}.json` trong thư mục `TRANSCRIPT_FOLDER`. Một danh sách các `video_id` đã xử lý được lưu lại trong file `transcribed_video_ids.txt` để tránh xử lý lại.
5. **Quản lý giới hạn API:** Để tránh vượt quá giới hạn lượt gọi của mỗi API, nhóm **sử dụng luân phiên** nhiều API key khác nhau. Mỗi API key được sử dụng cho **14 yêu cầu liên tiếp**, sau đó chuyển sang API key tiếp theo.

4.3.5 Trích xuất thông tin món ăn và địa điểm

Quy trình này được thực hiện trong file `03_FE_02_01-Extract_food_location_colab.ipynb` và dựa trên kết quả từ bước trước:

1. **Dữ liệu đầu vào:** Sử dụng cột `desc` (mô tả gốc) và cột `transcript` (văn bản từ audio) của các video đã được xử lý ở Mục 4.3.4 để trích xuất thông tin về món ăn, thành phố, và quận/huyện.
2. **Gọi Gemini API:** Tiếp tục sử dụng mô hình `gemini-2.0-flash`. Một **prompt** mới được thiết kế để yêu cầu mô hình phân tích nội dung `desc` và `transcript`, sau đó trích xuất:
 - Danh sách các món ăn được đề cập (`foods`: list of strings).
 - Tên thành phố (`city`: string).
 - Tên quận/huyện (`district`: string).
 - Prompt cũng yêu cầu mô hình trả về `None` cho các trường nếu video không thuộc chủ đề ẩm thực.
 - Kết quả phải trả về đúng **định dạng JSON**.
3. **Lưu kết quả:** Kết quả JSON chứa thông tin món ăn và địa điểm được lưu vào file `{video_id}.json` trong thư mục `new_food_location`. Tương tự, một danh sách các video đã xử lý được theo dõi trong file `preprocessed_video_ids.txt`.
4. **Quản lý giới hạn API:** Nhóm cũng sử dụng nhiều API key khác nhau để **tránh vượt quá giới hạn lượt gọi**. Mỗi API key được sử dụng cho **14 yêu cầu liên tiếp**, sau đó chuyển sang API key tiếp theo.

4.3.6 Trích xuất đặc trưng về thể loại của video

Và đặc trưng quan trọng cuối cùng mà ta cần rút trích là **thể loại của video**. Đây là các đặc trưng được sử dụng trực tiếp trong quá trình xây dựng công cụ hỗ trợ viết kịch bản cho video TikTok. Quy trình này sẽ được trình bày chi tiết trong Chương 5.

5 CHƯƠNG 5: CÔNG CỤ HỖ TRỢ VIẾT KỊCH BẢN CHO VIDEO TIKTOK

Quy trình tạo kịch bản video TikTok được xây dựng theo hướng **few-shot prompting** trên nền tảng mô hình **Gemini**. Hệ thống phân tích mô tả đầu vào từ người dùng để đề xuất bộ lọc phù hợp, sau đó lựa chọn một nhóm video tương đồng từ tập dữ liệu đã annotate. Các video mẫu này được đưa trực tiếp vào *prompt* dưới dạng ví dụ minh họa để mô hình sinh ra kịch bản thô dạng đoạn văn. Kịch bản sau đó được đưa vào một bước xử lý thứ hai để tự động phân đoạn và gắn nhãn nội dung, tạo thành bản kịch bản hoàn chỉnh có cấu trúc.

5.1 Trích xuất đặc trưng nội dung

Tên trường	Mô tả
categories	Thể loại chính của video. Bắt buộc. Nếu không liên quan đến ẩm thực, chỉ trả về trường này với giá trị “Không liên quan ẩm thực”.
structure_style	Các kỹ thuật trình bày nội dung như kể chuyện, mô tả đặc điểm, hướng dẫn, v.v..
hook_type	Cách mở đầu video nhằm thu hút người xem (ví dụ: gây tò mò, giật tít vào thẳng vấn đề, v.v.).
tone_of_voice	Giọng điệu và cảm xúc của người nói, như hài hước, chân thành, thân thiện, v.v..
pacing	Nhịp độ triển khai video: nhanh, chậm hoặc thay đổi.
has_cta, cta_type	Cho biết video có kêu gọi hành động không và nếu có thì đó là loại nào (ví dụ: follow, bình luận, chia sẻ, ghé quán, v.v.).
content_style	Phong cách tổng thể của nội dung: Gen Z, chuyên nghiệp, truyền thống, v.v..
audience_target	Nhóm khán giả mục tiêu mà video hướng tới (ví dụ: học sinh, dân văn phòng, người ăn chay, v.v.).

Bảng 5: Các trường được trích xuất từ nội dung video

Để phục vụ cho quá trình lọc video mẫu và sinh kịch bản, nhóm đã tiến hành trích xuất các đặc trưng nội dung từ *mô tả* và *transcript* của từng video TikTok. Việc trích xuất được thực hiện bằng mô hình ngôn ngữ lớn (LLM) với **prompt định hướng chi tiết**, nhằm gán nhãn cho từng trường nội dung theo một **schema chuẩn hoá**.

Trong giai đoạn đầu, nhóm không giới hạn tập nhãn cố định mà cho phép Gemini được **tự do gán nhãn** cho các trường nội dung (*free-form annotation*), không áp đặt trước số lượng hay tên nhãn cụ thể cho mỗi trường. Việc này cho phép thu thập linh hoạt các kiểu biểu hiện phong phú trong nội dung video thực tế. Sau đó, hệ thống tiến hành **thống kê các nhãn có số lượng video cao nhất ở mỗi trường** và thực hiện một vài bước **tinh chỉnh thủ công**, từ đó xác định một tập nhãn phổ biến và ổn định để sử dụng nhất quán trong toàn bộ *pipeline*.

Để đảm bảo sự đồng nhất trong định dạng và nội dung đầu ra khi *prompt* với Gemini, nhóm đã sử dụng một **response schema JSON** làm chuẩn. Schema này quy định rõ tên các trường được trích xuất, kiểu dữ liệu, danh sách các nhãn hợp lệ (nếu có), cũng như yêu cầu về tính bắt buộc của từng trường. Nhờ đó, mô hình có thể sinh đầu ra **đúng cấu trúc và dễ dàng xử lý** tiếp trong các bước lọc và sinh kịch bản. Một số trường nội dung quan trọng được trình bày trong Bảng 5.

5.2 Lọc video tương đồng theo mô tả người dùng

Sau khi người dùng nhập mô tả tự nhiên về nội dung video muốn tạo, hệ thống sẽ tiến hành phân tích mô tả để gán nhãn cho các trường nội dung đã chuẩn hoá trước đó. Quá trình này được thực hiện bằng mô hình Gemini, sử dụng *prompt* riêng kèm theo **response schema** định nghĩa các trường như *categories*, *structure_style*, *tone_of_voice*, v.v.. Ngoài ra, trong quá trình xử lý mô tả đầu vào, hệ thống cũng thực hiện **trích xuất thời lượng mong muốn** nếu người dùng có đề cập đến (ví dụ: “video khoảng 30 giây”, “tầm 1 phút rưỡi”, v.v.). Thời lượng này sẽ được sử dụng trong bước sinh kịch bản để kiểm soát độ dài hợp lý của nội dung đầu ra.

Các nhãn nội dung được sinh từ mô tả sẽ được sử dụng như điều kiện lọc để chọn ra các video có nội dung tương đồng trong tập dữ liệu đã annotate. Việc lọc được thực hiện bởi hàm *filter_by_multiple_labels_unified*, theo cơ chế:

1. Ban đầu áp dụng toàn bộ các điều kiện lọc.
2. Nếu không đủ số lượng video (mặc định là 20), hệ thống sẽ dần dần **nới lỏng điều kiện** theo thứ tự ưu tiên: nới nhãn trong từng trường, sau đó loại bỏ cả trường (trừ *categories*).
3. Quá trình dừng lại khi đạt đủ số lượng video tối thiểu hoặc khi đã nới tối đa có thể.

Nhờ cơ chế *lọc mềm*, hệ thống có thể tìm được một tập video mẫu có mức độ tương đồng cao với ý tưởng đầu vào của người dùng, ngay cả khi mô tả ban đầu quá cụ thể hoặc hiếm gặp trong dữ liệu.

Nếu kết quả gán nhãn ban đầu xác định rằng video không liên quan đến ẩm thực (giá trị “Không liên quan ẩm thực” trong trường *categories*), quy trình sẽ dừng và yêu cầu người dùng nhập lại mô tả phù hợp.

5.3 Sinh kịch bản thô từ video mẫu

Sau khi lọc được tập video có nội dung tương đồng với mô tả đầu vào, hệ thống tiến hành bước sinh kịch bản thô bằng phương pháp **few-shot prompting**. Cụ thể, hệ thống chọn tối đa **20 video mẫu** làm ví dụ đầu vào để hướng dẫn mô hình sinh ra một kịch bản mới phù hợp với bối cảnh yêu cầu. Việc sử dụng số lượng ví dụ lớn (20-shot) giúp mô hình có đủ dữ liệu tham chiếu để tái tạo phong cách nội dung phổ biến trong các video TikTok ẩm thực. Tập video mẫu được chọn dựa trên một công thức chấm điểm đơn giản, kết hợp giữa **mức độ phổ biến** và **độ gần thời gian** của video. Cụ thể, điểm số được tính theo công thức:

$$\text{Score} = \frac{\text{Lượt xem}}{(\text{Số ngày kể từ ngày đăng} + 1)^\alpha}, \quad \alpha = 0.7$$

Công thức này **ưu tiên các video có lượt xem cao**, đồng thời điều chỉnh theo thời gian để **tăng trọng số cho những nội dung mới** được đăng gần đây. Từ danh sách được sắp xếp theo điểm số giảm dần, hệ thống chọn ra 20 video đầu tiên để làm ví dụ minh họa trong *prompt*.

Để giới hạn độ dài hợp lý cho kịch bản đầu ra, hệ thống ước lượng **word count mục tiêu** dựa trên hai nguồn:

- Nếu người dùng có đề cập đến **thời lượng mong muốn** trong phần mô tả đầu vào, hệ thống sẽ trích xuất con số này và chuyển đổi thành số từ (*word count*) tương ứng.
- Nếu không có thông tin rõ ràng từ mô tả, hệ thống sử dụng **thời lượng trung bình của các video mẫu** đã lọc và nhân với **tốc độ nói trung bình** để tính ra số từ tối ưu.

Việc kiểm soát độ dài bằng word count giúp kịch bản sinh ra phù hợp với nhịp độ thực tế của video TikTok, tránh việc mô hình tạo nội dung quá dài hoặc quá ngắn so với kỳ vọng. Đây là một điểm then chốt trong pipeline, đảm bảo đầu ra có tính khả dụng cao và nhất quán với trải nghiệm người dùng.

Từ phần mô tả gốc và các ví dụ đã chọn, mô hình sẽ sinh ra một đoạn văn liền mạch, được xem là **kịch bản thô ban đầu**. Đoạn này chưa có cấu trúc rõ ràng, không phân chia các phần nội dung cụ thể như mở bài, nội dung chính hay CTA, nhưng vẫn đảm bảo đúng tone, nhịp điệu và đặc điểm của các video TikTok ẩm thực phổ biến.

5.4 Định dạng lại kịch bản có cấu trúc

Sau khi sinh được kịch bản thô từ mô tả và các video mẫu tương đồng, hệ thống tiếp tục định dạng lại nội dung đầu ra thành một cấu trúc rõ ràng hơn, nhằm phục vụ cho các bước hậu

kỳ và sản xuất video sau này.

Bước này được thực hiện bằng mô hình Gemini cùng với một **prompt chuyên biệt**, trong đó yêu cầu mô hình:

- Chuyển kịch bản dạng *plain* thành một JSON hợp lệ theo schema đã định nghĩa.
- Giữ nguyên lời thoại gốc, không viết lại nội dung.
- Chia nhỏ theo từng phần như mở đầu, mô tả món, cảm nhận, CTA, v.v..
- Dựa vào thời lượng ước tính của video, các đoạn mô tả mẫu và danh sách hashtag phổ biến để sinh phần mô tả video.

Mô hình được yêu cầu tạo ra một **kịch bản có chú thích rõ ràng cho từng phần nội dung**, dựa trên một *schema chuẩn hoá* gồm 5 trường chính:

- **video_description**: Mô tả video dựa trên nội dung kịch bản, có độ dài và số lượng hashtag điều chỉnh theo chuẩn trung bình từ các video mẫu.
- **duration**: Thời lượng tổng thể của video, định dạng “*x phút y giây*”.
- **setting**: Bối cảnh hoặc địa điểm quay video.
- **characters**: Nhân vật xuất hiện trong video.
- **main_content**: Danh sách các phần nội dung chính, mỗi phần bao gồm:
 - **title**: Tên đoạn, ví dụ “Giới thiệu quán ăn”.
 - **dialogue**: Lời thoại gốc từ kịch bản plain.
 - **visual_description**: Mô tả cảnh quay.
 - **time_range**: Mốc thời gian tương ứng trong video.

Do các mốc thời gian trong kịch bản do Gemini tạo ra không đảm bảo tính chính xác, hệ thống sẽ tiến hành tính toán lại **time_range** cho từng đoạn trong kịch bản **main_content**. Việc này được thực hiện dựa trên:

- **Số từ trong đoạn thoại (dialogue)**.
- **Tốc độ nói trung bình** được lấy từ các video mẫu (tính bằng từ/giây).

Cụ thể, thời lượng ước tính của mỗi đoạn được tính bằng công thức:

$$\text{Thời lượng (giây)} = \frac{\text{Số từ}}{\text{Tốc độ nói trung bình}}$$

Sau đó, hệ thống cộng dồn để tạo các `time_range` liên tiếp từ đầu tới cuối video. Điều này giúp đảm bảo rằng toàn bộ nội dung kịch bản được phân bố đều và hợp lý theo mốc thời gian, tạo điều kiện thuận lợi cho quá trình dựng video tự động hoặc chỉnh sửa thủ công.

5.5 Triển khai hệ thống trên giao diện web

Hệ thống được triển khai dưới dạng một ứng dụng web bằng **Streamlit**, cho phép người dùng nhập mô tả ý tưởng và nhận kết quả là kịch bản video TikTok được tạo tự động. Kịch bản sau khi sinh được hiển thị dưới dạng từng đoạn nội dung có cấu trúc rõ ràng và người dùng có thể chỉnh sửa trực tiếp nội dung từng đoạn ngay trên giao diện. Sau khi hoàn thiện, kịch bản có thể được sao chép hoặc tải về dưới định dạng **Markdown**, thuận tiện cho quá trình biên tập và sử dụng về sau.

Phân tích hiệu suất vi...
Phân tích nội dung
Phân tích xu hướng
Tổng quan
Phân tích món ăn và ...

Các công cụ hỗ trợ
Tổng quan
Nghiên cứu chủ đề
Đề xuất quay video
Viết kịch bản
Tối ưu kênh TikTok
View less

Chọn mô hình AI:
gemini-2.0-flash

Tạo kịch bản TikTok

Nhập mô tả video

Mô tả chi tiết về video TikTok bạn muốn tạo:

Làm video TikTok hướng dẫn nấu mì tôm trứng siêu nhanh, siêu dễ cho sinh viên. Cần nhấn mạnh vào sự tiện lợi và hướng dẫn từng bước...

Một số chi tiết có thể gợi ý cho hệ thống:

- Món ăn và mục tiêu video (review, nấu, chia sẻ...)
- Cách triển khai (kể chuyện, hướng dẫn, review...)
- Cách mở đầu, giọng điệu, tốc độ video
- Có CTA gì không (comment, chia sẻ, ghé quán...)
- Ai là người xem chính (học sinh, dân văn phòng, nội trợ...)

Ví dụ: Tôi muốn làm video quảng bá cho sản phẩm bánh tráng chấm phô mai của nhà tôi, cách nói chuyện gần gũi, có hướng dẫn cách ăn, giọng điệu từ tốn.

Tạo kịch bản

Hình 3: Giao diện chính của công cụ hỗ trợ viết kịch bản cho video TikTok

6 CHƯƠNG 6: DASHBOARD PHÂN TÍCH USER

Dashboard này nhằm khám phá và phân tích dữ liệu về các nhà sáng tạo nội dung trên TikTok. Phần này cung cấp cái nhìn tổng quan về bộ dữ liệu thu thập được.

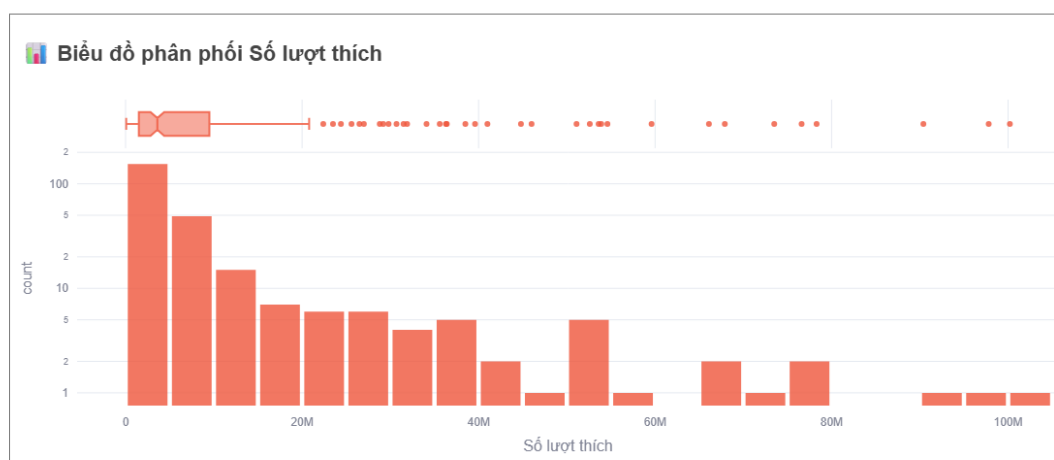
- Tổng số TikToker: 264 người
- Số người theo dõi trung bình: 336,473
- Số lượt thích trung bình: 10,214,598
- Số lượng video trung bình: 550 video

Dựa trên các con số này, nếu thu thập toàn bộ video của các TikToker, bộ dữ liệu sẽ bao gồm khoảng $264 \times 550 = 145,200$ video. Đây là một khối lượng dữ liệu đáng kể, đủ để xác định xu hướng ngành, đề xuất chiến lược nội dung, cũng như so sánh hiệu suất giữa các nhóm TikToker. Tuy nhiên, do hạn chế về thời gian và năng lực xử lý, phân tích trong báo cáo này sẽ tập trung vào các video mới và có nội dung liên quan trực tiếp đến chủ đề ẩm thực.

Nhận xét: Số lượt thích trung bình và số người theo dõi khá cao, đặt ra câu hỏi liệu dữ liệu có bị mất cân bằng hay không. Nếu có, cần áp dụng các phương pháp phân tích cẩn thận để đảm bảo kết quả thuyết phục.

6.1 Phân phối của dữ liệu

6.1.1 Phân Phối Số Lượt Thích

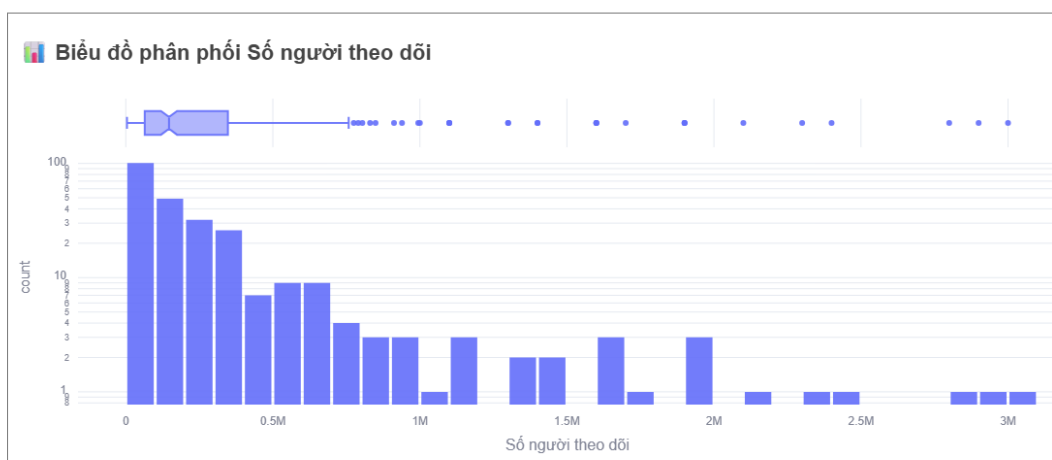


Hình 4: Phân Phối Số Lượt Thích

Phân phối số lượt thích (Hình 4) có dạng lệch phải, với các đặc điểm:

- Hầu hết video có số lượt thích thấp, nhưng giá trị trung bình (hơn 10 triệu) bị kéo lên bởi một số video có lượt thích cực cao.
- Khoảng tứ phân vị (IQR): từ 1.5 triệu (Q1) đến 9.5 triệu (Q3), cho thấy sự phân tán lớn.
- Giá trị tối đa: 100.2 triệu, nhấn mạnh sự hiện diện của các giá trị ngoại lai ảnh hưởng mạnh đến trung bình.

6.1.2 Phân Phối Số Người Theo Dõi



Hình 5: Phân phối số người theo dõi

Phân phối số người theo dõi (Hình 5) có tính bất đối xứng cao và lệch phải:

- Phần lớn tài khoản có số người theo dõi thấp, với đuôi phải kéo dài đến các giá trị lớn.
- Trung bình: ~336,000, cao hơn nhiều so với trung vị (~147,000), do ảnh hưởng của một số tài khoản rất nổi tiếng.
- Giá trị tối đa: 3 triệu, cho thấy sự chênh lệch lớn về mức độ nổi tiếng giữa các TikToker.

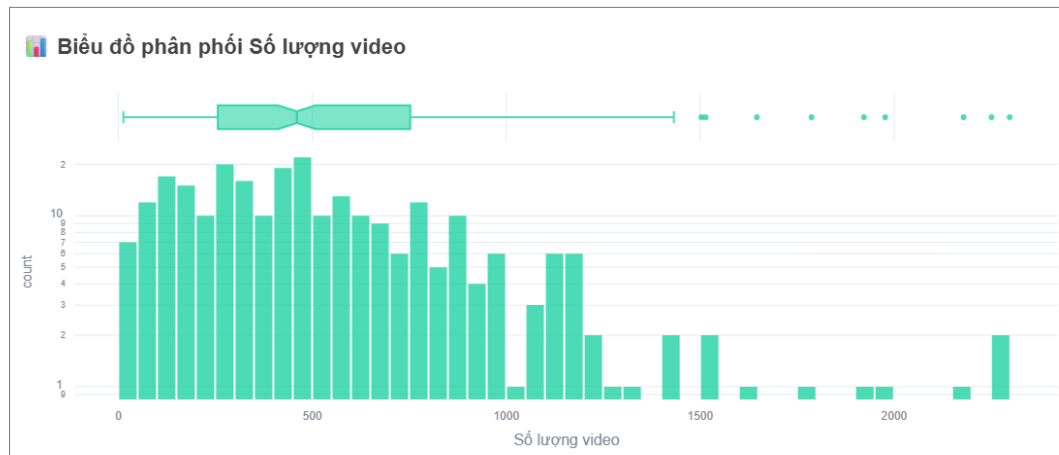
Nhận xét: Phân phối này tương tự phân phối mũ hoặc lũy thừa, nơi số người theo dõi tăng theo cấp số nhân ở một số ít tài khoản.

6.1.3 Phân Phối Số Lượng Video

Phân phối số lượng video (Hình 6) có dạng lệch phải, với các đặc điểm:

- Trung vị: 460 video.
- Khoảng tứ phân vị (IQR): từ 256 đến 752 video.

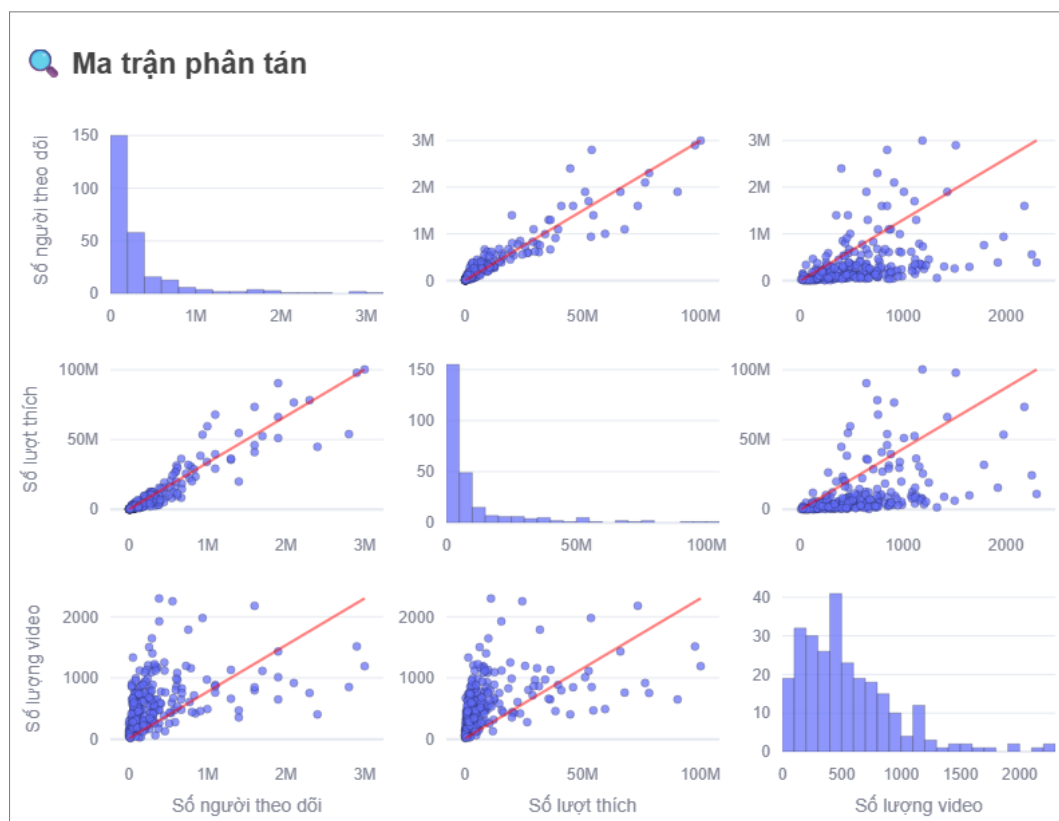
- Trung bình: 549.64, cao hơn trung vị, do một số TikToker có số lượng video rất lớn.
- Độ lệch chuẩn: 409.13, thể hiện sự khác biệt đáng kể về số lượng video.
- Giá trị tối đa: 2,298 video, cho thấy có những người dùng rất tích cực.



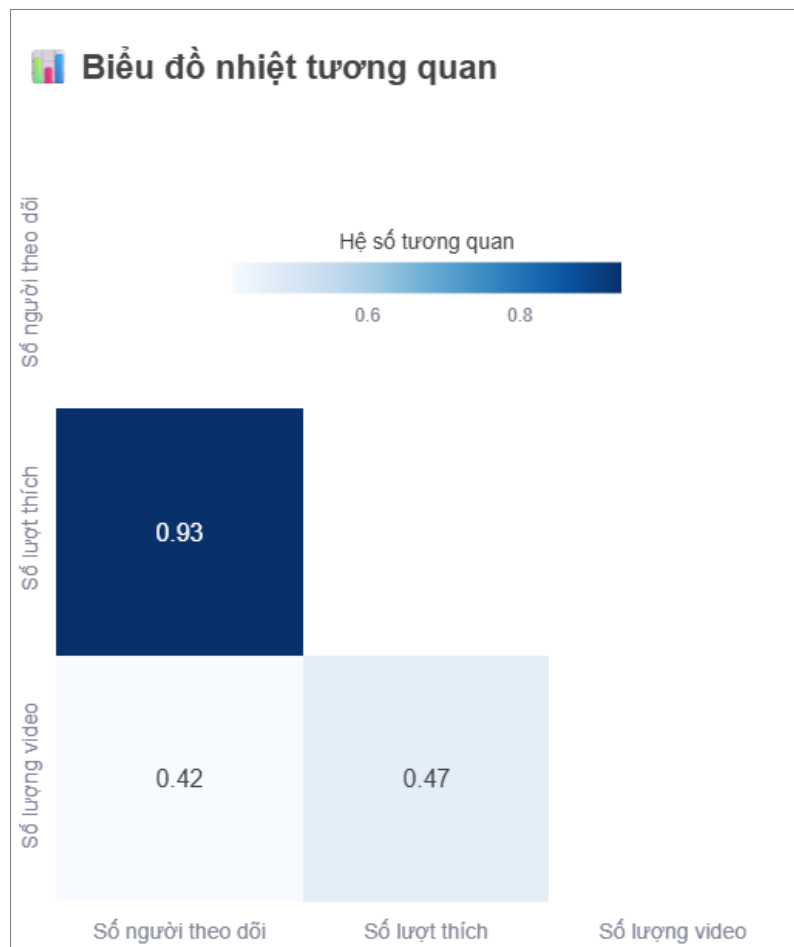
Hình 6: Phân phối số lượng video

6.2 Phân Tích Tương Quan

Cùng quan sát hai biểu đồ sau:



Hình 7: Ma Trận Phân Tán



Hình 8: Biểu Đồ Nhiệt Tương Quan

Mức độ tương quan giữa các cặp biến khác nhau đáng kể. Có thể thấy một tương quan mạnh mẽ giữa ‘Số người theo dõi’ và ‘Số lượt thích’, cùng với tương quan vừa phải giữa ‘Số người theo dõi’ và ‘Số lượng video’, cũng như ‘Số lượt thích’ và ‘Số lượng video’.

- **‘Số người theo dõi’ và ‘Số lượt thích’:** Mối tương quan giữa ‘Số người theo dõi’ và ‘Số lượt thích’ là rất mạnh (0.933). Điều này cho thấy mối quan hệ dương tính: khi một TikToker có nhiều người theo dõi, các video của họ cũng thường nhận được nhiều lượt thích hơn. Điều này có thể là do những người theo dõi thường xuyên xem và tương tác với nội dung của tài khoản, hoặc do các thuật toán ưu tiên hiển thị nội dung của các tài khoản có lượng theo dõi lớn, dẫn đến tăng lượt thích.
- **‘Số người theo dõi’ và ‘Số lượng video’:** Mối tương quan giữa ‘Số người theo dõi’ và ‘Số lượng video’ là trung bình (0.422). Tương quan dương cho thấy, nhìn chung, những TikToker đăng nhiều video hơn có xu hướng có lượng người theo dõi lớn hơn. Điều này có thể là do việc đăng tải thường xuyên giúp tăng khả năng hiển thị trên nền tảng và thu hút thêm người xem. Tuy nhiên, mối tương quan này không quá mạnh, có thể có những

TikToker với số lượng video hạn chế nhưng vẫn thu hút được lượng lớn người theo dõi nhờ chất lượng nội dung.

- **‘Số lượt thích’ và ‘Số lượng video’:** Mối tương quan giữa ‘Số lượt thích’ và ‘Số lượng video’ cũng là trung bình (0.473). Mối tương quan dương này ám chỉ những TikToker đăng nhiều video có xu hướng nhận được nhiều lượt thích hơn. Điều này có thể phản ánh sự hiện diện thường xuyên hơn trên nền tảng dẫn đến việc video có cơ hội hiển thị nhiều hơn và được nhiều lượt thích hơn.

Kết Luận và Gợi Ý: Người sáng tạo nội dung TikTok nên tập trung vào việc xây dựng lượng người theo dõi chất lượng, bởi vì nó có liên quan mật thiết với tương tác (lượt thích) trên các video. Đồng thời, việc đăng tải video thường xuyên, kết hợp với việc nâng cao chất lượng nội dung, có thể là một chiến lược hiệu quả để gia tăng cả lượng người theo dõi và lượt tương tác. Để đạt được kết quả tốt nhất, nhà sáng tạo nên tập trung vào việc tạo ra những video chất lượng cao, được đầu tư kỹ lưỡng về nội dung, hình ảnh và âm thanh để thu hút người xem.

6.3 Phân Tích Mức Độ Tương Tác Theo Nhóm Người Theo Dõi

Như vậy, qua phân tích tổng quát, chúng ta có thể thấy dữ liệu được thu thập có phân phối bất đối xứng và xu hướng lệch phải. Do đó để các phân tích có tính thuyết phục, chúng ta sẽ tiến hành phân chia các TikToker theo nhóm dựa trên lượng người theo dõi, để có thể khám phá ra các đặc điểm, tính chất để các kết luận đưa ra được tăng tính thuyết phục.

6.3.1 Phân Chia Nhóm Người Theo Dõi

- **Nhóm người theo dõi thấp:** Những người dùng có số người theo dõi thấp hơn 85,627. Những người dùng này có thể là những người mới bắt đầu hoặc chưa có nhiều nội dung nổi bật.
- **Nhóm người theo dõi trung bình:** Những người dùng có số người theo dõi nằm trong khoảng 85,627 - 276,358. Những người dùng này có thể là những người mới nổi hoặc có ảnh hưởng vừa phải trên TikTok.
- **Nhóm người theo dõi cao:** Những người dùng có số người theo dõi cao hơn 276,358. Những người dùng này có thể là những người nổi tiếng hoặc có ảnh hưởng lớn trên TikTok.

6.3.2 Bảng So Sánh Mức Độ Tương Tác

Chỉ số	Thấp	Trung bình	Cao
Số mẫu	87	87	90
Số người theo dõi	45,665	160,687	787,512
Số lượt xem trên mỗi video	195,285	371,054	729,571
Số bình luận trên mỗi video	99	176	238
Số lượt thích trên mỗi video	5,362	11,633	35,050
Số lượt chia sẻ trên mỗi video	556	1,039	960
Số lượt lưu trên mỗi video	766	1,293	1,772
Số video mỗi tuần	4.4 ± 4.0	4.5 ± 2.4	4.7 ± 2.4
Số hashtag trên mỗi video	7.5 ± 2.5	6.8 ± 2.5	5.7 ± 2.1
Thời lượng video (giây)	59.0 ± 36.3	74.5 ± 47.1	103.6 ± 59.6

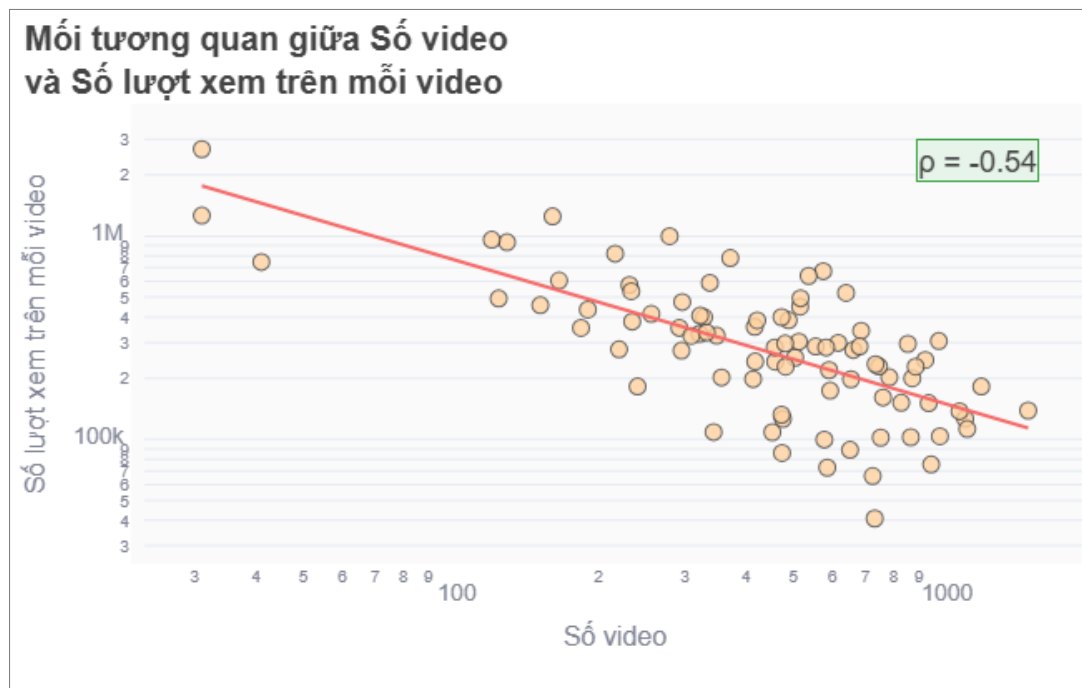
Bảng 6: So sánh mức độ tương tác của TikToker theo số người theo dõi

Nhận xét:

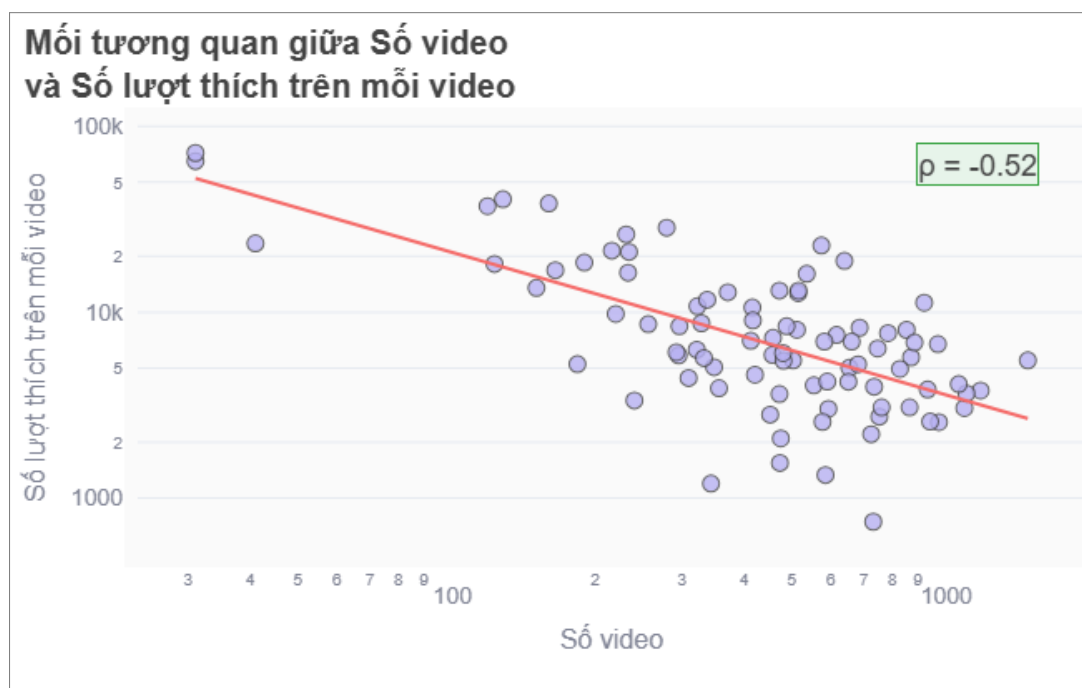
- **Tương tác tăng theo số người theo dõi:** TikToker có số người theo dõi cao (787,512) đạt mức tương tác vượt trội ở hầu hết các chỉ số (lượt xem, bình luận, lượt thích, lưu) so với nhóm trung bình (160,687) và thấp (45,665). Riêng số lượt chia sẻ của nhóm cao (960) thấp hơn nhóm trung bình (1,039), có thể do đặc điểm nội dung hoặc đối tượng khán giả.
- **Tần suất đăng video:** Số video mỗi tuần tăng nhẹ từ nhóm thấp (4.4) đến nhóm cao (4.7), với độ biến thiên giảm (từ ± 4.0 xuống ± 2.4), cho thấy nhóm có nhiều người theo dõi hơn có xu hướng duy trì lịch đăng ổn định hơn.
- **Số hashtag:** Nhóm thấp sử dụng nhiều hashtag hơn (7.5) so với nhóm trung bình (6.8) và cao (5.7), có thể do nỗ lực tăng khả năng hiển thị. Nhóm cao ít sử dụng hashtag hơn, có thể do đã có lượng khán giả sẵn có và ít phụ thuộc vào khám phá qua hashtag.
- **Thời lượng video:** Thời lượng video tăng rõ rệt từ nhóm thấp (59.0 giây) đến nhóm cao (103.6 giây), với độ biến thiên cũng tăng (từ ± 36.3 đến ± 59.6). Điều này cho thấy nhóm có nhiều người theo dõi hơn thường sản xuất video dài hơn và đa dạng hơn về độ dài.

6.4 Ảnh Hưởng của Số Lượng Video Đến Tương Tác

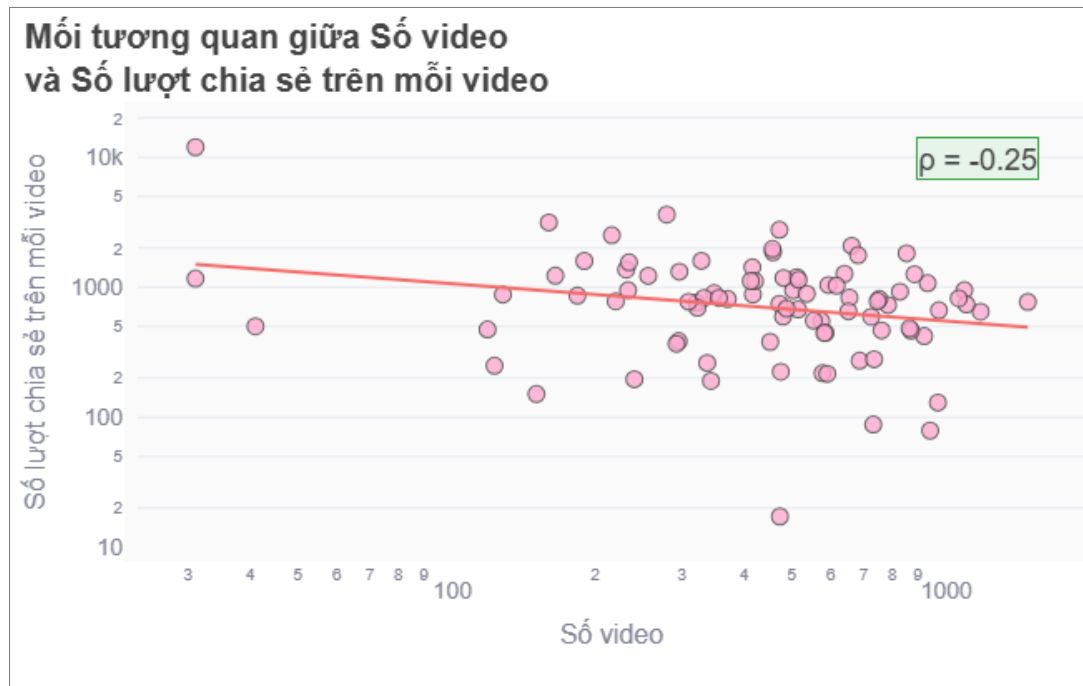
Câu hỏi: Với cùng một ngưỡng số người theo dõi nhất định, liệu việc tăng số lượng video có thực sự đem lại nhiều lượt xem và lượt tương tác hơn trên mỗi video, hay chỉ đơn thuần tạo ra nhiều lượt xem và tương tác tổng?



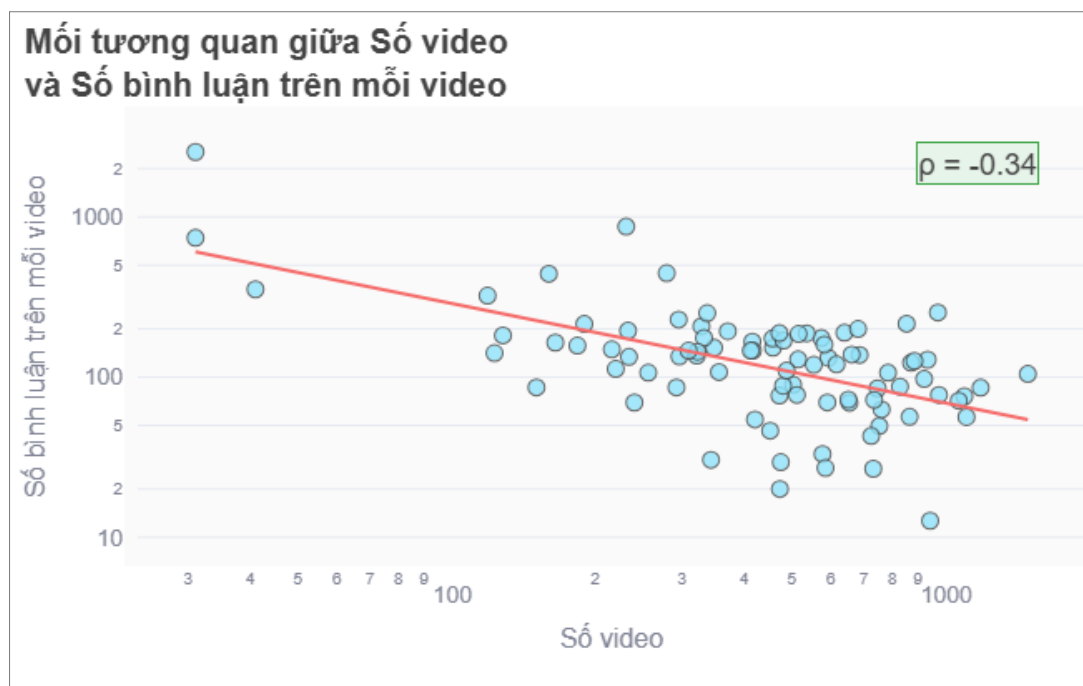
Hình 9: Mối tương quan giữa số Video và số Lượt xem ở nhóm Tiktoker có lượng người theo dõi trung bình



Hình 10: Mối tương quan giữa số Video và số Lượt thích ở nhóm Tiktoker có lượng người theo dõi trung bình



Hình 11: Mối tương quan giữa số Video và số Lượt chia sẻ ở nhóm Tiktoker có lượng người theo dõi trung bình



Hình 12: Mối tương quan giữa số Video và số Lượt bình luận ở nhóm Tiktoker có lượng người theo dõi trung bình

Qua phân tích mối tương quan giữa Tổng số video đăng tải và các chỉ số Lượt xem/Lượt thích/Lượt bình luận/Lượt chia sẻ trung bình trên mỗi video, một xu hướng nhất quán đã được quan sát và có thể tổng quát hóa cho cả ba nhóm người dùng (Thấp, Trung bình, Cao về số người theo dõi):

Mối tương quan nghịch giữa Số lượng video và Tương tác trên mỗi video:

Đối với tất cả các chỉ số tương tác được phân tích (Lượt xem, Lượt thích, Lượt bình luận, Lượt chia sẻ), đều tồn tại mối tương quan nghịch với tổng số lượng video đăng tải. Điều này có nghĩa là, nhìn chung, khi một TikToker đăng tải càng nhiều video, thì lượng tương tác trung bình (lượt xem, lượt thích, bình luận, chia sẻ) mà mỗi video riêng lẻ nhận được có xu hướng giảm xuống.

Mối tương quan nghịch này thể hiện rõ nhất ở Lượt xem và Lượt thích, và yếu dần ở Lượt bình luận và Lượt chia sẻ. Điều này cho thấy việc tăng số lượng video tác động mạnh hơn đến lượt xem và lượt thích trên từng video so với lượt bình luận và chia sẻ.

Đường xu hướng trên các biểu đồ phân tán đều minh họa xu hướng giảm của tương tác trên mỗi video khi số lượng video tăng lên.

Sự đánh đổi giữa Số lượng và Chất lượng (trên mỗi video):

Kết quả này nhấn mạnh sự đánh đổi tiềm ẩn giữa số lượng và hiệu quả tương tác trên từng video. Mặc dù việc đăng nhiều video hơn có thể làm tăng tổng số lượt xem/tương tác kênh nhận được (do số lượng nội dung nhiều hơn), nó dường như lại làm "loãng" mức độ tương tác trung bình mà mỗi video đơn lẻ có thể thu hút.

Quan sát về các TikToker có ít video nhưng đạt lượt tương tác trung bình trên mỗi video rất cao (nằm ở góc trên bên trái của các biểu đồ phân tán) củng cố nhận định này. Những nhà sáng tạo này có thể đang tập trung vào việc sản xuất nội dung chất lượng cao, độc đáo, hoặc tìm được thị trường ngách hiệu quả, giúp tối đa hóa hiệu quả tương tác của từng video mà không cần đăng bài ồ ạt.

Hàm ý cho Nhà sáng tạo nội dung (Tổng quát cho mọi nhóm): Dựa trên mối tương quan nghịch được quan sát, chiến lược "chất lượng hơn số lượng" (trên mỗi video) dường như là một yếu tố quan trọng để tối đa hóa mức độ tương tác trung bình cho từng nội dung.

- **Ưu tiên Chất lượng Nội dung:** Thay vì chỉ tập trung vào việc tăng tần suất đăng bài để có nhiều nội dung hơn, các nhà sáng tạo ở mọi nhóm (Thấp, Trung bình, Cao) nên ưu tiên đầu tư thời gian, công sức vào việc sản xuất các video có chất lượng cao, hấp dẫn, độc đáo và phù hợp với đối tượng mục tiêu.
- **Tối ưu hóa Tương tác trên từng Video:** Chú trọng vào các yếu tố giúp tăng tương tác trên mỗi video như nội dung giá trị, hình ảnh/âm thanh thu hút, kêu gọi hành động

(call-to-action) rõ ràng, tương tác với bình luận của người xem, v.v..

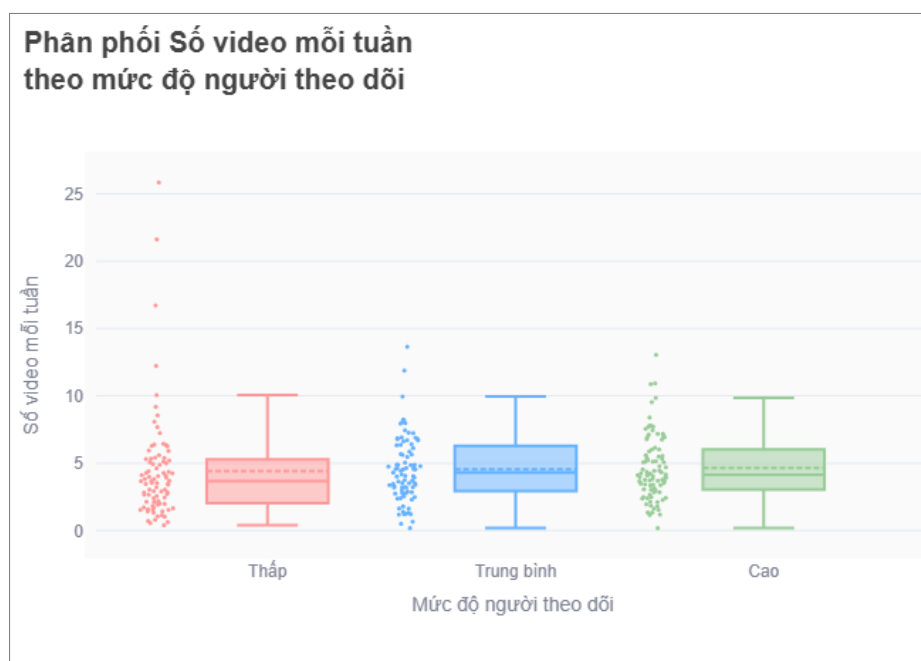
- **Cân bằng giữa Số lượng và Chất lượng:** Nhà sáng tạo cần tìm ra sự cân bằng phù hợp giữa tần suất đăng bài đều đặn và việc duy trì chất lượng cho từng video để tối ưu hóa cả tổng tương tác kênh và hiệu quả của từng nội dung.

Kết luận: Phân tích này cho thấy một xu hướng rõ ràng: việc tăng tổng số lượng video đăng tải có mối liên hệ tiêu cực với hiệu quả tương tác trung bình trên mỗi video. Điều này nhấn mạnh tầm quan trọng của chất lượng nội dung và chiến lược tối ưu hóa từng video đối với các nhà sáng tạo TikTok, bất kể họ đang ở mức độ người theo dõi nào.

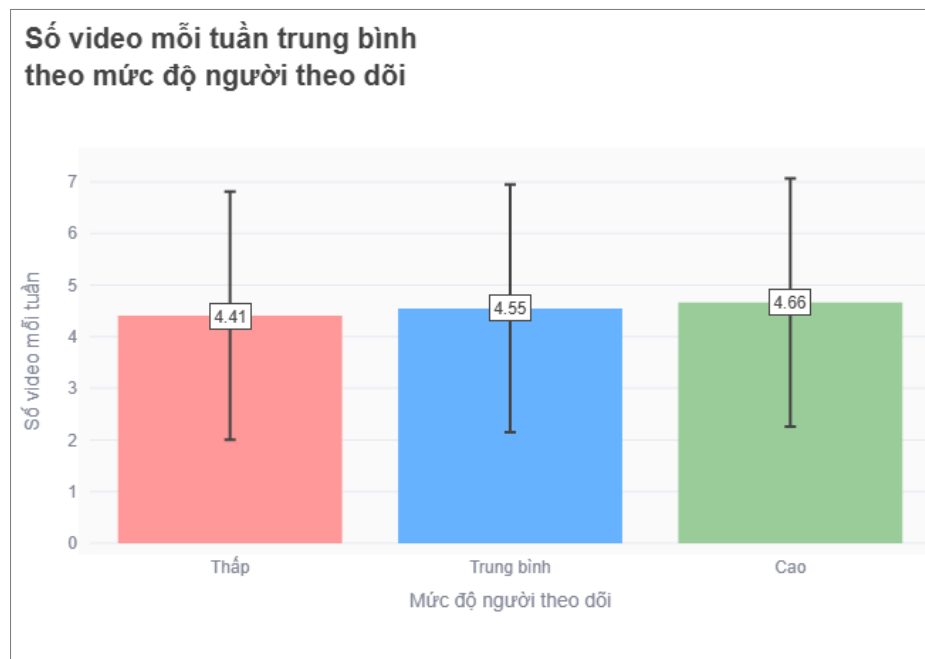
6.5 Phân tích ý nghĩa thống kê về tần suất đăng tải video mỗi tuần, số lượng hashtag trung bình trên mỗi video và thời lượng video trung bình giữa các nhóm người dùng

Câu hỏi nghiên cứu: Có sự khác biệt có ý nghĩa thống kê về tần suất đăng tải video mỗi tuần, số lượng hashtag trung bình trên mỗi video và thời lượng video trung bình giữa các nhóm người dùng có số người theo dõi thấp, trung bình và cao không?

6.5.1 Về Tần Suất Đăng Tải Video Mỗi Tuần



Hình 13: Phân phối Số Video Mỗi Tuần theo Mức Độ Người Theo Dõi



Hình 14: Số Video Trung Bình Mỗi Tuần theo Mức Độ Người Theo Dõi

Mức độ người theo dõi	Số lượng mẫu	Trung bình	Độ lệch chuẩn	Tối thiểu	Tối đa
Thấp	87	4.41	4.00	0.42	25.85
Trung bình	87	4.55	2.44	0.20	13.65
Cao	90	4.66	2.40	0.19	13.05

Bảng 7: Thống kê Mô tả Số Video Mỗi Tuần theo Mức Độ Người Theo Dõi

Kết quả kiểm định thống kê: Kiểm định Kruskal-Wallis: $p\text{-value} = 0.0856$

- **Kết luận:** Không có sự khác biệt có ý nghĩa thống kê về Số video mỗi tuần giữa các nhóm người dùng có số lượng người theo dõi khác nhau ($p = 0.0856 > 0.05$).

Nhận xét: Dựa trên kết quả kiểm định và các thống kê mô tả:

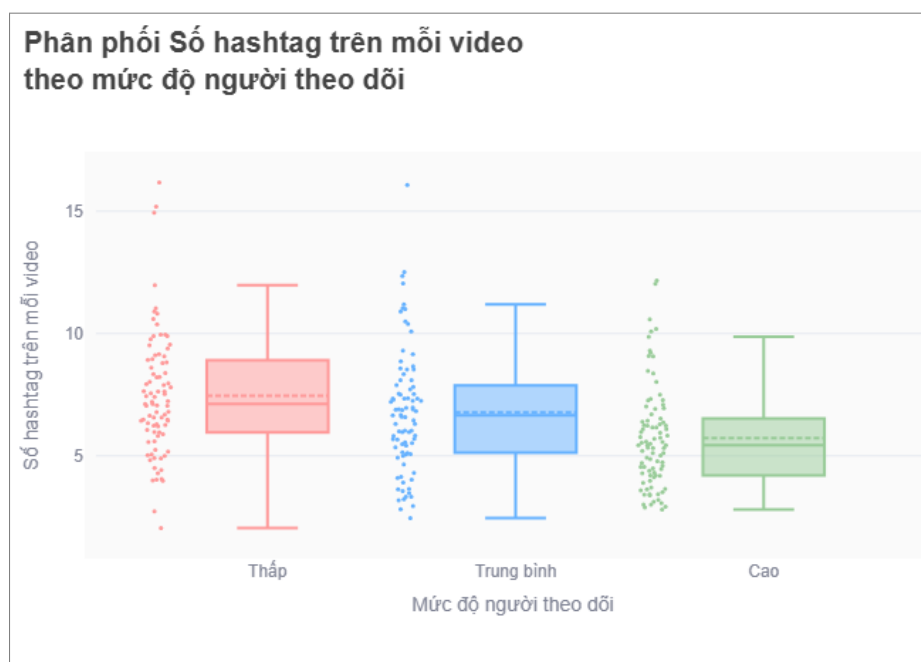
- **Không có sự khác biệt ý nghĩa:** Giá trị $p\text{-value} = 0.0856$ lớn hơn mức ý nghĩa thông thường (ví dụ: 0.05). Điều này cho thấy không có bằng chứng thống kê thuyết phục để kết luận rằng có sự khác biệt đáng kể về số video mỗi tuần giữa các nhóm người dùng có số lượng người theo dõi khác nhau (Thấp, Trung bình, Cao).
- **Xu hướng không rõ ràng:** Mặc dù không có ý nghĩa thống kê, các giá trị trung bình cho thấy một xu hướng tăng nhẹ về số video mỗi tuần khi số lượng người theo dõi tăng lên ($4.41 \rightarrow 4.55 \rightarrow 4.66$). Tuy nhiên, sự khác biệt này quá nhỏ và không đủ để kết luận chắc chắn.

- **Ý nghĩa đối với nhà sáng tạo nội dung:** Kết quả này gợi ý rằng số lượng người theo dõi hiện tại có thể không phải là yếu tố quyết định chính trong việc xác định tần suất đăng video của một người dùng. Điều này có thể vì các nhà sáng tạo nội dung ở tất cả các quy mô theo dõi đều đăng video ở tần suất tương tự.

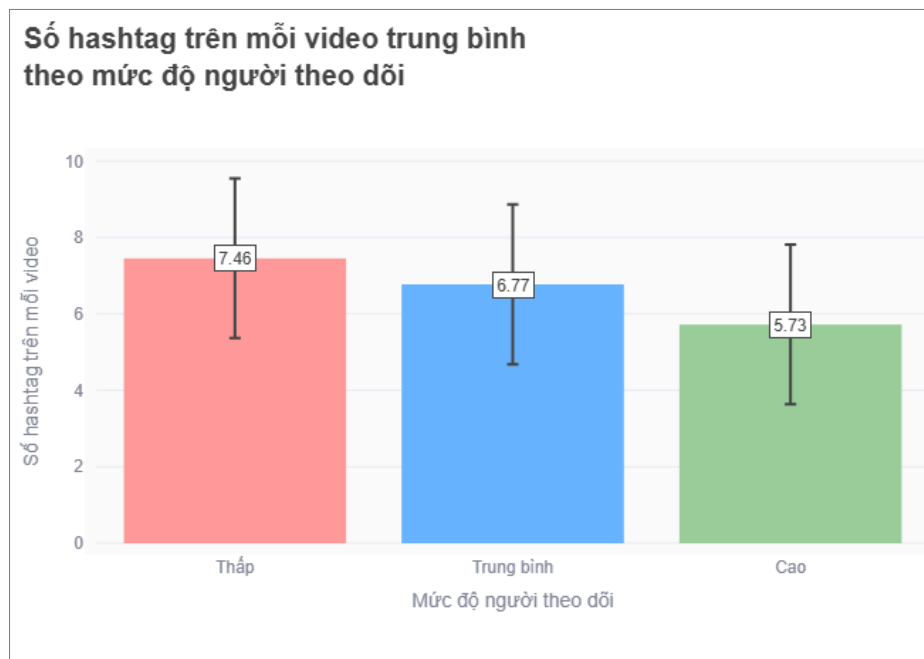
Chiến lược gợi ý:

- **Tập trung vào chất lượng:** Thay vì chỉ tập trung vào việc tăng tần suất đăng video, các nhà sáng tạo nên tập trung vào việc tạo ra nội dung chất lượng, thu hút và phù hợp với đối tượng khán giả mục tiêu.
- **Kiểm tra và thử nghiệm:** Cần tiếp tục theo dõi và phân tích dữ liệu để hiểu rõ hơn về hành vi của khán giả và tìm ra tần suất đăng video tối ưu. Có thể thử nghiệm các chiến lược khác, như đăng video vào các thời điểm khác nhau trong tuần hoặc thử nghiệm các loại nội dung khác nhau để xem xét ảnh hưởng.
- **Các yếu tố khác:** Ngoài số lượng người theo dõi, các yếu tố khác như tương tác (like, comment, share), nội dung, thời điểm đăng tải và thuật toán của TikTok có thể đóng vai trò quan trọng hơn trong việc quyết định sự thành công của một video.

6.5.2 Về Số Lượng Hashtag



Hình 15: Phân phối Số Hashtag trên Mỗi Video theo Mức Độ Người Theo Dõi



Hình 16: Số Hashtag Trung Bình trên Mỗi Video theo Mức Độ Người Theo Dõi

Mức độ người theo dõi	Số lượng mẫu	Trung bình	Độ lệch chuẩn	Tối thiểu	Tối đa
Thấp	87	7.46	2.50	2.04	16.18
Trung bình	87	6.77	2.52	2.46	16.07
Cao	90	5.73	2.09	2.80	12.16

Bảng 8: Thống kê Mô tả Số Hashtag trên Mỗi Video theo Mức Độ Người Theo Dõi

Kết quả kiểm định thống kê: Kiểm định Kruskal-Wallis: $p\text{-value} = 0.0000$

- **Kết luận:** Có sự khác biệt có ý nghĩa thống kê về Số hashtag trên mỗi video giữa các nhóm người dùng có số lượng người theo dõi khác nhau ($p = 0.0000 < 0.05$).

Nhận xét:

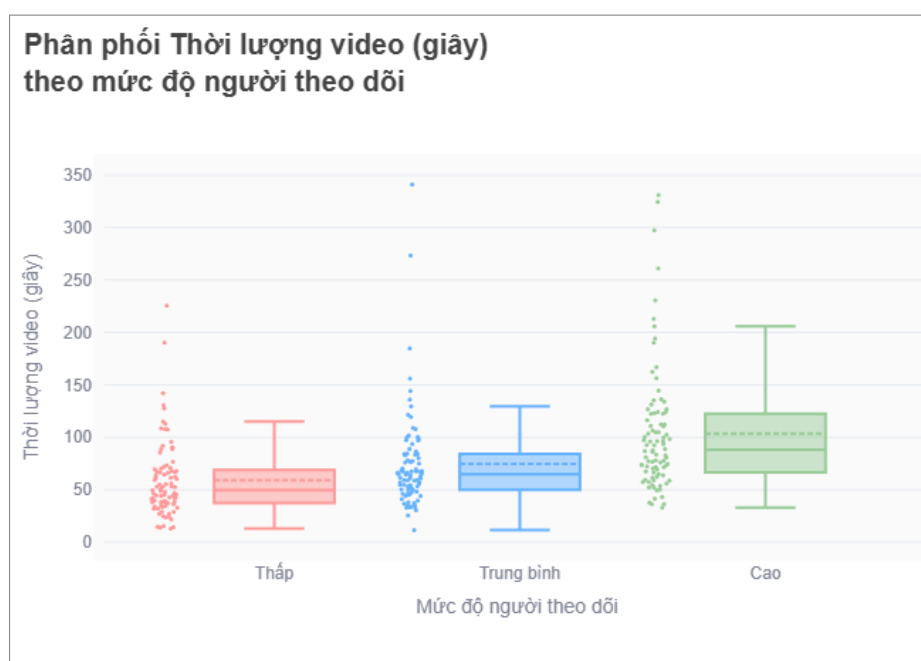
- **Sự khác biệt có ý nghĩa thống kê:** $p\text{-value} = 0.0000$ cho thấy có sự khác biệt có ý nghĩa thống kê giữa các nhóm người dùng về số lượng hashtag trên mỗi video. Điều này cho thấy mối quan hệ giữa số lượng người theo dõi và việc sử dụng hashtag không phải là ngẫu nhiên.
- **Số hashtag cao nhất/thấp nhất:**
 - Nhóm người theo dõi **Thấp**: Trung bình số hashtag là 7.46, cao nhất trong ba nhóm.
 - Nhóm người theo dõi **Cao**: Trung bình số hashtag là 5.73, thấp nhất trong ba nhóm.

- **Ý nghĩa đối với nhà sáng tạo nội dung:** Những nhà sáng tạo có ít người theo dõi có xu hướng sử dụng nhiều hashtag hơn. Điều này có thể là một nỗ lực để tăng khả năng hiển thị video của họ trong các kết quả tìm kiếm và tiếp cận đối tượng mới. Ngược lại, những nhà sáng tạo có nhiều người theo dõi có thể ít phụ thuộc vào hashtag hơn để tiếp cận khán giả của họ, vì họ đã có một lượng người xem trung thành.

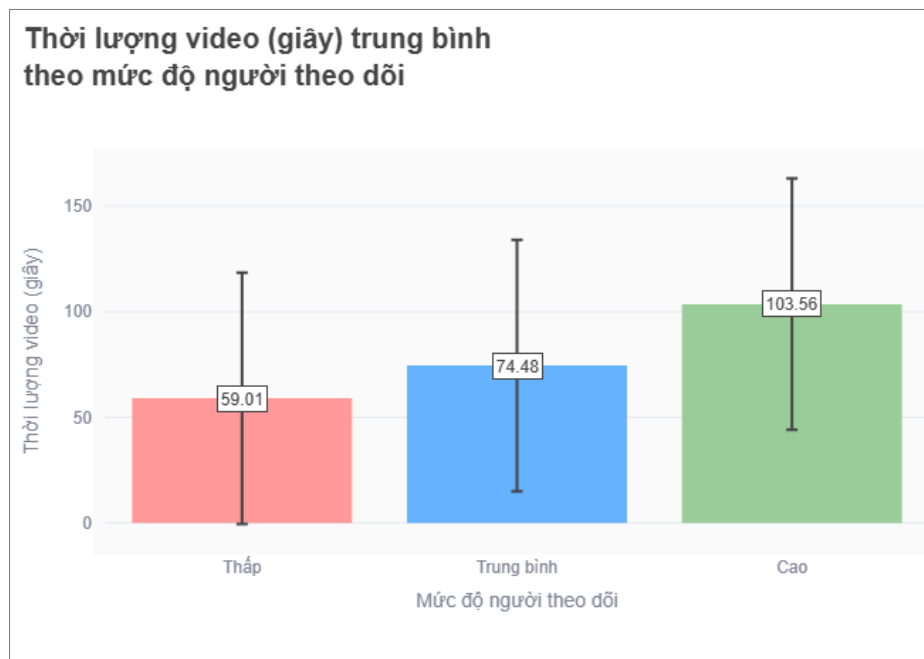
Chiến lược rút ra:

- **Đối với nhà sáng tạo có ít người theo dõi:** Việc sử dụng nhiều hashtag hơn có thể là một chiến lược hiệu quả để tăng khả năng hiển thị và thu hút người xem mới. Tuy nhiên, cần lựa chọn hashtag liên quan và phù hợp với nội dung video.
- **Đối với nhà sáng tạo có nhiều người theo dõi:** Có thể tập trung vào việc tạo nội dung chất lượng cao và tương tác với khán giả hiện tại, đồng thời sử dụng hashtag một cách có chọn lọc. Việc này giúp duy trì và phát triển cộng đồng của mình mà không bị phụ thuộc quá nhiều vào hashtag.
- **Chiến lược chung:** Cần cân bằng giữa việc sử dụng hashtag và việc tạo nội dung hấp dẫn. Việc thử nghiệm và theo dõi hiệu quả của hashtag là rất quan trọng để tìm ra chiến lược phù hợp nhất cho từng cá nhân.

6.5.3 Về Thời Lượng Video Trung Bình



Hình 17: Phân phối Thời Lượng Video theo Mức Độ Người Theo Dõi



Hình 18: Thời Lượng Video Trung Bình theo Mức Độ Người Theo Dõi

Mức độ người theo dõi	Số lượng mẫu	Trung bình	Độ lệch chuẩn	Tối thiểu	Tối đa
Thấp	87	59.01	36.30	6.00	239.00
Trung bình	87	74.48	47.10	7.00	298.00
Cao	90	103.56	59.60	10.00	599.00

Bảng 9: Thống kê Mô tả Thời Lượng Video (giây) theo Mức Độ Người Theo Dõi

Kết quả kiểm định thống kê: Kiểm định Kruskal-Wallis: $p\text{-value} = 0.0000$

- **Kết luận:** Có sự khác biệt có ý nghĩa thống kê về Thời lượng video (giây) giữa các nhóm người dùng có số lượng người theo dõi khác nhau ($p = 0.0000 < 0.05$).

Nhận xét:

- **Sự khác biệt có ý nghĩa thống kê:** Kết quả kiểm định Kruskal-Wallis với $p\text{-value} = 0.0000$ cho thấy có sự khác biệt có ý nghĩa thống kê về thời lượng video giữa các nhóm người theo dõi (Thấp, Trung bình, Cao).
- **Thời lượng video và số lượng người theo dõi:** Nhóm người theo dõi **Cao** có thời lượng video trung bình cao nhất (103.56 giây), tiếp theo là nhóm **Trung bình** (74.48 giây), và cuối cùng là nhóm **Thấp** (59.01 giây). Điều này cho thấy, xu hướng chung là người dùng có nhiều người theo dõi hơn có xu hướng đăng tải video dài hơn.

- **Ý nghĩa đối với nhà sáng tạo nội dung:** Kết quả này gợi ý rằng việc tăng thời lượng video có thể liên quan đến việc thu hút và giữ chân người xem, đặc biệt là đối với những tài khoản đã có lượng người theo dõi đáng kể. Tuy nhiên, cần lưu ý rằng mối quan hệ này có thể không phải là nguyên nhân - hệ quả. Có thể những người đã thành công với nội dung dài, dẫn đến thu hút nhiều người theo dõi hơn.

Chiến lược:

- **Đối với nhà sáng tạo mới:** Nên thử nghiệm với nhiều độ dài video khác nhau để xem loại video nào thu hút và giữ chân khán giả.
- **Đối với nhà sáng tạo có lượng người theo dõi nhất định:** Có thể cân nhắc tăng thời lượng video, nhưng cần đảm bảo nội dung đủ hấp dẫn và giá trị để giữ chân người xem. Cần theo dõi kỹ các chỉ số tương tác như lượt xem, thời gian xem trung bình để đánh giá hiệu quả của việc thay đổi thời lượng video.

6.6 Tổng quan Kỹ thuật

Phần này cung cấp cái nhìn tổng quan kỹ thuật toàn diện về một bộ các bảng (dashboards) dựa trên **Streamlit**, được thiết kế để phân tích đa dạng các khía cạnh của dữ liệu người dùng và video trên TikTok. Các ứng dụng này cho phép:

- **Phân tích Tương quan và Phân phối Chỉ số Người dùng:** Khám phá mối quan hệ và sự phân bố của các chỉ số người dùng cốt lõi như số lượng người theo dõi, tổng lượt thích (tim) và số lượng video đã đăng.
- **Phân tích Tương tác theo Cấp độ Người theo dõi:** Đánh giá mối quan hệ giữa số lượng video và các chỉ số tương tác trên mỗi video (lượt xem, lượt thích, bình luận, chia sẻ) và so sánh các mẫu sáng tạo nội dung (tần suất video, sử dụng hashtag, thời lượng video) giữa các nhóm người dùng có số lượng người theo dõi thấp, trung bình và cao.
- **Phân tích Chi tiết TikToker Cá nhân:** Đi sâu vào dữ liệu video của một TikToker cụ thể, bao gồm lịch sử đăng bài, số liệu tương tác, sở thích cá nhân (hashtag, âm nhạc), và đặc trưng nội dung (chủ đề, giọng điệu, cấu trúc).
- **Phân tích Người dùng TikTok Hàng đầu:** Xác định và trực quan hóa những người dùng TikTok hàng đầu dựa trên các chỉ số như lượt thích, số lượng video, số người theo dõi và tỷ lệ tương tác.

Các bảng điều khiển này tận dụng khả năng trực quan hóa dữ liệu tương tác của **Plotly**, xử lý và phân tích dữ liệu hiệu quả với **Pandas**, cung cấp thông tin chi tiết dựa trên AI thông qua **Google Gemini API**, và trong một số trường hợp, sử dụng **SciPy** cho các kiểm định thống kê. Giao diện người dùng được tùy chỉnh để nâng cao trải nghiệm và cung cấp các tính năng tương tác để khám phá dữ liệu.

6.6.1 Nguồn Dữ liệu

Định dạng Tập: Dữ liệu đầu vào chủ yếu được đọc từ các tập **Parquet** để tối ưu hiệu suất, với tùy chọn đọc từ tập **CSV**.

Đường dẫn Tập Ví dụ:

- **data/processed/cleaned_user_info.parquet:** Chứa thông tin tổng hợp của người dùng (sử dụng cho phân tích tương quan, phân tích tương tác, và phân tích người dùng hàng đầu).
- **data/processed/cleaned_video_info.parquet:** Chứa thông tin chi tiết của từng video (sử dụng cho phân tích **TikTok** cá nhân).
- **data/processed/content_features_6_users.parquet:** Chứa các đặc trưng về nội dung được phân tích trước (ví dụ: chủ đề, giọng điệu) cho một số người dùng (sử dụng cho phân tích **TikTok** cá nhân).

Cấu trúc Dữ liệu Chính:

Bảng 10: Danh sách các biến phân tích và ý nghĩa

Nhóm Biến	Tên Biến	Ý nghĩa
Thông tin Người dùng	<code>user.uniqueId</code>	ID duy nhất của người dùng TikTok
	<code>stats.followerCount</code>	Số lượng người theo dõi
	<code>stats.heartCount</code>	Tổng số lượt thích (thả tim) của người dùng
	<code>stats.videoCount</code>	Tổng số video đã đăng

(tiếp theo trang sau)

Nhóm Biến	Tên Biến	Ý nghĩa
Chỉ số Tương tác Trung bình	avg_comments_per_video avg_diggs_per_video avg_plays_per_video avg_shares_per_video avg_collects_per_video	Số bình luận trung bình mỗi video Số lượt thích trung bình mỗi video Số lượt xem trung bình mỗi video Số lượt chia sẻ trung bình mỗi video Số lượt lưu trung bình mỗi video
Chỉ số Nội dung	avg_videos_per_week avg_hashtags_per_video avg_video_duration	Số video trung bình mỗi tuần Số hashtag trung bình mỗi video Thời lượng trung bình mỗi video (giây)
Thông tin Video Chi tiết	createTime hashtags music.authorName desc	Thời điểm video được đăng Danh sách các hashtag được sử dụng Tên tác giả của âm nhạc trong video Mô tả nội dung video
Đặc trưng Nội dung/ Kịch bản	main_content_focus structure_style hook_type tone_of_voice pacing	Chủ đề chính của video Kiểu cấu trúc nội dung (kể chuyện, mô tả,...) Kiểu câu mở đầu thu hút người xem Tông giọng sử dụng trong video Tốc độ trình bày nội dung
Chỉ số Tính toán	engagement_rate engagement_ratio	Tỷ lệ tương tác trên tổng số lượt xem Tỷ lệ tương tác trên tổng số người theo dõi

6.6.2 Hằng số

Các hằng số được định nghĩa để đảm bảo tính nhất quán và khả năng tái sử dụng:

- **DARK_GRAY**: Mã màu (ví dụ: #444444) cho văn bản, tiêu đề và các yếu tố trực quan.
- **CLEANED_USER_DATA_FILE**, **CLEANED_VIDEO_INFO_FILE**, **CONTENT_FEATURES_FILE**: Đường dẫn đến các tệp dữ liệu.
- **COLORS_RGBA**: Danh sách các mã màu RGBA cho trực quan hóa (ví dụ: xanh dương, đỏ, xanh lá).
- **METRICS**: Danh sách các tên cột DataFrame để phân tích (ví dụ: stats.followerCount).
- **METRIC_LABELS**: Nhân tiếng Việt dễ đọc cho các chỉ số (ví dụ: Số người theo dõi).
- **COLUMN_TO_AXIS_TITLE**: Từ điển ánh xạ tên cột sang nhãn trục biểu đồ.

- **COLUMN_LABELS, COLUMN_METRICS:** Ảnh xạ các trường dữ liệu và chỉ số sang nhãn tiếng Việt.
- **STAT_TYPES:** Các loại thống kê (ví dụ: `count`, `mean`, `median`).
- **CHART_OPTIONS:** Ảnh xạ các lựa chọn chỉ số sang tên cột, bảng màu và nhãn trục.

6.6.3 Hàm Tiện ích Chung

Các hàm này xử lý các hoạt động cốt lõi, được chia sẻ hoặc có cấu trúc tương tự giữa các bảng điều khiển:

- **load_user_data() / load_data():** Tải dữ liệu từ tệp Parquet (hoặc CSV). Chuyển đổi các cột thời gian nếu cần. Thường được lưu trữ đệm với `@st.cache_data` (có thể kèm `persist="disk"`).
- **generate_insights(prompt, api_key):** Truy vấn Gemini API để tạo văn bản phân tích dựa trên một `prompt`. Xử lý ngoại lệ và trả về chuỗi rỗng nếu thất bại. Được lưu trữ đệm với `@st.cache_data`.
- **display_AI_generated_insights(prompt, api_key, ...):** Hiển thị các phân tích do AI tạo ra trong một `st.expander` của Streamlit, thường có một `st.spinner` để phản hồi người dùng trong quá trình tải.
- **apply_styles(), personal_styles():** Chèn CSS tùy chỉnh để cải thiện giao diện của bảng điều khiển (ví dụ: kiểu chữ tiêu đề, màu nút, định dạng `expander`) sử dụng `st.markdown(unsafe_allow_html=True)`.
- **calculate_engagement_ratio(df):** Tính toán tỷ lệ tương tác, ví dụ: `stats.heartCount / stats.followerCount` hoặc `stats.heartCount / stats.videoCount`. Xử lý các trường hợp chia cho không. Được lưu trữ đệm.

6.6.4 Hàm Tiện ích và Trực quan hóa theo Chức năng Bảng Điều khiển

Mỗi bảng điều khiển có các hàm chuyên biệt cho mục đích phân tích của nó:

Phân tích Tương quan và Phân phối Chỉ số Người dùng

- **get_correlation_matrix(df):** Tính ma trận tương quan Pearson.

- `select_columns(df, columns)`: Chọn các cột cụ thể từ DataFrame.
- `create_scatter_matrix(df, template)`: Tạo ma trận biểu đồ phân tán với biểu đồ tần suất trên đường chéo. Tùy chỉnh màu sắc, thêm đường hồi quy.
- `create_correlation_heatmap(df, template)`: Tạo heatmap của ma trận tương quan, thường ẩn tam giác trên. Xuất ma trận tương quan dưới dạng LaTeX cho AI prompt.
- `create_histogram(df, metric, bins, log_scale, ...)`: Tạo biểu đồ tần suất với biểu đồ hộp biên. Hỗ trợ tùy chỉnh số lượng bins và thang logarit.

Phân tích Tương tác theo Cấp độ Người theo dõi

- `calculate_avg_likes_per_video(df)`: Tính trung bình lượt thích trên mỗi video.
- `calculate_percentiles(df, percentiles)`: Tính các điểm phân vị cho `stats.follower-Count` để xác định ngưỡng thấp, trung bình, cao.
- `filter_data(df, follower_level, ...)`: Lọc DataFrame dựa trên cấp độ người theo dõi.
- `categorize_by_followers(df)`: Gán nhãn cấp độ người theo dõi (thấp, trung bình, cao).
- `plot_engagement_scatter(df, x_col, y_col, color)`: Tạo biểu đồ phân tán với đường xu hướng OLS (thường trên thang logarit) và hiển thị hệ số tương quan Pearson.
- *Biểu đồ hộp và biểu đồ cột*: Được dùng để so sánh các chỉ số (ví dụ: `avg_videos_per_week`) giữa các nhóm người theo dõi. Biểu đồ cột thường có thanh lỗi (error bars).
- *Kiểm định Kruskal-Wallis*: Được sử dụng để so sánh sự khác biệt thống kê giữa các nhóm người theo dõi.

Phân tích Chi tiết TikToker Cá nhân

- `filter_data_by_user_id(cleaned_video_info_df, ..., user_id)`: Lọc dữ liệu theo `userId` đã chọn.
- `plot_overall_posting_history(video_counts)`: Tạo biểu đồ diện tích thể hiện lịch sử đăng bài theo ngày, đánh dấu ngày có số lượng video cao nhất.

- `calculate_metrics(video_df)`: Tính toán các chỉ số hiệu suất chi tiết cho TikTok (ví dụ: tỷ lệ tương tác, lượt xem/lượt theo dõi).
- `determine_level(value, ref_range)`: Phân loại giá trị thành Thấp, Trung bình, Cao dựa trên khoảng tham chiếu.
- `display_dynamic_metrics_dashboard(video_df)`: Hiển thị bảng điều khiển với các đồng hồ đo (ví dụ: `go.Indicator`) cho các chỉ số hiệu suất, với màu sắc thay đổi theo mức độ.
- `plot_bar_chart(df, field, metric, stat_type, ...)`: Tạo biểu đồ cột ngang hiển thị thống kê theo trường và chỉ số, thường được sắp xếp.
- `analyze_scripts(data_df, title, user_context)`: Chức năng chính để phân tích và trực quan hóa đặc trưng nội dung, bao gồm bộ lọc đa lựa chọn, bảng điều khiển số liệu, biểu đồ cột, biểu đồ bánh (hashtags), và biểu đồ phân phối thời lượng.
- *Heatmap Lịch*: Hiển thị tần suất đăng bài theo ngày trong tuần và giờ trong ngày hoặc ngày trong tháng.
- *Treemap*: Hiển thị các hashtag phổ biến nhất.

Phân tích Người dùng TikTok Hàng đầu

- `filter_top_n_users(df, metric, n, sort_order)`: Lọc top N người dùng theo chỉ số được chọn và thứ tự sắp xếp.
- `create_bar_chart(data, metric, color_scale, y_label)`: Tạo biểu đồ cột hiển thị giá trị chỉ số cho top N người dùng, với thang màu động.
- `create_pie_chart(data, metric)`: Tạo biểu đồ tròn hiển thị phân phối chỉ số trong top N người dùng.

6.6.5 Bố cục Chung của các Bảng Điều khiển

Mặc dù mỗi bảng điều khiển có bố cục riêng, nhưng chúng cũng có một số cấu trúc chung:

- **Tiêu đề và Mô tả**: Sử dụng `st.title`, `st.header`, `st.markdown` hoặc `st.write` để giới thiệu mục đích của bảng điều khiển.
- **Thanh bên (Sidebar) / Khu vực Bộ lọc**:

- Cho phép người dùng lựa chọn (chẳng hạn như: `st.selectbox` để chọn TikTok, chỉ số; `st.slider` để chọn số lượng người dùng N hoặc phạm vi ngày).
- Các nút điều khiển như nút radio (`st.radio`) để chọn thứ tự sắp xếp.

- **Khu vực Hiển thị Chính:**

- **Tổng quan Dữ liệu/Số liệu Thống kê:** Thường sử dụng `st.metric` trong các cột (`st.columns`) để hiển thị các số liệu chính.
- **Trực quan hóa Dữ liệu:** Các biểu đồ Plotly được hiển thị bằng `st.plotly_chart`. Bố cục có thể là một cột hoặc nhiều cột.
- **Phân tích AI:** Các nhận xét từ Gemini API thường được hiển thị trong `st.expander`.
- **Bảng Dữ liệu Chi tiết:** Dữ liệu dạng bảng (thường là DataFrame của Pandas) có thể được hiển thị trực tiếp, trong `st.expander`, hoặc với phân trang tùy chỉnh.
- **Tải Dữ liệu:** Nút `st.download_button` để người dùng tải xuống dữ liệu đã xử lý hoặc đã lọc dưới dạng CSV.

6.6.6 Lưu trữ Đệm (Caching)

`@st.cache_data` được sử dụng rộng rãi để lưu trữ kết quả của các hàm tốn nhiều thời gian xử lý, bao gồm:

- Tải dữ liệu (`load_user_data`, `load_data`), thường với `persist="disk"` để lưu trữ giữa các phiên chạy nếu dữ liệu lớn và ít thay đổi.
- Tính toán ma trận tương quan (`get_correlation_matrix`).
- Tính toán các chỉ số phức tạp (`calculate_engagement_ratio`, `calculate_metrics`).
- Lọc dữ liệu (`filter_data`, `filter_top_n_users`).
- Tạo các biểu đồ Plotly (`create_scatter_matrix`, `create_bar_chart`, ...).
- Tạo nhận xét AI (`generate_insights`).

Lưu trữ đệm giúp cải thiện đáng kể hiệu suất và tốc độ phản hồi của ứng dụng khi người dùng tương tác.

Lưu ý: Cần đảm bảo các hàm quan trọng như tải dữ liệu luôn được cache, một số tài liệu gốc chỉ ra đây là điểm có thể cải thiện.

6.6.7 Tích hợp Trí tuệ Nhân tạo (AI Integration)

Google Gemini API (thường là mô hình `gemini-2.0-flash-lite` hoặc tương tự) được sử dụng để tạo ra các nhận xét, phân tích và tóm tắt dựa trên dữ liệu.

Cấu trúc Prompts: Các prompts được thiết kế cẩn thận để cung cấp ngữ cảnh cho AI, có thể bao gồm:

- Dữ liệu trực quan hóa (ví dụ: hình ảnh của biểu đồ được chuyển đổi bằng `fig.to_image()`).
- Bảng dữ liệu tóm tắt (ví dụ: DataFrame được chuyển đổi sang định dạng LaTeX hoặc markdown).
- Câu hỏi nghiên cứu hoặc hướng dẫn cụ thể về những gì cần phân tích.
- Các yêu cầu về độ dài hoặc định dạng của phản hồi (ví dụ: tránh các cụm từ như “Dựa trên dữ liệu”).

Xử lý Lỗi: Các hàm gọi API thường bao gồm khối `try-except` để xử lý các lỗi tiềm ẩn (ví dụ: API quá tải, lỗi mạng) và cung cấp phản hồi thân thiện cho người dùng.

6.6.8 Kiểm định Thống kê

Kiểm định Kruskal-Wallis: Được sử dụng trong "Bảng Điều khiển Phân tích Tương tác" để so sánh các chỉ số (ví dụ: tần suất đăng bài, sử dụng hashtag, thời lượng video) giữa các nhóm người theo dõi khác nhau (thấp, trung bình, cao). Đây là một kiểm định phi tham số, phù hợp khi giả định về phân phối chuẩn không được đáp ứng.

Xử lý Dữ liệu: Giá trị NaN thường được loại bỏ trước khi thực hiện kiểm định.

Hiển thị Kết quả: Giá trị p (p-value) được hiển thị, và kết luận được đưa ra dựa trên mức ý nghĩa (thường là $p < 0.05$).

6.6.9 Hướng dẫn tương tác trên trang web

- Sử dụng các bộ lọc trong thanh bên hoặc khu vực điều khiển để chọn người dùng, chỉ số, phạm vi ngày, số lượng kết quả (top N), v.v..
- Xem các biểu đồ tương tác, di chuột qua các điểm dữ liệu để xem chi tiết.

- Mở rộng các mục "Phân tích AI" hoặc "Chi tiết" để đọc các nhận xét hoặc xem dữ liệu dạng bảng.
- Sử dụng nút tải xuống để lưu dữ liệu đã lọc.

7 CHƯƠNG 7: DASHBOARD PHÂN TÍCH VIDEO

7.1 Giới thiệu

Dashboard này giúp người dùng dễ dàng phân tích hiệu suất video TikTok, hiểu rõ các yếu tố thành công như tương tác và hashtag, thông qua một ứng dụng web **Streamlit** tương tác với biểu đồ và nhận xét AI.

7.2 Kiến trúc Hệ thống và Công nghệ Sử dụng

Hệ thống dashboard được xây dựng chủ yếu bằng ngôn ngữ Python với các thư viện và công nghệ cốt lõi sau:

- **Framework Web App:** **Streamlit** được sử dụng làm nền tảng chính để xây dựng giao diện người dùng tương tác, hiển thị dữ liệu và điều hướng giữa các trang phân tích.
- **Xử lý và Phân tích Dữ liệu:**
 - **Pandas:** Là thư viện chủ đạo cho việc đọc dữ liệu Parquet trong `data_load.py`, làm sạch, biến đổi (ví dụ: trích xuất hashtag, chuẩn hóa thời gian), tính toán thống kê trong các module phân tích.
 - **NumPy:** Được sử dụng ngầm định bởi Pandas và có thể dùng trong các tính toán số học, ví dụ trong việc định nghĩa khoảng thời lượng video (`DURATION_BINS`) hoặc xử lý kiểu dữ liệu array.
- **Trực quan hóa Dữ liệu:** Plotly (bao gồm Plotly Express (px) và Plotly Graph Objects (go)) được dùng để tạo các biểu đồ tương tác đa dạng như biểu đồ đường, biểu đồ cột, biểu đồ phân tán, và biểu đồ kết hợp.
- **Tích hợp Trí tuệ Nhân tạo (AI):** Sử dụng API của Google Gemini thông qua thư viện `google.genai` để tự động tạo ra các báo cáo, nhận xét và insight dựa trên dữ liệu được hiển thị trong các biểu đồ.
- **Tối ưu hóa Hiệu năng:** Sử dụng kỹ thuật caching tích hợp của Streamlit:
 - `@st.cache_data`: Dùng để cache kết quả trả về của hàm tải dữ liệu (`load_data` trong `data_load.py`) và hàm tạo báo cáo AI (`generate_report` trong các module).

- `@st.cache_resource`: Dùng để cache các tài nguyên cần khởi tạo một lần, như client kết nối đến Gemini API (`get_genai_client` trong các module).

7.3 Luồng Dữ liệu và Tiền xử lý

Quá trình chuẩn bị dữ liệu cho dashboard bao gồm các bước chính, dữ liệu được tải vào ứng dụng thông qua `data_load.py`.

- **Nguồn Dữ liệu Đầu vào:** Dữ liệu thô ban đầu được đọc từ tệp `parquet` sau thu thập và tiền xử lý.
- **Xử lý Bổ sung khi Tải Dữ liệu:**
 - Trong `data_load.py`, cột `createTime` một lần nữa được đảm bảo là kiểu `datetime`.
 - Việc phân loại thời lượng video (cột `video.duration`) thành các khoảng thời gian (ví dụ: `<10s`, `10-30s`, ...) được thực hiện trong các module thay vì trong bước tiền xử lý chung.

7.4 Các Trang Phân tích

Dashboard được cấu trúc thành ba module (trang) phân tích chính, cho phép người dùng khám phá dữ liệu từ nhiều góc độ khác nhau. Dữ liệu được chia sẻ giữa các trang thông qua `st.session_state` sau khi được tải bởi `data_load.py` và khởi tạo trong `main.py`.

7.4.1 Phân tích Hiệu suất Video

Module này tập trung đánh giá hiệu suất tổng thể của video dựa trên các chỉ số tương tác chính, được triển khai trong tệp `video_performance.py`.

Mục tiêu: Cung cấp cái nhìn tổng quan về mức độ lan tỏa và tương tác của video, xác định các yếu tố ảnh hưởng như thời lượng `duration`, thời điểm đăng `createTime`.

Chỉ số và Tùy chọn: Người dùng có thể chọn chỉ số phân tích thông qua `st.selectbox` ở thanh bên bao gồm Lượt xem (`statsV2.playCount`), Lượt thích (`statsV2.diggCount`), Lượt bình luận (`statsV2.commentCount`), hoặc Lượt chia sẻ (`statsV2.shareCount`), được định nghĩa trong `METRIC_LABELS`. Kiểu tổng hợp dữ liệu (Tổng, Trung bình, Trung vị) cũng có thể được chọn qua `st.selectbox`, ảnh hưởng đến cách tính toán.

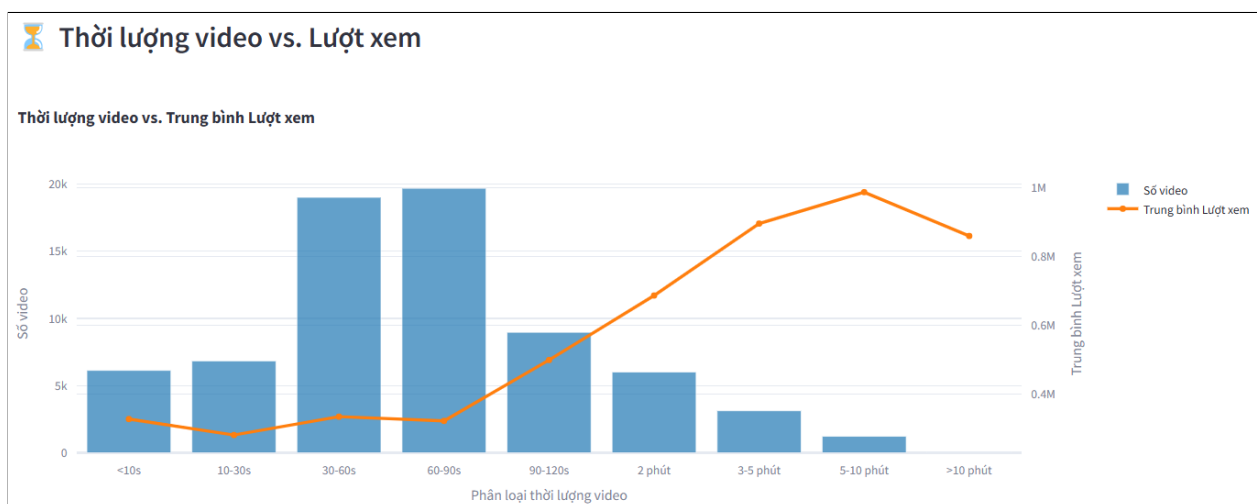
Trực quan hóa chính:

- **Thống kê Tổng quan:** Sử dụng `st.metric` để hiển thị các giá trị Tổng, Trung bình, Cao nhất của chỉ số được chọn, cùng với mô tả ngắn của video có giá trị cao nhất.



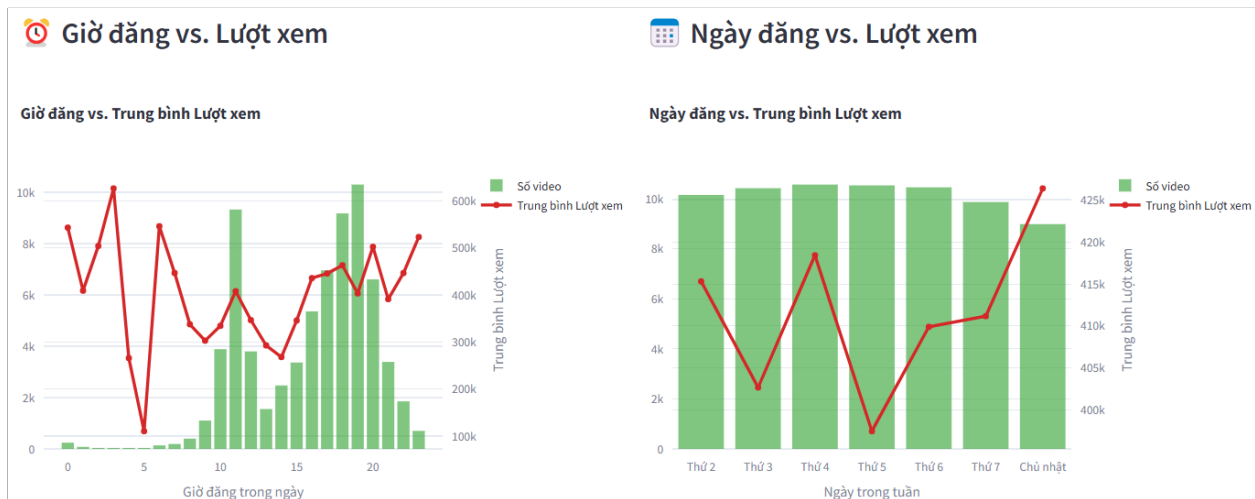
Hình 19: Thống kê tổng quan.

- **Hiệu suất theo Thời lượng Video:** Biểu đồ kết hợp hiển thị số lượng video (biểu đồ cột) và chỉ số tương tác đã chọn theo kiểu tổng hợp (biểu đồ đường) cho từng khoảng thời lượng video. Thời lượng được phân loại trước theo từng bins.



Hình 20: Thời lượng video vs. Lượt xem.

- **Hiệu suất theo Thời gian đăng:** Biểu đồ tương tự phân tích chỉ số theo giờ đăng trong ngày (`posting_hour` trích xuất từ `createTime.dt.hour`) và ngày đăng trong tuần (`posting_day` trích xuất từ `createTime.dt.day_name()`).



Hình 21: Thời gian đăng vs. Lượt xem.

- **Top 5 Video theo chỉ số hiệu suất:** Hiển thị bảng (sử dụng `st.dataframe`) liệt kê 5 video có hiệu suất cao nhất dựa trên chỉ số được chọn khi lọc filter ở thanh bên.

Top Video Hiệu Suất Cao		
Mô tả	Lượt xem	
Cơm với thịt luộc mằm nêm dầy ớt. #xuhuong #xuhuongtiktok #asmr #mukbang #ancungtiktok #viral #49lamdong #lethanhtuan	36400000	
ăn vặt tuổi thơ#anvat #tiktok #viral #learnontiktok #xuhuong #china #foryou	36000000	
Ăn cùng người lạ THCS Chu Văn An p33 #otreview #learnontiktok #ancungtiktok #xuhuong #wezmedia #WomenofTikTok #TikTokCommunit	26100000	
Cách làm cơm cuộn siêu ngon tại nhà, chỉ 10 phút #ancungtiktok #learnontiktok #xuhuong #LifebuoyChuaLifebuoyDi	25800000	
Ăn Nêm Rắn Siêu To Cùng Với Tui Nha #angithuongoi #mukbang #TràÔLong #TeaPlus #ĂnTếtNgonNhẹĐángSon #ancungtiktok #xuhuongtik	21900000	

Hình 22: Top Video Hiệu Suất Cao.

Tính năng bổ sung:

- **Nhận xét AI:** Tích hợp nhận xét tự động (sử dụng Gemini API) cho các biểu đồ phân tích theo thời lượng và thời gian đăng. Các nhận xét này được tạo dựa trên dữ liệu tóm tắt (`groupby(...).agg(...).to_string()`) của biểu đồ tương ứng.
- **Tải dữ liệu:** Nút `download_button` cho phép người dùng tải xuống toàn bộ dữ liệu đã xử lý dưới dạng file CSV.

7.4.2 Phân tích Hashtag Tổng quan

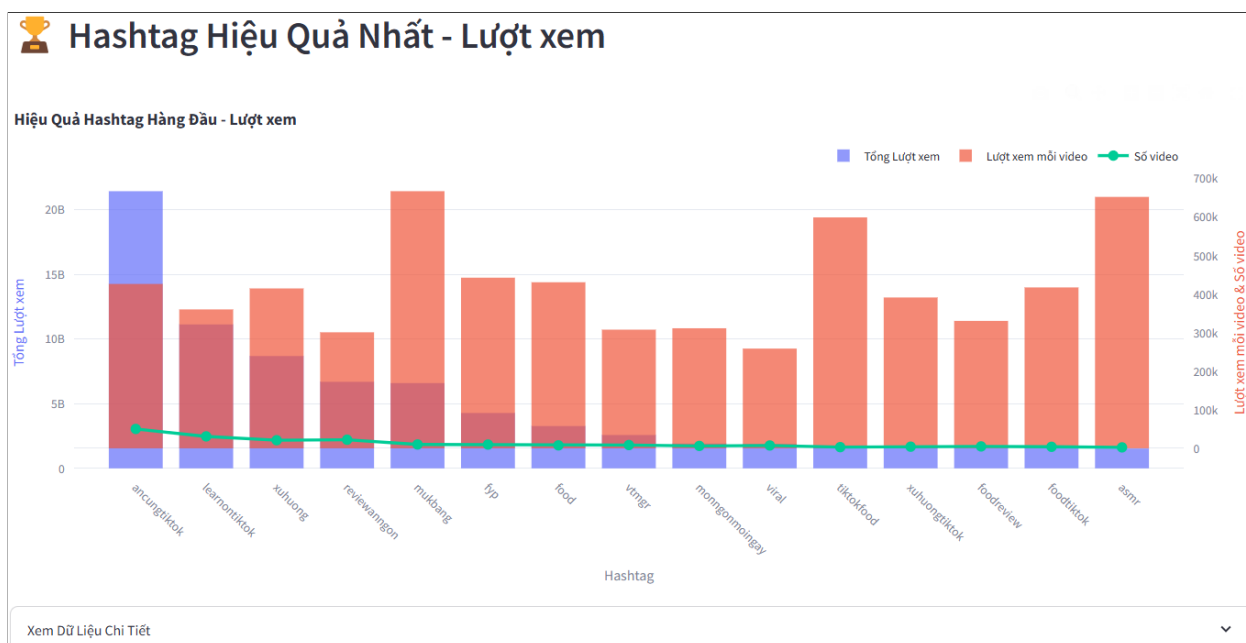
Module này cung cấp cái nhìn tổng thể về việc sử dụng và hiệu quả của các hashtag trong tập dữ liệu.

Mục tiêu: Khám phá các hashtag phổ biến, hiệu quả, xu hướng sử dụng và ảnh hưởng của số lượng hashtag đến hiệu suất video.

Tùy chọn / Filter: Người dùng chọn chỉ số, số lượng N (top_n) cho các bảng xếp hạng, và đơn vị thời gian (Ngày/Tuần/Tháng) cho biểu đồ xu hướng.

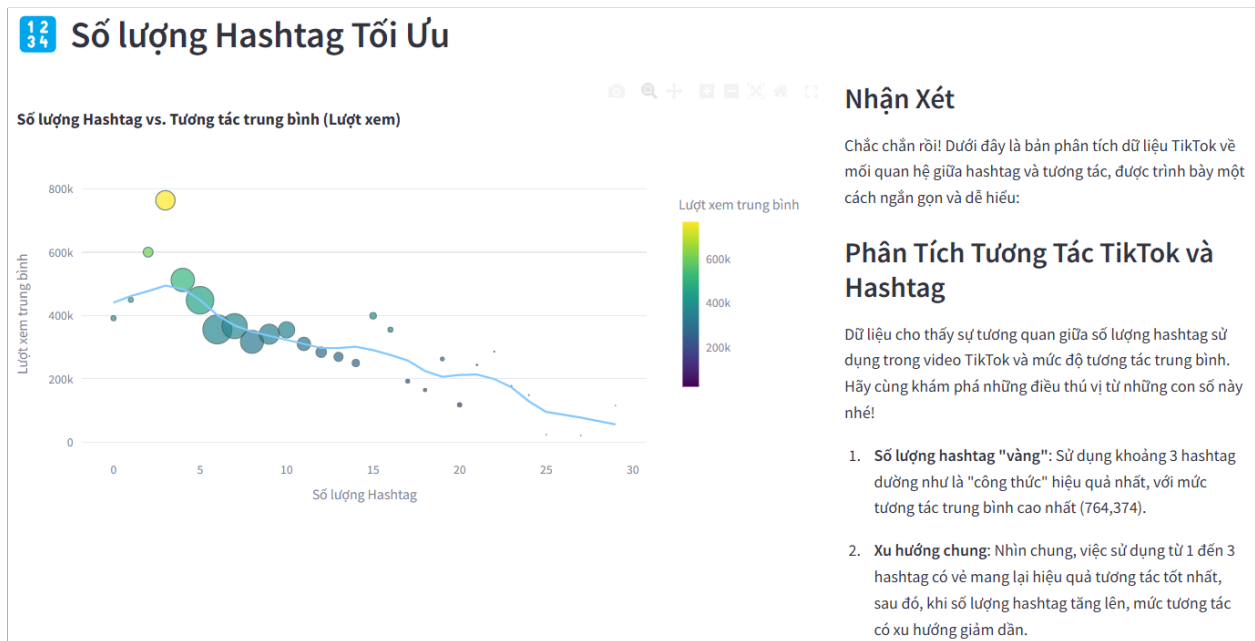
Phân tích chính:

- **Top N Hashtag hiệu quả:** Biểu đồ tính tổng chỉ số tương tác (total_engagement), số lượng video (video_count), và tương tác trung bình (avg_engagement) cho mỗi hashtag. Biểu đồ kết hợp tổng tương tác và tương tác/video vs cho số video sử dụng trực Y phụ.



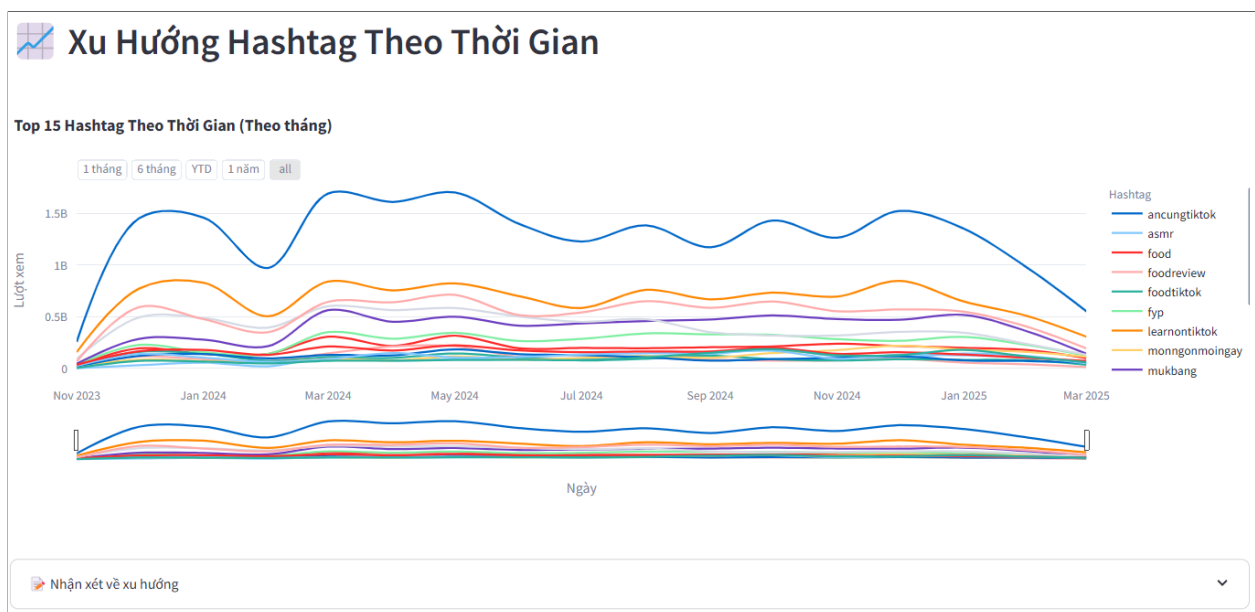
Hình 23: Hashtag Hiệu Quả Nhất.

- **Số lượng Hashtag tối ưu:** Biểu đồ sử dụng nhóm dữ liệu theo hashtag_count (số lượng hashtag mỗi video) và tính toán tương tác trung bình (avg_engagement), số lượng video (video_count), và trung vị (median_engagement). Biểu đồ trực quan hóa mối quan hệ giữa số lượng hashtag và tương tác trung bình, với kích thước điểm biểu thị số lượng video và có thêm đường xu hướng.



Hình 24: Số lượng Hashtag Tối Ưu.

- Xu hướng Hashtag theo thời gian:** Biểu đồ lọc ra top N hashtag hiệu quả nhất (dựa trên chỉ số phân tích), sau đó nhóm dữ liệu đã **explode** theo đơn vị thời gian được chọn và hashtag, rồi vẽ biểu đồ đường thể hiện tổng chỉ số tương tác theo thời gian cho từng hashtag. Có thể lựa chọn filter mốc thời gian trên biểu đồ.



Hình 25: Xu Hướng Hashtag Theo Thời Gian.

Tính năng bổ sung:

- Nhận xét AI:** Tích hợp nhận xét tự động từ Gemini cho biểu đồ phân tích số lượng hashtag và biểu đồ xu hướng hashtag, hiển thị trong window expander hoặc cột riêng.

7.4.3 Phân tích Hashtag Đơn lẻ

Module này cho phép người dùng đi sâu vào phân tích hiệu suất và bối cảnh sử dụng của một hashtag cụ thể.

Mục tiêu: Hiểu rõ hiệu suất theo thời gian, các hashtag liên quan và những người dùng (tác giả) sử dụng hashtag đó hiệu quả nhất.

Tùy chọn: Người dùng chọn hashtag cụ thể từ danh sách (danh sách drop-down được lấy từ dữ liệu), chọn chỉ số phân tích, đơn vị thời gian, và số lượng N (**top_n** qua slide filter ở thanh bên).

Phân tích chính (cho hashtag được chọn):

- **Lọc dữ liệu:** Thanh bên được sử dụng để tạo ra một filter con chỉ chứa các video có chứa hashtag được chọn (so sánh không phân biệt chữ hoa/thường). Nếu không có dữ liệu, cảnh báo (**warning**) sẽ được hiển thị.

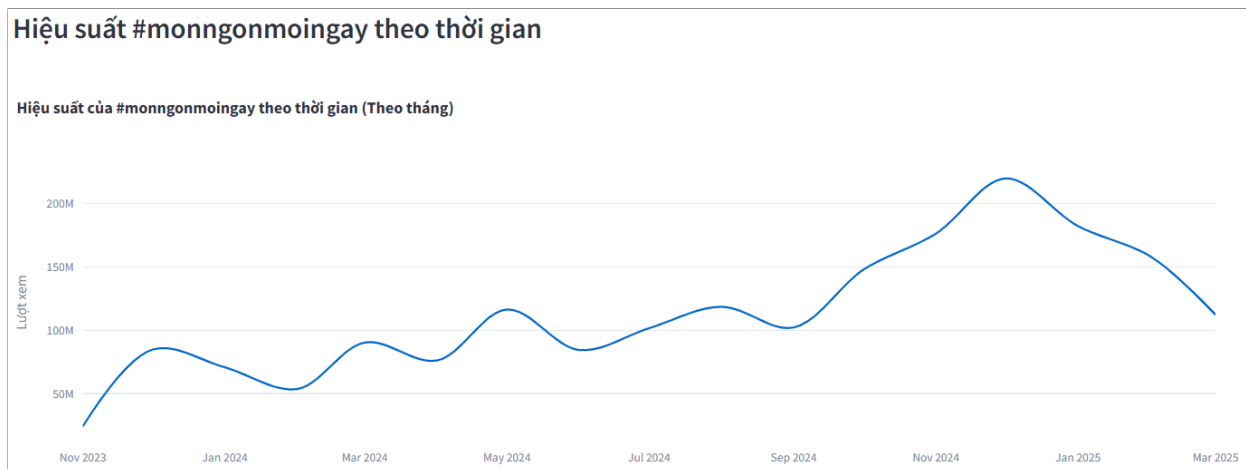
Hình 26: Cài đặt filter cho report

- **Thống kê tổng quan:** Hiển thị tổng quan tính toán các chỉ số tổng hợp (tổng video, tổng/trung bình/trung vị tương tác) cho hashtag đã lọc và hiển thị bằng thanh metric.

Phân tích hiệu suất của #monngonmoingay			
Tổng video	Tổng Lượt xem	Trung bình Lượt xem	Trung vị Lượt xem
6,154	1,917,329,712.0	311,558	64,150

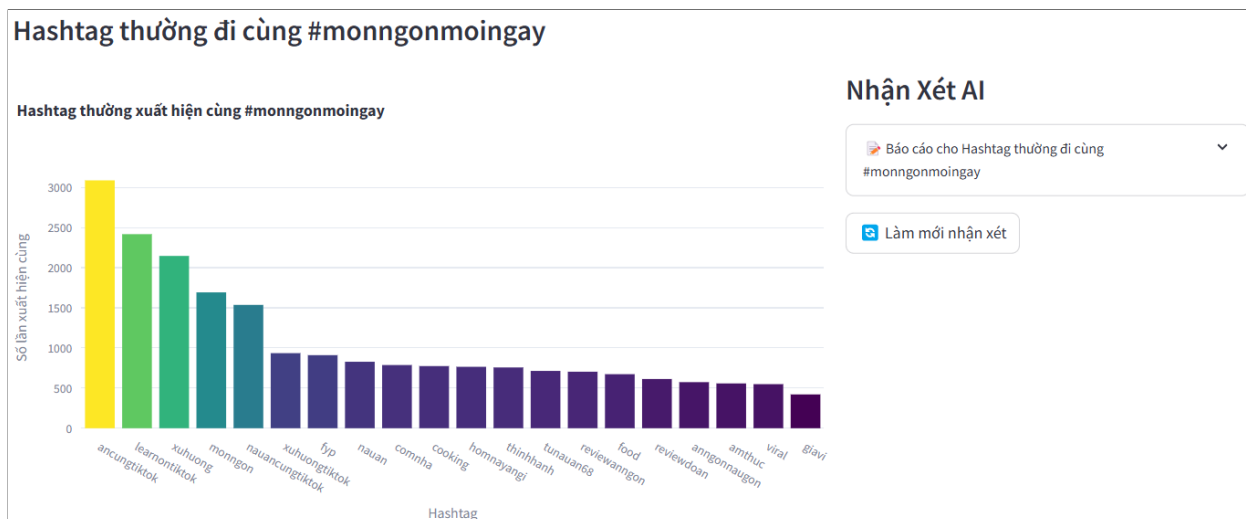
Hình 27: Phân tích hiệu suất của #monngonmoingay.

- **Xu hướng hiệu suất theo thời gian:** Biểu đồ nhóm dữ liệu đã lọc theo đơn vị thời gian (`time_group`) và vẽ biểu đồ đường thể hiện tổng chỉ số tương tác theo thời gian.



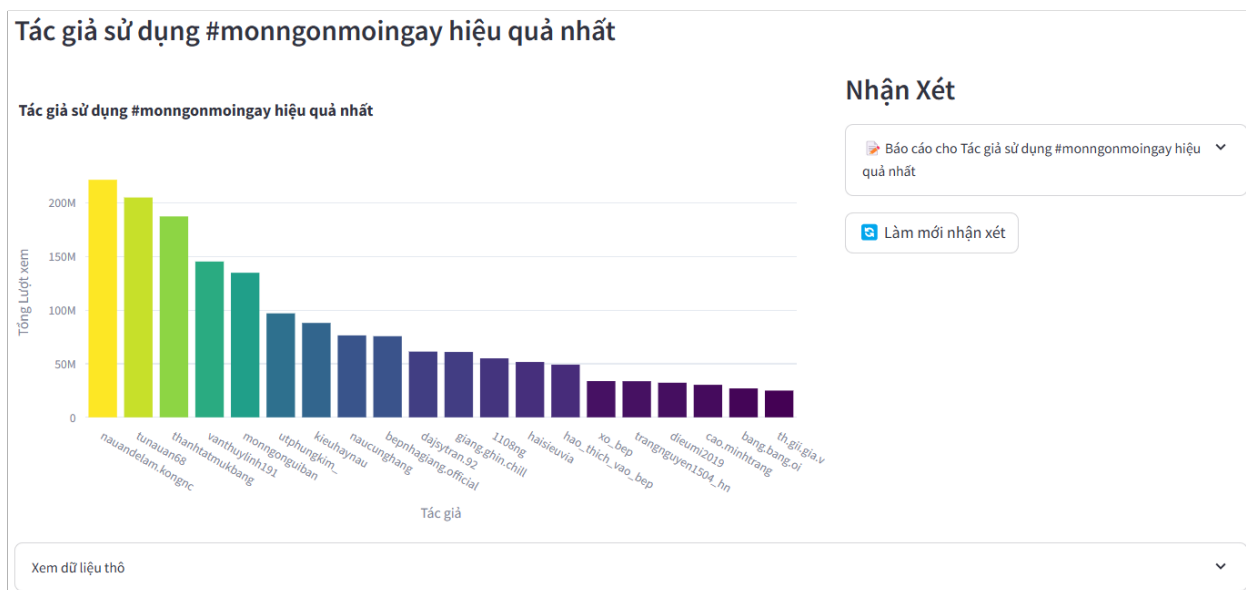
Hình 28: Hiệu suất #monngonmoingay theo thời gian.

- **Hashtag đồng xuất hiện (Co-occurring):** Biểu đồ hiển thị data về các video chứa hashtag mục tiêu, đầu tiên loại bỏ hashtag mục tiêu, đếm tần suất xuất hiện của các hashtag còn lại và vẽ biểu đồ cột hiển thị top N hashtag thường xuất hiện cùng nhất.



Hình 29: Hashtag thường đi cùng #monngonmoingay.

- **Tác giả hiệu quả nhất:** Biểu đồ sử dụng nhóm dữ liệu đã lọc theo `author.uniqueId`, tính tổng chỉ số tương tác (`total_engagement`), số video (`video_count`), và tương tác trung bình (`avg_engagement`), sau đó vẽ biểu đồ cột hiển thị top N tác giả có tổng tương tác cao nhất khi sử dụng hashtag này.



Hình 30: Tác giả sử dụng #monngonmoingay hiệu quả nhất.

Tính năng bổ sung:

- **Nhận xét AI:** Tích hợp nhận xét tự động từ Gemini cho phân tích hashtag đồng xuất hiện và phân tích tác giả hiệu quả, hiển thị trong window expander. Các prompt được thiết kế để yêu cầu Gemini phân tích về chủ đề liên quan, đặc điểm tác giả, khoảng cách hiệu suất, và gợi ý chiến lược.
- **Xem dữ liệu thô:** Cho phép người dùng xem bảng dữ liệu của các video chứa hashtag được chọn, sắp xếp theo chỉ số tương tác.

7.5 Tích hợp Trí tuệ Nhân tạo (AI)

Một tính năng nổi bật của dashboard là việc tích hợp mô hình ngôn ngữ lớn (LLM) Google Gemini để cung cấp các nhận xét và phân tích tự động, giúp người dùng nhanh chóng nắm bắt các insight quan trọng từ dữ liệu trực quan hóa.

- **Thư viện và Khởi tạo:** Sử dụng thư viện `google.genai`. Client kết nối đến API được khởi tạo và cache bằng hàm `get_genai_client` được đánh dấu `@st.cache_resource` trong mỗi module phân tích.
- **Hàm Tạo Báo cáo:** Đầu tiên nhận một chuỗi dữ liệu (`data_str`, thường là kết quả tóm tắt từ DataFrame hoặc sau khi áp dụng các bước filter / transformation và một prompt yêu cầu phân tích. Prompt được thiết kế bằng tiếng Việt, yêu cầu mô hình đóng vai trò

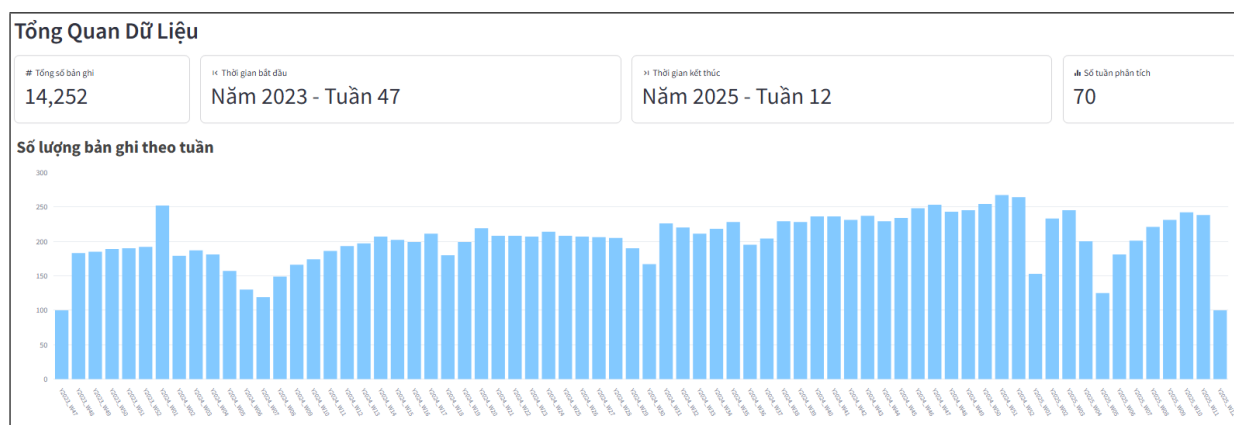
nhà phân tích dữ liệu, đưa ra các insight ngắn gọn, dễ hiểu dưới dạng Markdown. Kết quả trả về (văn bản nhận xét) được cache tối ưu cho **streamlit**.

- **Ứng dụng trong Dashboard:** Các nhận xét AI được tạo và hiển thị bên cạnh hoặc dưới các biểu đồ chính trong mỗi trang phân tích, thường nằm trong một window expander để tiết kiệm không gian. Ví dụ: nhận xét về mối quan hệ giữa thời lượng video và tương tác, gợi ý số lượng hashtag tối ưu, phân tích xu hướng hashtag, nhận định về các hashtag liên quan hoặc đặc điểm của các tác giả hàng đầu.

8 CHƯƠNG 8: DASHBOARD PHÂN TÍCH XU HƯỚNG

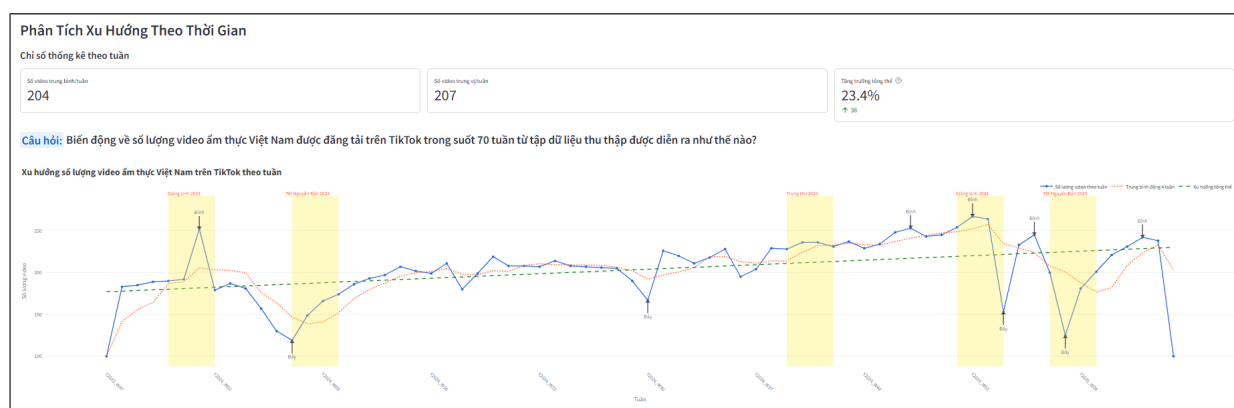
8.1 Giới thiệu và Tổng quan dữ liệu

Dashboard này được thiết kế nhằm cung cấp công cụ khám phá và theo dõi xu hướng đề cập của các món ăn và địa điểm ẩm thực dựa trên dữ liệu video TikTok. Bộ dữ liệu ban đầu được thu thập từ các video của các TikToker chuyên về ẩm thực, với tổng số **hơn 70,000** video. Quá trình thu thập diễn ra liên tục trong vòng **70 tuần**, từ **Tuần 47 Năm 2023** (tháng 11/2023) đến **Tuần 12 Năm 2025** (tháng 03/2025). Để thu gọn quy mô và đảm bảo bộ dữ liệu phản ánh các xu hướng nổi bật của từng tuần, nhóm đã tiến hành chọn lọc: mỗi tuần chỉ giữ lại **20%** video có **điểm số cao nhất** (tiêu chí điểm số được trình bày chi tiết tại Mục 4.3.3). Nhờ quá trình chọn lọc này, bộ dữ liệu cuối cùng phần nào đại diện cho các món ăn và địa điểm **đã và đang phổ biến, thịnh hành** trên nền tảng trong suốt giai đoạn nghiên cứu.



Hình 31: Tổng quan dữ liệu

8.2 Phân tích Xu hướng theo thời gian



Hình 32: Xu hướng theo Thời gian

Nhận xét:

1. Chỉ số thống kê theo Tuần:

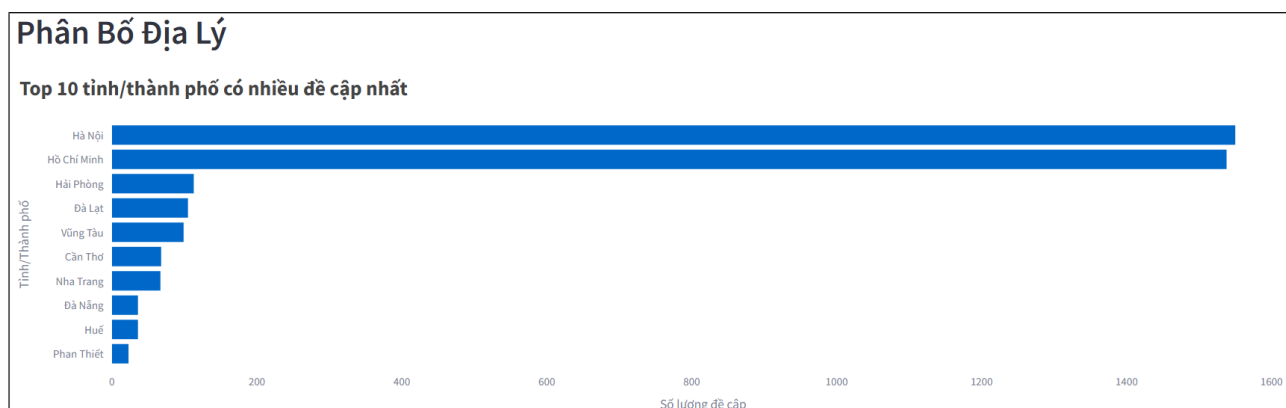
- Chỉ số thống kê theo tuần cho thấy số lượng video trung bình mỗi tuần đạt **204 video/tuần**, trung vị đạt **207 video/tuần**, thể hiện bộ dữ liệu không bị lệch quá nhiều và có thể kiểm tra lại với các phương pháp thống kê.
- Số lượng video cũng đạt mức tăng trưởng khoảng **23,4%** khi so sánh 4 tuần đầu và 4 tuần cuối, phản ánh xu hướng quan tâm đến các video ẩm thực được gia tăng.

- ### 2. Số lượng video qua từng giai đoạn:
- Xem xét về sự biến động số lượng video theo từng giai đoạn, ta có thể thấy được số lượng video về ẩm thực xuất hiện nhiều nhất ở các Tuần Giáng Sinh và có xu hướng giảm dần cho đến thời điểm Tết Nguyên Đán vào năm sau.

8.3 Phân tích Phân bố địa lý

8.3.1 Phân bố Tỉnh/Thành phố được đề cập

Tổng quan: Thông qua khảo sát dựa trên **14,252** bản ghi từ tập dữ liệu (xem Hình 33), chỉ khoảng **4,000** video có **đề cập đến địa điểm**. Trong đó, thành phố **Hà Nội** và thành phố **Hồ Chí Minh** là 2 thành phố được đề cập nhiều nhất trong toàn bộ dữ liệu ghi nhận được.



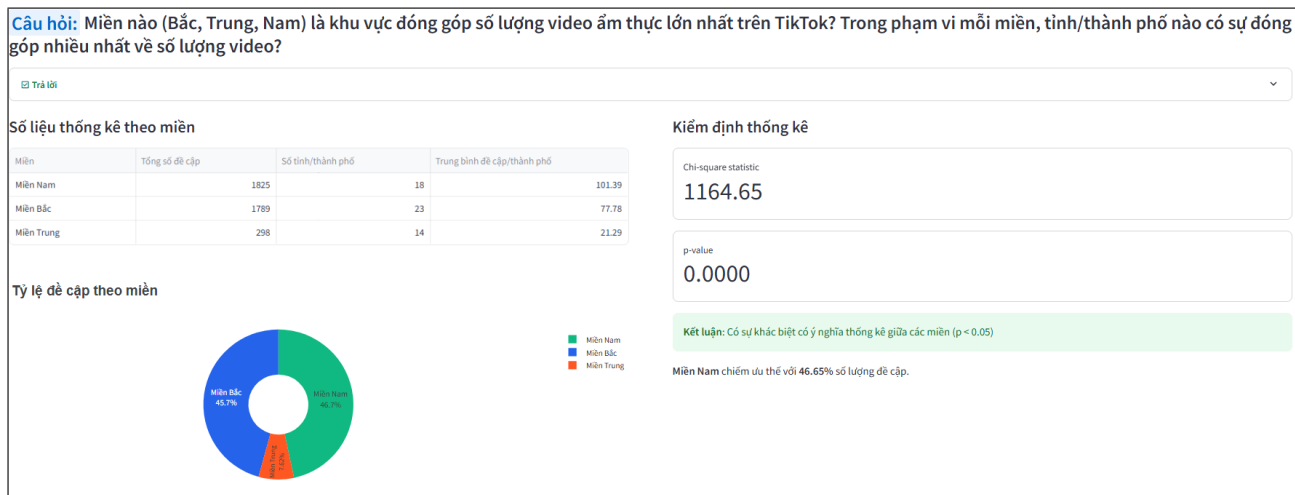
Hình 33: Phân bố các Tỉnh/Thành phố được đề cập

Phân tích chi tiết hơn: Hình 34 cho thấy: Số lượng video đề cập đến nhóm TP.Hồ Chí Minh và Hà Nội là **3088** (chiếm **78,94%**), cao hơn đáng kể, gấp **3,75 lần** so với **824** đề cập (chiếm **21,06%**) đến các địa phương khác. Kết quả kiểm định thống kê Mann-Whitney với $p = 0.009 < 0.05$ đã xác nhận sự khác biệt này là có ý nghĩa thống kê, cho thấy mức độ tập trung đề cập vào hai thành phố lớn này là vượt trội so với tổng các tỉnh/thành phố còn lại.

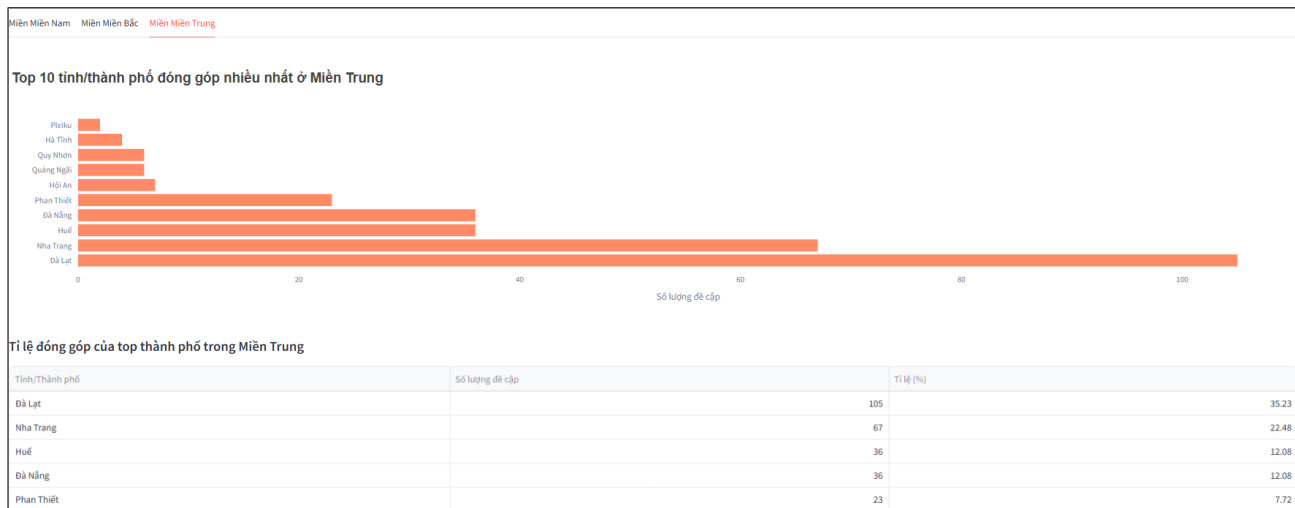


Hình 34: Số lượng video đề cập đến TP.Hồ Chí Minh và Hà Nội so với các địa phương khác

8.3.2 Phân bố Tỉnh/Thành phố thuộc từng vùng miền



Hình 35: Phân bố số lượng video theo từng miền

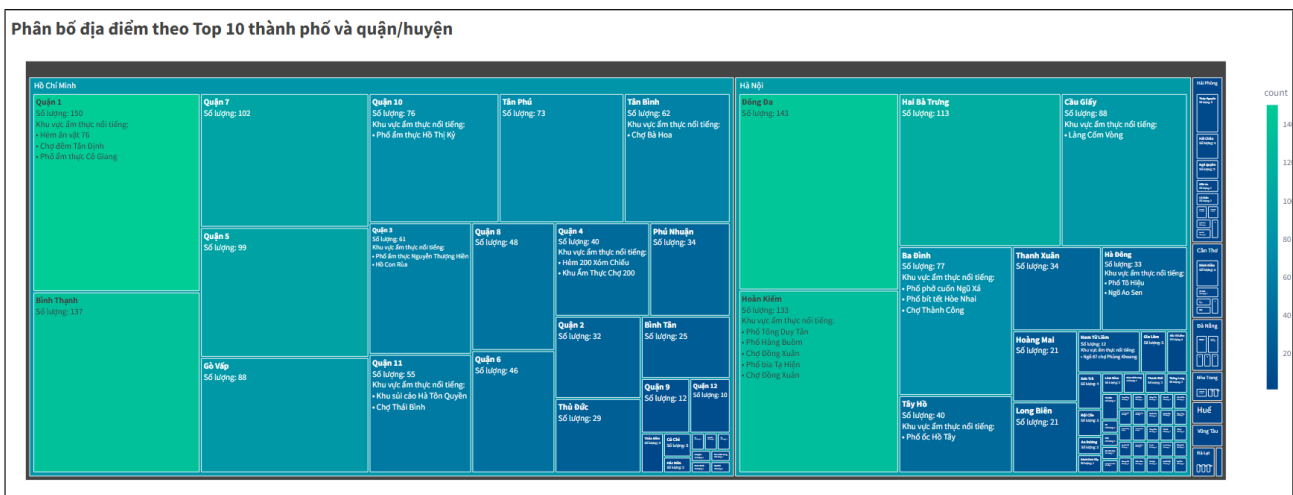


Hình 36: Phân bố tỉnh/thành phố trong từng miền

Nhận xét: Sau khi chuẩn hóa địa điểm theo từng vùng miền (sử dụng quá trình mapping), ta nhận thấy sự phân bố số lượng video có sự khác biệt rõ rệt. Cụ thể, Miền Trung có số lượng video ít nhất (xem Hình 35), chủ yếu tập trung vào các thành phố du lịch trọng điểm như Nha Trang, Đà Lạt, Huế (xem Hình 36).

Kiểm định thống kê: Để xác nhận liệu sự khác biệt trong phân bố số lượt đề cập giữa các vùng miền này có ý nghĩa thống kê hay không, nhóm đã tiến hành **kiểm định thống kê Chi-Square (Chi bình phương)**. Kết quả kiểm định đạt $\chi^2 = 1164.65$ và $p = 0.009 < 0.05$ (xem Hình 35). Điều này **khẳng định rằng sự khác biệt về tỷ lệ/phân bố số lượt đề cập giữa các danh mục (các vùng miền) là có ý nghĩa thống kê và không phải do ngẫu nhiên**. Dựa trên sự phân bố không đồng đều có ý nghĩa thống kê này, người dùng có thể xem xét phân tích và khai thác thị trường tiềm năng tại từng vùng miền.

8.3.3 Phân bố Quận/Huyện thuộc Tỉnh/Thành phố



Hình 37: Phân bố các Quận/Huyện thuộc Tỉnh/Thành phố

Địa điểm được khảo sát từ tập dữ liệu sẽ là các **địa điểm được đề cập** chứ không đảm bảo đây là địa điểm chính xác trong video. Ví dụ, khi xử lý video có đoạn **transcript** bao gồm thông tin “**Quán Bún đậu Hà Nội ở Quận 7**”, tập dữ liệu ghi nhận video chứa thông tin món ăn là Bún đậu, địa điểm thành phố Hà Nội và ở quận 7.

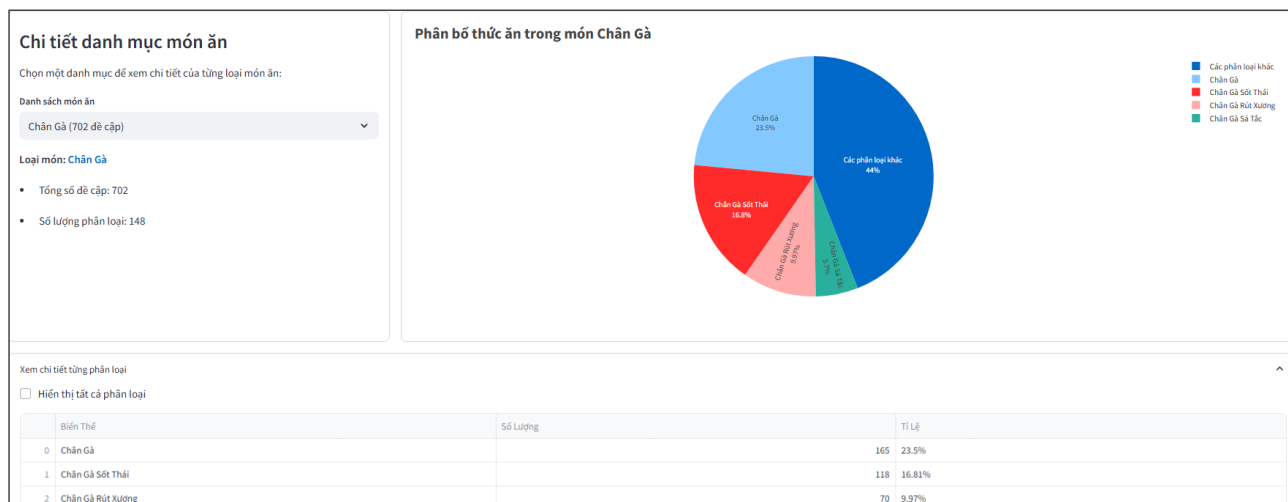
Thông qua phân tích và đánh giá thủ công từ các video, nhóm sẽ xử lý phân loại địa điểm dựa trên **Quận/Huyện để đưa về Tỉnh/Thành phố** do nội dung mọi người đề cập thường là các quán ăn ở các địa điểm thuộc Quận/Huyện và có **nguồn gốc** từ các Thành phố khác.

Tuy nhiên vẫn có các trường hợp, tên Quận/Huyện có thể trùng nhau ở các tỉnh, thành phố khác nhau nên nhóm chỉ có thể đảm bảo khảo sát dữ liệu Quận/Huyện tốt nhất trên 2 thành

phố **Hà Nội** và thành phố **Hồ Chí Minh**. Với các tỉnh/thành phố khác, nhóm sẽ giữ nguyên từ tập dữ liệu ban đầu chứ không thực hiện map về thành phố khác để đảm bảo tính toàn vẹn của dữ liệu.

8.4 Phân tích Món ăn theo danh mục và Xu hướng

8.4.1 Phân tích món ăn theo danh mục



Hình 38: Phân loại các loại **Chân gà** khác nhau

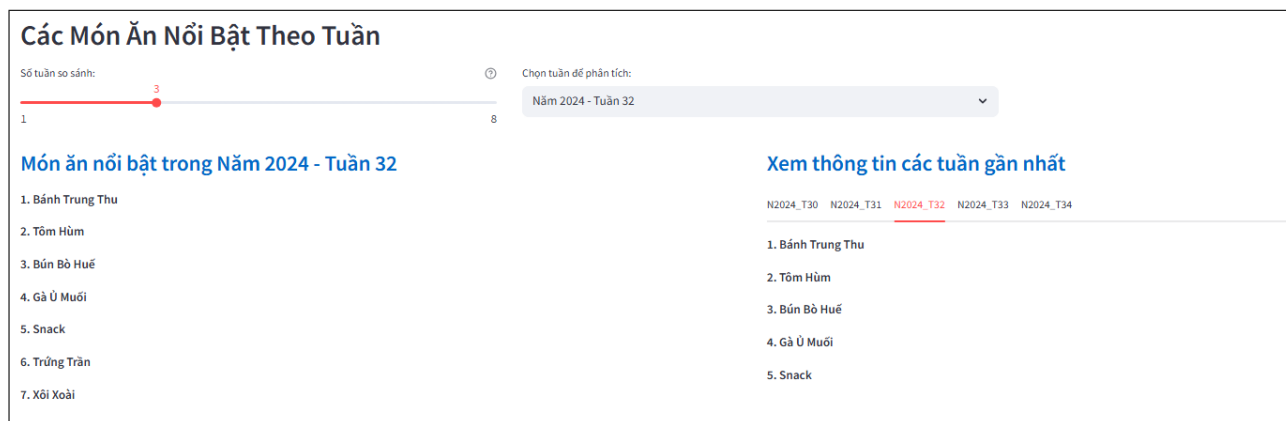
Để phân loại món ăn theo danh mục, thay vì sử dụng cách tiếp cận truyền thống với **dictionary** cố định, nhóm đã phát triển phương pháp phân loại linh hoạt hơn để giải quyết thách thức từ sự đa dạng và biến đổi liên tục của tên món ăn.

Phương pháp này ứng dụng kỹ thuật **explode** - một hàm cho phép tách các phần tử trong mảng thành các hàng riêng biệt trong DataFrame. Cụ thể:

1. Đầu tiên, tên các món ăn được tách thành các từ đơn.
2. Phương thức **explode** giúp biến đổi mỗi từ trong tên món ăn thành một hàng dữ liệu riêng.
3. Hệ thống sau đó tìm kiếm các cụm từ xuất hiện từ 2 lần trở lên trong tập dữ liệu.
4. Các cụm từ phổ biến này được sử dụng làm cơ sở để tự động tạo và nhóm các danh mục món ăn.

Cách tiếp cận này mang tính thích ứng cao, tự động phát hiện các mẫu hình trong tên món ăn và tạo ra hệ thống phân loại có khả năng mở rộng để đáp ứng với các món ăn mới, độc lạ mà không cần cập nhật thủ công các quy tắc phân loại, đảm bảo hiệu quả trong việc xử lý dữ liệu ẩm thực đa dạng và liên tục thay đổi theo thời gian.

8.4.2 Phân tích món ăn Nổi bật qua từng tuần



Hình 39: Món ăn nổi bật theo từng Tuần

Do các món ăn trong tập dữ liệu đã qua quá trình chọn lọc, đa phần đều có tỷ lệ tương tác cao và xuất hiện trong các video phổ biến, nên mục tiêu của thuật toán là phát hiện những món ăn **mới** và **đang thịnh hành** qua từng tuần cụ thể. Dựa trên nền tảng đó, nhóm đã thiết kế thuật toán phân tích dữ liệu ẩm thực theo thời gian nhằm xác định các món ăn đặc trưng cho từng tuần dựa trên phương pháp thống kê thích ứng.

Phương pháp này cho phép nhận diện các xu hướng ẩm thực nổi bật qua từng giai đoạn thời gian dựa trên các thông số: **tần suất xuất hiện trong tuần hiện tại, tỷ lệ xuất hiện so với các tuần trước đó, và tổng lượng đề cập.**

Quy trình thực hiện: Thuật toán được triển khai theo quy trình sau:

- Loại bỏ món ăn phổ biến kéo dài:** Xác định và loại trừ các món ăn xuất hiện trong hơn 60% tổng số tuần trong tập dữ liệu (70 tuần). Các món này được xem là phổ biến nhưng không phải xu hướng mới.
- Phân tích đặc trưng theo từng tuần:** Xử lý theo hai trường hợp chính:
 - Trường hợp tuần đầu tiên:** Khi không có dữ liệu để so sánh, thuật toán lấy 10 món ăn được đề cập nhiều nhất làm món đặc trưng.
 - Trường hợp các tuần tiếp theo:** So sánh dữ liệu với n tuần trước đó (người dùng có thể điều chỉnh n từ 1 đến 8 tuần). Sau đó áp dụng ba ngưỡng thích ứng:
 - Ngưỡng 1:** ≥ 3 lần xuất hiện, chiếm $\geq 60\%$ tổng số lần xuất hiện.
 - Ngưỡng 2:** ≥ 2 lần xuất hiện, chiếm $\geq 50\%$ tổng số lần xuất hiện.

– **Ngưỡng 3:** ≥ 1 lần xuất hiện, chiếm $\geq 40\%$ tổng số lần xuất hiện.

3. Nếu thu được đủ 5 món ăn đặc trưng tại một ngưỡng, thuật toán dừng xử lý. Nếu không, tiếp tục hạ ngưỡng để đảm bảo mỗi tuần đều có các món ăn đặc trưng được xác định.
4. Nếu không có món ăn nào đạt bất kỳ ngưỡng nào, thuật toán chọn 5 món có tần suất xuất hiện cao nhất trong tuần hiện tại.

Ưu điểm của phương pháp:

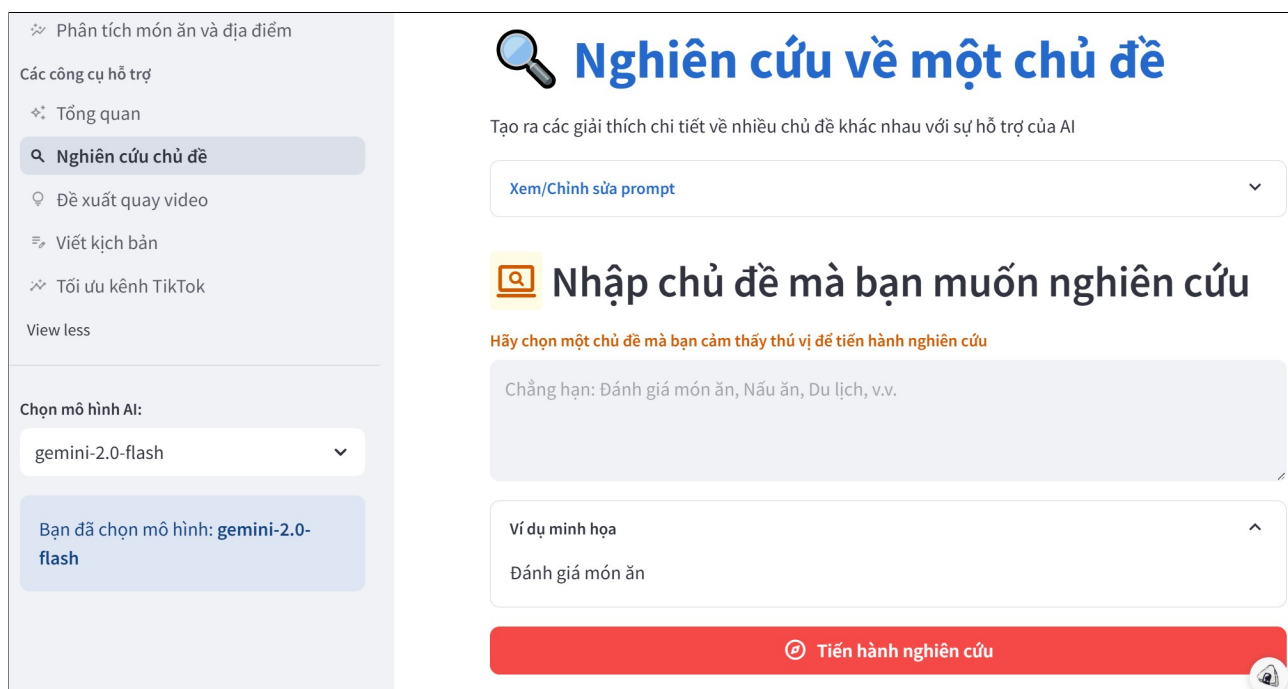
- Tự động điều chỉnh các ngưỡng phân tích phù hợp với đặc điểm phân phối dữ liệu, thích ứng với sự thay đổi của xu hướng ẩm thực theo thời gian.
- Tách biệt các món ăn phổ biến thường xuyên xuất hiện, tập trung vào các món mới nổi.
- Vẫn xác định được món đặc trưng ngay cả khi dữ liệu thiếu hoặc không đồng đều, đảm bảo tính liên tục trong phân tích.

9 CHƯƠNG 9: MỘT VÀI CÔNG CỤ BỔ TRỢ CHO VIỆC PHÁT TRIỂN KÊNH TIKTOK

9.1 Giới thiệu chung

Bên cạnh công cụ cốt lõi là **công cụ hỗ trợ viết kịch bản cho video TikTok** dựa trên phân tích dữ liệu (đã trình bày ở các chương trước), nhóm nhận thấy nhu cầu của các nhà sáng tạo nội dung còn mở rộng ra các giai đoạn khác trong quá trình sản xuất video. Để đáp ứng nhu cầu này và cung cấp một bộ giải pháp toàn diện hơn, nhóm đã phát triển thêm hai công cụ bổ trợ: **công cụ hỗ trợ nghiên cứu chủ đề** và **công cụ gợi ý cách quay video âm thực**.

9.2 Công cụ hỗ trợ nghiên cứu chủ đề



Hình 40: Giao diện chính của công cụ hỗ trợ nghiên cứu chủ đề

9.2.1 Tổng quan

Trong quá trình phát triển kênh TikTok, việc khám phá và thử nghiệm các chủ đề nội dung mới là yếu tố quan trọng để thu hút và giữ chân khán giả. Tuy nhiên, đối với những chủ đề mà nhà sáng tạo nội dung chưa có nhiều kinh nghiệm hoặc kiến thức nền tảng, giai đoạn nghiên cứu ban đầu có thể tốn nhiều thời gian và công sức. Để giải quyết vấn đề này, bên cạnh công cụ hỗ trợ viết kịch bản dựa trên phân tích dữ liệu TikTok, nhóm đã phát triển một công cụ bổ

trợ độc lập: **Công cụ Hỗ trợ Nghiên cứu Chủ đề.**

Công cụ này được thiết kế như một **trợ lý nghiên cứu ảo**, giúp người dùng nhanh chóng tổng hợp và cấu trúc thông tin về một chủ đề bất kỳ mà họ quan tâm. Điểm khác biệt chính so với công cụ hỗ trợ viết kịch bản là công cụ này **không dựa trên việc phân tích các đặc trưng dữ liệu đã được rút trích từ video TikTok**, mà thay vào đó, nó khai thác sức mạnh của các Mô hình Ngôn ngữ Lớn (Large Language Models - LLMs) thông qua **Gemini API** của Google. Phần này sẽ trình bày chi tiết về mục tiêu, kiến trúc, các kỹ thuật chính được sử dụng và quy trình hoạt động của công cụ này.

9.2.2 Mục tiêu và Lợi ích

Mục tiêu chính của công cụ này là cung cấp cho người dùng một phương tiện hiệu quả để:

1. **Nghiên cứu nhanh chóng:** Thu thập và tổng hợp thông tin tổng quan về một chủ đề một cách nhanh chóng, thay vì phải tìm kiếm thủ công qua nhiều nguồn.
2. **Hiểu biết có cấu trúc:** Thông tin trả về được trình bày một cách logic, có cấu trúc theo các khía cạnh quan trọng (tổng quan, điểm chính, ví dụ, thách thức, xu hướng, v.v.), giúp người dùng dễ dàng nắm bắt và ghi nhớ.
3. **Tiết kiệm thời gian và công sức:** Giảm thiểu đáng kể thời gian dành cho việc nghiên cứu sơ bộ, cho phép người dùng tập trung nhiều hơn vào việc lên ý tưởng và sáng tạo nội dung.
4. **Khám phá chủ đề mới:** Tạo điều kiện thuận lợi cho việc tìm hiểu các lĩnh vực mới mà người dùng chưa có nhiều kinh nghiệm.

9.2.3 Kiến trúc và Công nghệ sử dụng

Công cụ được xây dựng dưới dạng một ứng dụng web tương tác, sử dụng các công nghệ và kỹ thuật chính sau:

1. **Framework ứng dụng web:** Streamlit được lựa chọn làm framework để xây dựng giao diện người dùng (UI) một cách nhanh chóng và hiệu quả. Streamlit cho phép tạo các thành phần tương tác như nút bấm, hộp chọn, vùng nhập văn bản và hiển thị kết quả động chỉ với mã Python.
2. **Mô hình Ngôn ngữ Lớn (LLM):** Công cụ tận dụng khả năng **xử lý ngôn ngữ tự nhiên, tổng hợp thông tin và tạo văn bản** của các mô hình Gemini do Google cung

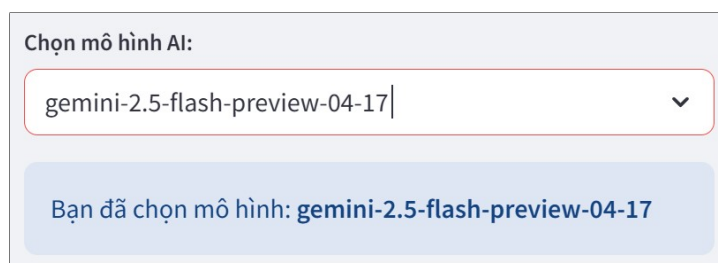
cấp thông qua **Gemini API**. Việc sử dụng LLM cho phép công cụ nghiên cứu về hầu hết mọi chủ đề dựa trên *kiến thức nền tảng rộng lớn và khả năng truy cập thông tin* của mô hình, thay vì bị giới hạn bởi dữ liệu TikTok đã thu thập.

3. **Thư viện tương tác với API:** Thư viện `google-genai` được sử dụng để gửi yêu cầu đến Gemini API và nhận kết quả trả về.
4. **Kỹ thuật prompting:** Đây là kỹ thuật cốt lõi để điều khiển LLM tạo ra kết quả mong muốn. Công cụ sử dụng một cấu trúc prompt được thiết kế cẩn thận để hướng dẫn mô hình AI cung cấp thông tin theo đúng định dạng và nội dung yêu cầu.
5. **Quản lý trạng thái: Streamlit Session State (`st.session_state`)** được dùng để lưu trữ trạng thái của ứng dụng, chẳng hạn như prompt hiện tại do người dùng chỉnh sửa và kết quả nghiên cứu gần nhất, đảm bảo trải nghiệm người dùng liền mạch.
6. **Caching:** Kỹ thuật caching (`@st.cache_data`) được áp dụng cho các hàm đọc file (danh sách mô hình, template prompt) để tăng tốc độ tải trang và giảm thiểu việc đọc lại file không cần thiết.

9.2.4 Thiết kế giao diện người dùng và Luồng tương tác

Giao diện người dùng: Được thiết kế đơn giản và trực quan, bao gồm các thành phần chính:

1. **Sidebar chọn mô hình:** Một menu thả xuống (`st.selectbox`) ở thanh bên (sidebar) cho phép người dùng chọn mô hình Gemini muốn sử dụng từ danh sách được đọc từ file `models/gemini_models.txt`. Thông tin về mô hình đã chọn được hiển thị ngay bên dưới thông qua `st.info` (xem Hình 41).



Hình 41: Giao diện sidebar chọn mô hình của công cụ hỗ trợ nghiên cứu chủ đề

2. **Khu vực tùy chỉnh prompt:** Sử dụng `st.expander` để tạo một khu vực có thể mở rộng/thu gọn (xem Hình 42), chứa:
 - Một vùng nhập văn bản lớn (`st.text_area`) hiển thị prompt hiện tại (mặc định hoặc đã tùy chỉnh) và cho phép người dùng chỉnh sửa.

- Hai nút bấm (`st.button`) được đặt trong hai cột (`st.columns`): "**Cập nhật prompt**" để lưu thay đổi và "**Khôi phục prompt mặc định**" để quay lại nội dung gốc từ file `user_prompt_template.md`.
- Thông báo thành công (`st.success`) được hiển thị khi cập nhật hoặc khôi phục prompt.

Xem/Chỉnh sửa prompt

Đây là hướng dẫn gửi đến mô hình AI. Bạn có thể sửa đổi nó để phù hợp hơn với nhu cầu của bạn.

Bạn có thể tùy chỉnh prompt bên dưới:

Phần giải thích về chủ đề đã chọn sẽ bao gồm các nội dung sau đây:

- Tổng quan: Giới thiệu về chủ đề và tầm quan trọng của nó.
- Các điểm chính: Liệt kê và giải thích các yếu tố quan trọng của chủ đề.

Cập nhật prompt **Khôi phục prompt mặc định**

✓ Đã khôi phục prompt mặc định!

Hình 42: Giao diện khu vực tùy chỉnh prompt của công cụ hỗ trợ nghiên cứu chủ đề

3. **Khu vực nhập chủ đề:** Một vùng nhập văn bản (`st.text_area`) để người dùng nhập chủ đề cần nghiên cứu. Có một ví dụ minh họa (`st.expander`) để hướng dẫn người dùng ngay bên dưới (xem Hình 43).
4. **Nút tạo kết quả:** Nút "**Tiến hành nghiên cứu**" (`st.button` với `type="primary"`) để tiến hành gửi yêu cầu đến API (xem Hình 43). Hệ thống sẽ kiểm tra xem người dùng đã nhập chủ đề hay chưa, nếu chưa sẽ hiển thị thông báo yêu cầu nhập chủ đề (`st.warning`).

Nhập chủ đề mà bạn muốn nghiên cứu

Hãy chọn một chủ đề mà bạn cảm thấy thú vị để tiến hành nghiên cứu

Chẳng hạn: Đánh giá món ăn, Nấu ăn, Du lịch, v.v.

Ví dụ minh họa

Đánh giá món ăn

Tiến hành nghiên cứu

⚡ Vui lòng nhập chủ đề để tiến hành nghiên cứu!

Hình 43: Giao diện khu vực nhập chủ đề của công cụ hỗ trợ nghiên cứu chủ đề

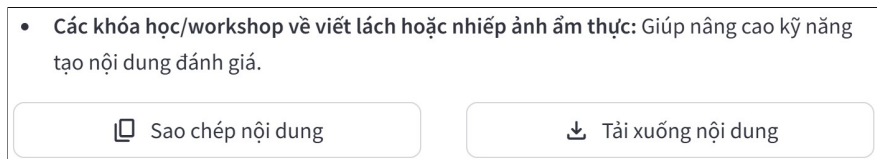
5. **Hiển thị kết quả:** Kết quả trả về từ API (sau khi chuẩn hóa) được hiển thị dưới dạng Markdown (`st.markdown`) trong khu vực chính (xem Hình 44). Khu vực này có thể cuộn để xem toàn bộ nội dung. Nội dung được phân chia thành các phần rõ ràng theo cấu trúc đã định nghĩa trong prompt, giúp người dùng dễ dàng theo dõi và hiểu thông tin.



Hình 44: Giao diện khu vực hiển thị kết quả của công cụ hỗ trợ nghiên cứu chủ đề

6. **Lưu kết quả:** Hai nút bấm (`st.button` và `st.download_button`) được đặt trong hai cột (`st.columns`) dưới phần kết quả (xem Hình 45), cho phép người dùng:

- Sao chép kết quả vào clipboard (sử dụng thư viện `pyperclip`).
- Tải kết quả xuống dưới dạng file Markdown (`.md`).



Hình 45: Giao diện khu vực lưu kết quả của công cụ hỗ trợ nghiên cứu chủ đề

Luồng tương tác của người dùng: Người dùng sẽ thực hiện các bước sau để sử dụng công cụ:

1. (Tùy chọn) Chọn mô hình AI từ sidebar.
2. (Tùy chọn) Mở rộng khu vực prompt, xem và chỉnh sửa nếu cần, sau đó nhấn "Cập nhật prompt" hoặc "Khôi phục prompt mặc định".
3. Nhập chủ đề muốn nghiên cứu vào ô văn bản.
4. Nhấn nút "Tiến hành nghiên cứu".
5. Chờ đợi trong khi hệ thống hiển thị trạng thái "Đang nghiên cứu..." (`st.spinner`).
6. Xem kết quả được hiển thị.
7. (Tùy chọn) Sao chép hoặc tải xuống kết quả.

9.2.5 Kỹ thuật Prompting

Hiệu quả của công cụ phụ thuộc rất lớn vào cách thiết kế prompt để giao tiếp với mô hình Gemini. Nhóm đã áp dụng phương pháp sử dụng kết hợp **System Prompt** và **User Prompt** để tối ưu hóa khả năng tạo nội dung của mô hình:

1. System Prompt (system_prompt_template.md):

- **Mục đích:** Định nghĩa vai trò và hướng dẫn tổng quát cho mô hình AI. Prompt này yêu cầu mô hình đóng vai là “*chuyên gia nghiên cứu với kiến thức sâu rộng*”.
- **Định dạng đầu ra:** Quan trọng nhất, system prompt yêu cầu mô hình trả lời theo định dạng Markdown với một cấu trúc gồm 7 phần rõ ràng: **Overview**, **Key Points**, **Examples**, **Challenges**, **Best Practices**, **Trends**, và **Resources**. Nó cũng nhấn mạnh các yêu cầu về tính rõ ràng, súc tích, chính xác và việc chỉ trả về nội dung bên trong khối Markdown.
- **Kỹ thuật:** Sử dụng kỹ thuật "**Role Playing**" (đóng vai) và "**Output Formatting**" (định dạng đầu ra) để kiểm soát hành vi và cấu trúc kết quả của LLM.

2. User Prompt (user_prompt_template.md và tùy chỉnh của người dùng):

- **Mục đích:** Cung cấp yêu cầu cụ thể hơn về nội dung cho từng phần trong cấu trúc đã định nghĩa ở system prompt. Prompt mặc định liệt kê lại các phần mong muốn và mô tả ngắn gọn nội dung của từng phần.
- **Tính linh hoạt:** Người dùng có thể tùy chỉnh user prompt này thông qua giao diện (`st.text_area`). Ví dụ, họ có thể yêu cầu tập trung sâu hơn vào phần "Challenges" hoặc bổ sung một phần mới như "Target Audience".
- **Kết hợp:** Khi gửi yêu cầu đến API, system prompt được gửi trước, theo sau là một câu dẫn (“*Hãy giúp tôi tạo ra một bài giải thích chi tiết về chủ đề [topic].*”) và cuối cùng là nội dung user prompt (mặc định hoặc đã tùy chỉnh).

3. Khôi phục Prompt: Chức năng khôi phục về prompt mặc định (default_user_prompt) đảm bảo người dùng luôn có thể quay lại trạng thái ban đầu nếu việc tùy chỉnh không như ý.

Kỹ thuật prompting này giúp đảm bảo tính nhất quán trong cấu trúc đầu ra, đồng thời mang lại sự linh hoạt cho người dùng để điều chỉnh nội dung theo nhu cầu nghiên cứu cụ thể.

9.2.6 Tương tác với Gemini API và Xử lý kết quả

1. **Lựa chọn mô hình:** Như đã đề cập, người dùng có thể chọn các mô hình Gemini khác nhau. Hàm `read_available_gemini_models` đọc danh sách các mô hình có sẵn từ file `models/gemini_models.txt` và hiển thị trong `st.selectbox`. Việc chọn mô hình nhanh hơn (`gemini-2.0-flash`, `gemini-2.5-flash`) sẽ cho kết quả nhanh hơn nhưng có thể kém chi tiết hơn so với mô hình mạnh hơn (`gemini-2.5-pro`).
2. **Gửi yêu cầu:** Khi người dùng nhấn nút "Tiến hành nghiên cứu", hàm `generate_content` (trong file `research.py`, gọi đến `client.models.generate_content` của thư viện `google-genai`) được thực thi. Hàm này nhận vào system prompt, user prompt (đã được định dạng cùng với chủ đề) và tên mô hình đã chọn. API key được sử dụng để xác thực yêu cầu. (*Quá trình này thường mất khoảng 15 giây.*)
3. **Nhận và Chuẩn hóa phản hồi:** Gemini API trả về một đối tượng response. Phần nội dung văn bản (`response.text`) thường được bao bọc trong khối mã Markdown (nội dung chính được bao bọc bởi các ký tự đánh dấu). Hàm `standardize_response` được sử dụng để loại bỏ các ký tự đánh dấu không cần thiết, chỉ giữ lại nội dung Markdown thuần túy.
4. **Hiển thị và Lưu trữ:** Kết quả Markdown đã chuẩn hóa được hiển thị trực tiếp trên giao diện bằng `st.markdown`. Đồng thời, kết quả này được lưu vào `st.session_state.last_research_response` để có thể sử dụng cho chức năng sao chép và tải xuống.
5. **Sao chép và Tải xuống:**
 - Nút "Sao chép nội dung" sử dụng thư viện `pyperclip` (`pyperclip.copy()`) để đưa nội dung trong `st.session_state.last_research_response` vào clipboard hệ thống (*tiện lợi cho việc dán vào tài liệu hoặc ứng dụng khác*).
 - Nút "Tải xuống nội dung" sử dụng `st.download_button`, truyền vào nội dung (có thể thêm tiêu đề "# Chủ đề: [topic]") và định dạng file là `research_topic.md` với kiểu MIME là `text/plain` (*phù hợp cho chỉnh sửa hoặc lưu trữ lâu dài*).


9.2.7 Minh họa cách sử dụng công cụ

Để minh họa cách công cụ hoạt động, nhóm đã quay video hướng dẫn sử dụng công cụ này, người dùng có thể tham khảo video demo tại [đây](#) (từ phút 4:10 đến 8:32). Trong ví dụ minh họa, người dùng muốn nghiên cứu về chủ đề "Đánh giá món ăn" (Food Review) để tạo video

TikTok. Ví dụ này cho thấy công cụ giúp người dùng nhanh chóng thu thập thông tin có cấu trúc, hỗ trợ quá trình sáng tạo nội dung.

9.3 Công cụ gợi ý cách quay video ẩm thực

Hệ thống gợi ý quay video chủ đề ẩm thực




Chọn loại video

Hãy chọn loại video mà bạn muốn thực hiện:

☒ Video review món ăn hoặc quán ăn

☐ Video mukbang

☐ Video hướng dẫn nấu ăn




Chọn mô hình AI

Chọn mô hình AI mà bạn muốn sử dụng:

gemini-2.5-pro-exp-03-25

Bạn đã chọn loại video: Video review món ăn hoặc quán ăn

Bạn đã chọn mô hình: gemini-2.5-pro-exp-03-25



Mô tả video của bạn

Hãy nhập mô tả chi tiết về video ẩm thực mà bạn muốn thực hiện:

Hình 46: Giao diện chính của công cụ gợi ý cách quay video ẩm thực

9.3.1 Tổng quan

Trong quy trình sáng tạo nội dung trên TikTok, sau giai đoạn lên ý tưởng và viết kịch bản, việc thực hiện các cảnh quay (filming) đóng vai trò then chốt, quyết định phần lớn chất lượng hình ảnh và cảm xúc của video thành phẩm. Đối với lĩnh vực ẩm thực, việc quay phim đòi hỏi sự chú ý đặc biệt đến các yếu tố như góc máy làm nổi bật món ăn, ánh sáng hấp dẫn, âm thanh sống động và cách thể hiện tương tác tự nhiên. Tuy nhiên, không phải nhà sáng tạo nội dung nào cũng có đủ kinh nghiệm hoặc kiến thức chuyên sâu về kỹ thuật quay phim cho từng thể loại video ẩm thực cụ thể.

Để giải quyết thách thức này và bổ sung vào bộ công cụ hỗ trợ phát triển kênh TikTok, nhóm đã xây dựng **Công cụ Gợi ý Cách quay Video Ẩm thực**. Công cụ này hoạt động như một **cố vấn sản xuất ảo**, cung cấp những lời khuyên và gợi ý kỹ thuật quay phim chi tiết, được cá nhân hóa dựa trên ý tưởng video và thể loại ẩm thực mà người dùng lựa chọn. Tương tự công cụ nghiên cứu chủ đề, công cụ này cũng khai thác sức mạnh của Mô hình Ngôn ngữ Lớn (LLM) tiên tiến thông qua **Gemini API**, tập trung vào kỹ thuật prompting để tạo ra các gợi ý chuyên sâu. Phần này sẽ đi sâu vào phân tích mục tiêu, kiến trúc, các kỹ thuật chính và quy trình hoạt động của công cụ gợi ý quay video.

9.3.2 Mục tiêu và Lợi ích

Công cụ được xây dựng với các mục tiêu cụ thể sau:

1. **Nâng cao chất lượng hình ảnh:** Cung cấp các gợi ý về góc quay, ánh sáng, bố cục để giúp video trông chuyên nghiệp và hấp dẫn hơn về mặt thị giác.
2. **Tối ưu hóa kỹ thuật quay:** Đề xuất các kỹ thuật quay phù hợp (slow-motion, cận cảnh, góc nhìn thứ nhất, v.v.) cho từng thể loại và nội dung cụ thể.
3. **Cải thiện âm thanh:** Gợi ý cách thu âm, xử lý tạp âm và lựa chọn nhạc nền phù hợp để tăng trải nghiệm nghe của khán giả.
4. **Tăng tính hấp dẫn của nội dung:** Đưa ra lời khuyên về cách tương tác, kể chuyện và trình bày món ăn một cách lôi cuốn.
5. **Hỗ trợ theo thể loại:** Cung cấp gợi ý chuyên biệt cho các thể loại video ẩm thực phổ biến (review, nấu ăn, mukbang).
6. **Tiết kiệm thời gian chuẩn bị:** Giúp người dùng, đặc biệt là những người mới, có sự chuẩn bị tốt hơn cho buổi quay, giảm thiểu thời gian thử nghiệm và sai sót.

9.3.3 Kiến trúc và Công nghệ sử dụng

Công cụ gợi ý quay video được triển khai dưới dạng ứng dụng web, sử dụng các thành phần công nghệ tương tự như công cụ nghiên cứu chủ đề:

1. **Framework ứng dụng web:** Streamlit tiếp tục được sử dụng để xây dựng giao diện người dùng thân thiện và các thành phần tương tác (tham khảo file `suggestion.py`).
2. **Mô hình Ngôn ngữ Lớn (LLM):** Công cụ dựa vào khả năng hiểu ngữ cảnh, kiến thức chuyên môn (được mô phỏng qua prompt) và khả năng tạo văn bản của các mô hình Gemini API để đưa ra các gợi ý quay phim.
3. **Thư viện tương tác với API:** Thư viện `google-genai` được dùng để giao tiếp với Gemini API.
4. **Kỹ thuật Prompting theo ngữ cảnh:** Điểm nhấn kỹ thuật của công cụ này là việc sử dụng các **prompt chuyên biệt** cho từng thể loại video ẩm thực, kết hợp với mô tả ý tưởng của người dùng để tạo ra gợi ý phù hợp nhất.

5. **Quản lý trạng thái: Streamlit Session State** (`st.session_state`) được dùng để lưu trữ gợi ý gần nhất được tạo ra, cho phép người dùng xem lại và thực hiện các thao tác lưu trữ.
6. **Caching:** Kỹ thuật caching (`@st.cache_data`) được áp dụng cho việc đọc danh sách mô hình và nội dung các file prompt template, tối ưu hóa hiệu năng ứng dụng.

9.3.4 Thiết kế giao diện người dùng và Luồng tương tác

Giao diện người dùng: Giao diện người dùng được thiết kế để người dùng dễ dàng nhập thông tin và nhận gợi ý:

1. Chọn loại video và mô hình AI: (Xem Hình 47)

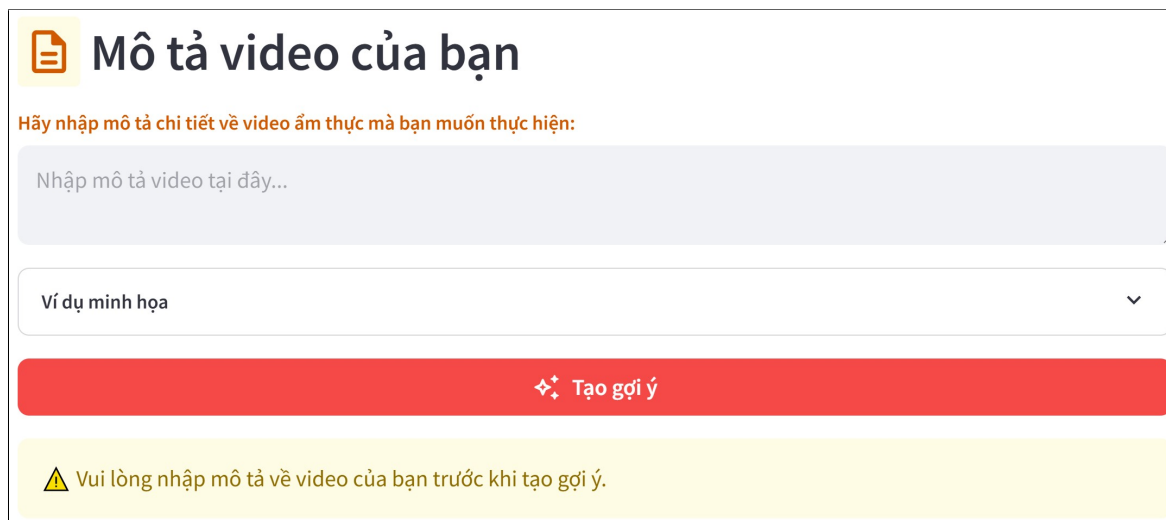
- Sử dụng hai cột (`st.columns`) để bố trí các lựa chọn.
- **Chọn loại video:** Người dùng chọn một trong ba loại video ẩm thực (*Video review món ăn hoặc quán ăn*, *Video mukbang*, *Video hướng dẫn nấu ăn*) thông qua nút `st.radio`. Danh sách các loại video được định nghĩa trong biến `VIDEO_TYPES`.
- **Chọn mô hình AI:** Người dùng chọn mô hình Gemini mong muốn từ danh sách (đọc từ file `models/gemini_models.txt`) bằng `st.selectbox`.
- Thông tin về lựa chọn của người dùng được hiển thị xác nhận bằng `st.info`.

Hình 47: Giao diện chọn loại video và mô hình AI của công cụ gợi ý cách quay video ẩm thực

2. Nhập mô tả video: (Xem Hình 48)

- Một vùng nhập văn bản lớn (`st.text_area`) cho phép người dùng mô tả chi tiết ý tưởng video của họ (món ăn, địa điểm, phong cách, mục tiêu, v.v.).
- Có một ví dụ minh họa (`st.expander`) để người dùng tham khảo cách mô tả hiệu quả.

3. **Nút tạo gợi ý:** Nút "Tạo gợi ý" (`st.button` với `type="primary"`) để bắt đầu quá trình tạo gợi ý. Hệ thống sẽ kiểm tra xem người dùng đã nhập mô tả hay chưa, nếu chưa sẽ hiển thị thông báo yêu cầu nhập mô tả thông qua `st.warning` (xem Hình 48).



Hình 48: Giao diện nhập mô tả video của công cụ gợi ý cách quay video ẩm thực

4. **Hiển thị gợi ý:** (Xem Hình 49)

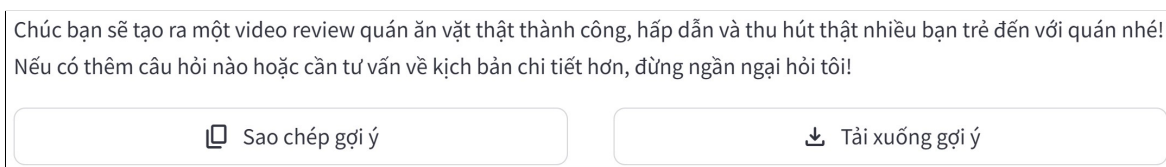
- Trong khi chờ đợi, `st.spinner` hiển thị thông báo "Đang tạo gợi ý..."
- Kết quả gợi ý (dạng Markdown) trả về từ API được lưu vào `st.session_state.suggestion` và hiển thị trên giao diện bằng `st.markdown`.



Hình 49: Giao diện hiển thị gợi ý của công cụ gợi ý cách quay video ẩm thực

5. **Lưu kết quả:** (Xem Hình 50) Tương tự công cụ nghiên cứu, hai nút bấm trong hai cột (`st.columns`) cho phép:

- Sao chép gợi ý vào clipboard (`pypyperclip.copy()`).
- Tải gợi ý xuống dưới dạng file Markdown (`.md`) bằng `st.download_button`.



Hình 50: Giao diện lưu kết quả của công cụ gợi ý cách quay video ẩm thực

Luồng tương tác của người dùng: Người dùng sẽ thực hiện các bước sau để sử dụng công cụ:

1. Chọn loại video ẩm thực muốn thực hiện.
2. (Tùy chọn) Chọn mô hình AI muốn sử dụng.
3. Nhập mô tả chi tiết về ý tưởng video.
4. Nhấn nút "Tạo gợi ý".
5. Xem các gợi ý chi tiết được hiển thị.
6. (Tùy chọn) Sao chép hoặc tải xuống các gợi ý.

9.3.5 Kỹ thuật Prompting chuyên biệt theo thể loại video

Đây là yếu tố kỹ thuật cốt lõi giúp công cụ đưa ra những gợi ý phù hợp và chuyên sâu. Thay vì dùng một prompt chung, công cụ sử dụng các prompt riêng biệt được thiết kế cho từng thể loại video ẩm thực:

1. **Cấu trúc Prompt:** Mỗi prompt template (lưu trong các file `.md` tương ứng: `food_review_template.md`, `cooking_template.md`, `mukbang_template.md`) đều có cấu trúc chung:
 - **Định nghĩa vai trò (Role Playing):** Yêu cầu LLM đóng vai là "chuyên gia sản xuất video chuyên nghiệp" với kinh nghiệm trong thể loại video tương ứng (review, nấu ăn, hoặc mukbang).
 - **Xác định đầu vào:** Nêu rõ rằng người dùng sẽ cung cấp thông tin/mô tả về video họ muốn làm.
 - **Yêu cầu đầu ra:** Liệt kê chi tiết các hạng mục cần gợi ý, được trình bày dưới dạng danh sách gạch đầu dòng với các tiêu đề in đậm (định dạng Markdown). Các hạng mục này bao gồm những khía cạnh quan trọng nhất của việc quay phim ẩm thực:
 - *Setup máy quay & Góc quay*

- *Ánh sáng & Âm thanh*
- *Giải thích & Hướng dẫn chi tiết* (cho video nấu ăn) hoặc *Chi tiết về phỏng vấn & giao tiếp* (cho video review) hoặc *Nêu cảm nhận & Tương tác với khán giả* (cho video mukbang)
- *Hiệu ứng & Chuyển cảnh*
- *Storytelling & CTA*
- *SEO & Thương hiệu* (cho video review)

- **Yêu cầu định dạng:** Yêu cầu trả lời bằng định dạng Markdown.

2. **Nội dung chuyên biệt:** Mặc dù có cấu trúc chung, nội dung yêu cầu chi tiết trong từng hạng mục của mỗi prompt được điều chỉnh để phù hợp với đặc thù của thể loại đó:

- **Review:** Nhấn mạnh vào góc quay đa dạng (toàn cảnh quán, cận cảnh món ăn), cách phỏng vấn chủ quán, xây dựng câu chuyện, tạo không khí đặc trưng của quán.
- **Nấu ăn:** Tập trung vào góc quay chi tiết các bước chế biến, ánh sáng làm nổi bật món ăn, giải thích rõ ràng, tạo không khí ấm cúng trong bếp.
- **Mukbang:** Chú trọng góc quay thể hiện cảm xúc khi ăn, âm thanh ASMR (nếu có), cách tương tác tự nhiên, tạo không khí vui vẻ, thân mật.

3. **Kết hợp prompt và mô tả từ người dùng:** Khi người dùng chọn loại video và nhập mô tả, hàm `generate_suggestion` (trong file `suggestion.py`) sẽ:

- Xác định file prompt template tương ứng dựa trên `video_type` (thông qua dictionary `VIDEO_TYPE_TO_PROMPT`).
- Đọc nội dung prompt từ file đó bằng hàm `read_prompt_file` (có sử dụng decorator `@st.cache_data`).
- Gửi nội dung prompt này (đóng vai trò system prompt/hướng dẫn) cùng với `user_description` (mô tả của người dùng) đến Gemini API.

Cách tiếp cận này cho phép LLM hiểu rõ ngữ cảnh (thể loại video) và yêu cầu cụ thể (mô tả của người dùng), từ đó đưa ra những gợi ý quay phim mang tính chuyên môn và sát với thực tế hơn.

9.3.6 Tương tác với Gemini API và Xử lý kết quả

Quy trình tương tác với API và xử lý kết quả tương tự như công cụ nghiên cứu chủ đề:

1. **Lựa chọn mô hình AI:** Người dùng chọn mô hình AI qua `st.selectbox`, mỗi mô hình có đặc điểm riêng về độ chi tiết và tốc độ phản hồi. Một số mô hình tiêu biểu bao gồm:

- **Gemini-2.5-pro:** Mô hình thử nghiệm tiên tiến nhất, được Google công bố vào tháng 3 năm 2025, cung cấp câu trả lời chi tiết nhưng có thể chậm hơn do tính chất thử nghiệm.
- **Gemini-2.0-flash:** Mô hình ổn định, được Google khuyến nghị cho các nhà phát triển, nổi bật với tốc độ phản hồi nhanh và chất lượng tốt trên nhiều tác vụ.
- **Gemini-2.5-flash:** Mô hình thử nghiệm mới, ra mắt vào giữa tháng 4 năm 2025, cân bằng giữa tốc độ của **Gemini-2.0-flash** và độ chi tiết của **Gemini-2.5-pro**.

2. **Gửi yêu cầu:** Hàm `generate_suggestion` gọi `client.models.generate_content` với prompt template phù hợp, mô tả của người dùng và tên mô hình đã chọn.

3. **Nhận và Hiển thị phản hồi:** Kết quả dạng text được trả về và hiển thị bằng `st.markdown`.

4. **Lưu trữ và Tương tác:** Kết quả được lưu vào `st.session_state` để hỗ trợ sao chép (`pypyperclip.copy`) và tải xuống (`st.download_button` với tên file `video_suggestion.md`).

9.3.7 Minh họa cách sử dụng công cụ

Để minh họa cách công cụ hoạt động, nhóm đã quay video hướng dẫn sử dụng công cụ này, người dùng có thể tham khảo video demo tại [đây](#) (từ phút 8:32 đến 11:04). Trong ví dụ minh họa, người dùng muốn quay video review một quán ăn vặt dành cho học sinh sinh viên. Công cụ gợi ý cách quay video sẽ giúp người dùng có được những gợi ý chi tiết về cách quay, ánh sáng, âm thanh và cách tương tác với chủ quán để tạo ra một video hấp dẫn và chất lượng.

10 CHƯƠNG 10: BÀN LUẬN VÀ KẾT LUẬN

10.1 Bàn luận

Đồ án “**Phân tích dữ liệu TikTok và Xây dựng công cụ hỗ trợ viết kịch bản dành cho các video TikTok**” đã được thực hiện với mục tiêu ứng dụng khoa học dữ liệu và trí tuệ nhân tạo để hỗ trợ các nhà sáng tạo nội dung, đặc biệt trong lĩnh vực ẩm thực. Qua quá trình thực hiện, nhóm đã đạt được những kết quả đáng kể và rút ra nhiều bài học kinh nghiệm giá trị.

10.1.1 Tóm tắt các kết quả đạt được

- Hoàn thiện Quy trình Khoa học Dữ liệu:** Nhóm đã thực hiện thành công một quy trình khoa học dữ liệu bài bản, bao gồm các giai đoạn: thu thập dữ liệu từ TikTok, tiền xử lý và làm sạch dữ liệu, rút trích các đặc trưng quan trọng (như hashtag, thông tin thời gian, nội dung audio, món ăn, địa điểm), phân tích khám phá dữ liệu thông qua dashboard trực quan, và cuối cùng là xây dựng các công cụ hỗ trợ. Quá trình này không chỉ tạo ra một tập dữ liệu có chất lượng về video ẩm thực trên TikTok mà còn giúp nhóm hiểu sâu hơn về đặc điểm và xu hướng của loại nội dung này.
- Xây dựng Công cụ Hỗ trợ Viết Kịch bản:** Công cụ cốt lõi của đồ án đã được phát triển thành công, có khả năng tự động tạo ra gợi ý kịch bản chi tiết cho video ẩm thực dựa trên các đặc trưng dữ liệu đã phân tích và mô tả ý tưởng do người dùng cung cấp. Công cụ này hứa hẹn sẽ là trợ thủ đắc lực, giúp giảm thiểu thời gian và công sức trong giai đoạn lên ý tưởng và xây dựng cấu trúc kịch bản.
- Phát triển Công cụ Bổ trợ:** Nhận thấy nhu cầu đa dạng của người dùng, nhóm đã phát triển thêm hai công cụ bổ trợ hữu ích:
 - Công cụ Hỗ trợ Nghiên cứu Chủ đề:* Sử dụng sức mạnh của Gemini API, công cụ này giúp người dùng nhanh chóng tìm hiểu và tổng hợp thông tin về các chủ đề mới lạ, cung cấp một cái nhìn tổng quan có cấu trúc.
 - Công cụ Gợi ý Cách quay Video Ẩm thực:* Cũng dựa trên Gemini API và các prompt chuyên biệt, công cụ này đưa ra những lời khuyên kỹ thuật và sáng tạo về quay phim cho ba thể loại video ẩm thực phổ biến (review, nấu ăn, mukbang).

4. **Bộ Giải pháp Toàn diện:** Ba công cụ trên kết hợp lại tạo thành một bộ giải pháp tương đối toàn diện, hỗ trợ người dùng từ giai đoạn nghiên cứu, lên ý tưởng, viết kịch bản cho đến chuẩn bị quay phim, giúp họ tự tin hơn trong việc tạo ra các video TikTok chất lượng cao, ngay cả khi chưa có nhiều kinh nghiệm.

10.1.2 Các phát hiện chính từ phân tích dữ liệu TikTok

Quá trình phân tích dữ liệu, chủ yếu thông qua các dashboard trực quan, đã mang lại nhiều **thông tin hữu ích** về các yếu tố ảnh hưởng đến sự thành công của video ẩm thực trên TikTok. Các phát hiện này (được tổng hợp chi tiết trong file `insights.md` và trình bày trên trang cuối cùng của sản phẩm) có thể tóm tắt như sau:

- **Sức hút của Chủ đề Ẩm thực:** Dữ liệu cho thấy ẩm thực vẫn là một chủ đề "nóng" và có xu hướng tăng trưởng về số lượng video qua các năm, khẳng định tiềm năng lớn cho các nhà sáng tạo nội dung trong lĩnh vực này.
- **Chất lượng quan trọng hơn Số lượng:** Phân tích tương quan giữa tần suất đăng bài và mức độ tương tác cho thấy việc đầu tư vào chất lượng của từng video (nội dung, hình ảnh, âm thanh) thường mang lại hiệu quả tương tác trung bình cao hơn so với việc chỉ tập trung tăng số lượng video.
- **Tối ưu Hashtag:** Số lượng hashtag tối ưu nên dao động từ 4-7 thẻ cho mỗi video. Việc sử dụng quá nhiều hashtag có thể làm giảm hiệu quả. Hashtag cần phù hợp với nội dung và việc đưa tên tài khoản vào hashtag là một chiến lược phổ biến của các kênh lớn. Người dùng mới hoặc có ít người theo dõi dường như phụ thuộc nhiều hơn vào hashtag để tăng khả năng hiển thị.
- **Tần suất và Thời điểm vàng:** Duy trì tần suất đăng bài đều đặn (3-4 video/tuần) là cần thiết. Chủ Nhật là ngày đăng tiềm năng do ít cạnh tranh hơn và nhu cầu giải trí cao. Các khung giờ "vàng" (11h-13h và 17h-19h) thường mang lại lượt xem tốt hơn. Thứ 7 có xu hướng là ngày có tương tác thấp nhất.
- **Thời lượng video:** Các kênh có lượng người theo dõi lớn hơn thường đăng các video có thời lượng dài hơn, có thể phản ánh sự đầu tư nội dung chuyên sâu và khả năng giữ chân khán giả tốt hơn.
- **Xu hướng theo mùa:** Lượng video tăng đột biến vào dịp Giáng sinh/Tết Dương lịch và

giảm mạnh vào tuần Tết Nguyên Đán, sau đó tăng trở lại. Điều này cho thấy sự ảnh hưởng của các kỳ nghỉ lễ và văn hóa đến hành vi sáng tạo nội dung.

- **Về mặt địa lý:** Hà Nội và TP. Hồ Chí Minh là hai trung tâm sản xuất nội dung ẩm thực lớn nhất. Miền Trung, mặc dù ít được đề cập hơn, nhưng lại cho thấy tiềm năng ở các thị trường ngách tại các thành phố du lịch (Đà Lạt, Nha Trang, Huế, Đà Nẵng), nơi sự cạnh tranh có thể thấp hơn.

10.1.3 Ý nghĩa và Đóng góp

Các kết quả đạt được của đồ án mang lại những ý nghĩa và đóng góp thiết thực:

- **Đối với nhà sáng tạo nội dung:** Cung cấp một bộ công cụ hữu ích giúp đơn giản hóa và tối ưu hóa quy trình sản xuất video TikTok ẩm thực, từ nghiên cứu, lên ý tưởng, viết kịch bản đến chuẩn bị quay phim. Đồng thời, các insights từ dữ liệu giúp họ hiểu rõ hơn các yếu tố then chốt để cải thiện hiệu suất kênh.
- **Đối với cộng đồng nghiên cứu:** Đồ án là một ví dụ về việc ứng dụng khoa học dữ liệu và AI vào lĩnh vực sáng tạo nội dung số, một lĩnh vực đang phát triển nhanh chóng. Các phương pháp thu thập, xử lý dữ liệu, rút trích đặc trưng và xây dựng ứng dụng có thể được tham khảo và phát triển thêm.
- **Đối với nhóm thực hiện:** Quá trình thực hiện đồ án giúp nhóm củng cố kiến thức về khoa học dữ liệu, kỹ thuật phần mềm, làm việc với API, xây dựng ứng dụng web và kỹ năng làm việc nhóm.

10.2 Kết luận và Hướng phát triển tương lai

10.2.1 Kết luận

Đồ án đã hoàn thành các mục tiêu đề ra: phân tích thành công dữ liệu video TikTok ẩm thực để rút ra các insights giá trị và xây dựng được một bộ ba công cụ hỗ trợ hiệu quả cho các nhà sáng tạo nội dung. Công cụ hỗ trợ viết kịch bản, cùng với hai công cụ bổ trợ về nghiên cứu chủ đề và gợi ý quay phim, tạo thành một hệ sinh thái hỗ trợ toàn diện, giúp người dùng nâng cao chất lượng và hiệu quả sản xuất video trên nền tảng TikTok. Các phát hiện từ dữ liệu cung cấp những định hướng chiến lược quan trọng cho việc phát triển kênh.

10.2.2 Hướng phát triển tương lai

Mặc dù đã đạt được những kết quả tích cực, sản phẩm vẫn còn nhiều tiềm năng để cải thiện và mở rộng trong tương lai. Nhóm đề xuất một số hướng phát triển chính như sau:

1. Mở rộng Phạm vi Chủ đề:

- Hiện tại, công cụ cốt lõi và công cụ gợi ý quay phim chủ yếu tập trung vào lĩnh vực ẩm thực. Hướng phát triển quan trọng là **mở rộng hỗ trợ sang các thể loại video phổ biến khác** trên TikTok như du lịch, thời trang, công nghệ, giáo dục, làm đẹp, v.v..
- Điều này đòi hỏi việc thu thập và phân tích dữ liệu đặc thù cho từng lĩnh vực, cũng như điều chỉnh các mô hình và prompt để phù hợp với yêu cầu của từng loại nội dung. Việc mở rộng này sẽ giúp sản phẩm tiếp cận được đối tượng người dùng rộng lớn hơn.

2. Tự động hóa Quy trình Dữ liệu với Airflow:

- Quy trình thu thập, tiền xử lý và rút trích đặc trưng dữ liệu hiện tại còn thực hiện thủ công hoặc bán tự động qua các script và notebook. Việc **tích hợp Apache Airflow** hoặc các công cụ điều phối quy trình (workflow orchestration) tương tự sẽ giúp **tự động hóa hoàn toàn pipeline dữ liệu**.
- Airflow cho phép lập lịch, giám sát và quản lý các tác vụ xử lý dữ liệu một cách hiệu quả, đảm bảo dữ liệu luôn được cập nhật, giảm thiểu lỗi thủ công và tiết kiệm đáng kể thời gian, công sức cho nhóm phát triển trong việc duy trì và cập nhật hệ thống.

3. Tích hợp Đa dạng Mô hình AI:

- Tất cả công cụ trong sản phẩm (kể cả việc tự động tạo nhận xét từ biểu đồ) **đều dựa vào mô hình AI của Google Gemini API**. Để tăng tính linh hoạt và đa dạng cho người dùng, nhóm có thể **tích hợp thêm các mô hình AI từ các nhà cung cấp khác** (ví dụ: OpenAI GPT, Claude, Llama, v.v.) hoặc các mô hình mã nguồn mở tiên tiến.
- Việc này cho phép người dùng lựa chọn mô hình phù hợp nhất với nhu cầu cụ thể về **chất lượng, tốc độ, hoặc phong cách** của kết quả đầu ra. Đồng thời, nó cũng giúp giảm sự phụ thuộc vào một nhà cung cấp API duy nhất.

Ngoài ra, các cải tiến khác có thể bao gồm việc tối ưu hóa hiệu năng của các công cụ, cải thiện giao diện người dùng dựa trên phản hồi thực tế, và tích hợp thêm các tính năng phân tích nâng cao (ví dụ: phân tích cảm xúc bình luận, dự đoán hiệu suất video).

Tóm lại, đề án đã đặt nền móng vững chắc cho một hệ thống hỗ trợ sáng tạo nội dung TikTok. Với những hướng phát triển tiềm năng được đề xuất, sản phẩm hoàn toàn có khả năng trở thành một công cụ mạnh mẽ và toàn diện hơn, đóng góp tích cực vào sự phát triển của cộng đồng nhà sáng tạo nội dung tại Việt Nam.

Tài liệu tham khảo

- [1] David Teather. *TikTokAPI*. Version 7.1.0. 2025. URL: <https://github.com/davidteather/tiktok-api>.
- [2] Wes McKinney. “Data Structures for Statistical Computing in Python”. In: *Proceedings of the 9th Python in Science Conference*. Ed. by Stéfan van der Walt and Jarrod Millman. 2010, pp. 56–61. DOI: [10.25080/Majora-92bf1922-00a](https://doi.org/10.25080/Majora-92bf1922-00a).
- [3] Charles R. Harris et al. “Array programming with NumPy”. In: *Nature* 585.7825 (Sept. 2020), pp. 357–362. DOI: [10.1038/s41586-020-2649-2](https://doi.org/10.1038/s41586-020-2649-2). URL: <https://doi.org/10.1038/s41586-020-2649-2>.
- [4] J. D. Hunter. “Matplotlib: A 2D graphics environment”. In: *Computing in Science & Engineering* 9.3 (2007), pp. 90–95. DOI: [10.1109/MCSE.2007.55](https://doi.org/10.1109/MCSE.2007.55).
- [5] Michael L. Waskom. “seaborn: statistical data visualization”. In: *Journal of Open Source Software* 6.60 (2021), p. 3021. DOI: [10.21105/joss.03021](https://doi.org/10.21105/joss.03021). URL: <https://doi.org/10.21105/joss.03021>.
- [6] Thomas Kluyver et al. “Jupyter Notebooks – a publishing format for reproducible computational workflows”. In: *Positioning and Power in Academic Publishing: Players, Agents and Agendas*. Ed. by F. Loizides and B. Schmidt. IOS Press. 2016, pp. 87–90.
- [7] Deepak Vohra. “Apache Parquet”. In: Sept. 2016, pp. 325–335. ISBN: 978-1-4842-2198-3. DOI: [10.1007/978-1-4842-2199-0_8](https://doi.org/10.1007/978-1-4842-2199-0_8).
- [8] Kehinde Abe. *Downloading and Converting YouTube Videos to MP3 using yt-dlp in Python*. 2024. URL: https://dev.to/_ken0x/downloading-and-converting-youtube-videos-to-mp3-using-yt-dlp-in-python-20c5 (visited on 05/08/2025).
- [9] yt-dlp. *yt-dlp*. 2012. URL: <https://github.com/yt-dlp/yt-dlp>.
- [10] Suramya Tomar. “Converting video formats with FFmpeg”. In: *Linux Journal* 2006.146 (2006), p. 10.
- [11] Plotly Technologies Inc. *Collaborative data science*. 2015. URL: <https://plot.ly>.
- [12] Snowflake Inc. *A faster way to build and share data apps*. URL: <https://streamlit.io/>.
- [13] Chanin Nantasenamat. *Building a dashboard in Python using Streamlit*. 2024. URL: <https://blog.streamlit.io/crafting-a-dashboard-app-in-python-using-streamlit/> (visited on 05/08/2025).

- [14] Snowflake Inc. *Prep and deploy your app on Community Cloud*. URL: <https://docs.streamlit.io/deploy/streamlit-community-cloud/deploy-your-app>.