BỘ GIÁO DỤC VÀ ĐÀO TẠO

TRƯỜNG ĐẠI HỌC KINH TẾ - TÀI CHÍNH

THÀNH PHỐ HỒ CHÍ MINH

CÔNG TRÌNH DỰ THI ĐỀ TÀI

"NGHIÊN CỨU KHOA HỌC SINH VIÊN"

PHÂN TÍCH DỮ LIỆU VIDEO YOUTUBE

Sinh viên thực hiện

Nguyễn Tú Như - 225210604 Lớp: 22D1DAS-FIN01

Nguyễn Thúy Vy - 225210789 Khoa: Công nghệ thông tin

Nguyễn Thị Chúc Ngọc - 225210041

Thành phố Hồ Chí Minh, tháng 03 năm 2025

BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC KINH TẾ - TÀI CHÍNH
THÀNH PHỐ HỒ CHÍ MINH

CÔNG TRÌNH DỰ THI ĐỀ TÀI
"NGHIÊN CỨU KHOA HỌC SINH VIÊN"

PHÂN TÍCH DỮ LIỆU VIDEO YOUTUBE

Sinh viên thực hiện

Nguyễn Tú Như - 225210604 Lớp: 22D1DAS-FIN01

Nguyễn Thúy Vy - 225210789 Khoa: Công nghệ thông tin

Nguyễn Thị Chúc Ngọc - 225210041

Người hướng dẫn: ThS. Nguyễn Thị Hoài Linh

Thành phố Hồ Chí Minh, tháng 03 năm 2025

MỤC LỤC

MỤC LỤC **2**
DANH MỤC CÁC TỪ VIẾT TẮT **4**
DANH MỤC CÁC BẢNG **7**
DANH MỤC CÁC BIỂU ĐỒ, ĐỒ THỊ, SƠ ĐỒ, HÌNH ẢNH **8**
LỜI NÓI ĐẦU **12**
LỜI CAM ĐOAN **13**
TÓM TẮT **14**
CHƯƠNG 1: TỔNG QUAN **15**
1.1. Đặt vấn đề 15
1.2. Lý do chọn đề tài 16
1.3. Mục tiêu của đề tài 17
1.4. Nội dung nghiên cứu 17
1.5. Phương pháp luận và phương pháp nghiên cứu 19
1.6. Đối tượng và phạm vi nghiên cứu 19
1.7. Kết cấu 21
CHƯƠNG 2: TỔNG QUAN VỀ LĨNH VỰC NGHIÊN CỨU **22**
CHƯƠNG 3: BỘ DỮ LIỆU **24**
3.1. Nguồn dữ liệu 24
3.2. Quy trình thu thập dữ liệu 24
3.3. Các đặc trưng của dữ liệu 24
3.4. Xử lý dữ liệu 25
3.4.1. Xử lý dữ liệu thiếu 25
3.4.2. Xử lý giá trị trùng lặp 26
3.4.3. Tạo cột dữ liệu mới 26
3.4.4. Chuẩn hóa dữ liệu 29
3.4.5. Loại bỏ cột không cần thiết 30
3.4.6. Loại bỏ giá trị nhiễu (dựa trên clustering DBSCAN) 30
CHƯƠNG 4: PHƯƠNG PHÁP **36**
4.1 EDA (Exploratory Data Analysis) 36
4.1.1. Đơn biến 36
4.1.2. Hai biến 47
4.1.3. Đa biến 50
4.2. Mô hình 59
4.2.1. RANDOM FOREST 68
4.2.1.1.Chuẩn bị dữ liệu 68
4.2.1.2. Triển khai 68
4.2.2. KNN 73
4.2.2.1. Chuẩn bị dữ liệu 73

4.2.2.2. Triển khai..... 73
4.2.3. K-MEANS..... 77
4.2.3.1. Chuẩn bị dữ liệu..... 78
4.2.3.2. Triển khai..... 78
4.2.4. HỒI QUY TUYẾN TÍNH..... 81
4.2.4.1. Chuẩn bị dữ liệu..... 81
4.2.4.2. Triển khai..... 82
CHƯƠNG 5: KẾT QUẢ VÀ THẢO LUẬN.....86
CHƯƠNG 6: KẾT LUẬN - ĐỀ NGHỊ..... 91
6.1. Kết luận..... 91
6.2. Đề nghị..... 92
CHƯƠNG 7: TÀI LIỆU THAM KHẢO..... 94

DANH MỤC CÁC TỪ VIẾT TẮT

Từ viết tắt	Tiếng Anh	Tiếng Việt
NLP	Natural Language Processing	Xử lý ngôn ngữ tự nhiên
EDA	Exploratory Data Analysis	Phân tích dữ liệu khám phá
DBI	Davies-Bouldin Index	Chỉ số Davies-Bouldin
PCA	Principal Component Analysis	Phân tích thành phần chính
SMOTE	Synthetic Minority Over-sampling Technique	Kỹ thuật tăng mẫu thiểu số tổng hợp
t-SNE	t-Distributed Stochastic Neighbor Embedding	Nhúng hàng xóm ngẫu nhiên phân phối t
KNN	k-Nearest Neighbors	Thuật toán K láng giềng gần nhất
MAE	Mean Absolute Error	Sai số tuyệt đối trung bình
MSE	Mean Squared Error	Sai số bình phương trung bình
RMSE	Root Mean Squared Error	Căn bậc hai sai số bình phương trung bình
API	Application Programming Interface	Giao diện lập trình ứng dụng
AI	Artificial Intelligence	Trí tuệ nhân tạo
ML	Machine Learning	Máy học
ID	Identifier	Định danh
DBSCAN	Density-Based Spatial Clustering of Applications with Noise	Phân cụm dựa trên mật độ

BIC	Bayesian Information Criterion	Tiêu chí thông tin Bayes
KDE	Kernel Density Estimation	Ước lượng mật độ hạt nhân
JB	Jarque-Bera Test	Kiểm định Jarque-Bera
WCSS	Within-Cluster Sum of Squares	Tổng bình phương nội cụm
VIF	Variance Inflation Factor	Hệ số phóng đại phương sai
OLS	Ordinary Least Squares	Bình phương nhỏ nhất thông thường
AIC	Akaike Information Criterion	Các tiêu chí thông tin Akaike
SVM	Support Vector Machine	Máy Vector hỗ trợ

DANH MỤC CÁC BẢNG

Bảng 4.1: Đánh giá hiệu quả phân cụm……..……...77

DANH MỤC CÁC BIỂU ĐỒ, ĐỒ THỊ, SƠ ĐỒ, HÌNH ẢNH

Hình 3.1: Kết quả trả về sau khi xử lý giá trị trùng lặp..... 22
Hình 3.2: Xử lý giá trị trùng lặp..... 23
Hình 3.3: Phát hiện ngôn ngữ bình luận..... 23
Hình 3.5: Làm sạch bình luận..... 24
Hình 3.6: Bỏ stopwords..... 25
Hình 3.7: Áp dụng mô hình sentiment analysis từ TextBlob..... 25
Hình 3.8: Tính thời gian cách biệt..... 26
Hình 3.9: Tạo hàm convert_duration..... 27
Hình 3.10:Hiển thị eps theo phương thức Elbow..... 28
Hình 3.11:Lựa chọn eps trong DBSCAN..... 28
Hình 3.12:Trực quan 3D kết quả DBSCAN..... 30
Hình 3.13:Trực quan 2D kết quả DBSCAN..... 31
Hình 3.14: Phân nhóm DBSCAN..... 31
Hình 4.1: Tính các thông số thống kê cơ bản..... 33
Hình 4.2: Hàm trực quan hóa bằng biểu đồ histogram..... 33
Hình 4.3: Hàm trực quan hóa bằng biểu đồ box..... 33
Hình 4.4: Kết quả thống kê mô tả của biến view_count..... 34
Hình 4.5: Biểu đồ trực quan hóa của biến view_count..... 35
Hình 4.6: Biểu đồ box plot của biến view_count..... 35
Hình 4.7: Kết quả thống kê mô tả của biến comment_count..... 36
Hình 4.8: Biểu đồ trực quan hóa của biến comment_count..... 36
Hình 4.9: Biểu đồ box plot của biến comment_count..... 37
Hình 4.10: Thống kê mô tả của biến like_count..... 37
Hình 4.11: Biểu đồ trực quan hóa của biến like_count..... 38
Hình 4.12: Biểu đồ box plot của biến like_count..... 38
Hình 4.13: Kết quả thống kê mô tả của biến converted_duration..... 39
Hình 4.14: Biểu đồ trực quan hóa của biến converted_duration..... 39
Hình 4.15: Biểu đồ box plot của biến converted_duration..... 40
Hình 4.16: Kết quả thống kê mô tả của biến time difference..... 40
Hình 4.17: Biểu đồ trực quan hóa của biến time_difference..... 41
Hình 4.18: Biểu đồ box plot của biến time_difference..... 42
Hình 4.19: Biểu đồ đếm của biến caption..... 42
Hình 4.20: Biểu đồ đếm của biến category_group..... 43
Hình 4.21: biểu đồ tròn thể hiện 10 ngôn ngữ phổ biến ở bình luận..... 43
Hình 4.22: Biểu đồ scatter của biến comment count..... 44
Hình 4.23: Biểu đồ scatter của biến like count..... 45
Hình 4.24: Biểu đồ scatter của biến converted duration..... 46
Hình 4.25: Biểu đồ scatter của biến time_difference..... 47
Hình 4.26: Biểu đồ pairplot theo từng biến..... 48

Hình 4.27: Ma trận hệ số tương quan..... 50
Hình 4.28: Ma trận hiệp phương sai..... 52
Hình 4.29: Tổng phương sai giải thích..... 54
Hình 4.30: Tổng tỷ lệ phương sai giải thích..... 55
Hình 4.31: Hàm xác định siêu tham số cho mô hình Random Forest..... 66
Hình 4.32: Đường cong học tập của mô hình Random Forest..... 67
Hình 4.33: Kết quả cross-validation của mô hình Random Forest.....67
Hình 4.34:Các chỉ số đánh giá mô hình Random Forest..... 67
Hình 4.35: Báo cáo phân loại mô hình Random Forest.....68
Hình 4.36: Confusion Matrix của mô hình Random Forest..... 68
Hình 4.37: Tầm quan trọng các đặc trưng trong mô hình Random Forest..69
Hình 4.38: Tìm K tối ưu bằng MSE với 200 giá trị k.....70
Hình 4.39: Tìm K tối ưu bằng R-Squared với 200 giá trị k.....71
Hình 4.40: Tìm K tối ưu bằng MSE với k từ 1 đến 19..... 71
Hình 4.41: Tìm K tối ưu bằng R-Squared với k từ 1 đến 19..... 72
Hình 4.42: Thực tế vs. Dự đoán của KNN Hồi quy (k=15)..... 73
Hình 4.43: Phân phối tần suất phần dư.....74
Hình 4.44: Biểu đồ subplot của phương thức Elbow.....76
Hình 4.45: Biểu đồ subplot của phương thức Silhouette.....76
Hình 4.46: Biểu đồ 3D scatter thể hiện Kmeans theo t-SNE.....78
Hình 4.47: Biểu đồ thể hiện mô hình hồi quy tuyến tính của tập dữ liệu kiểm tra..... 80
Hình 4.48: Kết quả của các thông số đánh giá mô hình..... 81

LỜI NÓI ĐẦU

Trước tiên, chúng em xin chân thành gửi lời cảm ơn sâu sắc đến với giảng viên ThS Nguyễn Thị Hoài Linh đã tận tình hướng dẫn, chia sẻ kiến thức và kinh nghiệm quý báu trong suốt quá trình thực hiện công trình nghiên cứu này. Em cũng xin bày tỏ lòng biết ơn tới các thầy cô khoa Công nghệ Thông tin đã trang bị cho em nền tảng kiến thức vững chắc về Khoa học Dữ liệu, cùng toàn thể bạn bè đã luôn động viên và hỗ trợ em hoàn thành công trình.

Bắt đầu công trình nghiên cứu 'Phân tích dữ liệu video Youtube' mang đến những kiến thức thực tiễn trong bối cảnh YouTube đang trở thành nền tảng video thống trị với hơn 2.5 tỷ người dùng hoạt động hàng tháng (Google, 2023), việc phân tích dữ liệu từ nền tảng này mở ra nhiều cơ hội nghiên cứu thú vị. Đặc biệt, khi 70% lượt xem đến từ hệ thống đề xuất (YouTube Internal Data, 2023), bài toán dự đoán xu hướng và phân tích sentiment người dùng trở nên có ý nghĩa thực tiễn cao.

Nhận thức được tiềm năng ứng dụng to lớn của dữ liệu YouTube, nghiên cứu này tập trung vào hai bài toán quan trọng: (1) Dự đoán xu hướng nội dung và (2) Phân tích sentiment (cảm xúc) người dùng từ bình luận video. Với sự phát triển của các kỹ thuật học máy và xử lý ngôn ngữ tự nhiên (NLP), công trình không chỉ mang ý nghĩa học thuật mà còn có giá trị thực tiễn cao, hỗ trợ các doanh nghiệp tối ưu hóa chiến dịch tiếp thị, giúp nhà sáng tạo nội dung nắm bắt xu hướng, và góp phần xây dựng các giải pháp quản lý nội dung thông minh.

Do kinh nghiệm và kiến thức còn hạn chế, nghiên cứu không tránh khỏi những thiếu sót. Em rất mong nhận được sự góp ý từ quý thầy cô và các bạn để đề tài được hoàn thiện hơn.

Thành phố Hồ Chí Minh, ngày 30 tháng 03 năm 2025

Người thực hiện

Nguyễn Tú Như

Nguyễn Thúy Vy

Nguyễn Thị Chúc Ngọc

LỜI CAM ĐOAN

Chúng tôi xin cam đoan đây là công trình nghiên cứu khoa học độc lập của riêng chúng tôi và giảng viên hướng dẫn. Các số liệu sử dụng trong công trình nghiên cứu có nguồn gốc rõ ràng, đã công bố theo đúng quy định. Các kết quả nghiên cứu trong công trình nghiên cứu do chúng tôi tự tìm hiểu, phân tích một cách trung thực, khách quan và phù hợp với thực tiễn. Các kết quả này chưa từng được công bố trong bất kỳ nghiên cứu nào khác.

Nếu có bất kỳ vi phạm nào, chúng tôi hoàn toàn chịu trách nhiệm trước nhà trường và người hướng dẫn.

Thành phố Hồ Chí Minh, ngày 30 tháng 03 năm 2025

Người thực hiện

Nguyễn Tú Như

Nguyễn Thúy Vy

Nguyễn Thị Chúc Ngọc

TÓM TẮT

YouTube - nền tảng chia sẻ video trực tuyến lớn nhất thế giới nay đã tích lũy hàng tỷ người dùng. Phân tích dữ liệu video YouTube giúp hiểu rõ hơn về xu hướng nội dung, đánh giá mức độ tương tác của người xem giúp các nhà sáng tạo nội dung tạo ra video phù hợp hơn với thị hiếu khán giả, các doanh nghiệp khai thác dữ liệu để xây dựng chiến lược marketing hiệu quả.

Tuy nhiên, việc thu thập dữ liệu YouTube gặp phải nhiều thách thức, bao gồm giới hạn API và quyền riêng tư, sự đa dạng và phức tạp của dữ liệu, cũng như vấn đề dữ liệu không đầy đủ hoặc gây nhiễu. Đặc biệt, dữ liệu trên YouTube rất đa dạng, bao gồm cả dữ liệu có cấu trúc và dữ liệu phi cấu trúc, đòi hỏi các kỹ thuật phân tích nâng cao.

Bởi đó, phần lớn các nghiên cứu chỉ tập trung vào các chỉ số đơn giản như lượt xem và lượt thích mà chưa xem xét sâu sắc về cảm xúc, động cơ của người xem và các yếu tố ảnh hưởng đến hiệu suất video. Bằng cách tích hợp NLP và các kỹ thuật học máy, đề tài hướng đến việc cung cấp một cách tiếp cận toàn diện hơn để phân tích dữ liệu video. Quá trình nghiên cứu bao gồm các bước từ thu thập dữ liệu, tiền xử lý đến áp dụng các mô hình học máy để phân tích dữ liệu. Dữ liệu được thu thập từ API YouTube, tập trung vào các video thịnh hành để đảm bảo tính cập nhật và độ phổ biến của nội dung, trải qua các bước tiền xử lý để chuẩn bị cho việc áp dụng các mô hình máy học: Random Forest giúp tìm ra các đặc trưng quan trọng; K-means Clustering nhằm phân nhóm video theo đặc điểm chung; KNN giúp xác định mức độ liên quan của đặc trưng và lượt xem; hồi quy tuyến tính hỗ trợ phân tích các yếu tố tác động đến lượt xem và mức độ tương tác của video, từ đó đưa ra dự đoán chính xác hơn về hiệu suất video trong tương lai.

Như vậy, bằng cách áp dụng các phương pháp phân tích dữ liệu tiên tiến, nghiên cứu này mong muốn thúc đẩy sự phát triển của truyền thông và tiếp thị số trong việc hiểu rõ hành vi người dùng trên nền tảng YouTube.

CHƯƠNG 1: TỔNG QUAN

1.1. Đặt vấn đề

Youtube là một nền tảng chia sẻ video trực tuyến của Hoa Kỳ (US) cho phép người dùng tải lên, xem, chia sẻ, bình luận và đánh giá các video. là một kho tàng nội dung đa dạng và phong phú như: giải trí, giáo dục, tin tức, âm nhạc, phim ảnh và nhiều hơn thế nữa. Youtube đã tạo ra một làn sóng ảnh hưởng sâu rộng trong xã hội, định hình lại văn hóa đương đại, dẫn dắt các trào lưu trên mạng và biến nhiều người dùng bình thường thành những ngôi sao triệu phú.

Dữ liệu Youtube cung cấp thông tin và hành vi của người dùng thông qua lượt xem, lượt thích, lượt không thích và nhiều chỉ số khác. Ngoài ra, Youtube có thể thu thập thông tin cá nhân của người dùng như tuổi, giới tính, sở thích để đưa ra kết quả quảng cáo chiến lược. Tuy nhiên, dữ liệu cá nhân từ người dùng có thể không thực sự chính xác.

Các nhà sáng tạo nội dung và nhà quảng cáo có thể sử dụng những dữ liệu đã thu thập được để nhằm hiểu rõ hơn về khán giả của họ. Qua đó, cải thiện nội dung phù hợp với nhu cầu của người xem.

Phân tích dữ liệu video trên nền tảng Youtube nhằm giúp nhận biết những chủ đề, từ khóa và loại video đang được người xem quan tâm. Từ đó, xác định được nội dung đang thịnh hành trong một khoảng thời gian nhất định, tạo ra nội dung kịp thời cho phù hợp với xu hướng hiện có.

Tuy nhiên, việc thu thập và phân tích dữ liệu từ YouTube có những thách thức như sau:

Thứ nhất, API giới hạn về số lượng dữ liệu có thể truy cập và tần suất gọi API trong một khoảng thời gian nhất định, gây khó khăn cho việc thu thập dữ liệu quy mô lớn hoặc theo thời gian thực. Bên cạnh đó, các quy định về quyền riêng tư cũng làm hạn chế khả năng thu thập thông tin người dùng, nhằm đảm bảo tuân thủ các chính sách bảo mật và bảo vệ dữ liệu cá nhân.

Hơn nữa, dữ liệu trên YouTube rất đa dạng và phức tạp, bao gồm cả dữ liệu có cấu trúc như lượt xem, lượt thích, thời lượng video, và dữ liệu phi cấu trúc như tiêu đề, mô tả, bình luận của người dùng. Trong khi dữ liệu có cấu trúc dễ dàng thu thập và phân tích, thì dữ liệu phi cấu trúc lại đòi hỏi các kỹ thuật xử lý phức tạp hơn như xử lý ngôn ngữ tự nhiên (NLP), trích xuất thông tin và phân tích cảm xúc để hiểu rõ nội dung và hành vi người dùng.

Bên cạnh đó, API vẫn tồn tại nhiều hạn chế đáng kể. Một trong những vấn đề phổ biến nhất là giới hạn về tốc độ và hạn ngạch (rate limiting), khiến các ứng dụng cần xử lý dữ liệu lớn gặp khó khăn. Ngoài ra, việc phụ thuộc vào nhà cung cấp cũng là rủi ro lớn, vì bất kỳ thay đổi nào về chính sách hoặc dịch vụ đều có thể làm gián đoạn hệ thống. Bảo mật cũng là một thách thức, do API có thể trở thành điểm yếu nếu không được kiểm soát chặt chẽ, dẫn đến rò rỉ dữ liệu hoặc tấn công mạng. Bên cạnh đó, nhiều API có tài liệu không đầy đủ hoặc chi phí sử dụng cao, gây khó khăn cho các nhà phát triển, đặc biệt là cá nhân hoặc doanh nghiệp nhỏ.

Riêng với YouTube API, các hạn chế còn rõ rệt hơn. Google áp dụng quota system, khiến mỗi yêu cầu API đều tiêu tốn một lượng quota nhất định, và giới hạn mặc định 10.000 quota/ngày dễ bị cạn kiệt nếu ứng dụng có lưu lượng truy cập cao. Dữ liệu trả về cũng bị giới hạn, chẳng hạn như API tìm kiếm chỉ cho phép lấy tối đa 500 kết quả, trong khi API bình luận khó truy xuất toàn bộ nếu video có tương tác lớn. Quy trình xác thực phức tạp với OAuth 2.0 hoặc API key cũng gây khó khăn cho người mới. Đặc biệt, Google thường thay đổi chính sách mà không báo trước, như việc siết chặt API bình luận năm 2020, khiến nhiều ứng dụng bị ảnh hưởng. Ngoài ra, YouTube API còn thiếu một số tính năng quan trọng, như không hỗ trợ tải video lên trực tiếp hoặc cung cấp dữ liệu lượt xem real-time.

1.2. Lý do chọn đề tài

Trong kỷ nguyên mà nội dung video trực tuyến chiếm lĩnh không gian số, sự phân tích dữ liệu các nền tảng như Youtube trở nên ngày càng quan trọng. Tuy nhiên, hiện tại lĩnh vực này vẫn tồn tại những khoảng trống nghiên cứu đáng kể. Trong khi các nghiên cứu về phân tích dữ liệu văn bản, hình ảnh đã có những bước tiến lớn, phân tích dữ liệu video vẫn còn hạn chế. Phần lớn các phương pháp hiện tại chỉ tập trung vào các chỉ số cơ bản như lượt xem, lượt thích, bỏ qua việc phân tích sâu sắc về nội dung, cảm xúc và tương tác của người xem. Điều này dẫn đến việc bỏ lỡ những thông tin giá trị có thể giúp chúng ta hiểu rõ hơn về hành vi người dùng và xu hướng thị trường. Thêm vào đó, việc tích hợp các kỹ thuật máy học (Machine Learning) tiên tiến, như xử lý ngôn ngữ tự nhiên (NLP) để phân tích bình luận, phân tích cảm xúc và nhận diện đối tượng trong video, vẫn chưa được khai thác đầy đủ. Sự phát triển của các công nghệ này mở ra tiềm năng to lớn để đào sâu vào dữ liệu video, mang đến những ứng dụng thực tiễn trong nhiều lĩnh vực như truyền thông, tiếp thị và nghiên cứu xã hội. Các doanh nghiệp, nhà sáng tạo nội dung và nhà nghiên cứu đều nhận ra tầm quan trọng của việc thấu hiểu hiệu quả video, xu

hướng khán giả và tác động của nội dung video.

Sự kết hợp giữa sự phát triển của nền tảng YouTube, nhu cầu hiểu rõ khán giả, sự phát triển của công cụ phân tích và ứng dụng của AI đã dẫn đến tầm quan trọng của việc phân tích dữ liệu video trên YouTube, đặc biệt là truyền thông, tiếp thị và trí tuệ nhân tạo (AI). Trong lĩnh vực truyền thông và tiếp thị, phân tích dữ liệu video giúp doanh nghiệp hiểu rõ hơn về đối tượng mục tiêu, từ sở thích, hành vi đến phản ứng cảm xúc của họ đối với nội dung. Điều này cho phép các nhà tiếp thị tạo ra các chiến dịch quảng cáo được cá nhân hóa, tối ưu hóa nội dung video để tăng tương tác và đo lường chính xác hiệu quả của các chiến dịch. Phân tích dữ liệu video cũng giúp nhận diện các xu hướng mới, nắm bắt cơ hội thị trường và tạo ra lợi thế cạnh tranh. Trong lĩnh vực AI, dữ liệu video đóng vai trò là nguồn tài nguyên vô giá để huấn luyện các mô hình học máy. Từ nhận đối tượng, phân tích hành động đến nhiều ngôn ngữ tự nhiên trong video, việc phân tích dữ liệu video mở ra những tiềm năng ứng dụng rộng lớn. các hệ thống giám sát thông minh, trợ lý ảo và robot tự động đều phụ thuộc vào khả năng phân tích dữ liệu video để hoạt động hiệu quả. Hơn nữa, việc phân tích dữ liệu cũng góp phần vào sự phát triển công nghệ tiên tiến như xe tự lái và thực tế ảo.

1.3. Mục tiêu của đề tài

Mục tiêu của đề tài phân tích dữ liệu trên YouTube bao trùm nhiều khía cạnh, từ việc thấu hiểu hành vi người xem đến việc ứng dụng trong các lĩnh vực chuyên sâu và tìm hiểu hiệu suất của việc áp dụng các mô hình máy học vào thông tin cơ bản của video. Trước hết, việc giải mã dữ liệu video giúp vẽ nên bức tranh rõ nét về đối tượng mục tiêu, từ sở thích, hành vi đến phản ứng cảm xúc, thông qua các chỉ số như lượt xem, bình luận và thời lượng xem. Dữ liệu từ YouTube còn là nguồn tài nguyên quý giá cho các nhà nghiên cứu xã hội, tâm lý học và truyền thông, giúp khám phá các xu hướng xã hội và đánh giá tác động của truyền thông. Bên cạnh đó, trong lĩnh vực máy học (Machine Learning), việc phân tích dữ liệu video mở ra tiềm năng lớn cho việc huấn luyện các mô hình học máy, từ nhận diện đối tượng đến phân tích hành động. Cuối cùng, phân tích dữ liệu video còn đóng vai trò quan trọng trong việc đo lường hiệu quả quảng cáo, giúp các doanh nghiệp tối ưu hóa ngân sách và đạt được lợi tức đầu tư tối đa.

1.4. Nội dung nghiên cứu

Thu thập và tiền xử lý dữ liệu:

- Thu thập dữ liệu:
    + Sử dụng API của YouTube để thu thập dữ liệu từ các video.

+ Thu thập các trường dữ liệu khác nhau, bao gồm: ID video, tiêu đề, mô tả, thời lượng, thời gian đăng tải, thẻ tag, độ phân giải, ID danh mục, lượt xem, lượt thích, bình luận, ID kênh, tên kênh.

- Tiền xử lý dữ liệu:
  + Xử lý dữ liệu thiếu, trùng lặp;
  + Phân nhóm video dựa trên lượt xem;
  + Tạo cột dữ liệu mới:
    - Phân tích cảm xúc.
    - Liên kết các mã số định danh của danh mục video (*category_id*) với tên gọi tương ứng của chúng (*category group*).
    - Mã hóa dữ liệu, bao gồm: ID thể loại, độ phân giải, phụ đề.
    - Tạo nhãn cho số lượt xem.
    - Tính thời gian từ lúc phát hành video đến thời điểm thịnh hành.
  + Chuẩn hóa dữ liệu
  + Loại bỏ cột không cần thiết, bao gồm: tiêu đề, mô tả, thời gian đăng tải, ID kênh, tên kênh, ID thể loại, thẻ tag, thời lượng, độ phân giải.
  + Loại bỏ giá trị nhiễu (dựa trên clustering DBSCAN).
- Áp dụng máy học để phân tích (gồm 4 mô hình):

+ K-means Clustering: phân nhóm các video dựa trên chủ đề, độ dài để hiểu rõ hơn về cấu trúc nội dung của kênh, và phân tích các bình luận để tìm ra các nhóm chủ đề đang được quan tâm, từ đó phát hiện xu hướng mới.

+ Random Forest: phân loại video dựa trên nội dung hoặc mức độ tương tác. Điều này giúp đề xuất video cho người xem dựa trên những video họ đã xem và thích, phân loại các bình luận thành tích cực, tiêu cực, hoặc trung lập để hiểu rõ hơn về phản hồi của người xem, và dự đoán lượt xem cho các video mới dựa trên các video có đặc điểm tương đương.

+ KNN: Dự đoán các yếu tố ảnh hưởng đến hiệu suất video (lượt xem, thời lượng video), xác định các đặc điểm quan trọng của video thành công. Điều này giúp dự đoán số lượt xem, xác định những yếu tố nào ảnh hưởng nhiều nhất đến hiệu suất video (tiêu đề, thời lượng), và tìm ra các đặc điểm của các video có khả năng lan truyền mạnh mẽ.

+ Regression: Dự đoán các giá trị số liên tục, chẳng hạn như lượt xem, và phân tích các yếu tố ảnh hưởng đến hiệu suất video. Điều này giúp dự đoán lượt xem

sử dụng các biến như tổng số lượt tương tác, và xác định những yếu tố nào (like, comment, duration) ảnh hưởng nhiều nhất đến hiệu suất video.

- So sánh tính khả thi của 4 mô hình trên đối với dữ liệu video YouTube.

1.5. Phương pháp luận và phương pháp nghiên cứu

Phương pháp luận:

- Phương pháp nghiên cứu định lượng: sử dụng kỹ thuật thống kê để phân tích các chỉ số định lượng, áp dụng mô hình học máy để dự đoán và phân loại dữ liệu.
- Phương pháp nghiên cứu định tính: phân tích nội dung bình luận để hiểu rõ hơn về cảm xúc và ý kiến của người xem.

Phương pháp nghiên cứu:

- Phương pháp thu thập dữ liệu: Sử dụng dữ liệu YouTube API để truy xuất dữ liệu công khai về video.
- Phương pháp phân tích dữ liệu:
    + Thống kê mô tả: tính toán các chỉ số thống kê cơ bản (trung bình, lớn nhất, nhỏ nhất, độ lệch chuẩn, hiệp phương sai, hệ số tương quan) để mô tả dữ liệu.
    + Trực quan hóa dữ liệu:
        - Sử dụng các công cụ như Matplotlib, Seaborns để tạo ra các biểu đồ và đồ thị trực quan.
        - Trực quan hóa các xu hướng, mối quan hệ và phân phối dữ liệu.
    + Phân tích EDA: Sử dụng phân tích đơn biến, hai biến, đa biến để xác định mối quan hệ giữa các biến.
    + NLP (xử lý ngôn ngữ tự nhiên): phân tích cảm xúc trong bình luận để đánh giá phản hồi của người xem.
    + Học máy (Machine Learning):
        - K-means Clustering.
        - KNN (K-Nearest Neighbors).
        - Random Forest.
        - Regression.

1.6. Đối tượng và phạm vi nghiên cứu

Đối tượng nghiên cứu: mối tương quan giữa thông tin, tương tác cơ bản của các video YouTube và sự thịnh hành của các video đó

Phạm vi nghiên cứu: tập trung 1000 video thịnh hành nhất, ngày 16/4/2025 tại 5 quốc gia: Hoa Kỳ, Canada, Mexico, Brazil, Argentina.

Dù YouTube không công khai thuật toán chi tiết để xác định video xuất hiện trong mục “Thịnh hành”, cộng đồng và các chuyên gia phân tích đã xác nhận các yếu tố ảnh hưởng sự “Thịnh hành” có thể bao gồm: Số lượt xem và tốc độ tăng trưởng, Thời gian xem, Tương tác người dùng, Nguồn gốc lượt xem, Độ tuổi video, Hiệu suất so với các video khác của kênh.

1.7. Kết cấu

Nghiên cứu này tập trung vào việc phân tích dữ liệu video trên YouTube nhằm khám phá các yếu tố ảnh hưởng đến mức độ phổ biến và tương tác của nội dung.

Chương 1: Tổng quan

Giới thiệu bối cảnh của YouTube trong hệ sinh thái nội dung số, nêu rõ mục tiêu, phạm vi và phương pháp nghiên cứu.

Chương 2: Tổng quan về lĩnh vực nghiên cứu

Cung cấp cái nhìn tổng quan về YouTube, thuật toán đề xuất nội dung và các yếu tố ảnh hưởng đến sự thành công của một video. Các lý thuyết về phân tích dữ liệu truyền thông kỹ thuật số cũng được đề cập, giúp xây dựng nền tảng cho việc áp dụng các công cụ như Pandas, Matplotlib và mô hình dự đoán đơn giản.

Chương 3: Bộ dữ liệu

Đi vào chi tiết phương pháp nghiên cứu từ cách thu thập dữ liệu YouTube API, tiền xử lý và chuẩn hóa dữ liệu.

Chương 4: Phương pháp

Trình bày kết quả phân tích với những khám phá quan trọng về xu hướng nội dung, thông qua phân tích EDA và áp dụng các mô hình máy học.

Chương 5: Kết quả và thảo luận

Tổng hợp những phát hiện chính về hiệu suất áp dụng mô hình trên dữ liệu video YouTube và các đặc điểm chung của video thịnh hành , đưa ra các đề xuất thực tiễn giúp tối ưu hóa chiến lược video trên YouTube. Các điểm mạnh và hạn chế của nghiên cứu cũng được thảo luận.

Chương 6: Kết luận - Đề nghị

Đưa ra kết luận về mức độ hoàn thành mục tiêu đã đặt ra và đề nghị hướng phát triển trong tương lai.

CHƯƠNG 2: TỔNG QUAN VỀ LĨNH VỰC NGHIÊN CỨU

Phân tích dữ liệu YouTube là một lĩnh vực nghiên cứu đang phát triển mạnh mẽ, đặc biệt trong bối cảnh sự bùng nổ của nội dung video trực tuyến và sự gia tăng nhu cầu hiểu biết về hành vi người dùng, xu hướng nội dung, và tác động của các video đến xã hội. YouTube là một trong những nền tảng video lớn nhất thế giới, với hàng tỷ lượt xem mỗi ngày, tạo ra một lượng dữ liệu khổng lồ có thể được khai thác để phân tích và đưa ra các quyết định kinh doanh, chiến lược tiếp thị, và nghiên cứu học thuật.

Các nghiên cứu trong lĩnh vực này thường tập trung vào các khía cạnh như:

- Phân tích xu hướng nội dung: Xác định các chủ đề, thể loại các video đang thịnh hàng và sự thay đổi của chúng theo thời gian.
- Phân tích hành vi người dùng: Hiểu cách người dùng tương tác với video, bao gồm lượt xem, lượt thích, bình luận và chia sẻ.
- Dự đoán mức độ phổ biến: Sử dụng các mô hình học máy để dự đoán mức độ phổ biến của video dựa trên các đặc trưng như tiêu đề, mô tả, thẻ tag và thông tin kênh.
- Phân tích cảm xúc và ý kiến: Phân tích bình luận của người dùng để hiểu cảm xúc và ý kiến của họ về video.

Nghiên cứu của tác giả Trịnh Vĩnh Phúc (2023) về phân tích dữ liệu các video thịnh hành trên nền tảng Youtube. Tác giả đã nghiên cứu thành công trong việc phân nhóm và dự đoán số lượt xem của video dựa trên các thuộc tính lượt xem, lượt thích, và lượt không thích. Kết quả nghiên cứu có thể hỗ trợ các quyết định trong việc tối ưu hóa hiệu quả và tăng cường tương tác của video trên nền tảng YouTube. Tuy nhiên, nghiên cứu cần được mở rộng với dữ liệu đa dạng hơn và tích hợp thêm các yếu tố quan trọng khác để nâng cao độ chính xác và tính ứng dụng.

Nghiên cứu của nhóm tác giả Đinh Quang Tiến (2024) về phân tích cảm xúc của người dùng về các chủ đề nóng trên nền tảng Youtube. Đây là một đồ án cuối kỳ trong môn phân tích dữ liệu lớn nhằm dự đoán các chủ đề có khả năng thịnh hành, xu hướng người xem có ảnh hưởng gì đến quyết định tiêu dùng trong tương lai hay không. Tác giả đã sử dụng các công cụ và kỹ thuật phân tích hiện đại trong lĩnh vực Big Data và khoa học dữ liệu. Bên cạnh đó, còn cung cấp dữ liệu có giá trị cho các nhà sáng tạo nội dung từ đó giúp tối ưu hóa nội dung truyền tải, hiệu quả tiếp cận và tương tác trên nền tảng Youtube. Tuy nhiên, nghiên cứu cần đáp ứng các kỹ năng xử lý và phân tích dữ liệu lớn để xử lý vấn đề thời gian thực thi cũng như giảm chiều dữ liệu để đưa ra các mô hình phù hợp hơn.

Nghiên cứu của nhóm tác giả Truong Le (2025) về EnTube: Exploring Key Video Features for Advancing YouTube Engagement. Nghiên cứu này khám phá các phương pháp dự đoán mức độ tương tác (engagement) của video trên YouTube bằng cách sử dụng mô hình học sâu đa phương thức (multimodal deep learning). Nó kết hợp các yếu tố như tiêu đề video, âm thanh, hình ảnh thu nhỏ (thumbnail), nội dung video và thẻ tag để phân loại video vào ba mức độ tương tác: Engage (Tương tác cao), Neutral (Trung bình), và Not Engage (Không tương tác). Nghiên cứu này có nhiều ưu điểm nổi bật, đặc biệt là việc sử dụng phương pháp đa phương thức và bộ dữ liệu EnTube. Tuy nhiên, nó cũng có một số hạn chế về tính tổng quát, độ phức tạp và khả năng áp dụng thực tế. Các nghiên cứu tiếp theo có thể tập trung vào mở rộng bộ dữ liệu, đơn giản hóa mô hình và cải thiện khả năng giải thích để tăng tính ứng dụng.

Nghiên cứu của nhóm tác giả Shiv Ratan Agrawal và Divya Mittal (2024) về Optimizing Marketing Strategy: A Video Analysis Approach. Nghiên cứu này khám phá video đánh giá sản phẩm của người có sự ảnh hưởng trên YouTube, tập trung vào lý do tại sao những video này lại phổ biến với khách hàng, đặc biệt trong lĩnh vực thiết bị điện và điện tử. Mục tiêu là hiểu cách người xem tương tác với nội dung này và cách các công ty có thể tối ưu hóa chiến lược marketing thông qua phân tích video. Đây là một trong những nghiên cứu đầu tiên tập trung vào video đánh giá sản phẩm của người có sự ảnh hưởng và sử dụng phương pháp phân tích video. Nghiên cứu đóng góp vào lĩnh vực marketing bằng cách cung cấp các hiểu biết thực tiễn về cách các công ty có thể tận dụng nội dung video để tăng cường tương tác với khách hàng. Tuy nhiên, nghiên cứu chỉ tập trung vào video bằng tiếng Anh và tiếng Hindi và chỉ xem xét các sản phẩm thiết bị điện và điện tử.

Nhìn chung, các nghiên cứu này đã chứng minh rằng phân tích dữ liệu YouTube không chỉ là công cụ mạnh mẽ để hiểu hành vi người dùng và xu hướng nội dung mà còn là nền tảng quan trọng để phát triển các chiến lược marketing hiệu quả, hoặc phục vụ cho các lĩnh vực khác như kinh tế, tài chính, y tế,... Tuy nhiên, để đạt được kết quả tối ưu, các nghiên cứu trong tương lai cần tập trung vào việc mở rộng dữ liệu, đơn giản hóa mô hình, và tích hợp thêm các yếu tố dữ liệu đa dạng để nâng cao tính ứng dụng và độ chính xác.

CHƯƠNG 3: BỘ DỮ LIỆU

3.1. Nguồn dữ liệu

Tập dữ liệu này được lấy vào ngày 16/4/2025 về các video ở 5 quốc gia: Hoa Kỳ (US), Canada, Mexico, Brazil, Argentina đang thịnh hành trên nền tảng Youtube. Dữ liệu bao gồm thông tin chi tiết về các video như tiêu đề, mô tả, lượt xem, lượt thích, lượt không thích, bình luận, thời lượng video, kênh đăng tải, thời gian đăng tải, và các thẻ tag liên quan. Ngoài ra, dữ liệu cũng bao gồm thông tin về người dùng như số lượng người đăng ký kênh, tương tác của người dùng với video, và các chỉ số về mức độ phổ biến của video trong khoảng thời gian nghiên cứu.

3.2. Quy trình thu thập dữ liệu

Để lấy được tập dữ liệu này, chúng tôi đã dùng Google Cloud mở trong miền Youtube API. Dưới đây là các bước chúng tôi đã thực hiện để lấy tập dữ liệu các video thịnh hành trên Youtube:

- Bước 1: Đăng nhập tài khoản Google Cloud. Sau đó lấy API key từ trường thuộc tính Youtube API.
- Bước 2: Sử dụng API key để truy cập vào Youtube API, lấy các trường thuộc tính khả dụng của các video đang thịnh hành
- Bước 3: Lưu dữ liệu vào tập CSV để phân tích.

3.3. Các đặc trưng của dữ liệu

Tập dữ liệu thứ nhất bao gồm 1000 dòng và 14 thuộc tính. Các thuộc tính gồm có:

1. Thông tin cơ bản về video:

- ID video (*video_id*): ID của video đăng tải.
- Tiêu đề (*title*): Tiêu đề của video.
- Mô tả (*description*): Mô tả ngắn về nội dung video.
- Thời lượng (*duration*): Độ dài của video tính bằng giây.
- Thời gian đăng tải (*published_at*): Ngày và giờ video được đăng tải.
- Thẻ tag (*tags*): Các từ khóa liên quan đến video.
- Độ phân giải (*definition*): Chất lượng của video.
- ID danh mục (*category_id*): mã danh mục.

2. Thông tin tương tác:

- Lượt xem (*view_count*): Tổng số lượt xem video.
- Lượt thích (*like_count*): Tổng số lượt thích video.
- Bình luận (*comment_count*): Tổng số bình luận trên video.

3. Thông tin về kênh:

- Channel ID (*channel_id*): ID của kênh đăng tải video.
- Tên kênh (*channel_title*): Tên của kênh đăng tải video.

Tập dữ liệu thứ hai dùng để lưu trữ các thông tin có liên quan đến bình luận của video gồm 1000 dòng dữ liệu với 2 trường thuộc tính:

- *video_id*: mã video.
- *comments*: các bình luận liên quan đến video.

3.4. Xử lý dữ liệu

3.4.1. Xử lý dữ liệu thiếu

Trong quá trình tiền xử lý dữ liệu, bước kiểm tra và xử lý dữ liệu thiếu là vô cùng quan trọng. Để xác định số lượng giá trị thiếu trong tập dữ liệu, chúng tôi đã sử dụng phương thức *main_df.isnull().sum()*.

Kết quả cho thấy, ngoại trừ cột *description* có 18 giá trị thiếu, tất cả các cột còn lại đều không chứa giá trị thiếu.

Hình 3.1: Kết quả trả về sau khi xử lý giá trị trùng lặp.

3.4.2. Xử lý giá trị trùng lặp

Để đảm bảo tính chính xác của dữ liệu, chúng tôi đã tiến hành kiểm tra và xử lý dữ liệu trùng lặp. Sử dụng hàm *main_df.duplicated().sum()*.

Kết quả cho thấy không có dòng nào bị trùng lặp trong tập dữ liệu:

```
main_df.duplicated().sum()
```

Hình 3.2: Xử lý giá trị trùng lặp.

3.4.3. Tạo cột dữ liệu mới

Phân tích cảm xúc bình luận

Sử dụng phương thức detect từ thư viện langdetect để xác định ngôn ngữ của từng bình luận.

```
def detect_language(text):
    try:
        if len(text) >= 5:  # Check if text has at least 5 characters
            language = detect(text)
            return language
        else:
            return 'unknown'  # Return 'unknown' for short comments
    except:
        return 'unknown'  # Return 'unknown' for comments that cannot be detected
```

Hình 3.3: Phát hiện ngôn ngữ bình luận.

Langdetect là một thư viện Python hỗ trợ phát hiện hơn 50 ngôn ngữ của văn bản dựa trên thư viện của Google.

Ở phương thức detect, văn bản đầu vào được tách thành các cụm từ hoặc đơn vị ngữ nghĩa nhỏ hơn, giúp quá trình phân tích trở nên dễ dàng hơn. Các mô hình N-gram thường được sử dụng để nhóm các ký tự liên tiếp lại với nhau. Những mẫu N-gram phổ biến trong từng ngôn ngữ sẽ được khai thác để trích xuất các đặc trưng, hỗ trợ việc nhận diện ngôn ngữ chính xác hơn.

Sau khi đặc trưng được trích xuất, văn bản sẽ được so sánh với mô hình ngôn ngữ đã được huấn luyện trước đó. Thư viện langdetect sử dụng bộ lọc Bayesian và dữ liệu huấn luyện thu thập từ Wikipedia để ước tính xác suất một văn bản thuộc về ngôn ngữ nào. Các văn bản từ nhiều ngôn ngữ khác nhau đã được dùng để huấn luyện mô hình, giúp nó dự đoán chính xác hơn. Khi thực hiện phương thức detect(), ngôn ngữ có xác suất cao nhất sẽ được lựa chọn.

Kết quả trả về là mã ngôn ngữ theo chuẩn ISO 639-1, giúp người dùng dễ dàng xác định ngôn ngữ của văn bản. Nếu văn bản chứa nhiều ngôn ngữ trộn lẫn, mô hình chỉ

trả về ngôn ngữ có xác suất cao nhất, thay vì xác định nhiều ngôn ngữ cùng lúc. Nhờ vào cách tiếp cận này, quá trình nhận diện ngôn ngữ trở nên hiệu quả và nhanh chóng hơn.

Sau đó, tạo các cột mới thể hiện cụ thể ngôn ngữ đã phát hiện được (thay vì mã ngôn ngữ) bằng cả tiếng Anh - đầu vào cho mô hình phân tích cảm xúc sử dụng TextBlob và tiếng Việt - chuẩn bị cho việc trực quan hóa.

Sau đó, làm sạch bình luận như sau:

```
def preprocess_text(df):
  def remove_emojis(words:list):
    for index in range(len(words)):
      emoji_pattern = re.compile("[\U0001F600-\U0001F64F"  # emoticons
                                 "\U0001F300-\U0001F5FF"  # symbols & pictographs
                                 "\U0001F680-\U0001F6FF"  # transport & map symbols
                                 "\U0001F700-\U0001F77F"  # alchemical symbols
                                 "\U0001F780-\U0001F7FF"  # Geometric Shapes Extended
                                 "\U0001F800-\U0001F8FF"  # Supplemental Arrows-C
                                 "\U0001F900-\U0001F9FF"  # Supplemental Symbols and Pictographs
                                 "\U0001FA00-\U0001FA6F"  # Chess Symbols
                                 "\U0001FA70-\U0001FAFF"  # Symbols and Pictographs Extended-A
                                 "\U00002702-\U000027B0"  # Dingbats
                                 "\U000024C2-\U0001F251"  # Enclosed characters
                                 "]+", flags=re.UNICODE)
      words[index] = emoji_pattern.sub(r'', words[index])
    return words

  df['cleaned_comment'] = ''

  for row in df.itertuples(index=False):
    id = row.video_id
    text = str(row.comment)
    lang = row.language
    text = text.lower()
    text = text.translate(str.maketrans('', '', string.punctuation))
    text = word_tokenize(text)
    text = remove_emojis(text)
    text = remove_stopwords(text, lang)
    df.loc[df['video_id'] == id, 'cleaned_comment'] = text
    index += 1
    if index % 1000 == 0:
      print(f'Đã xử lí {index} bình luận')
  return df
```

Hình 3.5: Làm sạch bình luận.

1. Chuyển văn bản về chữ thường.
2. Loại bỏ dấu câu bằng str.maketrans().
3. Chuyển các câu thành từng tiếng (Tách các từ bằng khoảng trắng).
4. Loại bỏ emoji bằng Unicode (các biểu tượng cảm xúc, các ký hiệu hình ảnh, biểu tượng giao thông, biểu tượng mở rộng, kí hiệu dingbats, …).
5. Lược bỏ các từ không ảnh hưởng đến ý nghĩa cảm xúc của bình luận tương ứng với ngôn ngữ đã xác định.

```
def remove_stopwords(comments, language):
  if language == 'unknown':
    return ''

  try:
    stop_words = set(stopwords.words(language))
    # comments_list = comments.split(',')
    cleaned_comments = []
    for comment in comments:
      tokens = word_tokenize(comment)

      filtered_tokens = [w for w in tokens if not w.lower() in stop_words]
      cleaned_comments.append(' '.join(filtered_tokens))
    return ' '.join(cleaned_comments)
  except:
    return ''
```

Hình 3.6: Bỏ stopwords.

Các bình luận đã làm sạch lưu tại DataFrame *sentiment_df*.

Sau đó, tiến hành phân tích cảm xúc. Ta áp dụng mô hình sentiment analysis từ TextBlob lên các bình luận đã được làm sạch.

```
def get_sentiment(comment):
  analysis = TextBlob(comment)
  return analysis.sentiment.polarity
```

Hình 3.7: Áp dụng mô hình sentiment analysis từ TextBlob.

- TextBlob là một thư viện Python nhằm xử lý ngôn ngữ tự nhiên (NLP), cung cấp các công cụ để phân tích văn bản, bao gồm phân tích cảm xúc.
- TextBlob sử dụng mô hình Pattern Analyzer để gán trọng số cảm xúc (polarity) cho từng từ hoặc cụm từ trong văn bản. Dựa trên một từ điển cảm xúc đã được huấn luyện trước, mỗi từ sẽ được đánh giá theo mức độ cảm xúc: từ tích cực nhận giá trị dương (gần 1), từ tiêu cực nhận giá trị âm (gần -1), từ trung lập có giá trị 0.

Tuy nhiên, TextBlob cũng có một số hạn chế, chẳng hạn như không xử lý tốt ngữ cảnh và có thể hiểu sai các câu phủ định (*"Không tệ" có thể bị nhận diện là tiêu cực*) và hỗ trợ chủ yếu cho tiếng Anh.

Đối với bộ dữ liệu sử dụng, ngôn ngữ ở bình luận phần lớn là tiếng Anh nên ít bị giới hạn bởi các hạn chế nêu trên.

Tạo cột thể hiện tên thể loại

Dựa trên thông tin trang web Mixed Analytics cung cấp về ID thể loại của các video YouTube, ta tạo cột thể hiện tên thể loại bằng phương thức map của Pandas. Dữ liệu này lưu tại cột *category_group* của DataFrame *main_df* đang sử dụng.

Encode ID thể loại, độ phân giải và phụ đề

Đối với cột ID thể loại có hàng chục giá trị, áp dụng One-Hot Encoder. Dữ liệu đã encode được lưu tại một DataFrame riêng và sẽ gộp chung ở bước sau.

Đối với độ phân giải (SD hoặc HD), phụ đề (True - có phụ đề hoặc False - không có phụ đề) áp dụng Label Encoder. Dữ liệu đã encode lưu đè lên dữ liệu gốc tại DataFrame *main_df*.

Tính thời gian từ lúc phát hành đến lúc video thịnh hành

Để xem xét tốc độ lan truyền của video, chúng tôi đã tính toán khoảng thời gian (phút) từ lúc video được phát hành đến một mốc thời gian cố định (ngày 15 tháng 4 năm 2025). lưu vào cột *time_difference*.

```
# Tính thời gian từ lúc phát hành đến lúc video thịnh hành
main_df['converted_published_at'] = pd.to_datetime(main_df['published_at'], utc=True).dt.tz_localize(None)
fixed_date = datetime(2025, 4, 15, 20, 0)
main_df['time_difference'] = (fixed_date - main_df['converted_published_at']).dt.total_seconds() / 60
```

Hình 3.8: Tính thời gian cách biệt.

Dữ liệu sau tính toán được lưu tại cột time_difference.

3.4.4. Chuẩn hóa dữ liệu

Để đảm bảo hiệu quả xử lý và phân tích dữ liệu, chúng tôi đã thực hiện một số bước chuyển đổi và tính toán quan trọng.

Đầu tiên, chuyển video ID thành kiểu chuỗi và chuyển độ dài tiêu đề thành kiểu số nguyên để thuận tiện cho tính toán. Các trường khác đã có kiểu dữ liệu thích hợp.

Tiếp theo, cột *duration* chứa thời lượng video được chuyển đổi từ định dạng ISO 8601 sang số phút dạng số thực bằng hàm *convert_duration* giúp chúng tôi dễ dàng thực hiện các phép tính toán và so sánh thời lượng video.

```
def convert_duration(duration):

  """Chuyển dữ liệu thời lượng từ chuỗi (ví dụ: PT15M48S) sang số phút"""

  hours = 0
  minutes = 0
  seconds = 0

  duration = duration.replace('PT', '')
  if 'H' in duration:
      hours = int(duration.split('H')[0])
      duration = duration.split('H')[1]
  if 'M' in duration:
      minutes = int(duration.split('M')[0])
      duration = duration.split('M')[1]
  if 'S' in duration:
      seconds = int(duration.replace('S', ''))

  return hours * 60 + minutes + seconds / 60
```

Hình 3.9: Tạo hàm convert_duration.

3.4.5. Loại bỏ cột không cần thiết

Sau khi hoàn thành các bước chuyển đổi và tính toán dữ liệu, chúng tôi tiến hành loại bỏ các cột không cần thiết để giảm thiểu dung lượng dữ liệu và tập trung vào các thông tin quan trọng cho phân tích.

Sau các bước trên, dữ liệu gồm các trường: *video_id, title, description, published_at, channel_id, channel_title, category_id, tags, duration, definition, caption, view_count, like_count, dislike_count, comment_count, category_group, view_count_label, converted_published_at, time_difference, converted_duration*

Chúng tôi sử dụng phương thức *drop()* để loại bỏ các cột *title, description, published_at, channel_id, channel_title, category_id, tags, duration, definition, converted_published_at*. Đây là những cột có thông tin mà trong quá trình phân tích được đánh giá là không thật sự cần thiết hoặc đã được xử lí, chuẩn hóa và lưu trữ lại ở vị trí khác.

Bộ dữ liệu có được sau khi thực hiện bước này bao gồm các cột như sau: *video_id, caption, view_count, like_count, dislike_count, comment_count, category_group, view_count_label, time_difference, converted_duration*

3.4.6. Loại bỏ giá trị nhiễu (dựa trên clustering DBSCAN)

Để loại bỏ giá trị nhiễu, chúng tôi đã sử dụng thuật toán clustering DBSCAN.

Trước khi áp dụng DBSCAN, gộp dữ liệu ID thể loại đã encode với các cột định lượng của *main_df* (*view_count, like_count, comment_count, converted_duration, và*

time_difference) thành 1 DataFrame mới là *dữ liệu*, sau đó chuẩn hóa dữ *dữ liệu* bằng StandardScaler.

Dựa vào bài báo "A density-based algorithm for discovering clusters in large spatial databases with noise", chọn **min_samples = số chiều dữ liệu + 1**

Tìm eps tối ưu:

Tham số eps là khoảng cách bán kính tối đa để xác định các điểm lân cận.

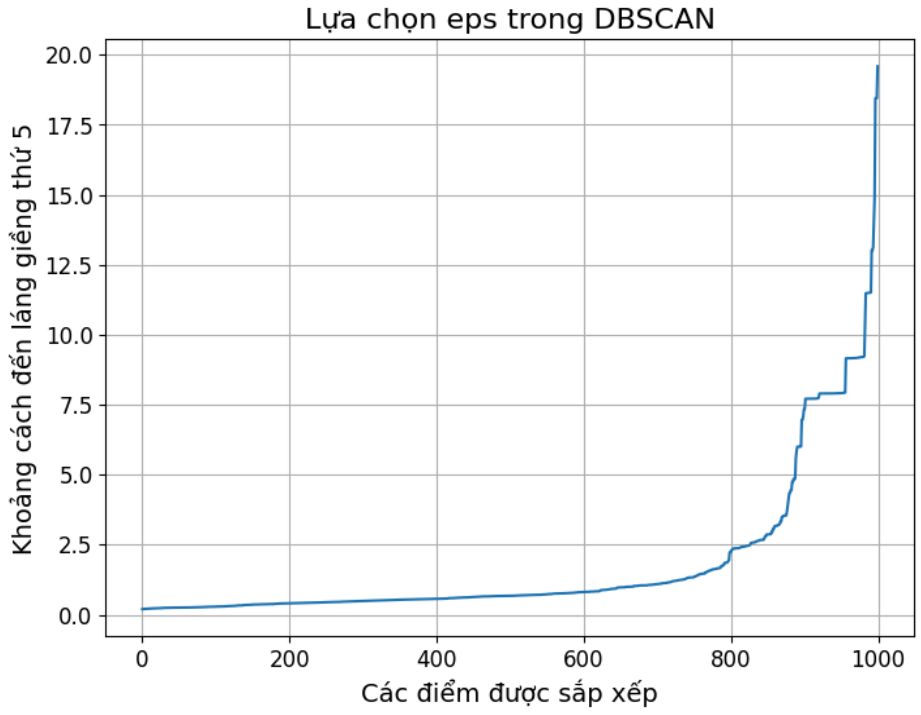
Để tìm giá trị eps tối ưu cho thuật toán DBSCAN, chúng tôi đã sử dụng phương pháp đồ thị Elbow khi triển khai *k-nearest neighbors* với *n_neighbors=min_sample*:

```
neighbors = NearestNeighbors(n_neighbors=5)
neighbors_fit = neighbors.fit(X)
distances, _ = neighbors_fit.kneighbors(X)
distances = np.sort(distances[:, -1])

plt.figure(figsize=(8, 6))
plt.plot(distances)
plt.xlabel("Các điểm được sắp xếp", fontdict={'fontsize': 14})
plt.ylabel("Khoảng cách đến láng giềng thứ 5", fontdict={'fontsize': 14})
plt.title("Lựa chọn eps trong DBSCAN", fontdict={'fontsize': 16})
plt.xticks(fontsize=12)
plt.yticks(fontsize=12)
plt.grid(True)
plt.show()
```

Hình 3.10:Hiển thị eps theo phương thức Elbow.

Đồ thị hiển thị khoảng cách đã sắp xếp cho thấy có một "khuỷu tay" rõ ràng ở giá trị eps bằng 2.5:



Hình 3.11:Lựa chọn eps trong DBSCAN.

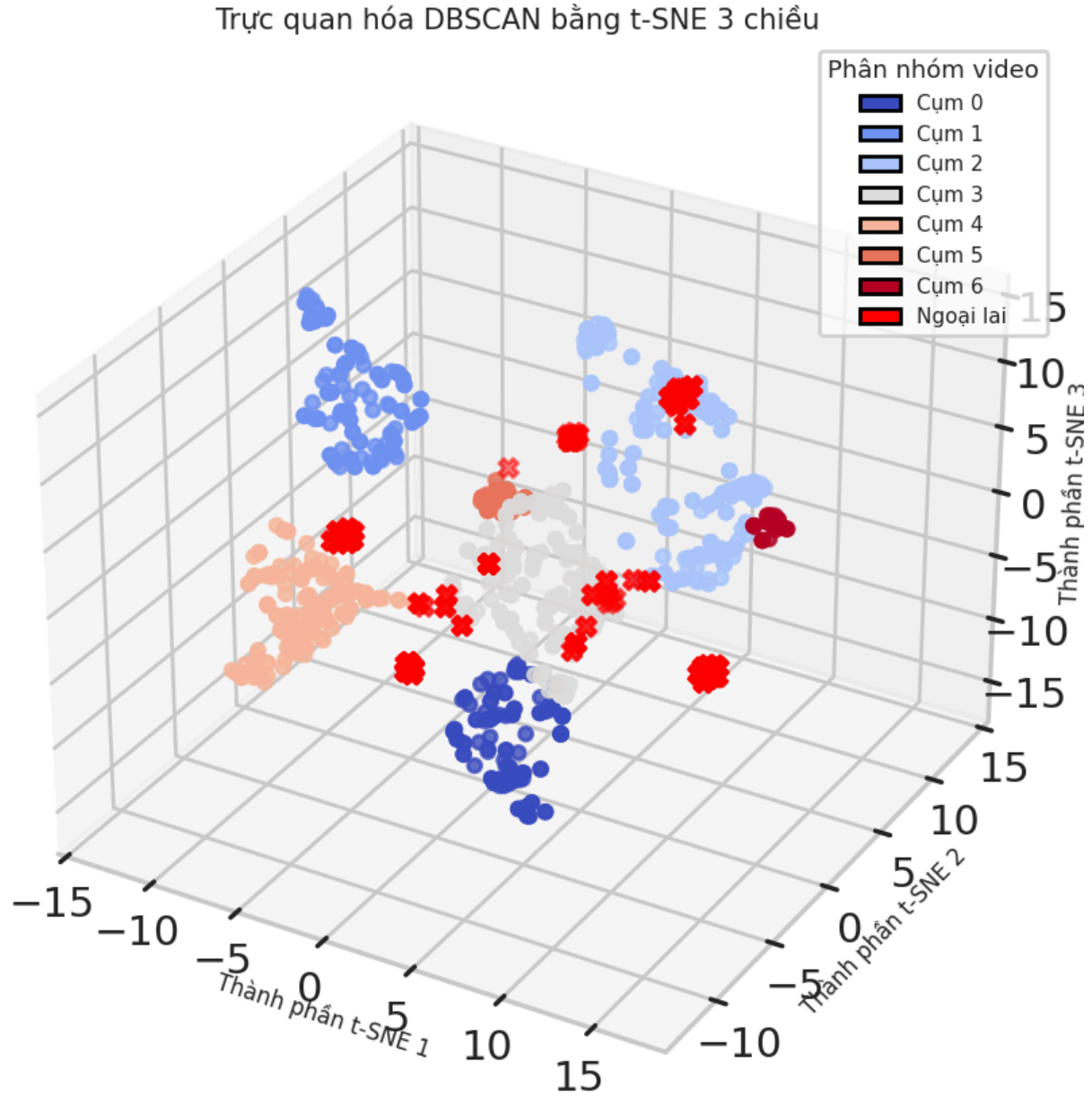
Điểm này cho thấy sự thay đổi đột ngột trong khoảng cách, cho thấy sự chuyển tiếp từ các điểm nằm trong cụm đến các điểm nhiễu. Do đó, chúng tôi chọn eps bằng 2.5 để đảm bảo DBSCAN có thể phân biệt rõ ràng giữa các cụm và các điểm nhiễu. Giá trị chính xác của eps trong khoảng này có thể được điều chỉnh dựa trên phân tích sâu hơn hoặc thử nghiệm thêm.

Với eps=2.5, min_samples =20, DBSCAN tìm thấy các cụm như sau:

- Cụm -1 với số lượng 127
- Cụm 0 với số lượng 118: các video có lượt xem từ 52,000 đến 875,000, lượt thích trung bình 65,000, lượt bình luận trung bình 4,000, thời lượng từ dưới 1 đến 60 phút với giá trị trung bình 18.37. Đặc biệt, các video này tập trung ở thể loại Trò chơi điện tử.
- Cụm 1 với số lượng 118: các video có lượt xem từ 45,000 đến 3,900,000, lượt thích trung bình 36,000, lượt bình luận trung bình 2,700, thời lượng từ dưới 1 đến 60 phút với giá trị trung bình 21.16. Đặc biệt, các video này tập trung ở thể loại Con người & Blog.
- Cụm 2 với số lượng 203: các video có lượt xem từ 26,000 đến 16,234,000, lượt thích trung bình 88,000, lượt bình luận trung bình 5,300, thời lượng từ dưới 1 đến 50 phút với giá trị trung bình 5,63. Đặc biệt, các video này tập trung ở thể loại Âm nhạc.
- Cụm 3 với số lượng 208: các video có lượt xem từ 49,000 đến 17,500,000, lượt thích trung binh 62,000, lượt bình luận trung bình 3,600, thời lượng từ dưới 1 đến 116 phút với giá trị trung bình 21,34. Đặc biệt, các video này tập trung ở thể loại Giải trí.
- Cụm 4 với số lượng 168: các video có lượt xem từ 36,000 đến 19,700,000, lượt thích trung bình 19,000, lượt bình luận trung bình 1,700, thời lượng từ dưới 1 đến 147 phút với giá trị trung bình 15,37. Đặc biệt, các video này tập trung ở thể loại Thể thao.
- Cụm 5 với số lượng 34: các video có lượt xem từ 107,000 đến 6,200,000, lượt thích trung bình 19,000, lượt bình luận trung bình 4,000, thời lượng từ dưới 1 đến 51 phút với giá trị trung bình 8,87. Các video này tập trung ở thể loại Tin tức & Chính trị.
- Cụm 6 với số lượng 24: các video có lượt xem từ 14,000 đến 14,600,000, lượt thích trung bình 86,000, lượt bình luận trung bình 6,000, thời lượng từ dưới 1 đến

30 phút với giá trị trung bình 5.16. Đặc biệt, các video này tập trung ở thể loại Phim & Hoạt hình.

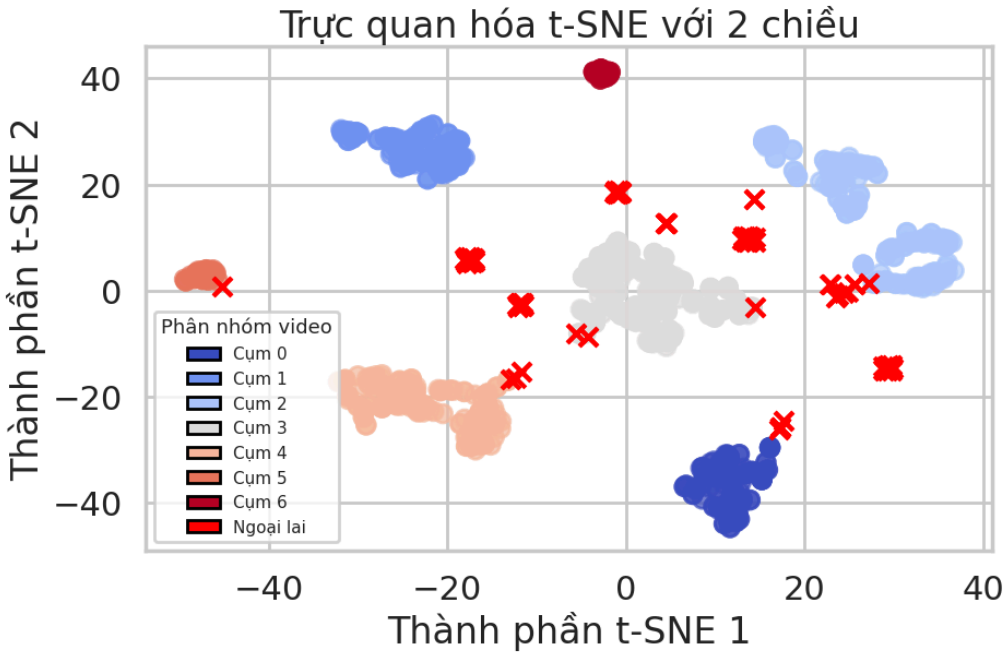
Để trực quan hơn, đây là đồ thị 3D thể hiện kết quả phân cụm bằng DBSCAN sau khi đã sử dụng t-SNE để giảm dữ liệu đầu vào xuống thành 3 chiều.



Hình 3.12:Trực quan 3D kết quả DBSCAN.

Biểu đồ cho thấy rõ ràng sự phân tách giữa các cụm. Điều này chứng tỏ rằng thuật toán DBSCAN đã hoạt động hiệu quả trong việc phân biệt các điểm dữ liệu thuộc về cụm và các điểm nhiễu. Việc trực quan hóa này giúp chúng tôi hiểu rõ hơn về cấu trúc của dữ liệu và đánh giá kết quả của thuật toán DBSCAN.

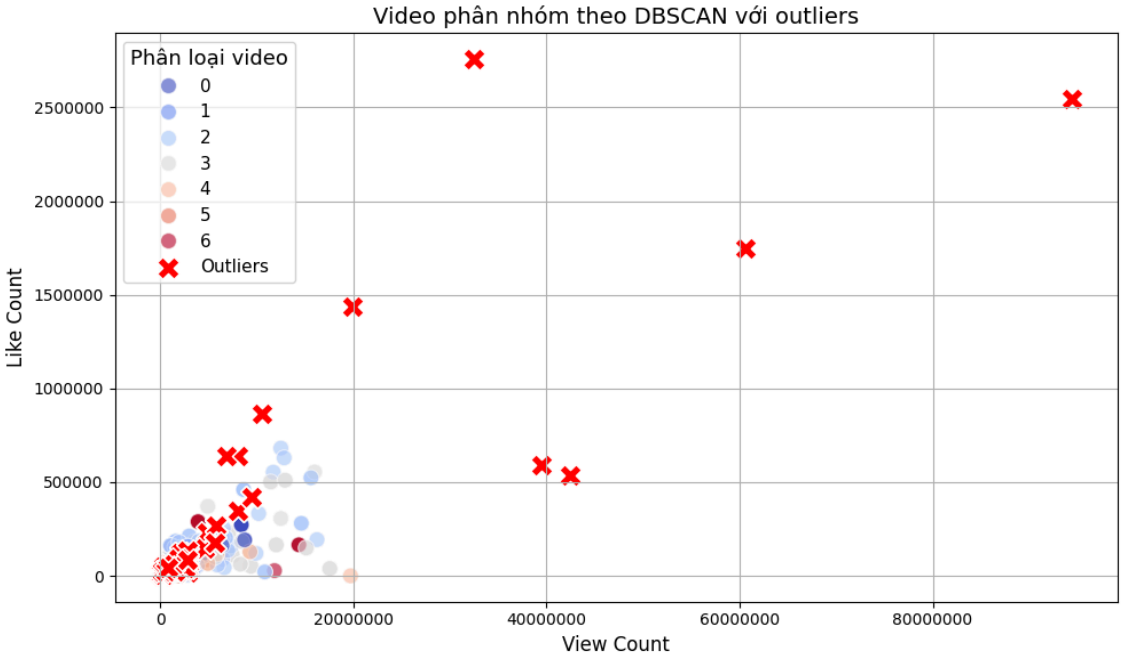
Tuy nhiên, vị trí tương đối của các điểm ngoại lai so với các điểm còn lại vẫn chưa được rõ ràng. Do đó, chúng tôi cũng đã vẽ biểu đồ phân tán 2D của dữ liệu.



Hình 3.13:Trực quan 2D kết quả DBSCAN.

Đồ thị cho thấy rõ ràng sự phân tách giữa cụm chính và các điểm nhiễu trong không gian 2D. Điều này chứng tỏ rằng thuật toán DBSCAN đã hoạt động hiệu quả trong việc phân biệt các điểm dữ liệu thuộc về cụm và các điểm nhiễu. Việc trực quan hóa này giúp chúng tôi hiểu rõ hơn về cấu trúc của dữ liệu và đánh giá kết quả của thuật toán DBSCAN.

Mặt khác, khi quan sát riêng các đặc trưng lượt xem, lượt thích, ta thấy các outliers cụ thể hơn:



Hình 3.14: Phân nhóm DBSCAN.

Đó là các video có lượt xem hàng triệu, lượt thích hàng trăm nghìn, lượt bình luận hàng chục nghìn, ngắn dưới 10 phút và phát hành từ rất lâu.

Tiếp theo, chúng tôi loại bỏ các điểm nhiễu tìm được, còn lại 993 điểm dữ liệu.

CHƯƠNG 4: PHƯƠNG PHÁP

4.1 EDA (Exploratory Data Analysis)

4.1.1. Đơn biến

Chúng tôi dùng các hàm sau để phân tích và trực quan các trường dữ liệu định lượng (*view_count, like_count, comment_count, converted_duration, và time_difference*).

***descriptive_stats(column)*:** Sử dụng phương thức describe() của pandas Series (một cột trong DataFrame) để tính các thống kê như trung bình, độ lệch chuẩn, giá trị nhỏ nhất, giá trị lớn nhất.

```
def descriptive_stats(column):
  descriptive_stats = column.describe()
  pd.options.display.float_format = '{:.2f}'.format
  print(descriptive_stats)
```

Hình 4.1: Tính các thông số thống kê cơ bản.

***distribution_plot(column)*:** Vẽ biểu đồ histogram với đường cong ước lượng mật độ (KDE) bằng *sns.hisplot*

```
def distribution_plot(column):
  print(f"Thống kê mô tả của biến: {column.name}")
  plt.figure(figsize=(8, 5))
  sns.histplot(column, kde=True)
  plt.ticklabel_format(style='plain', axis='x')
  plt.title(f"Phân bố của biến: {column.name}", fontdict={'fontsize': 16})
  plt.xlabel(column.name, fontsize=14)
  plt.ylabel('Count', fontsize=14)
  plt.xticks(fontsize=12)
  plt.yticks(fontsize=12)
  plt.show()
```

Hình 4.2: Hàm trực quan hóa bằng biểu đồ histogram.

***box_plot(column)*:** Vẽ biểu đồ hộp bằng *sns.boxplot(y=column)*.

```
def box_plot(column):
  plt.figure(figsize=(8, 5))
  sns.boxplot(y=column)
  plt.ticklabel_format(style='plain', axis='y')
  plt.title(f"Box plot của biến: {column.name}", fontdict={'fontsize': 16})
  plt.legend(fontsize=8)
  plt.ylabel(column.name, fontsize=14)
  plt.xticks([], [])  # Hide x-tick values
  plt.xticks(fontsize=12)
  plt.yticks(fontsize=12)
  plt.show()
```

Hình 4.3: Hàm trực quan hóa bằng biểu đồ box.

Kết quả thu được như sau:

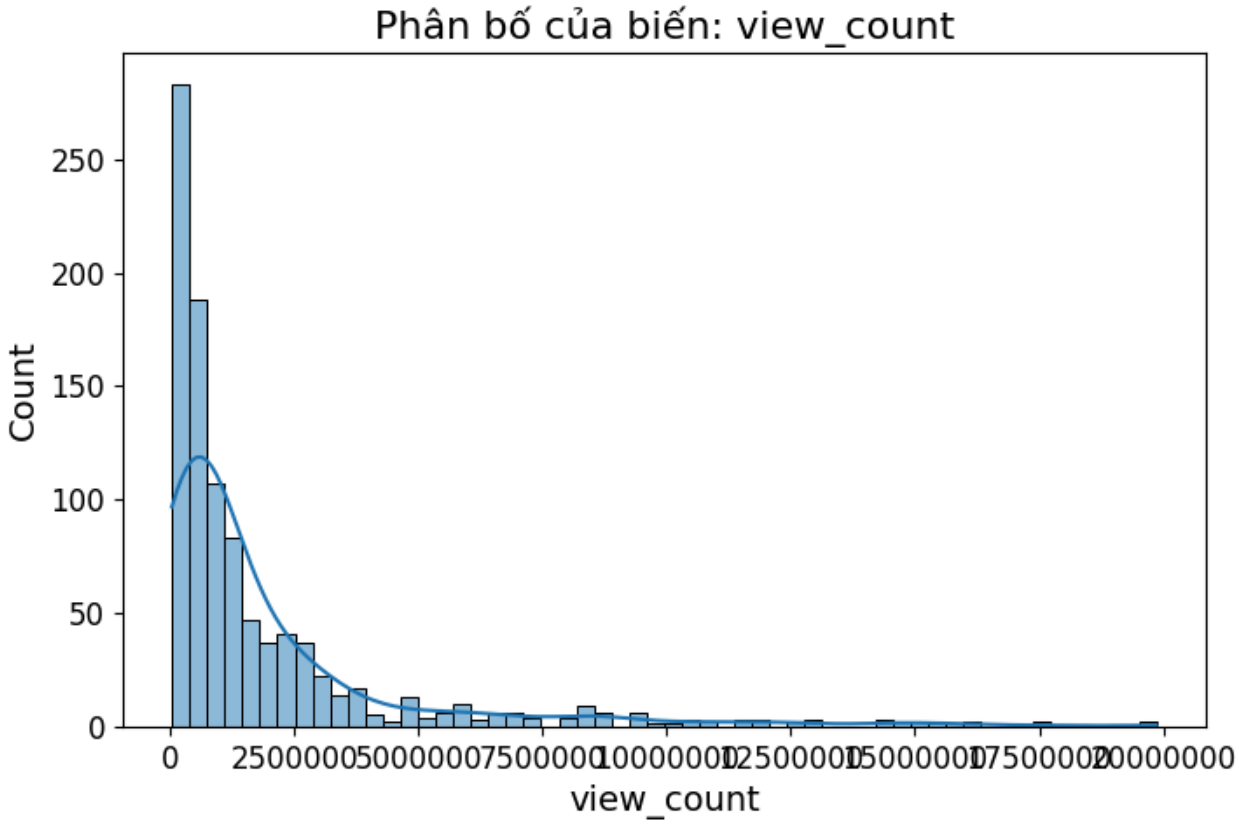
view_count:

- biến *view_count* cho thấy có tổng cộng 993 video được ghi nhận số lượt xem. Số lượt xem trung bình là khoảng 1.89 triệu, tuy nhiên, có sự khác biệt rất lớn giữa các video, thể hiện qua độ lệch chuẩn cao gần 2.88 triệu. Phạm vi số lượt xem trải dài từ tối thiểu 26.246 đến tối đa gần 19.87 triệu lượt xem. Giá trị trung vị (50%) là khoảng 794.399 lượt xem, thấp hơn đáng kể so với giá trị trung bình, cho thấy phân phối số lượt xem có xu hướng lệch phải, nghĩa là có một số video có số lượt xem cực kỳ cao kéo giá trị trung bình lên. Khoảng 50% số video có số lượt xem nằm giữa 334.340 (25%) và 2.11 triệu (75%).

```
count        993.00
mean     1889458.02
std      2879760.02
min        26246.00
25%       334340.00
50%       794399.00
75%      2106782.00
max     19865734.00
Name: view_count, dtype: float64
Thống kê mô tả của biến: view_count
```

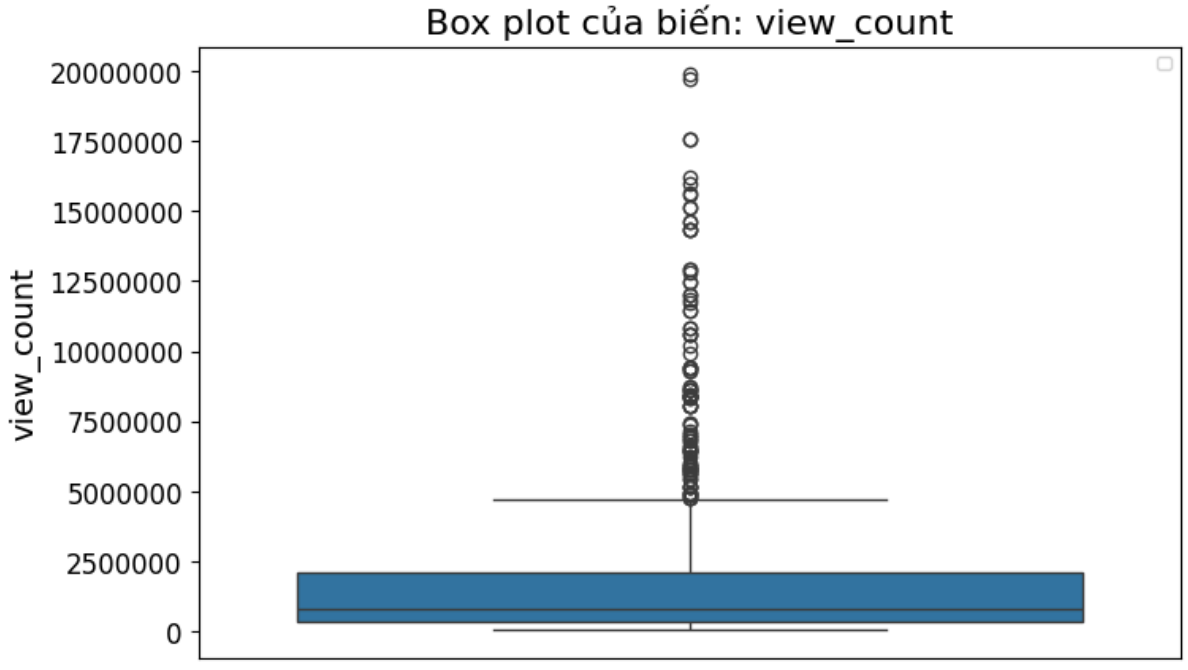
Hình 4.4: Kết quả thống kê mô tả của biến view_count.

- Biểu đồ phân phối *view_count* cho thấy phân phối lệch phải rõ rệt. Đa số video có số lượt xem thấp, tập trung ở gần 0, tạo đỉnh cao ở bên trái. Tần suất giảm nhanh khi số lượt xem tăng, với đuôi dài về bên phải cho thấy một số ít video có số lượt xem rất cao.



Hình 4.5: Biểu đồ trực quan hóa của biến view_count.

- Biểu đồ hộp *view_count* cho thấy phân phối lệch phải với hộp tập trung ở số lượt xem thấp và đường trung vị thấp hơn phạm vi tứ phân vị thứ ba. Rất nhiều điểm ngoại lai xuất hiện phía trên, cho thấy một số video có số lượt xem cao bất thường so với đa số video.



Hình 4.6: Biểu đồ box plot của biến view_count.

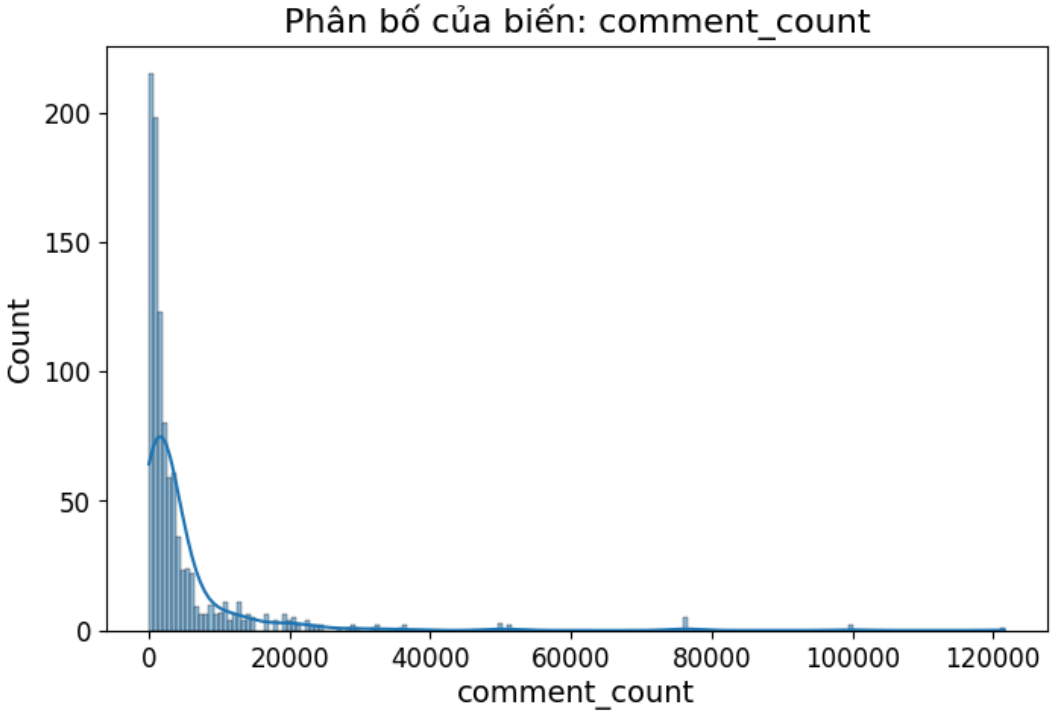
comment_count:

- Biến *comment_count* có 993 quan sát, với giá trị trung bình là 4608.21 bình luận. Độ lệch chuẩn cao (9919.48) cho thấy sự phân tán rộng của dữ liệu. Số lượng bình luận dao động từ 0 đến 121620. Giá trị trung vị (1797) thấp hơn nhiều so với giá trị trung bình, cho thấy dữ liệu lệch phải. 25% số quan sát có dưới 740 bình luận và 75% có dưới 4006 bình luận. Sự khác biệt lớn giữa các thống kê này cho thấy sự biến động đáng kể trong số lượng bình luận.

```
count      993.00
mean      4608.21
std       9919.48
min          0.00
25%        740.00
50%       1797.00
75%       4006.00
max     121620.00
Name: comment_count, dtype: float64
Thống kê mô tả của biến: comment_count
```

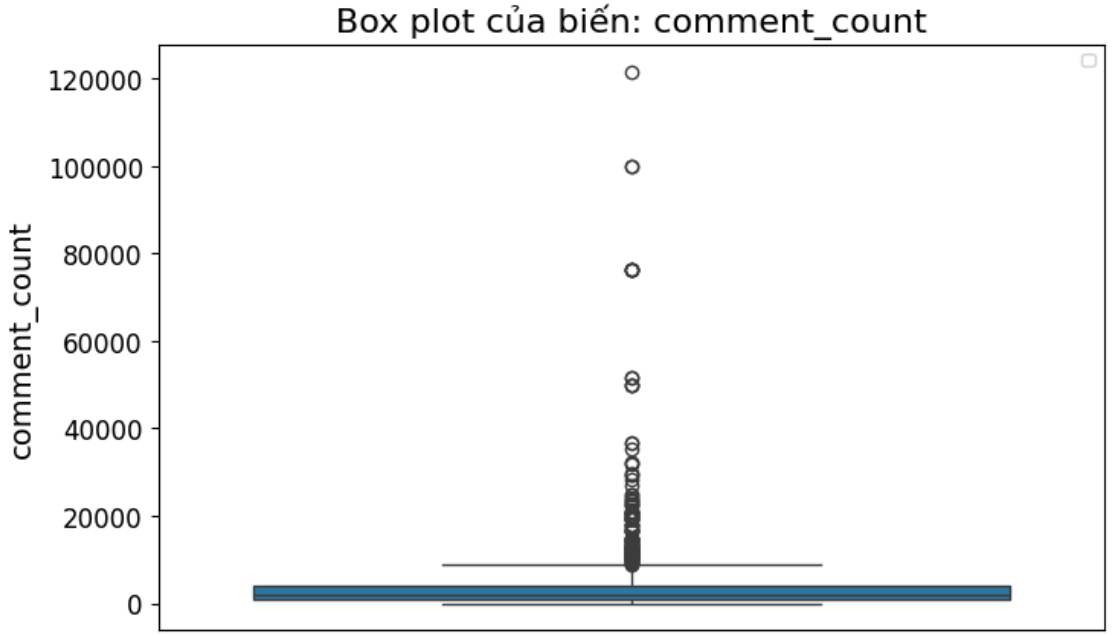
Hình 4.7: Kết quả thống kê mô tả của biến comment_count.

- Biểu đồ phân phối của biến *comment_count* bị lệch phải rõ rệt. Phần lớn các bài viết số lượng bình luận thấp, trong khi một số ít bài viết có số lượng bình luận rất cao, tạo nên đuôi dài bên phải của biểu đồ. Đường cong phân phối chuẩn không phù hợp với dữ liệu, cho thấy dữ liệu không tuân theo phân phối chuẩn.



Hình 4.8: Biểu đồ trực quan hóa của biến comment_count.

- Biểu đồ hộp (box plot) ho thấy sự phân bố lệch lạc của dữ liệu biến *comment_count* với phần lớn các giá trị tập trung ở mức thấp và một số ít giá trị rất cao tạo thành đuôi dài. Những giá trị cao này có thể được coi là các giá trị ngoại lai, phản ánh các bài viết có mức độ tương tác bình luận đặc biệt cao do các yếu tố như nội dung, sự chú ý hoặc khả năng lan truyền.



Hình 4.9: Biểu đồ box plot của biến comment_count.

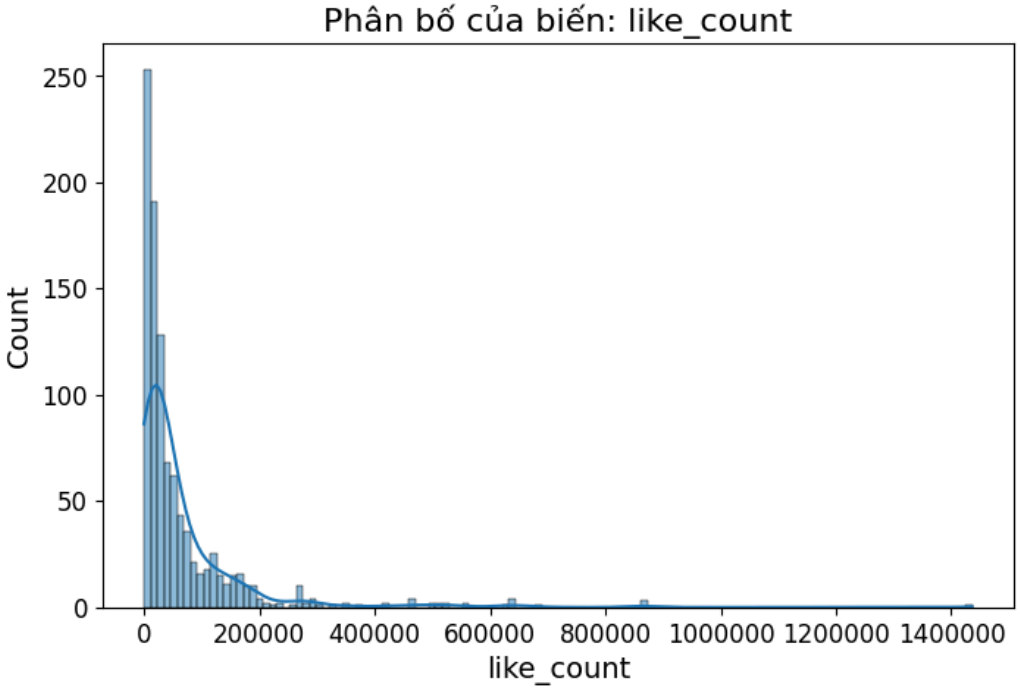
like_count:

- Biến *like_count* ghi nhận số lượt thích của 993 video, có phạm vi giá trị rộng, từ 0 đến hơn 1,4 triệu. Số lượt thích trung bình là khoảng 63.469, nhưng trung vị thấp hơn nhiều ở mức 27.036, cho thấy phân phối lệch phải, trong đó một số bài đăng có số lượt thích rất cao, kéo giá trị trung bình lên cao. Có sự thay đổi đáng kể về số lượt thích, như thể hiện bằng độ lệch chuẩn lớn.

```
count       993.00
mean      63469.44
std      110486.52
min          0.00
25%      11258.00
50%      27036.00
75%      68823.00
max     1436320.00
Name: like_count, dtype: float64
Thống kê mô tả của biến: like_count
```

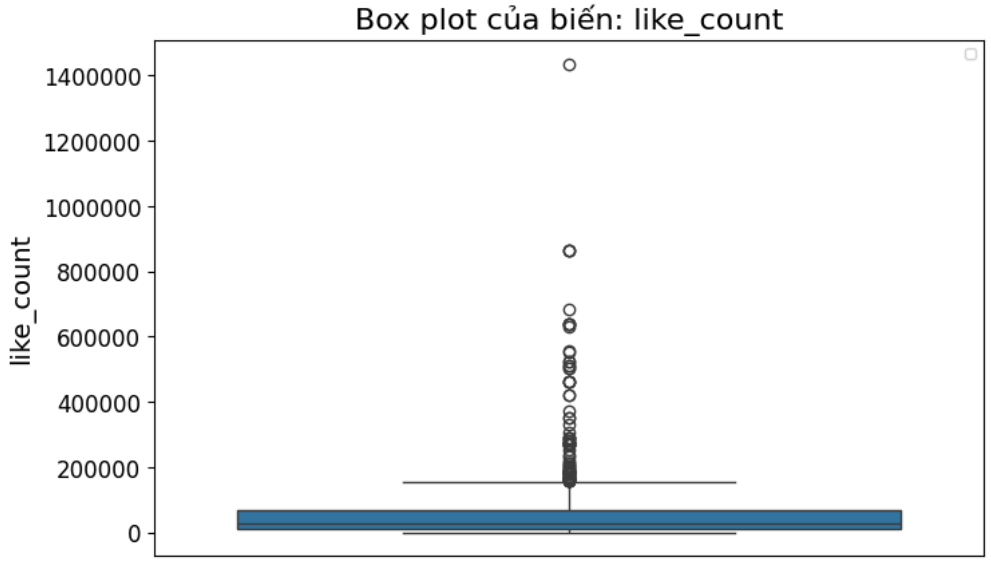
Hình 4.10: Thống kê mô tả của biến like_count.

- Biểu đồ phân phối cho thấy dữ liệu *like_count* bị lệch phải rõ rệt. Phần lớn các bài viết có số lượt thích thấp, trong khi một số ít bài viết có số lượt thích rất cao, tạo nên đuôi dài bên phải của biểu đồ. Đường cong phân phối chuẩn không phù hợp với dữ liệu, cho thấy dữ liệu không tuân theo phân phối chuẩn.



Hình 4.11: Biểu đồ trực quan hóa của biến like_count.

- Biểu đồ hộp (box plot) xác nhận sự hiện diện của nhiều giá trị ngoại lai. Các giá trị ngoại lai này cho thấy có một số video nhận được số lượt thích vượt trội so với phần lớn các video khác. Điều này có thể do các video này có nội dung đặc biệt hấp dẫn, gây sốt, hoặc được chia sẻ rộng rãi.



Hình 4.12: Biểu đồ box plot của biến like_count.

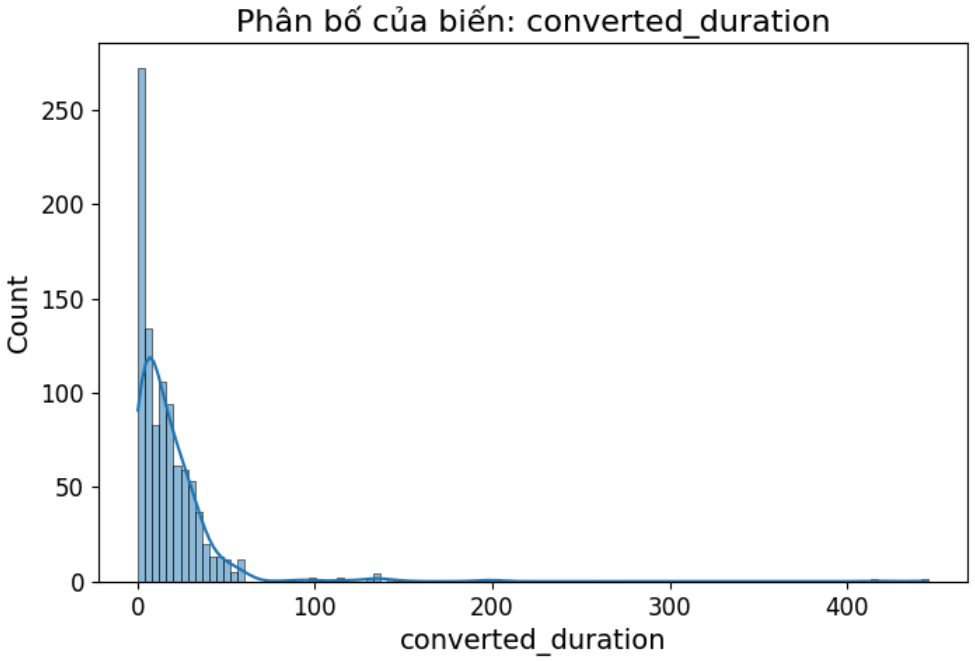
converted_duration:

- Biến *converted_duration* cho thấy dữ liệu có 993 quan sát. Thời lượng trung bình là 18.20 đơn vị. Tuy nhiên, có sự biến động lớn về thời lượng, thể hiện qua độ lệch chuẩn cao (27.90). Thời lượng video ngắn nhất là 0.27 và dài nhất lên đến 445.43, cho thấy phạm vi rất rộng. 50% số video có thời lượng dưới 12.53, trong khi 75% có thời lượng dưới 24.07.

```
count   993.00
mean     18.20
std      27.90
min       0.27
25%       3.88
50%      12.53
75%      24.07
max     445.43
Name: converted_duration, dtype: float64
Thống kê mô tả của biến: converted_duration
```

Hình 4.13: Kết quả thống kê mô tả của biến converted_duration.

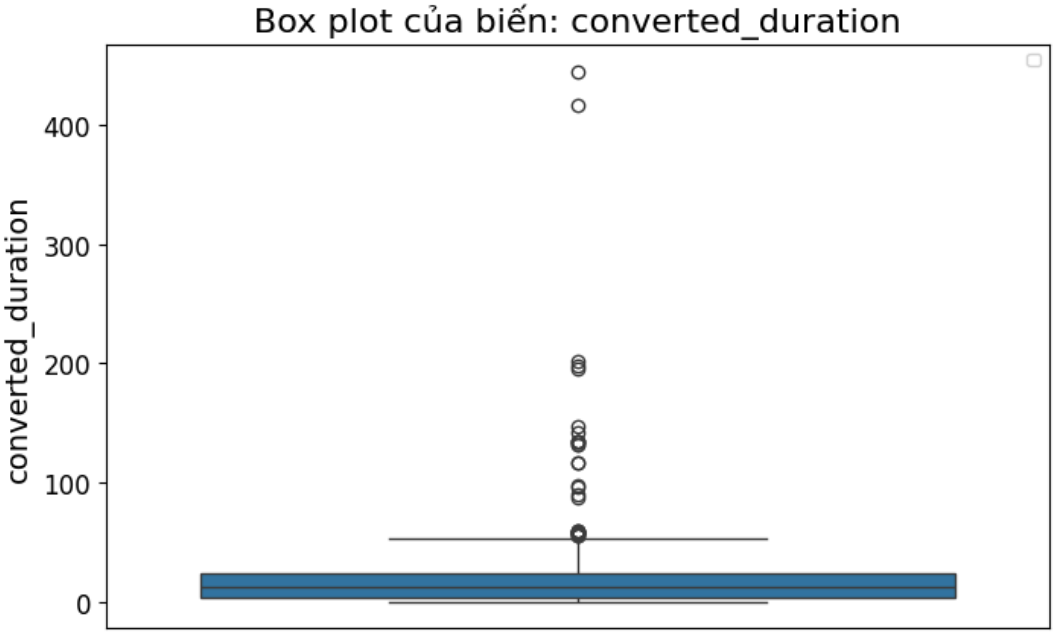
- Biểu đồ phân phối của *converted_duration* cho thấy phần lớn video có thời lượng rất ngắn, tập trung gần 0. Tần suất giảm nhanh khi thời lượng tăng, với một đuôi dài về bên phải cho thấy một số ít video có thời lượng rất lớn. Phân phối lệch phải này khẳng định rằng đa số video ngắn, trong khi video dài hơn thì ít phổ biến.



Hình 4.14: Biểu đồ trực quan hóa của biến converted_duration.

- Box plot của *converted_duration* cho thấy rõ ràng sự phân bố lệch phải của thời lượng video. Hộp biểu thị 50% dữ liệu trung tâm nằm ở khoảng thời lượng ngắn. Điều này cho thấy phần lớn video có thời lượng ngắn. Điểm ngoại lai cao nhất

thậm chí vượt quá 400 đơn vị thời gian. Sự tồn tại của các ngoại lai này khẳng định rằng trong khi đa số video có thời lượng ngắn, vẫn có một số ít video có thời lượng rất dài, góp phần vào sự lệch phải của phân phối đã quan sát được từ các thống kê mô tả và biểu đồ phân phối trước đó.



Hình 4.15: Biểu đồ box plot của biến converted_duration.

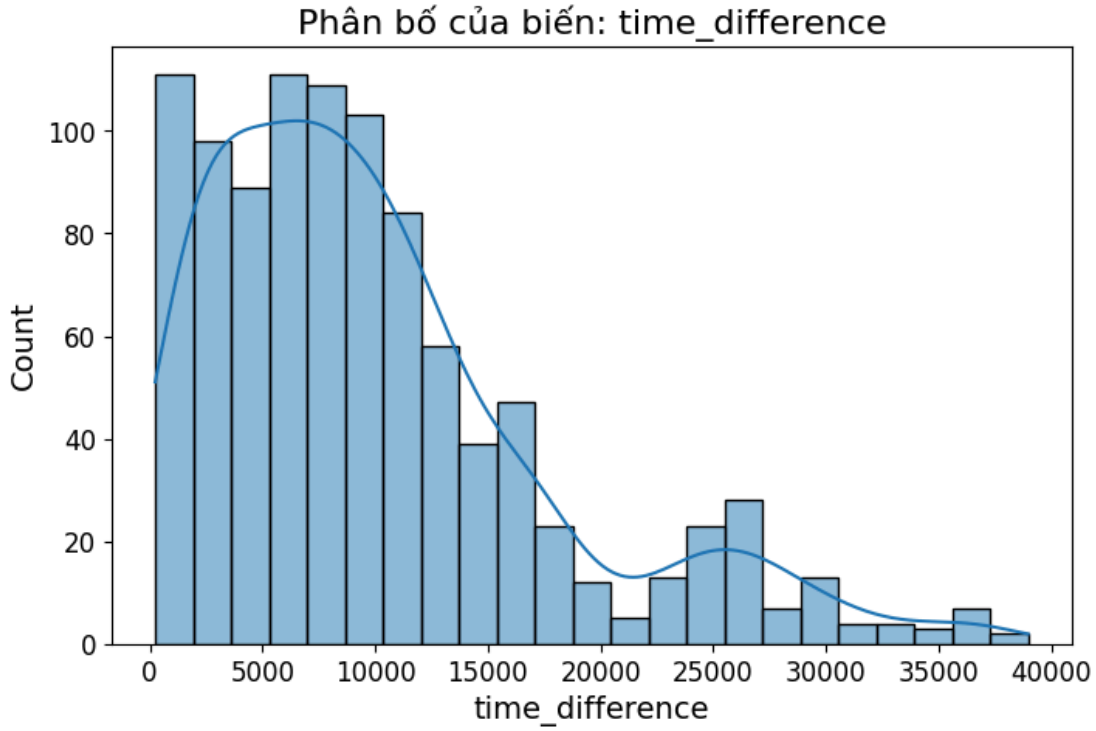
time_difference:

- Biến *time_difference* cho thấy có 993 quan sát với giá trị trung bình là 10077.95 đơn vị thời gian. Độ lệch chuẩn là 7772.99, cho thấy sự biến động đáng kể xung quanh giá trị trung bình. Phạm vi giá trị khá rộng, từ giá trị nhỏ nhất là 239.90 đến giá trị lớn nhất là 38970.32. Giá trị trung vị (50%) là 8481.52, thấp hơn giá trị trung bình, gợi ý về một phân phối lệch phải, nơi có một số giá trị lớn kéo giá trị trung bình lên cao hơn giá trị trung tâm. Khoảng 50% dữ liệu nằm giữa 4440.53 (25%) và 13120.10 (75%).

```
count     993.00
mean    10077.95
std      7772.99
min       239.90
25%      4440.53
50%      8481.52
75%     13120.10
max     38970.32
Name: time_difference, dtype: float64
Thống kê mô tả của biến: time_difference
```

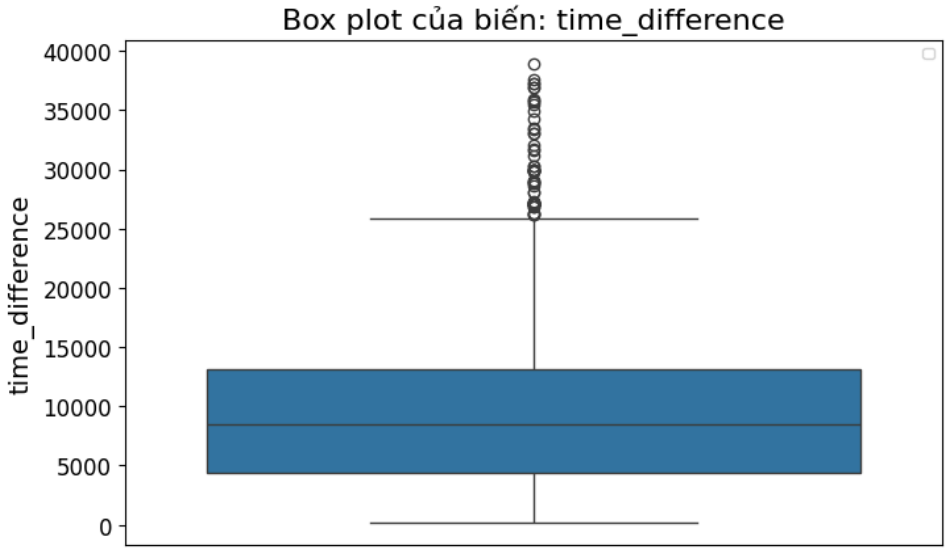
Hình 4.16: Kết quả thống kê mô tả của biến time difference.

- Biểu đồ phân phối của *time_difference* cho thấy một phân phối phức tạp với ít nhất hai vùng tập trung dữ liệu. Đỉnh chính nằm ở giá trị thấp, cho thấy phần lớn quan sát có *time_difference* nhỏ. Một đỉnh hoặc vai thứ hai xuất hiện ở giá trị cao hơn, cho thấy một nhóm quan sát khác có giá trị *time_difference* lớn hơn. Đuôi bên phải kéo dài cho thấy vẫn có một số giá trị rất lớn.



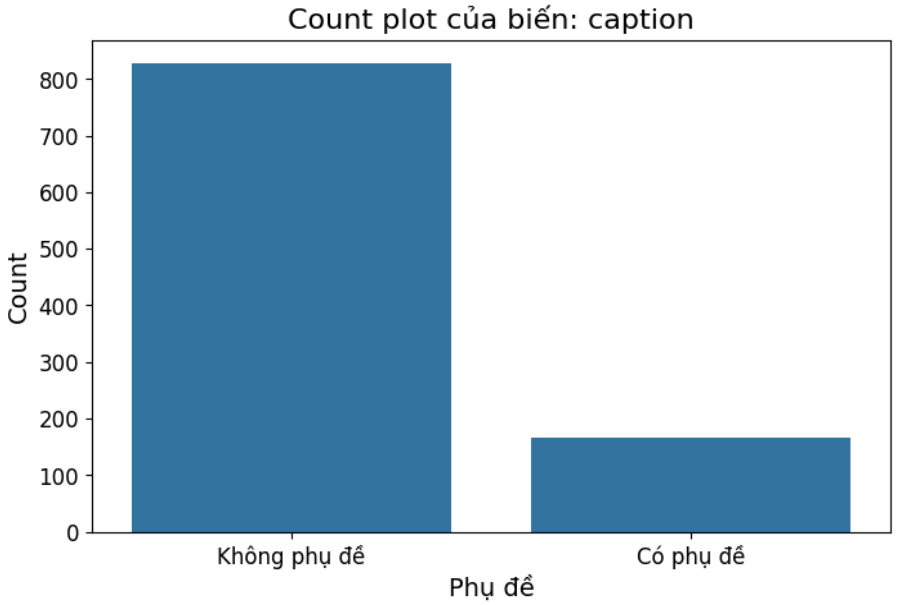
Hình 4.17: Biểu đồ trực quan hóa của biến time_difference.

- Box plot của *time_difference* cho thấy sự phân tán của dữ liệu và xác định các giá trị ngoại lai. Hộp biểu thị 50% dữ liệu trung tâm nằm giữa phạm vi tứ phân vị thứ nhất (khoảng 4000-5000) và phạm vi tứ phân vị thứ ba (khoảng 13000). Đường kẻ ngang bên trong hộp, biểu thị giá trị trung vị, nằm ở khoảng 8000-9000. Các "râu" kéo dài từ hộp cho thấy phạm vi của dữ liệu không phải là ngoại lai.



Hình 4.18: Biểu đồ box plot của biến time_difference.

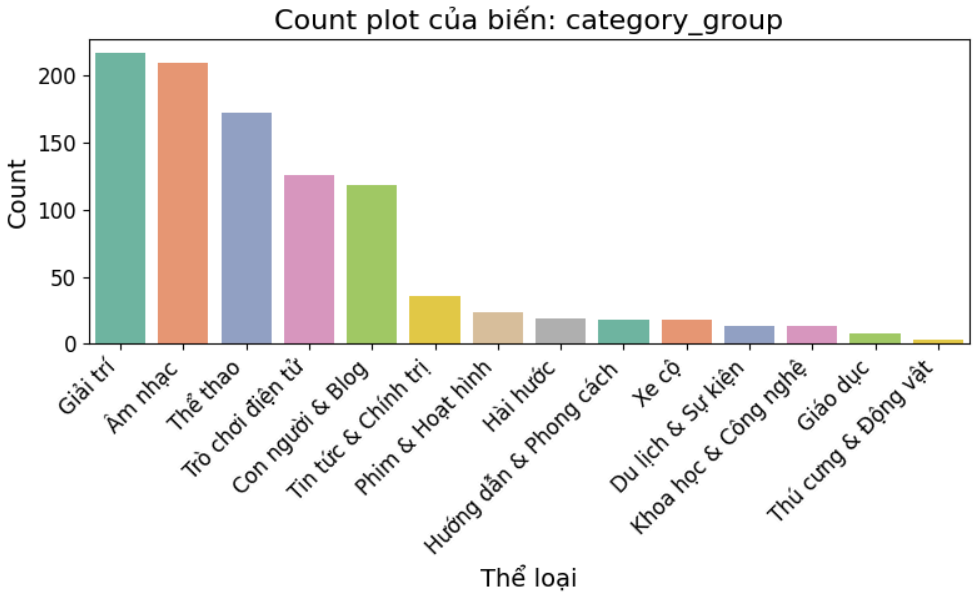
Biểu đồ đếm của biến *caption* hiển thị số lượng quan sát cho từng danh mục. Cột bên trái, tương ứng với "Không phụ đề", có chiều cao khoảng 830, cho thấy có khoảng 830 video không có phụ đề. Cột bên phải, tương ứng với "Có phụ đề", có chiều cao khoảng 170, cho thấy có khoảng 170 video có phụ đề. Như vậy, số lượng video không có phụ đề lớn hơn đáng kể so với số lượng video có phụ đề trong tập dữ liệu này.



Hình 4.19: Biểu đồ đếm của biến caption.

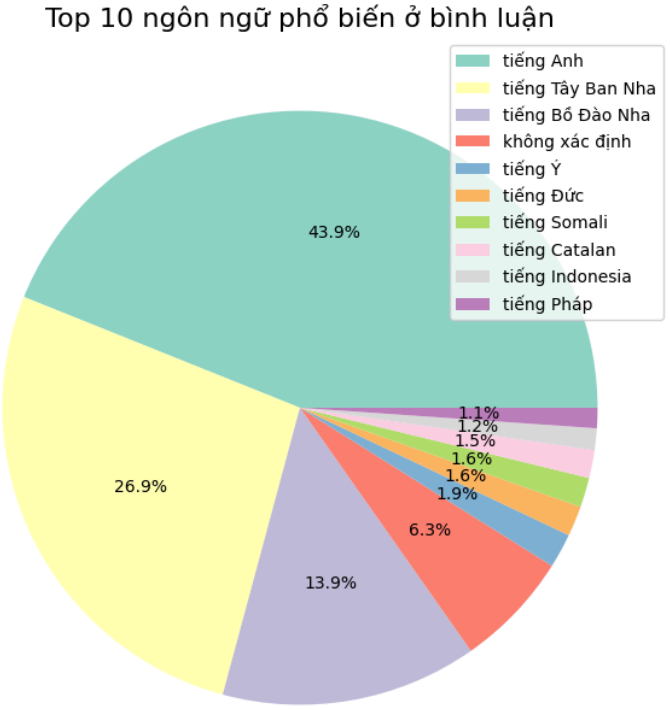
Nhóm "Giải trí" có số lượng video cao nhất, tiếp theo là "Âm nhạc" và "Thể thao". Số lượng video giảm dần ở các nhóm tiếp theo như "Trò chơi điện tử", "Con người & Blog", và "Tin tức & Chính trị". Các nhóm như "Phim & Hoạt hình", "Hướng dẫn & Phong cách", "Hài hước", "Xe cộ", "Du lịch & Sự kiện", "Khoa học & Công nghệ", "Giáo dục", và "Thú cưng & Động vật" có số lượng video ít hơn đáng kể so với các

nhóm dẫn đầu. Điều này cho thấy sự phân bố không đồng đều giữa các nhóm danh mục video trong tập dữ liệu, với một số nhóm phổ biến hơn nhiều so với các nhóm khác.



Hình 4.20: Biểu đồ đếm của biến category_group.

Biểu đồ tròn "Top 10 ngôn ngữ phổ biến ở bình luận" cho thấy tiếng Anh là ngôn ngữ được sử dụng nhiều nhất (43.9%), theo sau là tiếng Tây Ban Nha (26.9%) và tiếng Bồ Đào Nha (13.9%). Một phần (6.3%) là các bình luận không xác định được ngôn ngữ. Các ngôn ngữ khác trong top 10 (tiếng Ý, Đức, Somali, Catalan, Indonesia, Pháp) chiếm tỷ lệ nhỏ hơn (1.1% - 1.9%). Tiếng Anh là ngôn ngữ chiếm ưu thế trong các bình luận.



Hình 4.21: biểu đồ tròn thể hiện 10 ngôn ngữ phổ biến ở bình luận.

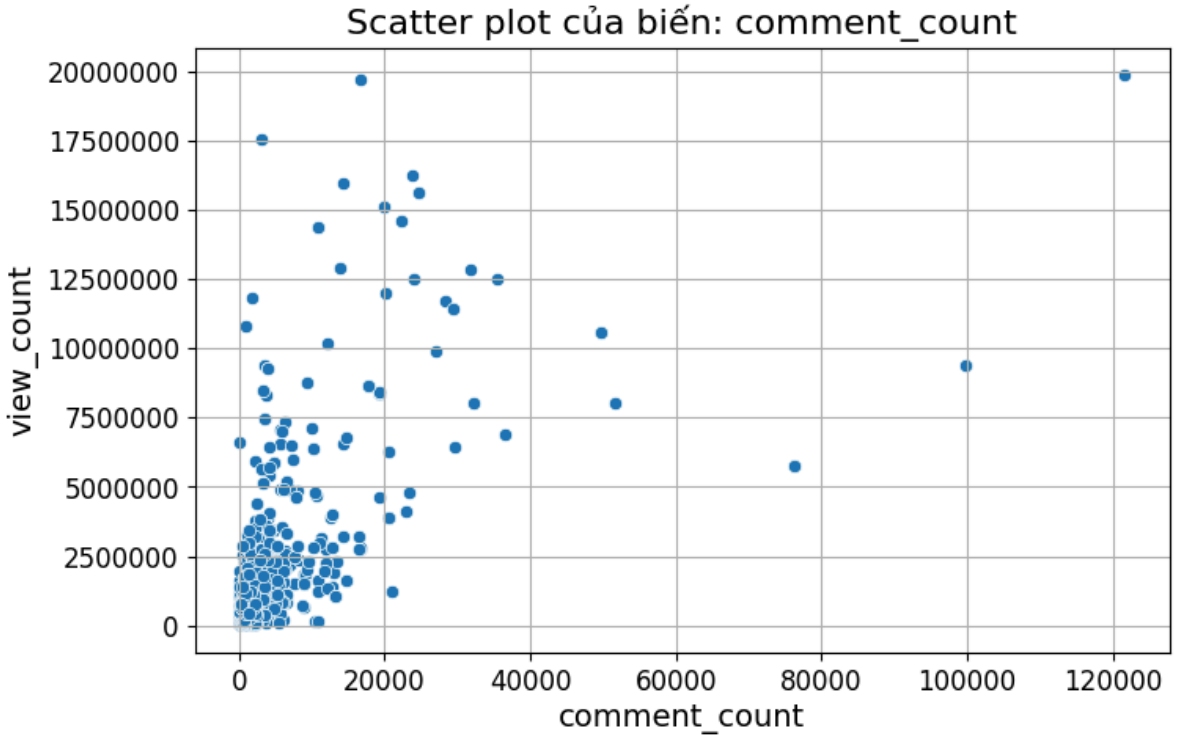
4.1.2. Hai biến

Lượt xem video trên nền tảng YouTube là một chỉ số quan trọng phản ánh mức độ phổ biến và khả năng tiếp cận của nội dung. Số lượt xem có thể chịu ảnh hưởng bởi nhiều yếu tố như thời lượng video, số lượt thích, số bình luận, danh mục nội dung, thời điểm đăng tải và mức độ tương tác của người xem. Phân tích mối quan hệ giữa số lượt xem và các biến liên quan không chỉ giúp nhận diện các xu hướng trong hành vi người dùng mà còn cung cấp cơ sở dữ liệu quan trọng cho việc tối ưu hóa nội dung và chiến lược tiếp thị trên nền tảng này.

Phần này tập trung vào phân tích hai biến nhằm đánh giá mức độ tương quan giữa lượt xem và từng yếu tố tác động thông qua biểu đồ scatter.

Kết quả nhận được như sau:

comment_count



Hình 4.22: Biểu đồ scatter của biến comment count.

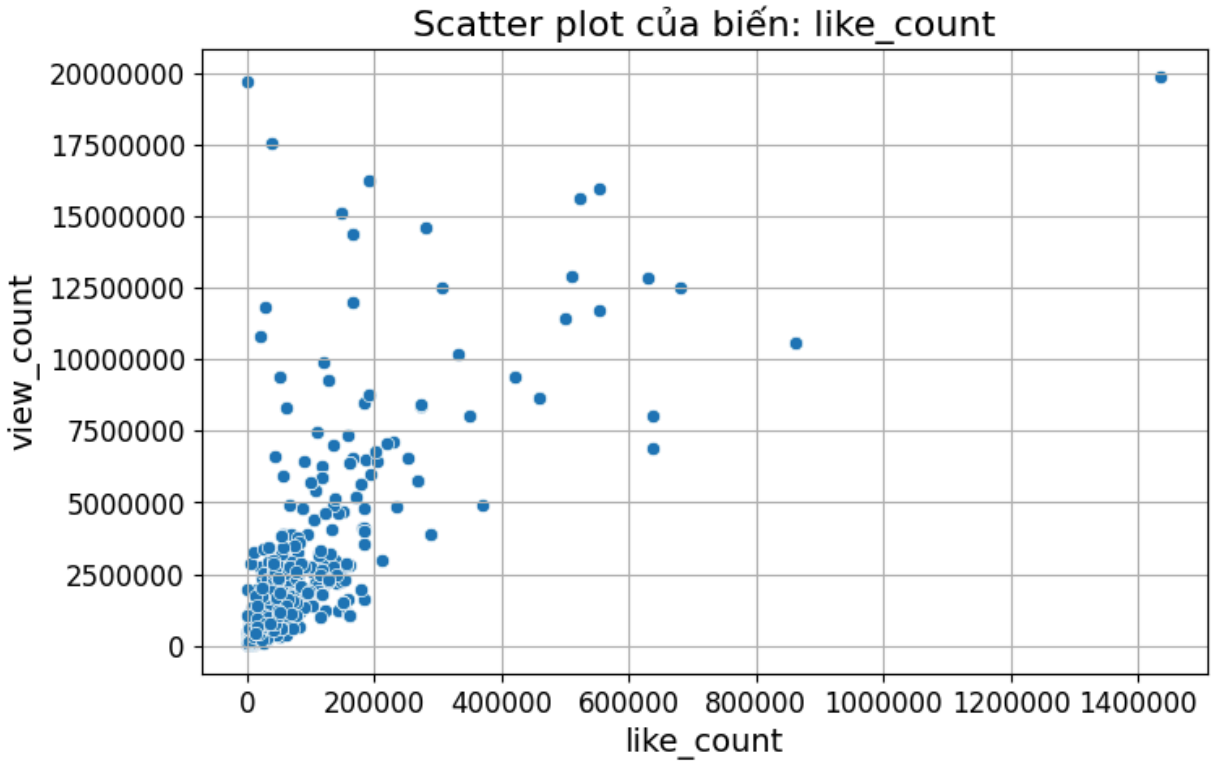
Biểu đồ cho thấy phần lớn các điểm dữ liệu tập trung ở góc dưới bên trái của biểu đồ, cho thấy đa số các video có số lượng bình luận thấp (dưới khoảng 20,000) và số lượt xem tương đối thấp (dưới khoảng 1 triệu). Điều này phản ánh rằng phần lớn các video trên nền tảng có mức độ tương tác trung bình hoặc thấp.

Tuy nhiên, có một số điểm dữ liệu phân tán ở vùng phía trên và bên phải, cho thấy có một số video nhận được số lượng bình luận hoặc số lượt xem cao hơn đáng kể. Đáng chú ý là có một vài điểm ngoại lai rõ rệt với số lượt xem rất cao (gần 2 triệu) nhưng số

lượng bình luận không đặc biệt lớn (dưới 20,000), và ngược lại, có một điểm ngoại lai với số lượng bình luận rất cao (khoảng 120,000) nhưng số lượt xem ở mức trung bình (khoảng 2 triệu).

Nhìn chung, biểu đồ cho thấy một xu hướng tăng nhẹ của số lượt xem khi số lượng bình luận tăng lên, tuy nhiên, mối quan hệ này không mạnh mẽ và có nhiều trường hợp ngoại lệ. Điều này gợi ý rằng số lượng bình luận có thể là một trong những yếu tố đóng góp vào sự phổ biến của video (được đo bằng số lượt xem), nhưng không phải là yếu tố duy nhất và mối quan hệ giữa chúng có thể bị ảnh hưởng bởi nhiều yếu tố khác như nội dung video, thời điểm đăng tải, và chiến lược quảng bá.

like_count



Hình 4.23: Biểu đồ scatter của biến like count.

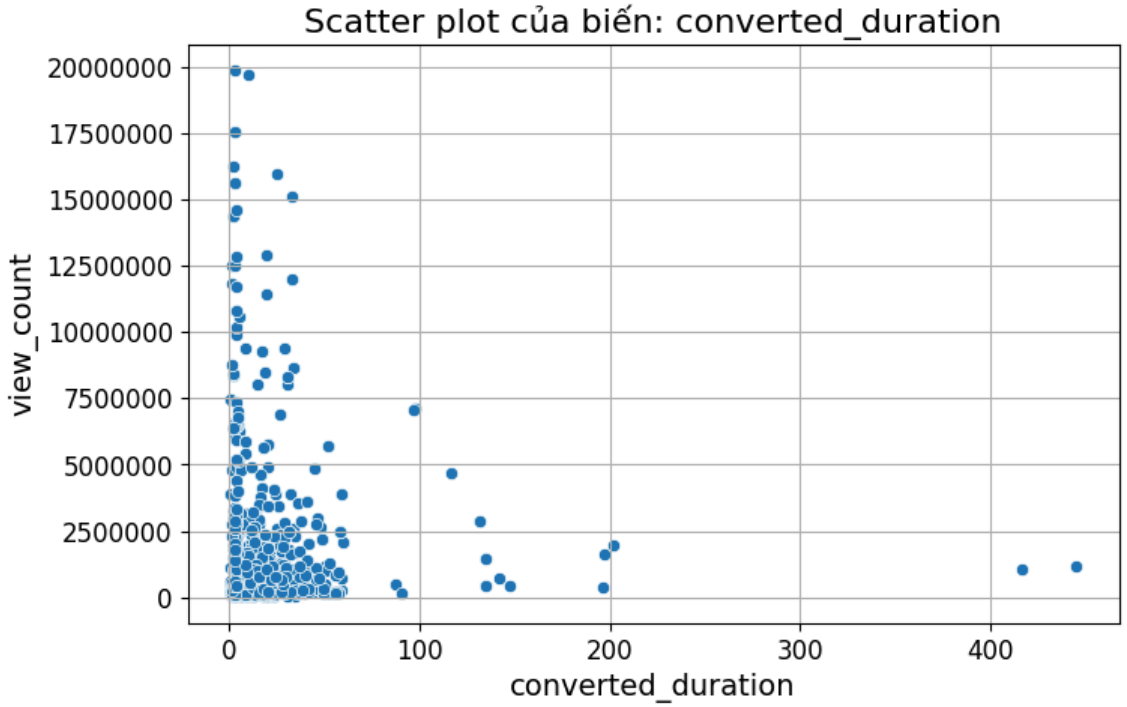
phần lớn các điểm dữ liệu tập trung ở góc dưới bên trái, cho thấy đa số các video có cả số lượt thích và số lượt xem ở mức thấp. Điều này cho thấy rằng phần lớn các video có mức độ tương tác (cả thích và xem) ở mức trung bình hoặc thấp.

Tuy nhiên, khi số lượt thích tăng lên, có một xu hướng chung là số lượt xem cũng tăng theo, cho thấy một mối tương quan dương giữa hai biến này. Các video có số lượt thích cao hơn thường có xu hướng nhận được nhiều lượt xem hơn. Dù vậy, mối quan hệ này không hoàn toàn tuyến tính và có sự phân tán đáng kể.

Đáng chú ý là có một số điểm ngoại lai ở góc trên bên phải, biểu thị các video có cả số lượt thích và số lượt xem rất cao. Ví dụ, có một video có số lượt thích rất lớn (gần

1.5 triệu) và số lượt xem cũng rất cao (gần 2 triệu). Ngoài ra, cũng có một số video có số lượt xem cao nhưng số lượt thích lại ở mức trung bình, và ngược lại. Điều này cho thấy số lượt thích là một yếu tố quan trọng đóng góp vào số lượt xem, nhưng các yếu tố khác như nội dung, thời điểm đăng tải và khả năng lan tỏa cũng đóng vai trò quan trọng.

converted_duration



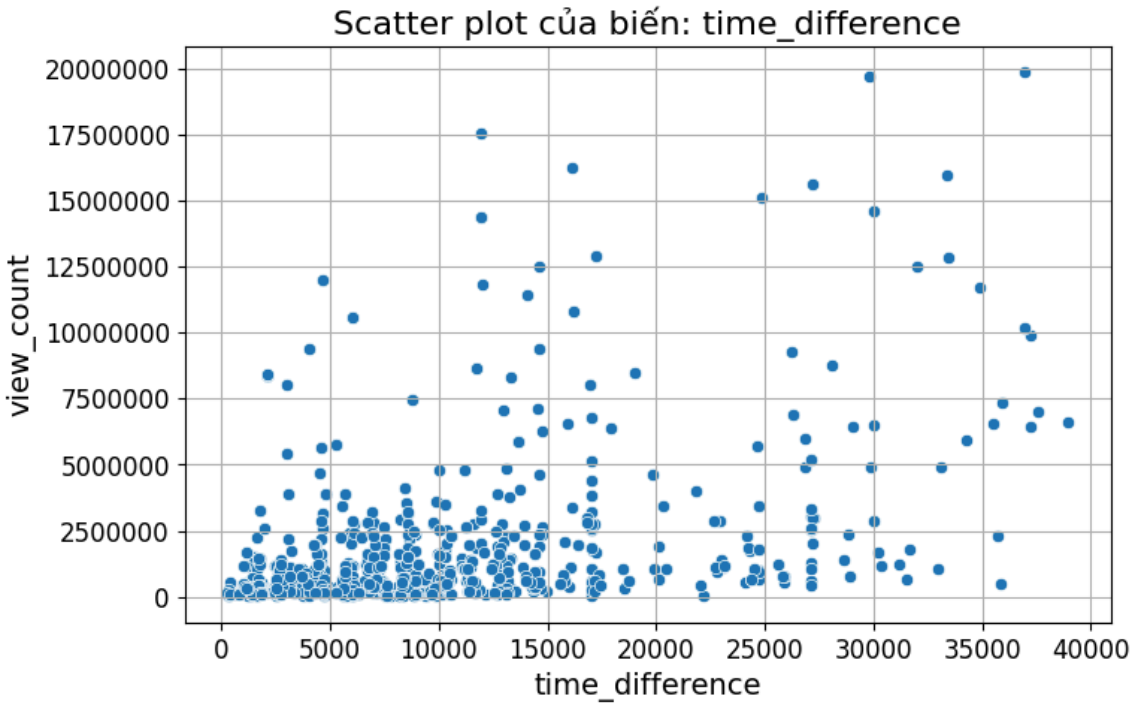
Hình 4.24: Biểu đồ scatter của biến converted duration.

Quan sát biểu đồ cho thấy Phần lớn các điểm dữ liệu tập trung ở vùng có thời lượng video ngắn, chủ yếu dưới 100 đơn vị thời gian, và trải rộng trên nhiều mức độ lượt xem khác nhau, từ thấp đến cao. Điều này cho thấy rằng có cả những video ngắn có ít lượt xem và những video ngắn có rất nhiều lượt xem.

Khi thời lượng video tăng lên, số lượng điểm dữ liệu giảm đáng kể. Có vẻ như không có một xu hướng rõ ràng giữa thời lượng video và số lượt xem. Một số video có thời lượng trung bình (khoảng 100-200 đơn vị thời gian) có thể có số lượt xem cao hoặc thấp. Đáng chú ý là có một vài điểm ngoại lai với thời lượng video rất dài (trên 400 đơn vị thời gian), nhưng số lượt xem của chúng lại không quá cao so với một số video ngắn.

Nhìn chung, biểu đồ này gợi ý rằng thời lượng video không phải là một yếu tố quyết định trực tiếp đến số lượt xem. Các video ngắn có thể đạt được lượng xem lớn, và ngược lại, các video dài cũng có thể có lượng xem cao hoặc thấp. Có vẻ như các yếu tố khác như nội dung, chủ đề, và cách quảng bá video có thể đóng vai trò quan trọng hơn trong việc thu hút người xem so với độ dài của video.

time_difference



Hình 4.25: Biểu đồ scatter của biến time_difference.

Quan sát dữ liệu cho thấy phần lớn video được đăng tải không lâu, tập trung trong khoảng 0 - 15.000 phút (0 - 4 giờ) trước khi đạt mức viral. Điều này có thể phản ánh thực tế rằng YouTube thường ưu tiên các video mới trong thuật toán đề xuất, giúp chúng tiếp cận nhiều khán giả hơn trong thời gian đầu sau khi đăng tải.

Khi sự khác biệt thời gian tăng lên, số lượng điểm dữ liệu có xu hướng giảm, nhưng vẫn có các video với sự khác biệt thời gian lớn hơn đạt được nhiều lượt xem. Không có một xu hướng rõ ràng nào cho thấy sự khác biệt thời gian có ảnh hưởng trực tiếp và nhất quán đến số lượt xem. Các video đã được đăng tải trong một khoảng thời gian dài (sự khác biệt thời gian lớn) vẫn có thể thu hút được lượng lớn người xem, tương tự như các video mới đăng tải.

Sự phân tán rộng rãi của các điểm dữ liệu trên biểu đồ cho thấy rằng sự khác biệt thời gian, một mình nó, không phải là một yếu tố quyết định chính đến số lượt xem của video. Các yếu tố khác như nội dung, chủ đề, tính thời vụ và các nỗ lực quảng bá có thể đóng vai trò quan trọng hơn trong việc thu hút người xem, bất kể video đã được đăng tải bao lâu.

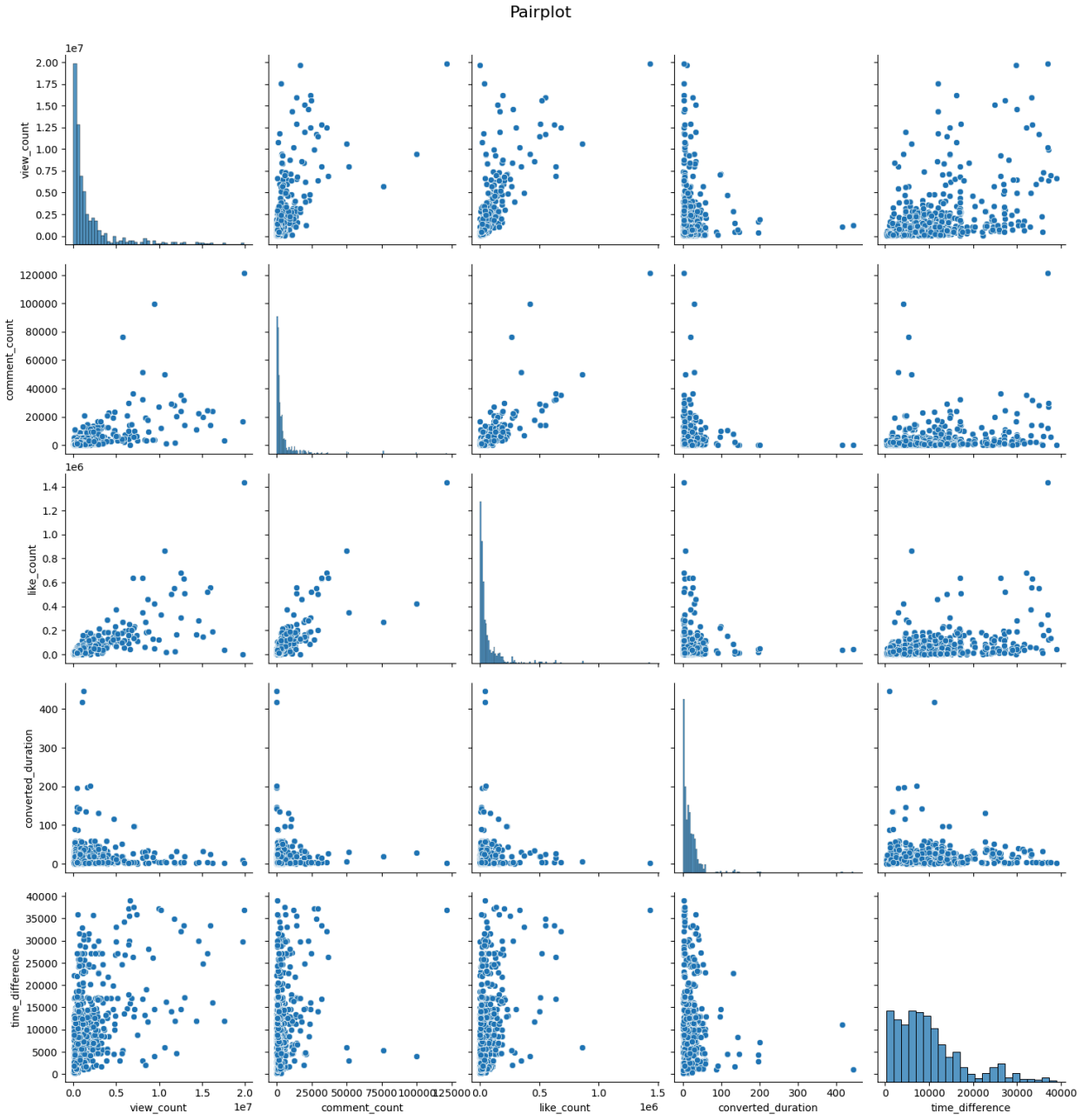
4.1.3. Đa biến

Phân tích đa biến giúp khám phá mối quan hệ phức tạp giữa các yếu tố trong dữ liệu. Phần này tập trung vào phân tích các mối quan hệ như vậy thông qua pairplot, ma trận hệ số tương quan, ma trận hiệp phương sai và giảm chiều dữ liệu thông qua PCA.

Kết quả thu được như sau:

Biểu đồ pairplot

Pairplot thể hiện mỗi cặp biến bằng một biểu đồ phân tán (scatter plot) trên đường chéo chính của ma trận biểu đồ. Điều này cho phép xem xét mối quan hệ giữa các biến dọc theo đường chéo.



Hình 4.26: Biểu đồ pairplot theo từng biến.

Sau khi trực quan hóa dữ liệu, quan sát cho thấy các mẫu tương tác không đồng đều với sự xuất hiện của nội dung viral, mối quan hệ phức tạp giữa các chỉ số tương tác, và sự đa dạng về phản hồi cảm xúc từ người dùng.

Các biểu đồ trên đường chéo chính hiển thị phân phối của từng biến riêng lẻ dưới dạng histogram. Có thể thấy rõ rằng *view_count*, *comment_count*, và *like_count* đều có

phân phối lệch phải mạnh, với phần lớn dữ liệu tập trung ở các giá trị thấp và một số ít giá trị rất cao. *converted_duration* cũng cho thấy sự lệch phải tương tự, trong khi *time_difference* có vẻ phân bố rộng hơn.

Các biểu đồ ngoài đường chéo chính là biểu đồ tán xạ (scatter plot) thể hiện mối quan hệ giữa từng cặp biến. Có một mối tương quan dương rõ rệt giữa *view_count*, *comment_count*, và *like_count*, cho thấy rằng các video có nhiều lượt xem thường cũng có xu hướng có nhiều bình luận và lượt thích hơn. Tuy nhiên, mối quan hệ này không hoàn toàn tuyến tính và có nhiều điểm phân tán, đặc biệt ở các giá trị cao.

Mối quan hệ giữa *converted_duration* và các biến khác không rõ ràng bằng. Dường như không có mối tương quan mạnh mẽ giữa thời lượng video và số lượt xem, bình luận hoặc lượt thích. Tương tự, mối quan hệ giữa *time_difference* và các biến khác cũng không cho thấy xu hướng mạnh mẽ nào. Các điểm dữ liệu có vẻ phân tán rộng rãi, gợi ý rằng thời gian đăng tải video không phải là yếu tố quyết định trực tiếp đến mức độ tương tác (lượt xem, bình luận, thích) hoặc thời lượng của video.

Nhìn chung, ta có thể kết luận dữ liệu thể hiện rõ tính chất "viral" của nội dung mạng xã hội, với phân bố không đồng đều về tương tác. Tính đa ngôn ngữ là đặc điểm quan trọng của nền tảng, phản ánh tầm ảnh hưởng toàn cầu.

Và nghiên cứu có thể đưa ra khuyến nghị như là phân tích sâu hơn về nội dung viral bằng cách nghiên cứu đặc điểm chung của các nội dung có tương tác cao bất thường. Phân tích theo phân khúc bằng việc chia dữ liệu thành các nhóm (theo thời lượng, ngôn ngữ, v.v.) để phát hiện các mẫu tương tác đặc trưng. Mô hình dự đoán để phát triển mô hình dự đoán khả năng viral của nội dung dựa trên các đặc điểm ban đầu. Cuối cùng là phân tích theo chuỗi thời gian khám phá sự thay đổi trong mẫu tương tác theo thời gian để hiểu rõ hơn về chu kỳ sống của nội dung.

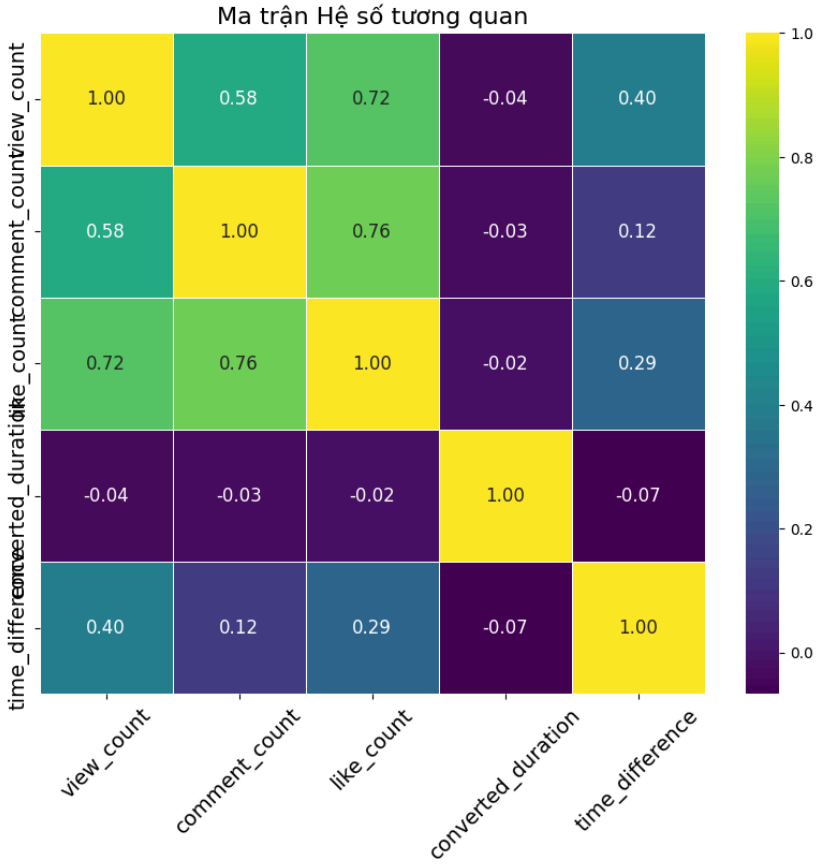
Ma trận hệ số tương quan

Ma trận hệ số tương quan dùng để đo lường và sự liên quan của hai hay nhiều biến trong tập dữ liệu. Khi hai biến số có sự tương quan cao thì khi một biến số thay đổi có xu hướng kéo theo biến số còn lại có giá trị thay đổi cùng chiều.

Biểu đồ này là một ma trận hệ số tương quan (correlation matrix) giữa các biến khác nhau. Hệ số tương quan đo lường mối quan hệ tuyến tính giữa hai biến, với giá trị từ -1 đến 1. Giá trị 1 cho thấy mối quan hệ dương hoàn hảo, -1 cho thấy mối quan hệ âm hoàn hảo, và 0 cho thấy không có mối quan hệ tuyến tính.

Ta sử dụng ma trận hệ số tương quan trong trường hợp đánh giá sự tương quan giữa nhiều biến. Trong bài nghiên cứu này, chúng tôi muốn tìm sự liên quan giữa các trường dữ liệu với nhau cũng như đánh giá mức độ biến động của các biến khi một trong các biến thay đổi cùng hoặc ngược chiều với nhau.

Để trực quan hóa ma trận tương quan, chúng tôi lựa chọn biểu đồ dải nhiệt Heatmap để thể hiện mật độ của dữ liệu tại các điểm khác nhau trên một mặt phẳng. Càng nhiều dữ liệu tập trung tại một điểm, màu sắc tại điểm đó càng đậm, ngược lại, càng ít dữ liệu thì màu sắc càng nhạt.



Hình 4.27: Ma trận hệ số tương quan.

Dựa trên ma trận tương quan và biểu đồ heatmap, ta có thể đánh giá mức độ liên quan giữa các biến với nhau. Các ô dữ liệu có giá trị trên 0.6 nghĩa là tương quan dương mạnh, các biến có sự ảnh hưởng liên tục với nhau. Các ô dữ liệu tiến đến giá trị 1.0 có sự tương quan hoàn hảo với chính nó. Còn các giá trị từ 0.3 trở xuống có sự tương quan nhưng không đáng kể.

mối tương quan dương mạnh giữa *view_count* và *like_count* (0.72), và giữa *comment_count* và *like_count* (0.76). Mối tương quan giữa *view_count* và *comment_count* cũng khá mạnh (0.58). Điều này cho thấy các video có nhiều lượt xem thường cũng có xu hướng nhận được nhiều lượt thích và bình luận hơn, và ngược lại.

Tuy nhiên, mối tương quan giữa *converted_duration* và các biến khác là rất yếu (gần 0 hoặc âm nhẹ), cho thấy thời lượng video không có mối tương quan tuyến tính đáng kể với số lượt xem, bình luận hoặc lượt thích.

Tương tự, mối tương quan giữa *time_difference* và các biến khác cũng tương đối yếu. Mặc dù có một tương quan dương nhẹ giữa *time_difference* và *view_count* (0.40) và *like_count* (0.29), nhưng mối tương quan này không đủ mạnh để kết luận về một mối quan hệ tuyến tính đáng kể. Mối tương quan giữa *time_difference* và *comment_count* thậm chí còn yếu hơn (0.12).

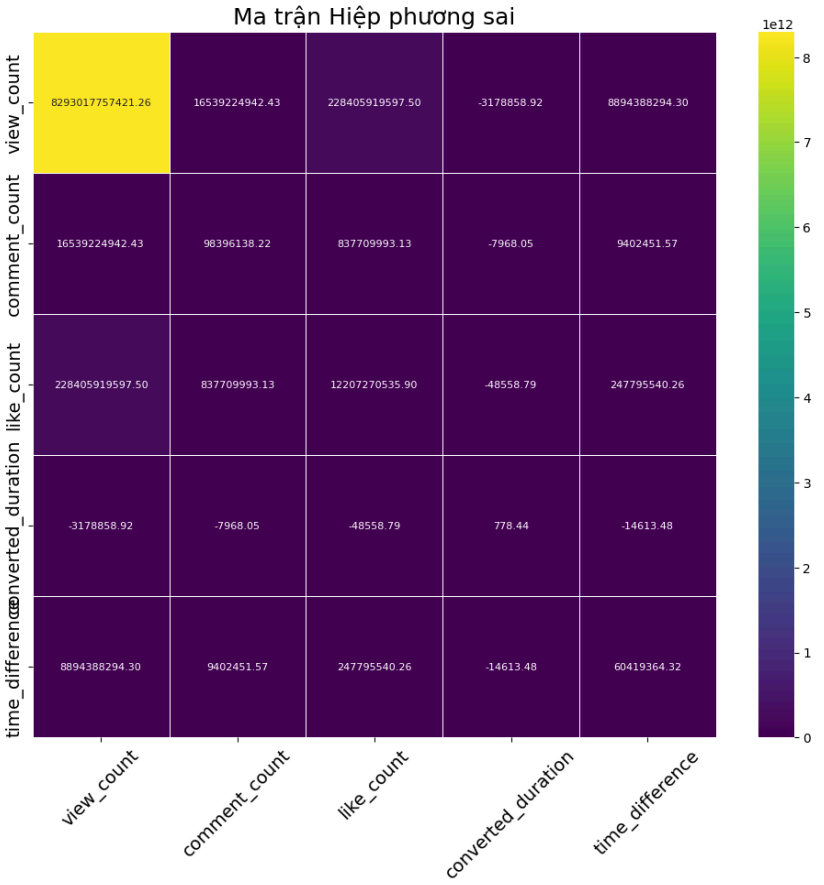
Thang màu ở bên phải cho biết cường độ và hướng của các tương quan, với màu vàng đại diện cho tương quan dương mạnh, màu xanh đậm đại diện cho tương quan yếu hoặc không có tương quan, và màu tím đại diện cho tương quan âm.

Ma trận tương quan này giúp xác định các chỉ số nào có xu hướng biến động cùng nhau và chỉ số nào có mối quan hệ ngược lại. Đối với người sáng tạo nội dung hoặc nhà phân tích, những mối quan hệ này có thể cung cấp thông tin chi tiết về mô hình tương tác, động lực cảm xúc và cách các chỉ số khác nhau ảnh hưởng lẫn nhau.

Ma trận hiệp phương sai

Hiệp phương sai là một trong những khái niệm cơ bản trong khoa học dữ liệu và học máy. Nó được sử dụng để đánh giá mức độ tương quan giữa các biến trong quá trình phân tích dữ liệu và xây dựng các mô hình dự đoán.

Hiệp phương sai xác định hướng của mối quan hệ giữa hai biến. Nó không suy nghĩ về sức mạnh của mối quan hệ. Nó cho chúng ta biết tỷ lệ giữa hai biến. Hiệp phương sai có thể là bất kỳ số thực nào. Nó phụ thuộc vào phương sai của các biến và tỷ lệ của ánh xạ. Nó có thể được tính bằng tích của tổng các chênh lệch trung bình từ tập hợp biến chia cho tổng số phần tử. Hiệp phương sai trong khoa học dữ liệu được sử dụng để phân tích dữ liệu để hiểu những diễn biến trong quá khứ. Hành vi của các biến khác nhau thay đổi với sự thay đổi trong một yếu tố. Điều đó có thể được sử dụng để hiểu rõ hơn những gì đang xảy ra. Hiệp phương sai có thể cung cấp một sự hiểu biết cơ bản về mối quan hệ giữa các biến. Biến có thể tỉ lệ thuận hoặc tỉ lệ nghịch.Để trực quan hóa ma trận hiệp phương sai, chúng tôi sử dụng biểu đồ dải nhiệt Heatmap để thể hiện sự tương quan tuyến tính giữa các biến với nhau. Với các dải màu thay đổi từ nhạt sang đậm thể hiện chiều vector của các biến định lượng.



Hình 4.28: Ma trận hiệp phương sai.

Các giá trị trên đường chéo chính biểu thị phương sai của từng biến. Có thể thấy rằng *view_count*, *comment_count*, và *like_count* có phương sai rất lớn, cho thấy sự phân tán rộng rãi của dữ liệu cho các biến này, phù hợp với các phân tích trước về phân phối lệch phải và sự tồn tại của các giá trị ngoại lai. Phương sai của *converted_duration* và *time_difference* nhỏ hơn đáng kể.

Các giá trị ngoài đường chéo chính thể hiện hiệp phương sai giữa các cặp biến. Có hiệp phương sai dương lớn giữa *view_count* và *like_count* (2.28e+12), giữa *view_count* và *comment_count* (1.65e+12), và giữa *comment_count* và *like_count* (8.38e+11). Điều này cho thấy khi số lượt xem tăng, số lượt thích và bình luận cũng có xu hướng tăng theo, và tương tự cho các cặp biến khác.

Hiệp phương sai giữa *converted_duration* và các biến khác rất nhỏ, gần 0 hoặc âm nhẹ, cho thấy mối quan hệ tuyến tính yếu giữa thời lượng video và các chỉ số tương tác hoặc thời gian đăng tải. Tương tự, hiệp phương sai giữa *time_difference* và các biến khác cũng tương đối nhỏ so với hiệp phương sai giữa các chỉ số tương tác chính.

Nhìn chung, về mặt tương tác người dùng có sự chênh lệch lớn trong mức độ tương tác giữa các nội dung. Các biến như *like_count* và *comment_count* có mối quan hệ chặt chẽ với nhau. Về mặt thời gian thì thời gian có ảnh hưởng đáng kể đến mức độ tương tác, còn thời lượng video ít ảnh hưởng đến các yếu tố khác. Còn về mặt cảm xúc, các chỉ số cảm xúc có độ biến thiên thấp. Dựa trên các kết luận trên, ta có thể khuyến nghị các người sáng tạo nội dung trên Youtube tập trung vào việc tối ưu hóa thời điểm đăng nội dung, có thể xem xét tăng tính đa dạng ngôn ngữ để tăng tương tác.

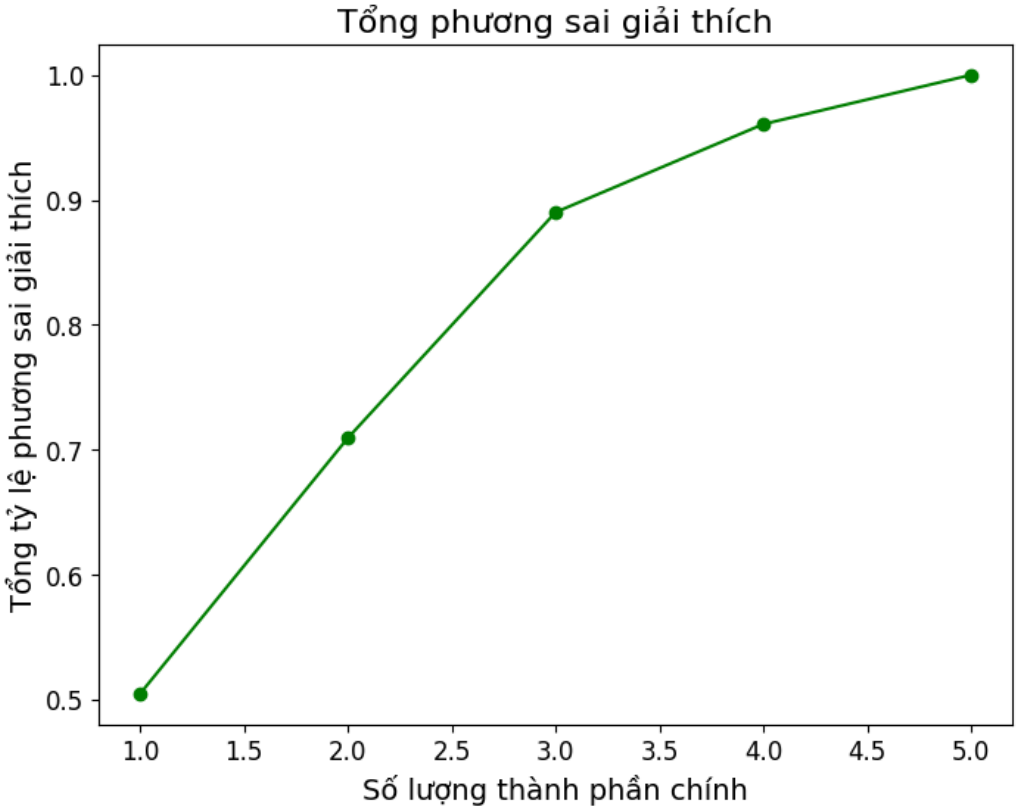
PCA

PCA là một thuật toán thống kê được sử dụng rộng rãi nhất trong việc giảm kích thước của một tập dữ liệu. Phương pháp PCA được giới thiệu lần đầu tiên bởi Karl Pearson vào năm 1901, được Harold Hotelling phát triển độc lập và đặt tên vào những năm 1930.

Ý tưởng chính của phương pháp PCA là giảm kích thước của một tập dữ liệu gồm một lượng lớn các biến có liên quan lẫn nhau, đồng thời, giữ lại càng nhiều càng tốt sự biến thiên trong tập dữ liệu. Sự giảm thiểu này đạt được bằng cách chuyển đổi thành một tập hợp các biến mới, các thành phần chính (PC) bằng sự kết hợp tuyến tính các biến ban đầu trong dữ liệu, được chuẩn hóa về giá trị trung bình bằng 0 và phương sai đơn vị. Các PC được chọn sao cho chúng không có mối liên hệ với nhau và chúng được sắp xếp sao cho thành phần đầu tiên nắm bắt được lượng biến động lớn nhất có thể trong

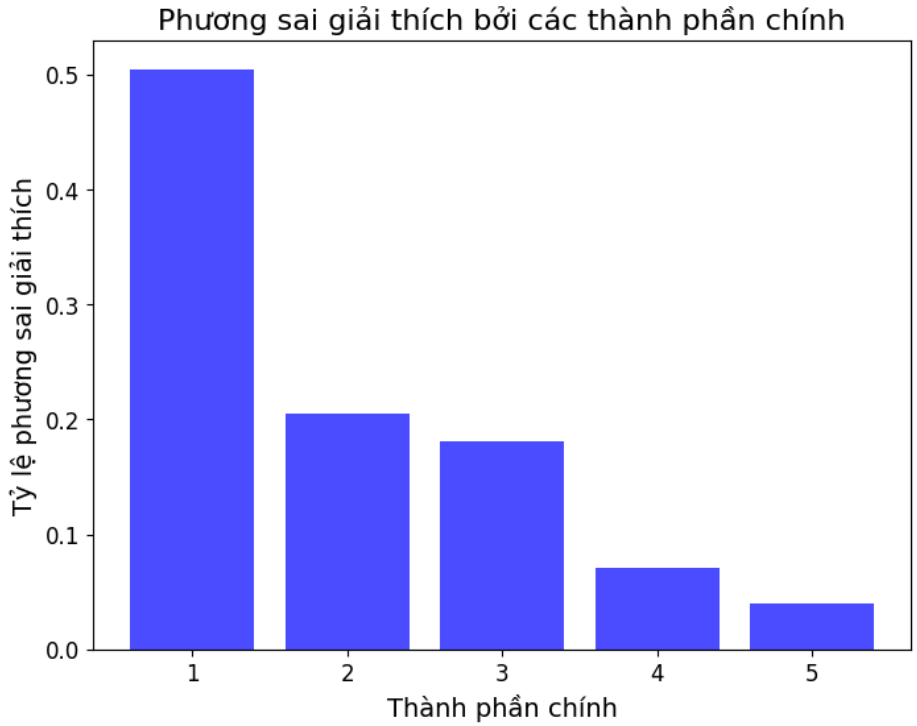
dữ liệu và các thành phần tiếp theo nắm bắt ngày càng ít hơn. Thông thường, các tính năng chính trong dữ liệu chỉ có thể được nhìn thấy từ hai hoặc ba PC đầu tiên.

Nói một cách đơn giản hơn, phương pháp PCA sử dụng phép biến đổi trực giao để biến đổi một tập hợp dữ liệu từ một không gian nhiều chiều sang một không gian mới ít chiều hơn (2 hoặc 3 chiều) nhằm tối ưu hóa việc thể hiện sự biến thiên của dữ liệu.



Hình 4.29: Tổng phương sai giải thích.

Tổng phương sai giải thích (Explained Variance Ratio): Đây là tỷ lệ phương sai được giải thích bởi mỗi thành phần chính. Trục tung (từ 0.5 đến 1.0) biểu thị tỷ lệ phương sai được giải thích, trong khi trục hoành biểu thị số lượng thành phần chính. Đường cong này cho thấy mức độ thông tin (phương sai) được giữ lại khi số lượng thành phần chính tăng lên.



Hình 4.30: Tổng tỷ lệ phương sai giải thích.

Biểu đồ cột "Phương sai giải thích bởi các thành phần chính" trực quan hóa tỷ lệ phương sai mà mỗi thành phần chính nắm giữ sau khi thực hiện phân tích PCA. Cột đầu tiên, tương ứng với Thành phần chính 1 (PC1), cao nhất, cho thấy nó giải thích khoảng 50.5% tổng phương sai của dữ liệu. Điều này có nghĩa là PC1 là thành phần quan trọng nhất, nắm bắt phần lớn sự biến động hoặc thông tin trong dữ liệu gốc.

Cột thứ hai, đại diện cho Thành phần chính 2 (PC2), thấp hơn đáng kể, giải thích khoảng 20.5% phương sai. Điều này cho thấy PC2 vẫn đóng góp một lượng thông tin quan trọng, nhưng ít hơn nhiều so với PC1.

Thành phần chính thứ ba (PC3) giải thích khoảng 18% phương sai, tiếp tục xu hướng giảm dần về lượng thông tin được nắm giữ bởi các thành phần chính kế tiếp.

Hai thành phần chính cuối cùng, PC4 và PC5, có tỷ lệ phương sai giải thích nhỏ hơn nhiều, lần lượt là khoảng 7% và 4%. Điều này cho thấy chúng chỉ nắm bắt được một phần nhỏ còn lại của sự biến động trong dữ liệu.

Nhìn chung, biểu đồ này cho thấy rằng các thành phần chính đầu tiên (đặc biệt là PC1 và PC2) là quan trọng nhất trong việc tóm tắt thông tin của dữ liệu. Việc giảm chiều dữ liệu bằng cách chỉ giữ lại một vài thành phần chính đầu tiên có thể giữ lại được phần lớn phương sai và thông tin quan trọng, đồng thời loại bỏ nhiễu hoặc các chiều ít quan trọng hơn. Trong trường hợp này, có thể cân nhắc giữ lại 2 hoặc 3 thành phần chính, vì

chúng đã giải thích được một tỷ lệ đáng kể tổng phương sai (khoảng 71% với 2 PC và 89% với 3 PC).

4.2. Mô hình

Công cụ sử dụng

NumPy

NumPy là một thư viện mã nguồn mở mạnh mẽ dành cho ngôn ngữ lập trình Python, cung cấp hỗ trợ cho các mảng đa chiều lớn và các phép toán số học hiệu quả. Được viết chủ yếu bằng Python và C, NumPy cung cấp một giao diện Python đơn giản với tốc độ của mã biên dịch. Thư viện này đóng vai trò là nền tảng cho nhiều thư viện khoa học dữ liệu và máy học khác như Pandas, SciPy và Scikit-learn.

Pandas

Pandas là một thư viện mã nguồn mở mạnh mẽ và linh hoạt dành cho Python, được sử dụng rộng rãi trong thao tác và phân tích dữ liệu. Thư viện này cung cấp các cấu trúc dữ liệu như Series (một mảng một chiều có nhãn) và DataFrame (một bảng hai chiều với các cột có thể có các kiểu dữ liệu khác nhau), giúp xử lý dữ liệu dạng bảng một cách hiệu quả. citeturn0search6 Pandas hỗ trợ nhiều chức năng như làm sạch dữ liệu, chuyển đổi dữ liệu, phân tích dữ liệu, và trực quan hóa dữ liệu. Được xây dựng trên nền tảng NumPy, Pandas tối ưu hóa hiệu suất và tích hợp tốt với các thư viện khác trong hệ sinh thái Python.

Thư viện scikit-learn

Scikit-learn là thư viện machine learning quan trọng nhất trong Python cho hồi quy tuyến tính. Nó cung cấp lớp LinearRegression để triển khai phương pháp bình phương tối thiểu (OLS), cùng các phiên bản mở rộng như Ridge, Lasso với chức năng regularization giúp giảm overfitting. Thư viện này còn bao gồm Polynomial Features để tạo các biến đa thức từ biến gốc, giúp mô hình bắt được các quan hệ phi tuyến tính.

Stats Models

Statsmodels bổ sung khả năng phân tích thống kê chuyên sâu cho scikit-learn. Hàm OLS() của nó tạo mô hình hồi quy với báo cáo chi tiết thông qua phương thức summary(), cung cấp đầy đủ các chỉ số như p-value, khoảng tin cậy và hệ số xác định. Thư viện này đặc biệt mạnh trong việc kiểm tra các giả định hồi quy thông qua các kiểm định chẩn đoán như Breusch-Pagan (kiểm tra phương sai sai số không đổi) và Durbin-Watson (phát hiện tự tương quan).

Matplotlib/Seaborn

Hai thư viện trực quan hóa này đóng vai trò thiết yếu trong việc kiểm tra mô hình. Scatter plot giúp quan sát mối quan hệ giữa các biến, trong khi residual plot cho phép đánh giá tính ngẫu nhiên của sai số. Q-Q plot là công cụ hữu hiệu để kiểm tra giả định phân phối chuẩn của phần dư. Seaborn còn cung cấp regression plot với khả năng hiển thị đường hồi quy cùng khoảng tin cậy một cách trực quan.

KMeans từ sklearn.cluster

KMeans trong thư viện scikit-learn là một thuật toán phân cụm phổ biến, giúp chia tập dữ liệu thành các nhóm dựa trên sự tương đồng giữa các điểm dữ liệu.

Bao gồm các tham số chính mà bạn có thể tùy chỉnh khi sử dụng KMeans:

- *n_clusters*: Số lượng cụm (k) mà bạn muốn chia tập dữ liệu. Giá trị mặc định là 8.
- *init*: Phương pháp khởi tạo các tâm cụm ban đầu. Các tùy chọn bao gồm:
    + 'k-means++': Phương pháp khởi tạo thông minh giúp tăng tốc độ hội tụ và cải thiện chất lượng phân cụm.
    + 'random': Chọn ngẫu nhiên các tâm cụm từ tập dữ liệu.
    + Mảng numpy: Bạn có thể cung cấp trực tiếp các tọa độ của tâm cụm ban đầu.
- *n_init*: Số lần thuật toán được chạy với các khởi tạo tâm cụm khác nhau. Kết quả cuối cùng sẽ là lần chạy có tổng bình phương khoảng cách trong cụm nhỏ nhất (inertia). Giá trị mặc định là 'auto', trong đó số lần chạy phụ thuộc vào giá trị của init: 10 nếu sử dụng 'random' hoặc hàm do người dùng định nghĩa; 1 nếu sử dụng 'k-means++' hoặc mảng numpy.
- *max_iter*: Số lần lặp tối đa cho mỗi lần chạy của thuật toán k-means. Giá trị mặc định là 300.
- *tol*: Ngưỡng dung sai để xác định sự hội tụ. Nếu sự thay đổi của tổng bình phương khoảng cách trong cụm giữa hai lần lặp liên tiếp nhỏ hơn giá trị này, thuật toán sẽ dừng. Giá trị mặc định là 1e-4.
- *algorithm*: Thuật toán được sử dụng để giải quyết bài toán k-means. Các tùy chọn bao gồm:
    + *'lloyd'*: Thuật toán Lloyd tiêu chuẩn.
    + *'elkan'*: Sử dụng biến đổi Elkan để tăng tốc độ hội tụ (chỉ áp dụng cho khoảng cách Euclidean).
    + *'auto'*: Tự động chọn giữa 'lloyd' và 'elkan' dựa trên bản chất của dữ liệu.

silhouette_score, davies_bouldin_score từ sklearn.metrics

Silhouette score (điểm silhouette) là một chỉ số dùng để đánh giá chất lượng của việc phân cụm trong phân tích dữ liệu. Giá trị của điểm silhouette dao động từ -1 đến 1, phản ánh mức độ mà một điểm dữ liệu thuộc về cụm của nó:

- Gần 1: điểm dữ liệu được phân cụm tốt, nằm gần trung tâm của cụm.
- Gần 0: điểm dữ liệu nằm trên biên giới giữa hai cụm, có thể thuộc về cụm này hoặc cụm kia.
- Gần -1: điểm dữ liệu có thể đã bị phân vào cụm sai, vì nó gần với cụm khac hơn.

Điểm silhouette s(i) được tính bằng công thức:

$$s(i) = \frac{a(i) - b(i)}{max\{a(i), b(i)\}}$$

Trong đó:

- $a(i)$: khoảng cách trung bình giữa điểm dữ liệu i và tất cả các điểm khác trong cùng cụm.
- $b(i)$: khoảng cách trung bình giữa điểm dữ liệu i và tất cả các điểm trong cụm gần nhất khác (cụm lân cận).

Chỉ số Davies-Bouldin (Davies-Bouldin Index - DBI) là một thước đo được sử dụng để đánh giá chất lượng của các thuật toán phân cụm. Được giới thiệu bởi David L. Davies và Donald W. Bouldin vào năm 1979, chỉ số này giúp xác định mức độ phân tách và sự chặt chẽ của các cụm trong tập dữ liệu. Giá trị DBI càng thấp cho thấy các cụm càng được phân tách rõ ràng và chặt chẽ.

DBI được tính dựa trên tỷ lệ giữa khoảng cách nội cụm (intra-cluster-distance) và khoảng cách liên cụm (inter-cluster-distance). Cụ thể, đối với mỗi cụm C_i ta tính:

- Độ phân tán nội cụm (S_i): là khoảng cách trung bình giữa các điểm dữ liệu trong cụm C_i và tâm cụm của nó.
- Khoảng cách liên cụm ($M_{i,j}$): là khoảng cách giữa tâm C_i và tâm của cụm C_i.

Sau đó, với mỗi cụm (i, j) ta tính tỷ lệ:

$$R_{i,j} = \frac{S_i + S_j}{M_{ij}}$$

Chỉ số $R_{i,j}$ đại diện cho mức độ tương đồng giữa hai cụm. Tiếp theo, với mỗi cụm i, ta xác định:

$$D_i = max_{j \neq i} R_{ij}$$

Cuối cùng, chỉ số Davies-Bouldin được tính bằng trung bình của tất cả các D_i:

$$DB = \frac{1}{N}\sum_{i=1}^{N} D_i$$

Trong đó, N là số lượng cụm.

KNeighborsRegressor từ sklearn.neighbors

KNeighborsRegressor là một lớp trong thư viện scikit-learn, được sử dụng để thực hiện hồi quy dựa trên thuật toán k-láng giềng gần nhất (k-Nearest Neighbors - KNN). Phương pháp này dự đoán giá trị mục tiêu cho một điểm dữ liệu mới bằng cách nội suy từ các giá trị mục tiêu của k láng giềng gần nhất trong tập huấn luyện.

Các tham số chính của *KNeighborsRegressor*:

- *n_neighbors* (mặc định=5): Số lượng láng giềng được sử dụng để dự đoán.
- *weights* (mặc định=*'uniform'*): Hàm trọng số được sử dụng trong dự đoán. Có thể là:
    + *'uniform'*: Tất cả các điểm trong mỗi láng giềng được trọng số như nhau.
    + *'distance'*: Trọng số các điểm theo nghịch đảo của khoảng cách, nghĩa là các láng giềng gần hơn sẽ có ảnh hưởng lớn hơn đến dự đoán.
    + Hàm do người dùng định nghĩa nhận một mảng khoảng cách và trả về một mảng trọng số cùng hình dạng.
- *algorithm* (mặc định=*'auto'*): Thuật toán được sử dụng để tính toán các láng giềng gần nhất:
    + *'ball_tree'*: Sử dụng cấu trúc dữ liệu BallTree.
    + *'kd_tree'*: Sử dụng cấu trúc dữ liệu KDTree.
    + *'brute'*: Sử dụng tìm kiếm vét cạn.
    + *'auto'*: Tự động chọn thuật toán phù hợp dựa trên dữ liệu huấn luyện.
- *leaf_size* (mặc định=30): Kích thước lá được truyền vào BallTree hoặc KDTree. Tham số này có thể ảnh hưởng đến tốc độ xây dựng và truy vấn, cũng như bộ nhớ cần thiết để lưu trữ cây. Giá trị tối ưu phụ thuộc vào bản chất của vấn đề.
- *p* (mặc định=2): Tham số quyền hạn cho khoảng cách Minkowski metric. Khi *p=1*, nó tương đương với khoảng cách Manhattan, và khi *p=2*, nó tương đương với khoảng cách Euclidean.

mean_absolute_error (MAE), mean_squared_error (MSE), r2_score từ sklearn.metrics

Trong scikit-learn, các hàm mean_absolute_error, mean_squared_error và r2_score trong module sklearn.metrics được sử dụng để đánh giá hiệu suất của các mô hình hồi quy.

Mean Absolute Error (MAE):

MAE đo lường giá trị trung bình của các sai số tuyệt đối giữa giá trị thực tế và giá trị dự đoán.

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

Trong đó:

- n à số lượng điểm dữ liệu.
- y_i là giá trị thực tế tại điểm i.
- $\hat{y}_i$ là giá trị dự đoán tại điểm i.
- $|y_i - \hat{y}_i|$ là sai số tuyệt đối của từng điểm.

MAE phản ánh mức độ sai lệch trung bình mà mô hình dự đoán so với giá trị thực tế. Giá trị MAE càng nhỏ, mô hình dự đoán càng chính xác.

Mean Squared Error (MSE):

MSE tính giá trị trung bình của bình phương sai số giữa giá trị thực tế và giá trị dự đoán.

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

Do bình phương sai số, MSE nhạy cảm hơn với các sai số lớn, tức là các sai số lớn sẽ ảnh hưởng nhiều hơn đến MSE. Giá trị MSE càng nhỏ, mô hình dự đoán càng chính xác.

R-squared (R^2) Score:

R^2 là một thước đo thống kê cho biết mức độ các biến độc lập giải thích được phương sai của biến phụ thuộc trong mô hình hồi quy.

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

Trong đó:

- $\bar{y}$ là giá trị trung bình của tất cả các giá trị thực tế.

Giá trị R^2 nằm trong khoảng từ 0 đến 1, với giá trị gần 1 cho thấy mô hình giải thích được phần lớn phương sai của dữ liệu, trong khi giá trị gần 0 cho thấy mô hình không giải thích được nhiều. Tuy nhiên, trong một số trường hợp, R^2 có thể âm, cho thấy mô hình không phù hợp với dữ liệu.

matplotlib.pyplot, mpl_toolkits.mplot3d

Trong scikit-learn, các hàm mean_absolute_error, mean_squared_error và r2_score trong module sklearn.metrics được sử dụng để đánh giá hiệu suất của các mô hình hồi quy.

T-NSE từ sklearn.manifold

t-SNE là một kỹ thuật giảm chiều dữ liệu phi tuyến tính, đặc biệt hữu ích trong việc trực quan hóa các tập dữ liệu có chiều cao. Phương pháp này chuyển đổi các điểm dữ liệu từ không gian nhiều chiều xuống không gian hai hoặc ba chiều, trong khi cố gắng bảo toàn mối quan hệ tương đồng giữa các điểm dữ liệu.

Các bước hoạt động của t-SNE:

1. Tính toán xác suất tương đồng trong không gian cao chiều:

   với mỗi cặp điểm dữ liệu x_i và x_j, t-SNE tính xác suất $p_{j|i}$ thể hiện mức độ tương đồng giữa chúng:

$$p_{ij} = \frac{p_{j|i} + p_{i|j}}{2N}$$

Trong đó:

- $||x_i - x_j||$ là khoảng cách Euclide giữa hai điểm dữ liệu.
- σ_i là tham số điều chỉnh mức độ "lân cận" (phụ thuộc vào tham số perplexity).

Xác suất đối xứng được tính như sau:

$$p_{ij} = \frac{p_{j|i} + p_{i|j}}{2N}$$

Với N là số điểm dữ liệu.

2. Tính toán xác suất trong khong gian thấp chiều:

   Tại không gian thấp chiều, xác suất tương đồng được tính bằng phân phối t-Student với một bậc tự do:

$$q_{ij} = \frac{(1 + ||y_i - y_j||^2)}{\sum_{k \neq l} (1 + ||y_k + y_l||^2)^{-1}}$$

Trong đó:

- y_i và y_j là các điểm trong không gian thấp chiều.
- q_{ij}: xác suất tương đồng.

3. Tối ưu hóa để giảm thiểu sự khác biệt giữa hai phân phối:

Hàm mất mát của t-SNE là độ lệch Kullback-Leibler (KL Divergence):

$$C = \sum_{i \neq j} p_{ij} log \frac{p_{ij}}{q_{ij}}$$

Mục tiêu là tối thiểu hóa hàm mất mát thông qua phương pháp Gradient Descent, sao cho phân phối trong không gian thấp chiều q_{ij} gần giống với phân phối p_{ij} trong không gian cao chiều.

t-SNE được biết đến rất hiệu quả trong việc phân nhóm và trực quan hóa dữ liệu phi tuyến, giúp hiển thị trực quan các mẫu ẩn, cụm và cấu trúc trong dữ liệu mà các phương pháp tuyến tính như PCA có thể không phát hiện được. Tuy nhiên, t-SNE không bảo toàn khoảng cách toàn cục, do đó khoảng cách giữa các cụm trong không gian thấp chiều không nhất thiết phản ánh khoảng cách thực tế trong không gian cao chiều.

StandardScaler

StandardScaler là một công cụ trong thư viện scikit-learn của Python, được sử dụng để chuẩn hóa các đặc trưng (features) bằng cách loại bỏ giá trị trung bình và chia cho độ lệch chuẩn, giúp các đặc trưng có trung bình bằng 0 và phương sai bằng 1.

$$x' = \frac{x - \mu}{\sigma}$$

Trong đó:

- x: giá trị của dữ liệu gốc.
- μ: giá trị trung bình feature.
- σ: độ lệch chuẩn.

StandardScaler hoạt động hiệu quả không chỉ trong cải thiện hiệu suất mô hình mà còn đảm tính chính xác của kết quả dự đoán.

Việc chuẩn hóa dữ liệu được coi là cần thiết đối với các mô hình nhạy cảm với thang đo do một vài mô hình phụ thuộc vào khoảng cách giữa các điểm dữ liệu, nên nếu dữ liệu không được chuẩn hóa, kết quả phân tích có thể bị sai lệch và hiệu suất của mô hình sẽ giảm sút đáng kể.

Ngoài ra, trong tình huống dữ liệu không tuân theo phân phối chuẩn, MinMaxScaler được khuyến nghị sử dụng vì công cụ này có khả năng đưa các đặc trưng về cùng một thang đo, bất kể hình dạng của phân phối dữ liệu. Do đó, việc lựa chọn công

cụ chuẩn hóa phải được thực hiện một cách cẩn thận, dựa trên đặc điểm cụ thể của dữ liệu và yêu cầu của mô hình.

PCA

PCA (Principal Component Analysis) là một phương pháp giảm chiều dữ liệu bằng cách biến đổi các đặc trưng ban đầu thành các các thành phần chính (principal components) mà vẫn giữ được phần lớn thông tin của dữ liệu. Cụ thể, phương pháp này sẽ tìm ra các hướng (thành phần chính) mà dữ liệu có độ phân tán lớn nhất. Mục tiêu là chuyển đổi dữ liệu từ không gian ban đầu sang một không gian mới có ít chiều hơn nhưng vẫn giữ được thông tin quan trọng nhất.

Công thức tính trong PCA:

1. Tính ma trận hiệp phương sai (Covariance Matrix):

$$C = \frac{1}{N} X'^{T} X'$$

Trong đó:

- X' là dữ liệu đã chuẩn hóa.
- C là ma trận hiệp phương sai kích thước $d \times d$ (với d là số đặc trưng), cho biết mối quan hệ tuyến tính giữa các đặc trưng trong dữ liệu.

2. Tính các vector riêng (Eigenvectors) và giá trị riêng (Eigenvalues):

Giải phương trình:

$$Cv = \lambda v$$

Trong đó:

- v là vector riêng (principal component - thành phần chính).
- λ là giá trị riêng tương ứng, thể hiện độ biến thiên mà thành phần đó giải thích.

3. Sắp xếp các vector riêng theo giá trị riêng giảm dần:

   Chọn ra k vector riêng tương ứng với k giá trị riêng lớn nhất (k là số chiều mong muốn sau khi giảm chiều dữ liệu).

4. Chiếu dữ liệu lên không gian mới:

$$Z = X'W$$

Trong đó:

- W là ma trận chứa các vector riêng (principal components).
- Z là dữ liệu mới trong không gian k chiều.

  PCA (Phân tích Thành phần Chính) có những ưu và nhược điểm đáng chú ý.

Về ưu điểm, PCA giúp giảm chiều dữ liệu trong khi vẫn giữ được thông tin quan trọng, từ đó loại bỏ các đặc trưng dư thừa và cải thiện hiệu suất của các mô hình học máy. Ngoài ra, PCA còn hỗ trợ hiệu quả trong việc trực quan hóa dữ liệu, đặc biệt khi xử lý các tập dữ liệu có nhiều chiều.

Tuy nhiên, PCA cũng có những hạn chế. Phương pháp này chỉ phát hiện được các mối quan hệ tuyến tính giữa các đặc trưng, do đó không phù hợp với các dữ liệu có cấu trúc phi tuyến tính. Bên cạnh đó, PCA dễ bị ảnh hưởng bởi dữ liệu nhiễu, làm giảm độ chính xác và hiệu quả của kết quả phân tích.

SMOTE

SMOTE (Synthetic Minority Over-sampling Technique) là một kỹ thuật được sử dụng trong học máy để xử lý vấn đề mất cân bằng dữ liệu. Khi một tập dữ liệu có sự chênh lệch lớn giữa số lượng mẫu của các lớp, các mô hình học máy thường có xu hướng thiên vị về phía lớp đa số, dẫn đến hiệu suất kém trong việc dự đoán lớp thiểu số.

Cách hoạt động của SMOTE:

SMOTE tạo ra các mẫu tổng hợp mới bằng cách chọn ngẫu nhiên k láng giềng gần nhất sau đó tạo các điểm dữ liệu mới bằng cách nội suy giữa mẫu hiện tại và các láng giềng này.

Phương pháp này giúp mở rộng không gian mẫu của lớp thiểu số, làm cho ranh giới quyết định của mô hình trở nên tổng quát hơn và giảm thiểu overfitting.

SMOTE đem lại nhiều lợi ích trong xử lý tập dữ liệu mất cân bằng. Cụ thể, SMOTE cải thiện hiệu suất mô hình qua việc tạo thêm dữ liệu cho lớp thiểu số, qua đó triệt tiêu thiên vị về phía lớp đa số. Hơn nữa, kỹ thuật này củng cố khả năng tổng quát hóa, giúp mô hình nhận diện hiệu quả các mẫu thuộc lớp thiểu số.

Dù vậy, SMOTE cũng vấp phải một số trở ngại. Thứ nhất, nguy cơ phát sinh các mẫu không thực tế khi dữ liệu lớp thiểu số phân tán rộng, dẫn đến nội suy lệch khỏi phân phối gốc. Thứ hai, SMOTE bất lực trước vấn đề chồng chéo giữa các lớp, khiến các mẫu tổng hợp dễ rơi vào vùng giao thoa, gây trở ngại cho mô hình trong việc phân biệt các lớp.

Inertia

Inertia (quán tính) là một thước đo thể hiện mức độ chặt chẽ của các điểm dữ liệu trong cùng một cụm. Nó được tính bằng tổng bình phương khoảng cách giữa mỗi điểm dữ liệu và tâm cụm của nó.

$$Inertia = \sum_{i=0}^{n} \min_{\mu_j \in C} ||x_i - \mu_j||^2$$

Với:

- x_i là một điểm dữ liệu.
- μ_j là tâm cụm của cụm C.

Inertia giúp đánh giá mức độ gắn kết của các điểm dữ liệu trong từng cụm. Khi giá trị inertia càng thấp, các điểm dữ liệu càng gần tâm cụm, đồng nghĩa với việc cụm được hình thành chặt chẽ hơn và có cấu trúc rõ ràng hơn. Tuy nhiên, Inertia luôn giảm khi số cụm tăng, bởi vì khi có nhiều cụm hơn, các điểm dữ liệu sẽ được phân chia nhỏ hơn, dẫn đến khoảng cách đến tâm cụm cũng giảm. Quan trọng hơn, inertia giả định rằng tất cả các cụm đều có hình dạng lồi và kích thước tương đồng, trong khi trên thực tế, dữ liệu có thể phân bố theo nhiều kiểu phức tạp hơn.

Phương thức Elbow

Phương pháp Elbow (Khuỷu tay) là một kỹ thuật phổ biến trong phân cụm K-Means, được sử dụng để xác định số lượng cụm tối ưu kk cho một tập dữ liệu. Mục tiêu của phương pháp này là tìm ra điểm mà việc tăng thêm số cụm không còn mang lại lợi ích đáng kể trong việc giảm tổng bình phương khoảng cách trong cụm (Within-Cluster Sum of Squares - WCSS).

Elbow (khuỷu tay) là điểm mà sau đó, việc tăng số lượng cụm k không làm giảm WCSS đáng kể nữa và điểm này thường được coi là số lượng cụm tối ưu.

4.2.1. RANDOM FOREST

4.2.1.1.Chuẩn bị dữ liệu

Dữ liệu đã được xử lý loại bỏ outliers, noise, … của các cột có giá trị định lượng (bao gồm: *comment_count, like_count, converted_duration, time_difference*) sau khi được cân bằng số lượng mẫu (dựa trên *dbscan_label*, thông qua SMOTE) được lưu trữ tại biến x.

Tương tự, dữ liệu cột *dbscan_label* được xử lý và lưu vào biến y.

Ta có tất cả 1664 điểm dữ liệu. Sau đó ta chia tập dữ liệu với tỉ lệ tập kiểm tra: tập huấn luyện = 0.3 : 0.7

4.2.1.2. Triển khai

Đầu tiên, triển khai hàm tune_hyperparameter() để tìm ra giá trị các siêu tham số cho kết quả mô hình tối ưu.

Hàm tune_hyperparameter() định nghĩa như sau:

```
def tune_hyperparameters(x_train, y_train, scoring):
  rfmodel = RandomForestClassifier(random_state=42)
  param_grid = {
      'n_estimators': np.arange(50,201,10),
      'max_depth': np.arange(5,15),
      'min_samples_split': np.arange(2,21,2),
      'min_samples_leaf': [1, 2, 4]
  }
  # Perform Random Search
  random_search = RandomizedSearchCV(rfmodel, param_grid, n_iter=20, cv=5,
                                     scoring=scoring, random_state=42, n_jobs=-1, verbose=2)
  random_search.fit(x_train, y_train)

  # Best parameters
  print("Siêu tham số tối ưu:", random_search.best_params_)
  print("Độ chính xác tốt nhất:", random_search.best_score_)
```

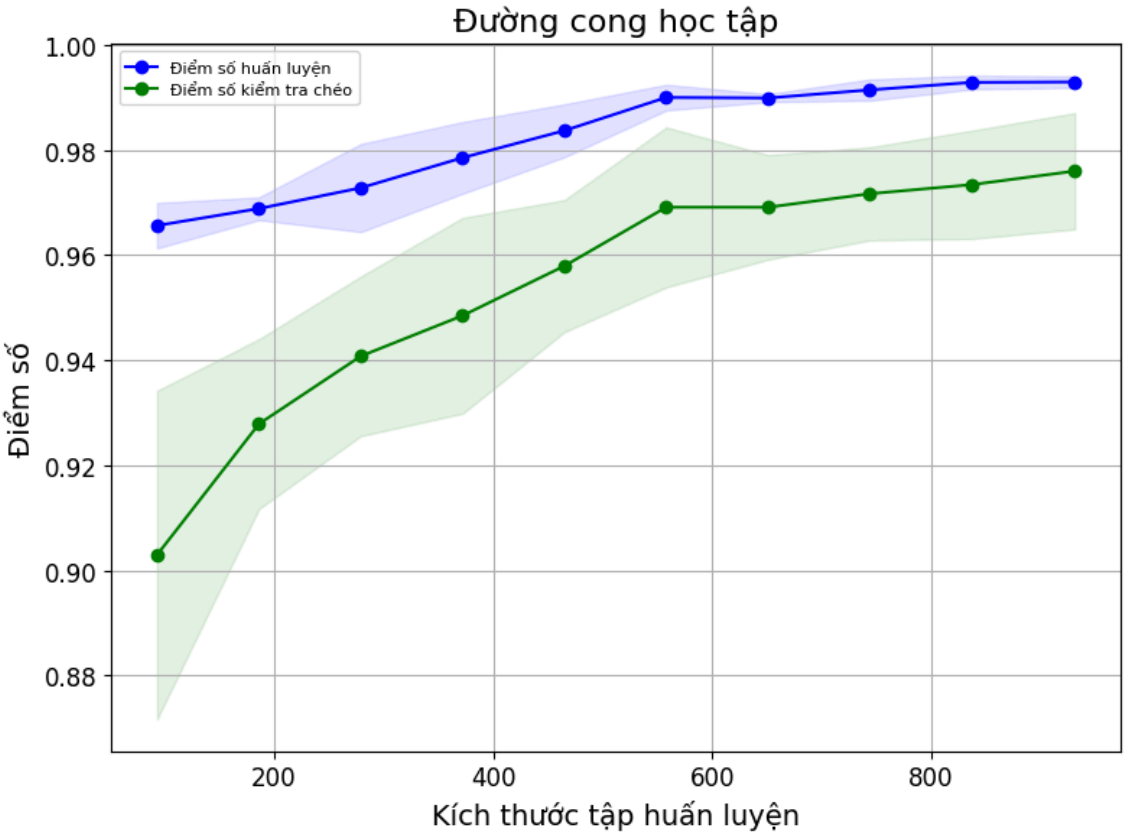
Hình 4.31: Hàm xác định siêu tham số cho mô hình Random Forest.

Trong đó, cách chọn các giá trị *n_estimators, max_depth, min_namples_split, min_sample_leaf* nhằm đảm bảo số lượng cây trong mô hình đủ mạnh để học được đặc trưng dữ liệu nhưng không gây overfitting và vẫn tối ưu được thời gian huấn luyện.

Kết quả cho thấy mô hình có thể đạt độ chính xác cao nhất ở ngưỡng 0.98 với các giá trị siêu tham số như sau:

- *n_estimator:* 140
- *min_samples_split:* 4
- *min_samples_leaf:* 4
- *max_depth:* 12

Tiếp theo, ta huấn luyện mô hình với các siêu tham số vừa tìm được. Hiệu suất mô hình như sau:



Hình 4.32: Đường cong học tập của mô hình Random Forest

Đồ thị chứng tỏ mô hình học rất tốt trên tập huấn luyện (gần 100%). Khoảng cách giữa 2 đường cũng thu hẹp dần khi tăng kích thước dữ liệu, cho thấy overfitting đã giảm đáng kể với nhiều dữ liệu hơn.

```
KẾT QUẢ CROSS-VALIDATION (CV=5):
Độ chính xác trung bình: 0.9768 ± 0.0117
Điểm số từng lần: 0.9571, 0.9700, 0.9871, 0.9828, 0.9871
```

Hình 4.33: Kết quả cross-validation của mô hình Random Forest

Tất cả các điểm số đều trên 0.96, chứng tỏ sự ổn định của mô hình. Độ lệch nhỏ giữa các chỉ số này cũng cho thấy mô hình ít bị ảnh hưởng bởi dữ liệu huấn luyện khác nhau.

```
-----CHỈ SỐ ĐÁNH GIÁ HIỆU SUẤT MÔ HÌNH-----
Độ chính xác:  0.9740
Độ chính xác tuyệt đối: 0.9757
Độ nhạy:    0.9740
Điểm F1:  0.9739
```

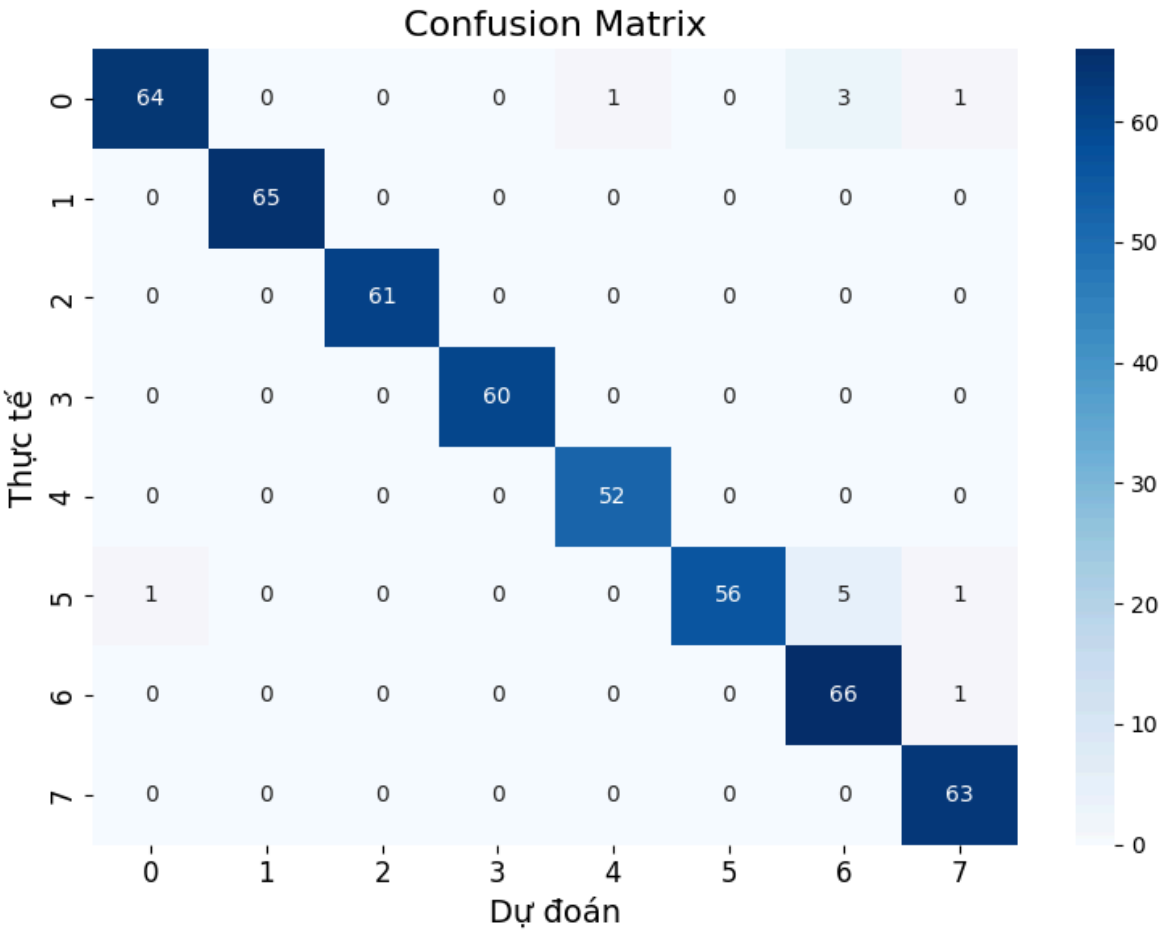
Hình 4.34:Các chỉ số đánh giá mô hình Random Forest.

-----BÁO CÁO PHÂN LOẠI-----

	precision	recall	f1-score	support
-1	0.9846	0.9275	0.9552	69
0	1.0000	1.0000	1.0000	65
1	1.0000	1.0000	1.0000	61
2	1.0000	1.0000	1.0000	60
3	0.9811	1.0000	0.9905	52
4	1.0000	0.8889	0.9412	63
5	0.8919	0.9851	0.9362	67
6	0.9545	1.0000	0.9767	63
accuracy			0.9740	500
macro avg	0.9765	0.9752	0.9750	500
weighted avg	0.9757	0.9740	0.9739	500

Hình 4.35: Báo cáo phân loại mô hình Random Forest.

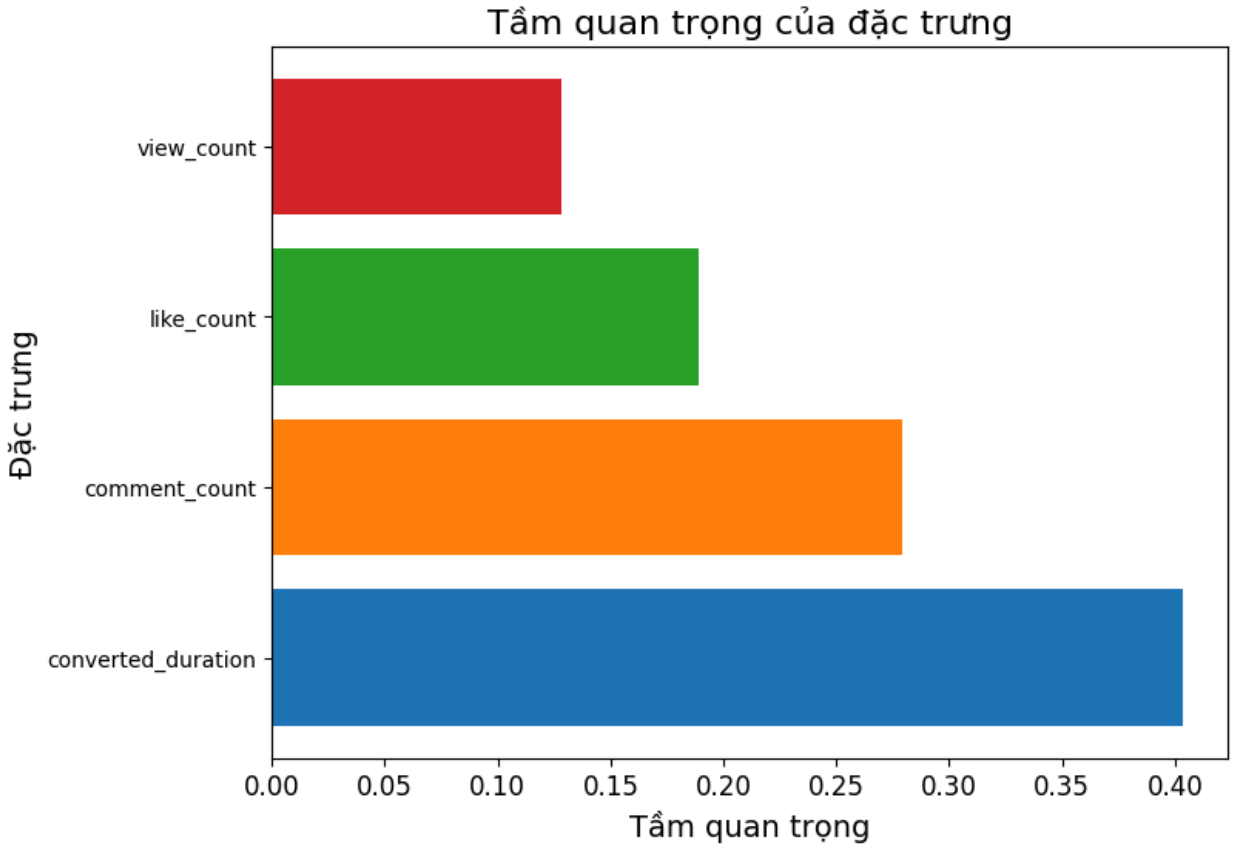
Các lớp 0, 1, 2 đạt điểm tuyệt đối 1.0; các lớp còn lại có điểm số đều trên 0.9.



Hình 4.36: Confusion Matrix của mô hình Random Forest.

Ở confusion matrix, phần lớn giá trị nằm ở đường chéo, tức dự đoán đúng hầu hết. Các trường hợp phân loại sai xảy ra với lớp 0, 1, 4, 5, 6, 7.

Với hiệu suất như trên, ta có mức độ quan trọng của các đặc trưng như sau:



Hình 4.37: Tầm quan trọng các đặc trưng trong mô hình Random Forest.

Kết quả này cho thấy, mặc dù video thịnh hành bằng phương thức nào và có các đặc điểm khác như thế nào, thời lượng video vẫn là yếu tố quan trọng nhất phía sau những video thịnh hành. Trên thực tế, các video ngắn dễ tiếp cận hơn và các video dài có cơ hội tạo ra nhiều tương tác hơn.

Số lượng bình luận cũng là chỉ số quan trọng giúp đánh giá mức độ thảo luận và phản hồi từ cộng đồng. Kết quả mô hình cho thấy YouTube đã ưu tiên những video có nhiều bình luận hơn trong việc đề xuất.

So với 2 yếu tố trên, lượt xem và số lượt xem có tầm quan trọng thấp hơn. Lượt thích có thể không phản ánh toàn bộ mức độ phổ biến. Một số video có thể nhận được nhiều lượt xem nhưng ít lượt thích vì chủ đề gây tranh cãi. Còn lượt xem hiển nhiên là một yếu tố cơ bản để xác định mức độ phổ biến của video, nhưng do sự chênh lệch lượt xem giữa các mã vùng mà có sự không chính xác khi phân loại như vậy.

4.2.2. KNN

4.2.2.1. Chuẩn bị dữ liệu

Dữ liệu đã được xử lý loại bỏ outliers, noise, … của các cột có giá trị định lượng (bao gồm: *comment_count, like_count, converted_duration, time_difference)* được lưu trữ tại biến x.

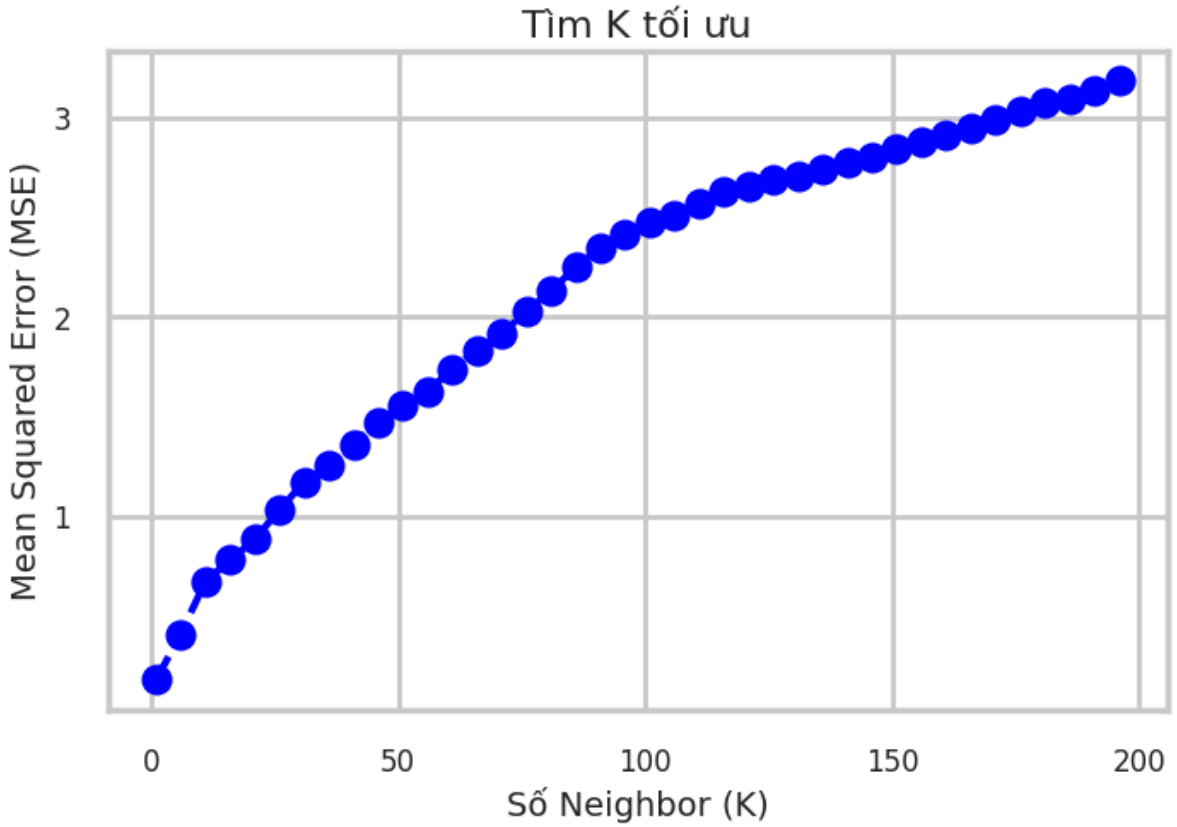
Tương tự, dữ liệu cột *view_count* được xử lý và lưu vào biến y.

Tiếp theo, bộ dữ liệu được chia làm 2 phần: 0.2 tập kiểm tra : 0.8 tập huấn luyện.

4.2.2.2. Triển khai

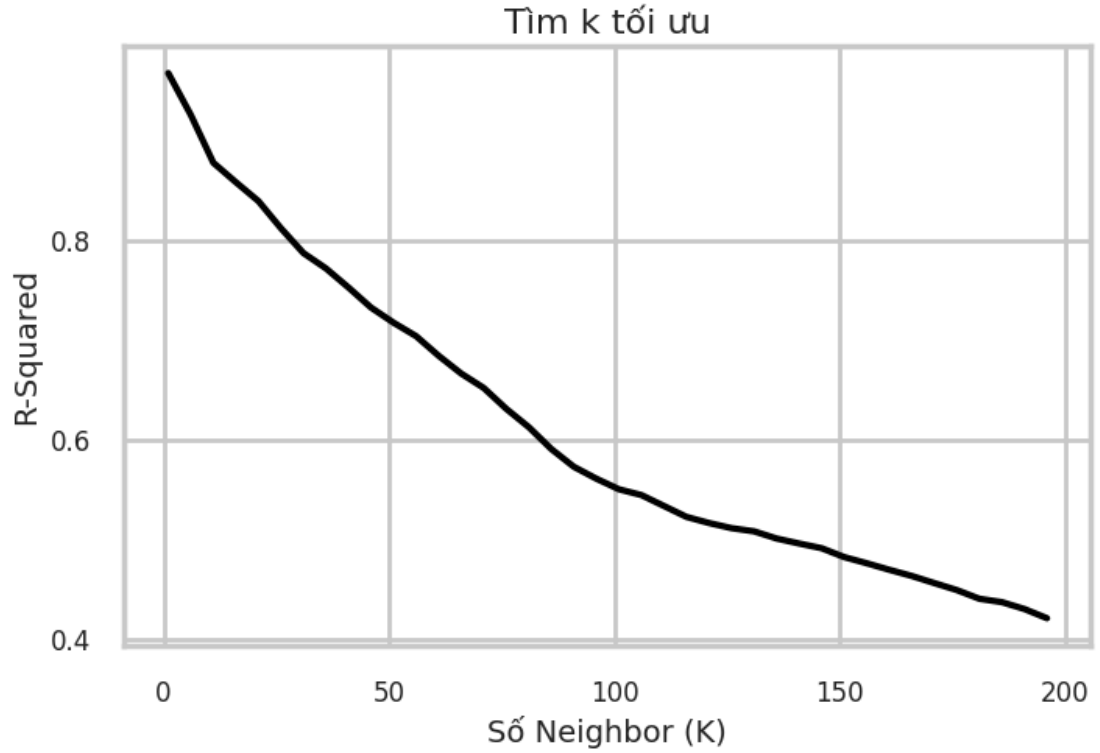
Tìm giá trị k tối ưu thông qua xem xét MSE và R-Squared

Ta triển khai mô hình KNN với giá trị k từ 1 đến 200 và lưu lại giá trị MSE, R-Squared tại các giá trị đó. Kết quả như sau:



Hình 4.38: Tìm K tối ưu bằng MSE với 200 giá trị k

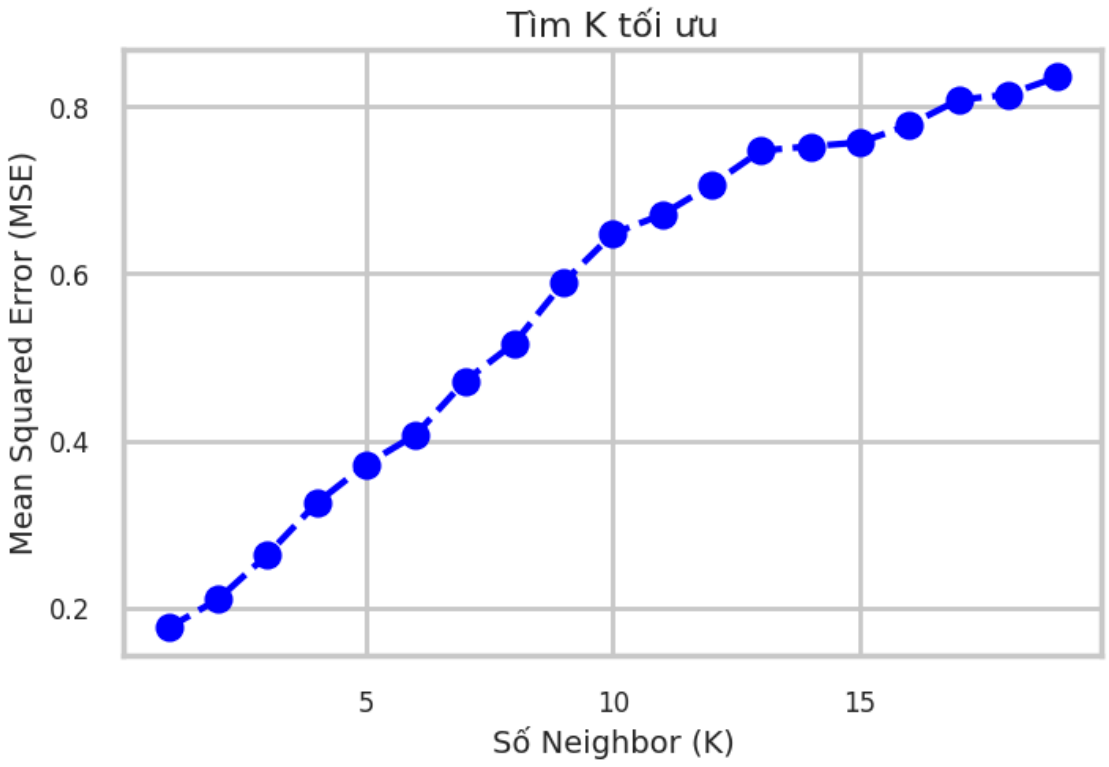
Tại đây nhận thấy MAE có xu hướng tăng liên tục khi k tăng.



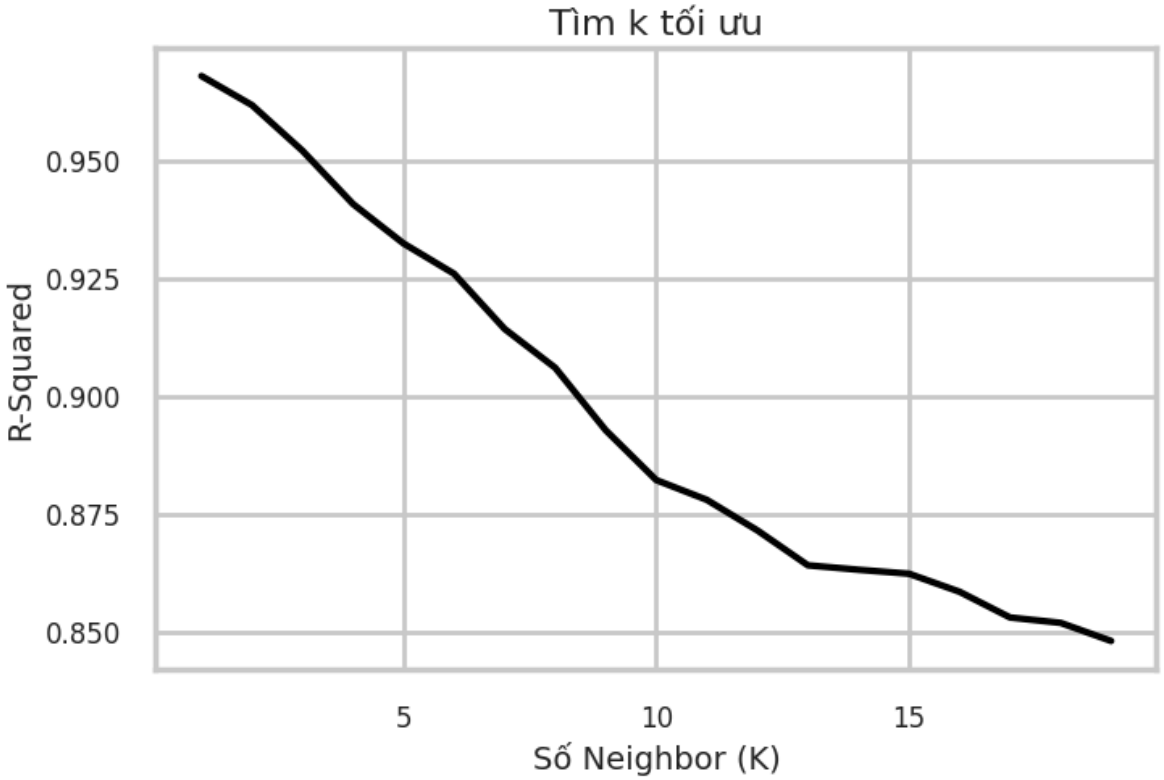
Hình 4.39: Tìm K tối ưu bằng R-Squared với 200 giá trị k.

Còn giá trị R^2 giảm dần khi k tăng.

Để chính xác hơn, ta làm tương tự như trên với giá trị k từ 1 đến 20. Kết quả thu được như sau:



Hình 4.40: Tìm K tối ưu bằng MSE với k từ 1 đến 19.



Hình 4.41: Tìm K tối ưu bằng R-Squared với k từ 1 đến 19.

Biết rằng biến phụ thuộc y có khoảng giá trị từ 26,246 đến 19,865,734 và có giá trị trung bình 1,889,458. Các giá trị MSE ở đây là rất thấp.

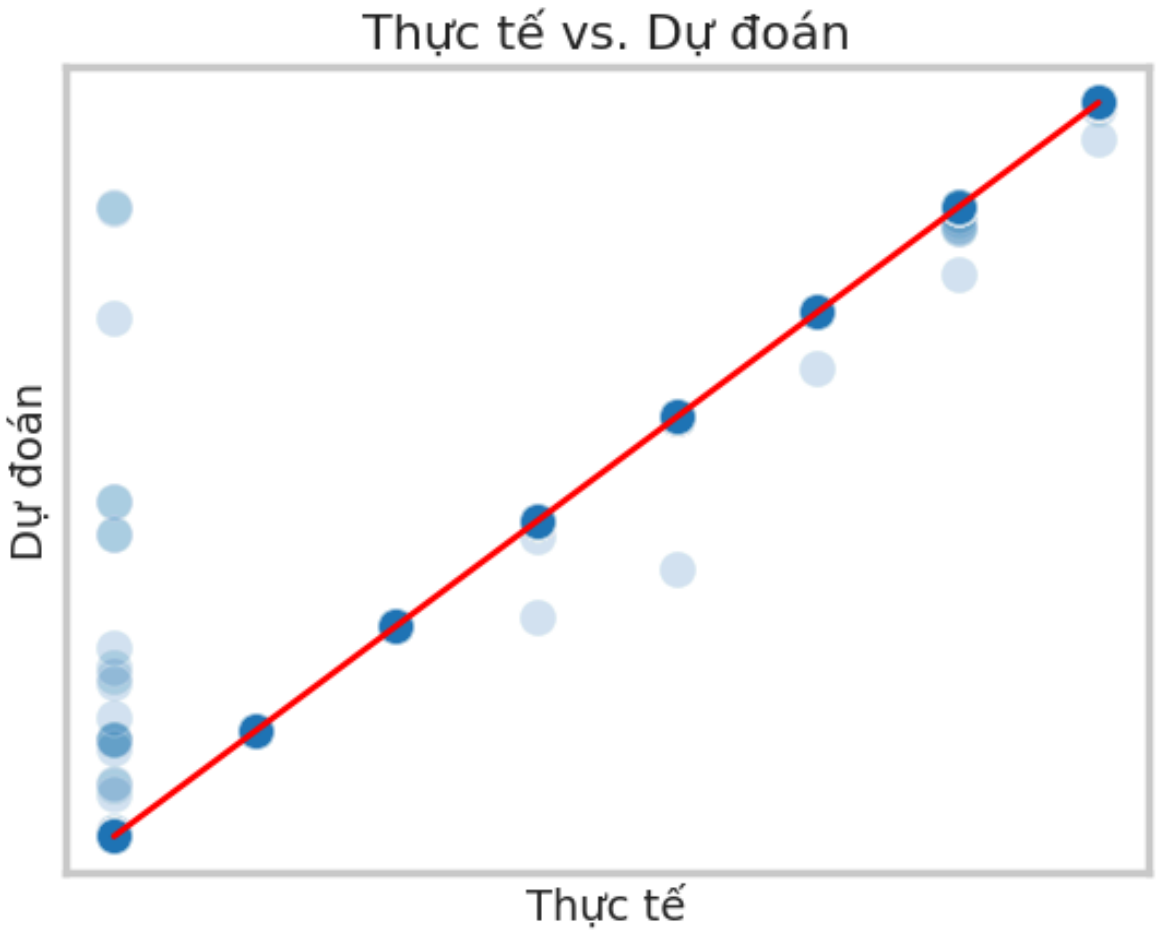
Mặt khác, với tập dữ liệu 1000 video, các giá trị k ở đây là quá nhỏ, dễ dẫn đến việc thiếu tính tổng quát của kết quả.

Dù vậy, theo phương thức Elbow, ta chọn k=15. Giá trị k này cũng cân bằng được tốt nhất R-Squared và MSE.

Đánh giá

Mô hình được đánh giá thông qua các chỉ số MAE, MSE, RMSE, R-Squared và biểu đồ phân phối của phần dư. Phần dư được tính bởi hiệu của giá trị thực tế và giá trị dự đoán.

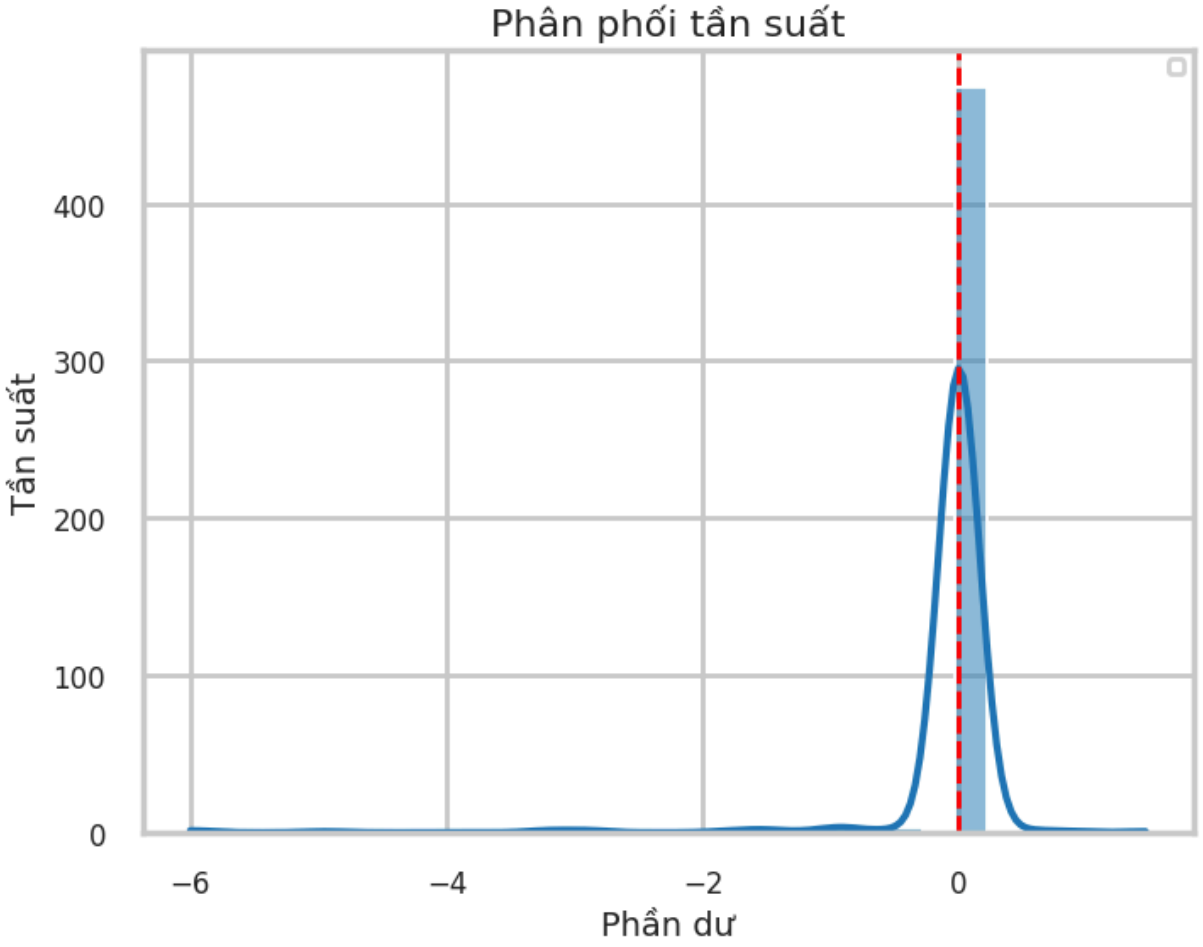
Tại $k = 15$:



Hình 4.42: Thực tế vs. Dự đoán của KNN Hồi quy (k=15).

Các điểm dữ liệu có sự phân tán khá lớn, nhiều điểm nằm xa khỏi đường lý tưởng. Điều này cho thấy mô hình có độ chính xác chưa cao.

Mô hình có xu hướng dự đoán thấp hơn giá trị thực tế khi giá trị thực tế cao (các điểm nằm dưới đường lý tưởng) và dự đoán cao hơn khi giá trị thực tế thấp (các điểm nằm trên đường lý tưởng ở khu vực giá trị thấp). Đây là dấu hiệu của sự thiếu chính xác và độ biến động lớn trong dự đoán.



Hình 4.43: Phân phối tần suất phần dư.

Phần dư chỉ tập trung quanh 0, cho thấy mô hình có xu hướng dự đoán cao hơn thực tế đối với một số giá trị. Tuy nhiên, giá trị phần dư vẫn khá nhỏ (dưới 2) nhưng xảy ra với tần suất hơn 400 lần trong khi tập kiểm tra chỉ có 500 điểm dữ liệu.

Kết hợp các thông số

Mean Absolute Error (MAE): 0.0952

Mean Squared Error (MSE): 0.3079

Root Mean Squared Error (RMSE): 0.5548

R-squared (R^2): 0.944,

có thể kết luận rằng mô hình KNN Regression với $k = 15$ có độ chính xác cao và các yếu tố thời lượng, lượt xem, lượt thích, lượt bình luận đóng vai trò quan trọng trong thuật toán đề xuất của YouTube.

4.2.3. K-MEANS

K-Means là một thuật toán phân cụm dựa trên khoảng cách. Nó hoạt động bằng cách tính toán khoảng cách giữa các điểm dữ liệu và các centroid (tâm cụm) để phân chia dữ liệu thành các cụm khác nhau.

Cách hoạt động của thuật toán KMeans:

- Khởi tạo: Chọn ngẫu nhiên k trung tâm cụm ban đầu.
- Gán cụm: Gán mỗi điểm dữ liệu vào trung tâm cụm gần nhất dựa trên khoảng cách.
- Cập nhật trung tâm cụm: Tính toán lại trung tâm cụm bằng cách lấy trung bình các điểm dữ liệu trong mỗi cụm.
- Lặp lại: Lặp lại quá trình gán cụm và cập nhật trung tâm cho đến khi trung tâm cụm không thay đổi hoặc đạt đến số lần lặp tối đa.

Mô hình KMeans được áp dụng ở đây hướng tới việc tìm ra đặc điểm của các nhóm video riêng biệt nhưng đều thịnh hành.

4.2.3.1. Chuẩn bị dữ liệu

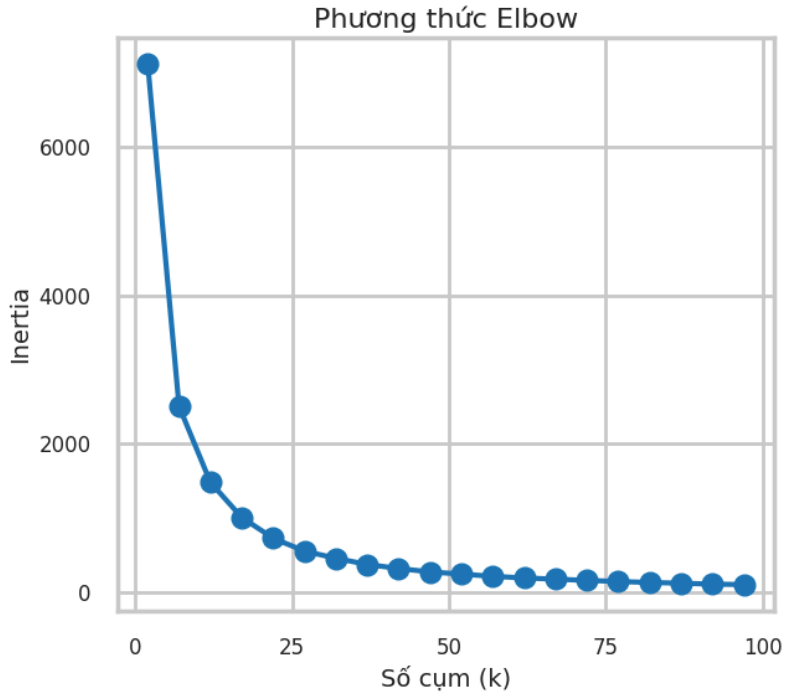
Dữ liệu đã được xử lý loại bỏ outliers, noise, … của các cột có giá trị định lượng (bao gồm: *comment_count, like_count, converted_duration, time_difference, min_sentiment, max_sentiment, mean_sentiment, num_language*) được chuẩn hóa bởi StandardScaler() nhằm tránh thiên vị do chênh lệch giá trị của các trường có đơn vị đo lường khác nhau, làm cân bằng các lớp bởi SMOTE, sau đó lưu trữ tại biến x.

4.2.3.2. Triển khai

Tìm số cụm tối ưu (k)

Áp dụng K-Means với các giá trị k từ 1 đến 20-24, sau đó đánh giá bằng Inertia Index (Phương thức Elbow) và Silhouette Score với mỗi giá trị k.

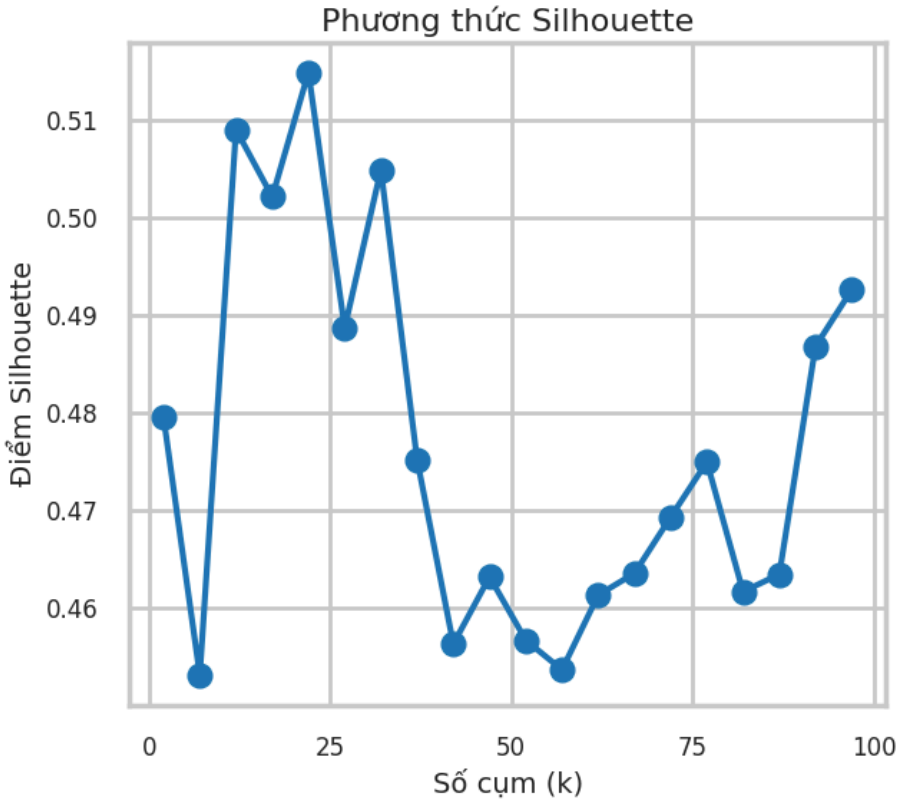
Dựa trên phương thức Elbow, phát hiện độ suy giảm lớn nhất xảy ra khoảng k=20-25, sau đó độ giảm Inertia chậm dần.



Hình 4.44: Biểu đồ subplot của phương thức Elbow.

Đối với Phương thức Silhouette Score, k=20-24, Có một số đỉnh cục bộ, nhưng không có một giá trị k nào cho thấy điểm Silhouette vượt trội rõ rệt so với các giá trị khác..

Do đó quyết định thử k=20,21,22,23,24



Hình 4.45: Biểu đồ subplot của phương thức Silhouette.

Huấn luyện K-Means và đánh giá

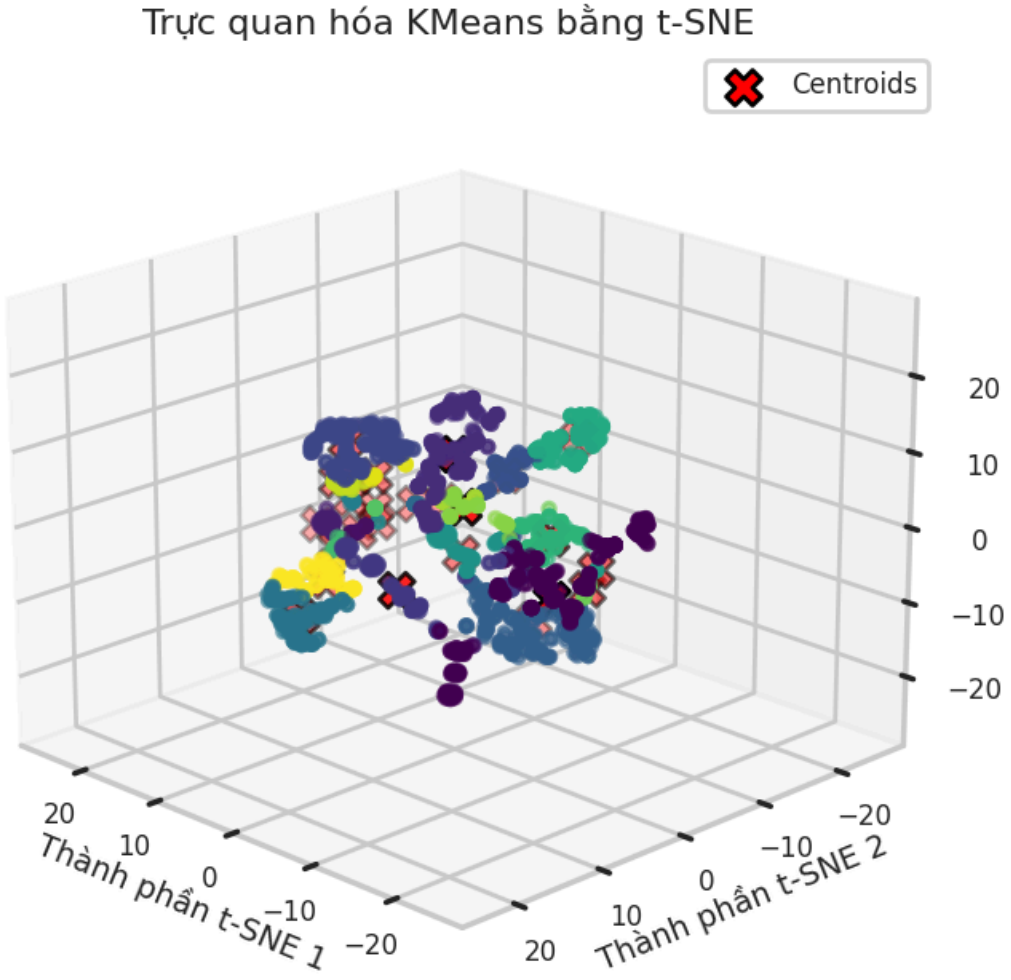
Với mỗi giá trị k trên, đánh giá hiệu quả phân cụm của mô hình thông qua Điểm Silhouette và Chỉ số Davies-Bouldin.

Số cluster (k)	Điểm Silhouette	Chỉ số Davies-Bouldin
20	0.4899	0.7528
21	0.4941	0.7578
22	0.5149	0.6901
23	0.5099	0.7270
24	0.5176	0.6811

Bảng 4.1: Đánh giá hiệu quả phân cụm.

Tương tự, kết quả đánh giá thông qua Điểm Silhouette và Chỉ số Davies-Bouldin cũng cho thấy tại k = 22 và k = 24 (có thể lựa chọn), mô hình hoạt động tốt hơn, với Chỉ số Davies-Bouldin thấp nhất cho thấy các cụm ít chống chéo nhau hơn so với tại các giá trị k khác.

Sử dụng t-SNE để trực quan hóa kết quả áp dụng mô hình KMeans lên bộ dữ liệu với 24 cho kết quả như sau:



Hình 4.46: Biểu đồ 3D scatter thể hiện Kmeans theo t-SNE.

Dữ liệu ban đầu là dữ liệu đa chiều. Do đó, khi giảm xuống 3 chiều và trực quan thông qua t-SNE có thể gây khó hình dung. Tuy nhiên, t-SNE, một kỹ thuật trực quan hóa phi tuyến tính, cho phép nắm bắt các mối quan hệ cục bộ có thể là tối ưu để trực quan dữ liệu ở bài này.

4.2.4. HỒI QUY TUYẾN TÍNH

4.2.4.1. Chuẩn bị dữ liệu

Để có thể sử dụng mô hình tuyến tính một cách tối ưu nhất, chúng tôi quyết định thực hiện quá trình xác định tính tuyến tính của dữ liệu.

Xác định tính tuyến tính của dữ liệu là bước quan trọng trong phân tích để lựa chọn mô hình phù hợp. Đầu tiên, chúng tôi cần phân biệt giữa dữ liệu tuyến tính và phi tuyến. Dữ liệu tuyến tính thể hiện mối quan hệ có thể biểu diễn bằng đường thẳng (Y = aX + b), trong khi dữ liệu phi tuyến có quan hệ phức tạp hơn như dạng parabol, hàm mũ hoặc phân nhánh. Để xác định tính chất này, chúng tôi đã sử dụng các phương pháp EDA

như vẽ scatter plot, kiểm tra hệ số tương quan Pearson đã đề cập tại chương 4, danh mục EDA.

Từ kết quả cho thấy, chúng tôi quyết định sử dụng các biến như view_count, like_count, time_difference, converted_duration cho mô hình hồi quy tuyến tính. Tiếp đó, chúng tôi sẽ bắt đầu quá trình xây dựng mô hình.

Sau khi hoàn thành khâu chuẩn bị dữ liệu, chúng tôi bắt đầu sử dụng mô hình để dự đoán kết quả trên tập kiểm tra. Trong quá trình đánh giá, có thể xảy ra các lỗi không mong muốn. Chúng tôi cần tìm và xử lý các ngoại lệ này để đảm bảo chương trình không bị dừng đột ngột.

4.2.4.2. Triển khai

Đầu tiên, nghiên cứu bắt đầu kiểm tra xem mô hình có đa cộng tuyến hay không. Đa cộng tuyến (Multicollinearity) là hiện tượng các biến độc lập trong mô hình hồi quy có mối quan hệ tuyến tính chặt chẽ với nhau. Điều này có thể làm giảm độ chính xác của các ước lượng hệ số hồi quy, gây khó khăn trong việc đánh giá tác động riêng lẻ của từng biến.

VIF (Variance Inflation Factor) là một chỉ số dùng để đo lường mức độ đa cộng tuyến. Giá trị VIF càng cao, mức độ đa cộng tuyến càng nghiêm trọng. Thông thường:

- VIF < 5: Mức độ đa cộng tuyến thấp, có thể chấp nhận được.
- 5 ≤ VIF < 10: Mức độ đa cộng tuyến trung bình, cần xem xét.
- VIF ≥ 10: Mức độ đa cộng tuyến cao, cần loại bỏ hoặc xử lý.

Để tính VIF cho từng biến trong tập dữ liệu, chúng tôi đã sử dụng thư viện statsmodels và hàm *variance_inflation_factor*. Và cho ra được kết quả như dưới:

- *converted_duration*: VIF = 1.19 (< 5) → Không có vấn đề đa cộng tuyến.
- *time_difference*: VIF = 1.90 (< 5) → Không có vấn đề đa cộng tuyến.
- *view_count*: VIF = 3.26 (< 5) → Không có vấn đề đa cộng tuyến.
- *like_count*: VIF = 4.47 (< 5) → Không có vấn đề đa cộng tuyến.
- *comment_count*: VIF = 2.99 (< 5) → Không có vấn đề đa cộng tuyến.

Ta có thể đưa ra kết luận như sau: Tất cả các biến đều có giá trị VIF nhỏ hơn 5, điều này cho thấy không có hiện tượng đa cộng tuyến nghiêm trọng trong mô hình. Các biến có VIF cao nhất là *like_count* (4.47) và *view_count* (3.26), nhưng vẫn nằm trong ngưỡng chấp nhận được (VIF < 5). Trong trường hợp này, không cần xử lý đa cộng tuyến vì tất cả các biến đều có VIF < 5.

Trong phần này, chúng tôi sẽ bắt đầu trực quan hóa mối quan hệ giữa biến độc lập (*like_count*) và biến phụ thuộc (*view_count*) bằng cách sử dụng biểu đồ scatter plot và đường hồi quy.

Chúng tôi sử dụng dữ liệu DataFrame new_df, với:

- Y: Biến phụ thuộc (view_count).
- X: Biến độc lập (like_count).

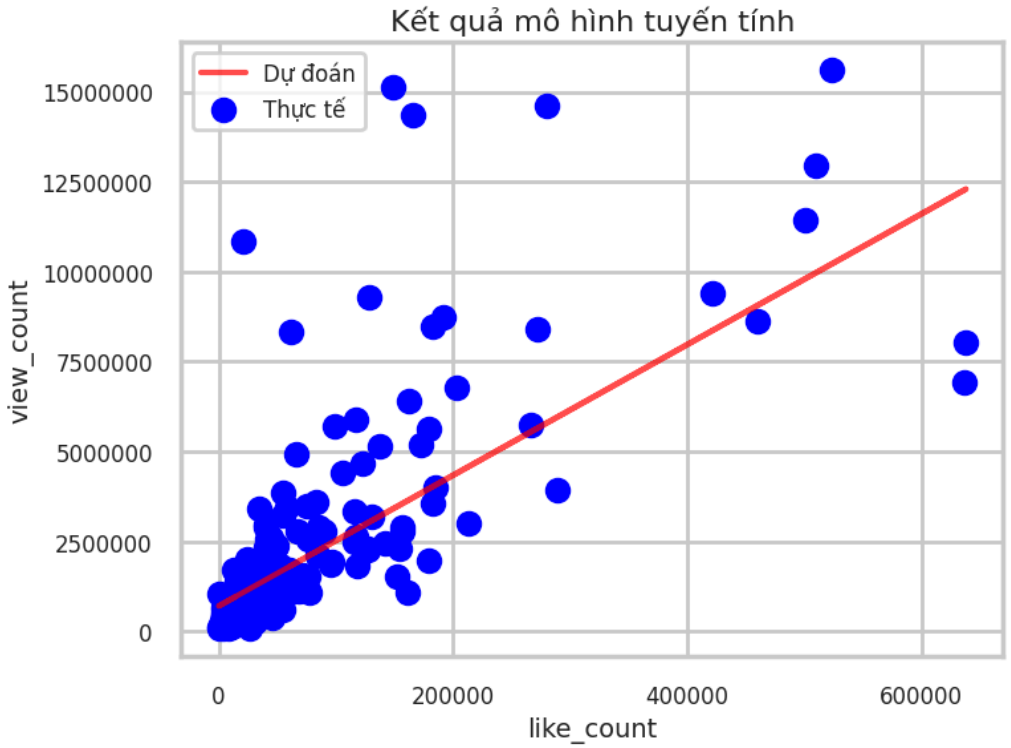
Dữ liệu được chia thành tập huấn luyện (80%) và tập kiểm tra (20%).

Chúng ta sử dụng scatter plot để hiển thị dữ liệu kiểm tra, với:

- Trục x: *like_count*.
- Trục y: *view_count*.

Chúng tôi sử dụng thư viện scikit-learn để huấn luyện mô hình hồi quy tuyến tính trên tập huấn luyện. Sau khi huấn luyện mô hình, chúng tôi vẽ đường hồi quy lên cùng biểu đồ với dữ liệu kiểm tra để đánh giá mức độ phù hợp của mô hình:

- Scatter plot (màu xanh): Hiển thị mối quan hệ thực tế giữa *like_count* và *view_count* trong tập kiểm tra.
- Đường hồi quy (màu đỏ): Đường thẳng thể hiện dự đoán của mô hình hồi quy tuyến tính.



Hình 4.47: Biểu đồ thể hiện mô hình hồi quy tuyến tính của tập dữ liệu kiểm tra.

Mối quan hệ giữa *like_count* và *view_count*:

- Mô hình hồi quy tuyến tính cho thấy có một mối quan hệ tuyến tính dương giữa *like_count* và *view_count*. Khi số lượt like tăng, số lượt xem cũng có xu hướng tăng theo.
- Đường hồi quy (màu đỏ) trong biểu đồ scatter plot cho thấy mô hình dự đoán có sự phân tán cao.

Sau khi tính toán các chỉ số đánh giá, chúng tôi bắt đầu phân tích kết quả để hiểu rõ hơn về hiệu suất của mô hình hồi quy tuyến tính. Phần này sẽ tập trung vào việc đánh giá các chỉ số như MAE (Mean Absolute Error), MSE (Mean Squared Error), R^2 (R-Squared), và Adjusted R^2 để đưa ra nhận định về chất lượng của mô hình.

```
Mean Absolute Error (MAE): 1097474.4534443668
Mean Squared Error (MSE): 4016428537723.395
R-Squared (R²): 0.5472213628462844
Adjusted R-Squared: 0.5449229941297682
```

Hình 4.48: Kết quả của các thông số đánh giá mô hình.

Kết quả đánh giá mô hình cho thấy:

- Mean Absolute Error (MAE): 1097474.45
  - + MAE đo lường sai số tuyệt đối trung bình giữa giá trị thực tế và giá trị dự đoán. Giá trị MAE càng thấp, mô hình càng chính xác.
  - + MAE = 1097474.45 cho thấy trung bình, mô hình sai lệch khoảng 1097474,45 views so với giá trị thực tế. Đây là một sai số khá lớn, đặc biệt khi giá trị *view_count* nằm trong khoảng hàng triệu.
- Mean Squared Error (MSE): 4016428537723.4
  - + MSE đo lường bình phương sai số trung bình. MSE nhạy cảm với các sai số lớn hơn so với MAE.
  - + MSE = 4016428537723.4 cho thấy có một số điểm dữ liệu có sai số rất lớn, điều này có thể do các giá trị *view_count* cực cao hoặc cực thấp.
- R-Squared (R^2): 0.5472.
  - + R^2 đo lường mức độ phù hợp của mô hình với dữ liệu. Giá trị R^2 càng gần 1, mô hình càng giải thích được nhiều biến động của dữ liệu.
  - + R^2 = 0.5472 cho thấy mô hình giải thích được 54.72% biến động của *view_count*. Điều này có nghĩa là còn 45.28% biến động của *view_count* không được giải thích bởi biến *like_count*.

- Adjusted R-Squared: 0.5449.
    + Adjusted R^2 điều chỉnh R^2 để phản ánh số lượng biến đầu vào trong mô hình.
    + Adjusted R^2 = 0.5449 gần với R^2, cho thấy mô hình không bị ảnh hưởng nhiều bởi số lượng biến.

Ta có thể đánh giá mô hình như sau:

- Mối quan hệ giữa *like_count* và *view_count*:
    + Mô hình hồi quy tuyến tính cho thấy có một mối quan hệ tuyến tính dương giữa *like_count* và *view_count*. Khi số lượt like tăng, số lượt xem cũng có xu hướng tăng theo.
    + Tuy nhiên, với R^2 = 0.5472, mô hình chỉ giải thích được 54.72% biến động của *view_count*, điều này cho thấy *like_count* không phải là yếu tố duy nhất ảnh hưởng đến *view_count*.
- Độ chính xác của mô hình:
    + Với MAE = 1097474.45 và MSE = 4016428537723.4, mô hình có sai số khá lớn, đặc biệt là ở các giá trị *view_count* cao. Điều này có thể do các yếu tố khác như *comment_count*, *duration*, hoặc *time_difference* cũng ảnh hưởng đến *view_count*.

Từ đó, có thể thấy các hạn chế của mô hình như:

- Thiếu các biến độc lập khác:
    + Mô hình hiện tại chỉ sử dụng *like_count* để dự đoán *view_count*. Chúng tôi có thể cần xem xét thêm các biến độc lập khác như *comment_count*, *duration*, hoặc *time_difference* để cải thiện độ chính xác của mô hình.
- Hiện tượng phương sai không đồng đều (Heteroscedasticity):
    + Khi các điểm dữ liệu trên biểu đồ scatter plot có xu hướng phân tán nhiều hơn khi *like_count* tăng, điều này có thể chỉ ra hiện tượng phương sai không đồng đều. Điều này có thể làm giảm độ tin cậy của mô hình.

Mô hình hồi quy tuyến tính hiện tại cho thấy một mối quan hệ tuyến tính dương giữa *like_count* và *view_count*, nhưng chỉ giải thích được 54.72% biến động của *view_count*. Sai số lớn cho thấy mô hình cần được cải thiện bằng cách thêm các biến độc lập khác và xem xét các phương pháp xử lý phương sai không đồng đều.

Và khi so sánh với các mô hình kể trên, có thể thấy phương thức lựa chọn các biến số thực hiện rất quan trọng trong quá trình sử dụng mô hình máy học.

CHƯƠNG 5: KẾT QUẢ VÀ THẢO LUẬN

Qua các phân tích trên, có thể đúc kết được các đặc điểm chung của một video YouTube nổi bật tại mã vùng US như sau:

Dựa trên kết quả mô hình Random Forest, các yếu tố như lượt tương tác (thích, bình luận, …) và thời lượng mang tính quyết định cao về lượt xem. Cụ thể hơn:

- Entertainment, Music, và Gaming có lượt xem cao nhất, phản ánh sự hấp dẫn mạnh mẽ của các thể loại này đối với khán giả.
- Tồn tại sự khác biệt trong cách khán giả tiếp cận nội dung, trong đó tính chất nội dung quan trọng hơn số lượt xem trong việc thúc đẩy tương tác.
- Những video viral hoặc có cộng đồng fan mạnh mẽ thường có lượng thích cao ngay cả khi không có lượt xem quá nổi bật, cho thấy sự kết nối cảm xúc đóng vai trò quan trọng.
- Video ngắn (dưới 10 phút) có xu hướng thu hút lượt xem cao hơn, đặc biệt trong danh mục Âm nhạc và Giải trí.
- Video dài trên 200 phút hiếm khi đạt lượt xem lớn, có thể vì chúng chủ yếu là livestream hoặc nội dung tổng hợp.
- Video mới, đặc biệt trong lĩnh vực Âm nhạc và Giải trí, có thể đạt viral rất nhanh. Tuy nhiên, thời gian tồn tại lâu không đồng nghĩa với việc thu hút nhiều lượt xem hơn – chính nội dung mới là yếu tố quyết định.

Như vậy, số lượt xem không phải là chỉ số duy nhất phản ánh sự thành công của video. Mức độ tương tác, tính chất nội dung, sự kích thích thảo luận, và khả năng kết nối cảm xúc với khán giả mới là những yếu tố quan trọng hơn trong việc đánh giá mức độ phổ biến của một video.

Từ các kết quả kể trên, chúng ta có thể đưa ra các chiến lược thu hút người xem cho người sáng tạo nội dung như:

- Tăng cường tương tác bình luận: Khuyến khích người xem để lại bình luận bằng cách đặt câu hỏi hoặc tạo các cuộc thảo luận trong video.
- Tối ưu thời lượng video: Đảm bảo thời lượng video phù hợp với nội dung và đối tượng mục tiêu để tối đa hóa lượt xem.
- Giảm tập trung vào lượt thích: Mặc dù lượt thích là một chỉ số tương tác, nhưng nó không ảnh hưởng nhiều đến lượt xem. Do đó, nên tập trung vào các yếu tố khác như bình luận và thời lượng.

Điểm mạnh

- Điểm mạnh đầu tiên của nghiên cứu là về các phương pháp luận khoa học:
    + Sử dụng phân tích hồi quy tuyến tính: Nghiên cứu sử dụng phương pháp phân tích hồi quy tuyến tính, một phương pháp phổ biến và hiệu quả trong việc xác định mối quan hệ giữa các biến. Điều này giúp đảm bảo tính chính xác và độ tin cậy của kết quả nghiên cứu.
    + Kiểm định giả định mô hình: Nghiên cứu đã kiểm tra các giả định của mô hình hồi quy, bao gồm kiểm tra hiện tượng đa cộng tuyến và phân phối của phần dư, giúp tăng cường độ tin cậy của kết quả.
- Tiếp đó là sự lựa chọn dữ liệu phù hợp với mục đích nghiên cứu:
    + Dữ liệu đa dạng: Nghiên cứu sử dụng dữ liệu từ nhiều video khác nhau, giúp kết quả có tính đại diện cao và có thể áp dụng rộng rãi.
    + Biến số phù hợp: Các biến số được lựa chọn (như *comment_count, converted_duration, like_count*) là những yếu tố quan trọng và có ý nghĩa trong việc phân tích lượt xem và tương tác.
- Cuối cùng là các kết quả khả quan mà nghiên cứu đã đạt được:
    + Kết quả rõ ràng: Nghiên cứu đưa ra các kết luận rõ ràng về các yếu tố ảnh hưởng đến lượt xem, giúp các nhà sản xuất nội dung có thể tập trung vào các yếu tố quan trọng nhất.
    + Ứng dụng thực tiễn: Kết quả nghiên cứu có thể được áp dụng trực tiếp vào việc tối ưu hóa chiến lược nội dung, giúp tăng lượt xem và tương tác trên các nền tảng video.

Điểm yếu

- Thiếu thông tin về nhân khẩu học và hành vi người xem:
    + Dữ liệu hiện tại chủ yếu tập trung vào thông tin video và tương tác cơ bản. Thiếu các thông tin về: nhân khẩu học (tuổi, giới tính, vị trí địa lý)., hành vi người xem (thời gian xem trung bình, tỷ lệ giữ chân người xem, nguồn lưu lượng truy cập).
    + Thông tin này rất quan trọng để hiểu rõ hơn về đối tượng mục tiêu và cách họ tương tác với video.
- Hạn chế về phân tích bình luận:

+ Chỉ có trường *comments* chứa văn bản thô. Điều này gây khó khăn cho việc phân tích cảm xúc, chủ đề hoặc xu hướng trong bình luận.
+ Cần có các kỹ thuật xử lý ngôn ngữ tự nhiên (NLP) để trích xuất thông tin có giá trị từ bình luận.

- Thiếu thông tin về thuật toán đề xuất của video:
  + Thuật toán đề xuất của YouTube đóng vai trò quan trọng trong việc quyết định video nào được hiển thị cho người xem.
  + Thiếu thông tin về các yếu tố ảnh hưởng đến thuật toán này (ví dụ: tỷ lệ nhấp, tỷ lệ giữ chân người xem) sẽ hạn chế khả năng hiểu rõ về lý do tại sao một số video thành công hơn những video khác.
- Các điểm yếu khác:
  + *description*: mô tả có thể có các thông tin quan trọng, nhưng dạng văn bản tự do này sẽ rất khó cho các phân tích định lượng nếu không có quá trình xử lý ngôn ngữ tự nhiên.
  + *tags*: các thẻ tag có thể bị sai lệch hoặc không đầy đủ, vì vậy không phải lúc nào cũng phản ánh chính xác nội dung video.
  + *category_id*: mã danh mục có thể không đủ chi tiết để phân tích sâu hơn về các chủ đề.

Về hiệu suất mô hình, sau quá trình phân tích cho thấy Random Forest nổi bật với khả năng phân loại video thịnh hành vượt trội, đạt độ chính xác 0.98 và thể hiện sự ổn định cao qua các chỉ số đánh giá cùng đường cong học tập tốt. Trong khi đó, KNN cho thấy tiềm năng trong việc dự đoán số lượt xem với R-squared ấn tượng là 0.944, tuy nhiên, độ chính xác trong dự đoán giá trị cụ thể vẫn cần được xem xét kỹ lưỡng hơn. K-Means đã thành công trong việc phân cụm dữ liệu, làm sáng tỏ cấu trúc nhóm của các video thịnh hành. Ngược lại, mô hình hồi quy tuyến tính, khi chỉ dựa trên số lượt thích, cho thấy những hạn chế đáng kể trong việc dự đoán số lượt xem. Tóm lại, đối với mục tiêu phân loại video thịnh hành, Random Forest là lựa chọn mô hình ổn định và hiệu quả nhất.

Từ các phân tích đã triển khai, có thể kết luận như sau về các video thịnh hành ở khu vực châu Mỹ (cụ thể là Hoa Kỳ, Mexico, Canada, Brazil, Argentina):

- Thời lượng video đóng vai trò quan trọng: Cả video ngắn (dễ tiếp cận) và video dài (có tiềm năng tương tác cao) đều có cơ hội trở nên thịnh hành.

- Tương tác cao là yếu tố then chốt: Số lượng bình luận có tác động đáng kể đến khả năng video được đề xuất, cho thấy sự ưu tiên của nền tảng đối với các nội dung khơi gợi thảo luận.
- Lượt thích có liên quan đến lượt xem: Mối tương quan dương cho thấy video được yêu thích thường có xu hướng được xem nhiều hơn, góp phần vào sự thịnh hành.
- Sự đa dạng về nội dung: Các video thịnh hành không đồng nhất mà có thể thuộc nhiều nhóm đặc trưng khác nhau, cho thấy sự phong phú trong sở thích của người xem.
- Nhiều yếu tố cùng tác động: Sự phổ biến của video không chỉ dựa vào một yếu tố duy nhất mà là sự kết hợp của nhiều đặc điểm nội dung và tương tác.

Do đó, đối với thị trường châu Mỹ, nhóm đề xuất các chiến lược cho người sáng tạo nội dung như: tạo ra nội dung hấp dẫn, khuyến khích tương tác, thử nghiệm nhiều định dạng và chủ đề, đồng thời liên tục phân tích dữ liệu để tối ưu hóa chiến lược.

Ngoài ra, bài báo đã giải quyết được một số vấn đề tồn tại ở các nghiên cứu trước đó, bao gồm:

- Sử dụng phương pháp luận khoa học có độ tin tưởng cao: Các mô hình được lựa chọn (Random Forest, KNN, K-Means, Hồi quy tuyến tính) là những thuật toán học máy phổ biến và đã được chứng minh hiệu quả trong nhiều bài toán tương tự. Việc đánh giá mô hình bằng các chỉ số phù hợp (độ chính xác, R-squared, Silhouette Score, Davies-Bouldin) và sử dụng cross-validation đảm bảo tính khách quan và độ tin cậy của kết quả.
- Sử dụng dữ liệu phù hợp với mục đích nghiên cứu: Dữ liệu đã được xử lý kỹ lưỡng để loại bỏ nhiễu và ngoại lệ, đồng thời được cân bằng mẫu bằng SMOTE để tránh hiện tượng thiên vị mô hình. Việc lựa chọn các đặc trưng (thời lượng, lượt xem, lượt thích, bình luận, v.v.) có liên quan đến sự thịnh hành của video trên YouTube đảm bảo dữ liệu đầu vào phù hợp với mục tiêu phân tích và dự đoán.
- Đạt được các kết quả khả quan, tổng quát có khả năng áp dụng thực tế:
    + Mô hình Random Forest đạt độ chính xác cao trong việc phân loại video thịnh hành, cho thấy khả năng xác định các yếu tố quan trọng.
    + Mô hình KNN cho thấy tiềm năng trong việc dự đoán số lượt xem.

+ Mô hình K-Means giúp phân cụm và hiểu rõ hơn về các nhóm video thịnh hành khác nhau.
+ Phân tích hồi quy tuyến tính làm sáng tỏ mối quan hệ giữa lượt thích và lượt xem. Những kết quả này cung cấp thông tin hữu ích cho người sáng tạo nội dung (về các yếu tố cần tập trung) và các nhà nghiên cứu (về hiệu quả của các mô hình trong lĩnh vực này). Các đề xuất chiến lược cho người sáng tạo nội dung dựa trên kết quả phân tích cũng cho thấy khả năng ứng dụng thực tế của nghiên cứu.

Tuy nhiên, bài báo cũng gặp phải các hạn chế như sau:

- Hạn chế về lượng dữ liệu: Do các giới hạn của API Google Cloud, chỉ có thể thu thập các thông tin cơ bản của tối đa 200 video từ mỗi mã vùng trong quá trình nghiên cứu. Hạn chế này ảnh hưởng đến khả năng thu thập dữ liệu toàn diện hơn.
- Thiếu thông tin về nhân khẩu học và hành vi người xem: Dữ liệu hiện tại tập trung chủ yếu về các thông tin của video và các tương tác cơ bản, còn thông tin về người dùng và hành vi người dùng, giúp hiểu rõ hơn về đối tượng mục tiêu và cách họ tương tác với video, chưa được khai thác.
- Hạn chế về phân tích bình luận: dữ liệu bình luận chỉ bao gồm các văn bản thô gây khó khăn cho việc phân tích chủ đề, xu hướng trong bình luận.
- Thiếu thông tin về thuật toán đề xuất của YouTube: YouTube không công bố đầy đủ cơ chế hoạt động của hệ thống đề xuất – vốn được cá nhân hóa theo hành vi người dùng, thời gian, vị trí và các yếu tố khác – nên trong khuôn khổ bài nghiên cứu này, không thể kiểm soát hoặc mô phỏng lại quá trình mà người dùng tiếp cận các video. Điều này gây khó khăn trong việc xác định liệu dữ liệu bình luận thu thập được có thực sự mang tính ngẫu nhiên, đại diện hay không, hay chỉ phản ánh một nhóm người dùng cụ thể được hệ thống ưu tiên tiếp cận. Hạn chế này ảnh hưởng đến độ khách quan và tổng quát của kết quả phân tích cảm xúc.
- Hạn chế trong đánh giá ứng dụng thực tiễn: Nhóm chưa thực hiện được bước chuyển giao và thử nghiệm mô hình trong môi trường thực tế. Điều này khiến việc đánh giá mức độ khả thi và hiệu quả áp dụng trong các bối cảnh vận hành thực tế còn chưa được kiểm chứng đầy đủ.

CHƯƠNG 6: KẾT LUẬN - ĐỀ NGHỊ

6.1. Kết luận

Công trình nghiên cứu đã hoàn thành các mục tiêu đề ra, mang đến những hiểu biết sâu sắc về các nhân tố tác động đến sự phổ biến và tương tác của video trên nền tảng YouTube. Bằng việc thu thập, xử lý sơ bộ và phân tích dữ liệu từ 1000 video thịnh hành tại năm quốc gia (Hoa Kỳ, Canada, Mexico, Brazil, Argentina) vào ngày 16 tháng 04 năm 2025, nghiên cứu đã làm nổi bật một số khía cạnh quan trọng. Phân tích thống kê mô tả và trực quan hóa dữ liệu đã xác định được những đặc điểm chung của các video thịnh hành, bao gồm xu hướng về thời lượng, mức độ tương tác (lượt thích, bình luận) và các yếu tố liên quan đến khoảng thời gian từ khi phát hành đến khi trở nên thịnh hành.

Thêm vào đó, việc phân tích ngôn ngữ và cảm xúc trong các bình luận đã cung cấp những thông tin giá trị về phản ứng của người xem đối với nội dung video. Số lượng ngôn ngữ khác nhau trong phần bình luận và các chỉ số về mức độ cảm xúc (tích cực, tiêu cực, trung lập) có thể có mối liên hệ với khả năng lan tỏa và mức độ tương tác của video. Nghiên cứu cũng đã ứng dụng và so sánh hiệu quả của bốn mô hình học máy khác nhau như K-means Clustering, Random Forest, KNN, Hồi quy tuyến tính trong việc phân tích và dự đoán các khía cạnh khác nhau của dữ liệu video YouTube.

Mô hình Random Forest đã chứng minh khả năng xác định các đặc trưng quan trọng ảnh hưởng đến một biến mục tiêu cụ thể. Mô hình K-means Clustering giúp phân loại các video dựa trên sự tương đồng về đặc điểm, mở ra khả năng khám phá các phân khúc nội dung được ưa chuộng. Mô hình KNN được sử dụng để dự đoán các giá trị hoặc phân loại dựa trên các video gần nhất, cho thấy tiềm năng trong việc đề xuất các video tương tự. Cuối cùng, mô hình Hồi quy tuyến tính đã được áp dụng để phân tích mối quan hệ giữa các biến số và dự đoán các giá trị số như số lượt xem.

Nghiên cứu cũng đã đánh giá và so sánh tính phù hợp của từng mô hình học máy đối với các bài toán phân tích dữ liệu video YouTube khác nhau, từ đó đưa ra những nhận xét về ưu điểm và hạn chế của từng phương pháp.

Nhìn chung, công trình nghiên cứu này đã cung cấp những phân tích bước đầu về dữ liệu video YouTube thịnh hành, tạo nền tảng cho các nghiên cứu chuyên sâu hơn trong tương lai. Việc kết hợp các phương pháp phân tích thống kê, xử lý ngôn ngữ tự nhiên và học máy đã mang lại những hiểu biết đa chiều về dữ liệu video và hành vi của người dùng trên nền tảng này.

Từ đó rút ra các ứng dụng thực tiễn về các đặc trưng chung của những video phổ biến bậc nhất tại năm quốc gia thuộc Châu Mỹ, mức độ quan trọng của các đặc trưng đó và một số đề xuất cho chiến lược nội dung đối với nhà sản xuất nội dung cũng như giúp các doanh nghiệp khai thác hiệu quả hơn các tài nguyên trong các chiến dịch quảng cáo.

Mặc dù phần lớn các mục tiêu nghiên cứu đã được đáp ứng, một số hạn chế lớn vẫn còn tồn tại. Do đó, mở ra nhiều khả năng và phương hướng cải tiến nhằm nâng cao chất lượng nghiên cứu trong tương lai, như những điều chỉnh về phạm vi dữ liệu, mô hình phân tích và phương pháp xử lý ngôn ngữ tự nhiên trong các nghiên cứu tiếp theo.

6.2. Đề nghị

Chúng tôi thành thật thừa nhận rằng nghiên cứu này vẫn còn nhiều hạn chế đáng kể. Mô hình phân tích của chúng tôi chưa thể nắm bắt đầy đủ sự phức tạp của thuật toán đề xuất của YouTube và các yếu tố ảnh hưởng đến hiệu suất video. Dữ liệu thu thập còn hạn chế về quy mô và đa dạng, chưa đại diện đầy đủ cho toàn bộ hệ sinh thái YouTube. Phương pháp xử lý dữ liệu ngoại lai và biến đổi dữ liệu của chúng tôi vẫn còn đơn giản, chưa tận dụng được các kỹ thuật tiên tiến nhất trong lĩnh vực khoa học dữ liệu. Mối quan hệ phi tuyến giữa các biến chưa được khám phá và mô hình hóa một cách triệt để, dẫn đến khả năng dự đoán còn hạn chế.

Việc phân tích dữ liệu YouTube hiện nay vẫn tồn tại nhiều hạn chế như đã đề cập ở phần trước. Để nâng cao chất lượng phân tích, chúng tôi có các đề nghị có thể sử dụng trong tương lai như sau:

Để mở rộng và làm giàu dữ liệu nhằm tăng chiều sâu của phân tích, chúng tôi quyết định kết hợp nhiều nguồn dữ liệu hoặc áp dụng các phương pháp thu thập khác như crawling theo từ khóa, theo thời gian hoặc theo kênh cụ thể, có thể giúp mở rộng phạm vi dữ liệu. Việc thu thập thêm thông tin về nhân khẩu học và hành vi người xem thông qua tích hợp dữ liệu từ các nền tảng bổ sung hoặc tiến hành khảo sát người dùng sẽ giúp hiểu rõ hơn về đối tượng khán giả và cách họ tương tác với nội dung video. Bên cạnh đó, việc áp dụng các kĩ thuật NLP tiên tiến hơn sẽ giúp khai thác triệt để nội dung bình luận và mô tả video, từ đó phân tích tình cảm và chủ đề chính xác hơn.

Nhằm cải thiện mô hình để tăng độ chính xác và khả năng tổng quát, chúng tôi đề xuất áp dụng các phương pháp biến đổi dữ liệu phù hợp như biến đổi logarit để chuẩn hóa phân phối của các biến như *view_count*, *like_count*. Việc phát triển các mô hình hồi quy phi tuyến hoặc mô hình học máy nâng cao như Random Forest, Gradient Boosting sẽ

giúp nắm bắt tốt hơn mối quan hệ phức tạp giữa các biến. Đồng thời, xây dựng mô hình riêng biệt cho từng thể loại nội dung như Giải trí, Trò chơi điện tử, Âm nhạc sẽ tăng độ chính xác của dự đoán, do mỗi thể loại có đặc điểm và xu hướng khác nhau. Việc tích hợp phân tích chuỗi thời gian cũng rất quan trọng để nắm bắt xu hướng theo thời gian và tính mùa vụ trong dữ liệu YouTube.

Để giảm sự không chắc chắn do thiếu thông tin về thuật toán đề xuất của YouTube, chúng tôi cần tiếp cận được thêm các chỉ số như tỷ lệ nhấp (CTR), tỷ lệ giữ chân người xem (retention rate) hoặc nguồn lưu lượng truy cập (referrer), hoặc trực tiếp xây dựng các mô hình mô phỏng hệ thống đề xuất của YouTube. Đây là những số liệu thực tế mang tính xác thực mà chúng tôi đề xuất sử dụng trong tương lai để nâng cao hiệu suất của công trình. Việc phân tích tương quan giữa các biến sau khi biến đổi sẽ giúp hiểu rõ hơn về cách YouTube xếp hạng và đề xuất video.

Cải thiện phương pháp xử lý dữ liệu ngoại lai và dữ liệu thiếu là một khía cạnh quan trọng khác. Chúng tôi đề xuất phát triển các phương pháp phát hiện và xử lý dữ liệu ngoại lai (outliers) tinh vi hơn, như phương pháp dựa trên mật độ hoặc khoảng cách Mahalanobis. Việc áp dụng các kỹ thuật nội suy và ngoại suy tiên tiến sẽ giúp xử lý dữ liệu thiếu hoặc không đầy đủ một cách hiệu quả. Đồng thời, sử dụng phương pháp tái lấy mẫu (resampling) sẽ giúp cân bằng dữ liệu giữa các thể loại nội dung khác nhau, từ đó cải thiện độ chính xác của mô hình.

Cuối cùng, việc bổ sung bước chuyển giao công nghệ và tiến hành thử nghiệm thực tế là không thể thiếu. Chúng tôi đề xuất xây dựng bảng điều khiển (dashboard) trực quan và thân thiện với người dùng để nhà sáng tạo nội dung có thể dễ dàng áp dụng các phát hiện từ phân tích dữ liệu. Việc tiến hành thử nghiệm thực tế với các nhà sáng tạo nội dung sẽ giúp xác định tính ứng dụng, độ tin cậy cũng như những điều chỉnh cần thiết khi triển khai trong môi trường ngoài phòng thí nghiệm. Đồng thời, phát triển các hướng dẫn và khuyến nghị cụ thể dựa trên phân tích dữ liệu sẽ giúp nhà sáng tạo nội dung tối ưu hóa chiến lược YouTube của họ một cách hiệu quả.

Những đề xuất trên nhằm khắc phục các hạn chế hiện tại và nâng cao chất lượng phân tích dữ liệu YouTube trong tương lai, từ đó cung cấp những hiểu biết sâu sắc và có giá trị hơn cho nhà sáng tạo nội dung và các bên liên quan. Chúng tôi tin rằng với những cải tiến này, việc phân tích dữ liệu YouTube sẽ trở nên toàn diện và chính xác hơn, mang lại giá trị thực tiễn cao hơn cho người dùng.

CHƯƠNG 7: TÀI LIỆU THAM KHẢO

[1] Michael Keenan (2024), "The YouTube Algorithm: How it Works in 2025", truy cập tại: https://www.shopify.com/blog/youtube-algorithm?utm_source=chatgpt.com

[1] Martin, Ester, Hans-Peter Kriegel, Jiirg Sander, Xiaowei Xu (1996), "A density-based algorithm for discovering clusters in large spatial databases with noise", truy cập tại: https://studylib.net/doc/13739397/a-density-based-algorithm-for--discovering-clusters

[2] Mixed Analytics (2023), "List of YouTube Category IDs", truy cập tại: https://mixedanalytics.com/blog/list-of-youtube-video-category-ids

[3] Kristi Kates (07/03/2025), "Secrets of the YouTube Algorithm", truy cập tại: https://fourthwall.com/blog/secrets-of-the-youtube-algorithm#how-does-the-youtube-algorithm-rank-videos?

[4] Wikipedia (2024), "Youtube", truy cập tại: https://vi.wikipedia.org/wiki/YouTube

[5] Studocu (2024), "Đồ án cuối kì Big Data", truy cập tại: https://www.studocu.vn/vn/document/truong-dai-hoc-kinh-te-thanh-pho-ho-chi-minh/khoa-hoc-du-lieu/do-an-big-data-do-an-cuoi-ki-big-data/95860795?origin=content-sidebar-recent

[6] Github (2023), "Phân tích dữ liệu trên nền tảng Youtube", truy cập tại: https://github.com/trinhvinhphuc/youtube_data_analysis/blob/master/Report.pdf -

[7] Sciencedirect (2025), "EnTube: Exploring Key Video Features for Advancing YouTube Engagement", truy cập tại: https://www.sciencedirect.com/science/article/abs/pii/S187595212500014X#preview-section-abstract

[8] Emerald (2024), "Optimizing marketing strategy: a video analysis approach", truy cập tại: https://www.emerald.com/insight/content/doi/10.1108/mip-12-2023-0655/full/html

[9] TextBlob (2024), "TextBlob: Simplified Text Processing" , truy cập tại: https://textblob.readthedocs.io/

[10] Scikit-learn (2024), "KMeans", truy cập tại: https://scikit-learn.org/stable/modules/generated/sklearn.cluster.KMeans.html

[11] Scikit-learn (2024), "silhouette-score", truy cập tại:

https://scikit-learn.org/stable/modules/generated/sklearn.metrics.silhouette_score.html

[12] DataCamp (2024), "Introduction to t-SNE: Nonlinear Dimensionality Reduction and Data Visualization", truy cập tại: Introduction to t-SNE: Nonlinear Dimensionality Reduction and Data Visualization | DataCamp

[13] Analytics Vidhya (2024), "SMOTE for Imbalanced Classification with Python", truy cập tại: https://www.analyticsvidhya.com/blog/2020/10/overcoming-class-imbalance-using-smote-techniques/

[14] Geeks for Geeks (2024), "Elbow Method for optimal value of k in KMeans", truy cập tại: Elbow Method for optimal value of k in KMeans - GeeksforGeeks

[15] Scikit-learn (2024), "K Neighbors Regression", truy cập tại: https://scikit-learn.org/stable/modules/generated/sklearn.neighbors.KNeighborsRegressor.html

[16] Geeks for Geeks (2024), "Regression Metrics", truy cập tại: https://www.geeksforgeeks.org/regression-metrics

[17] ResearchGate (), "A Density Based Algorithm for Discovering Density Varied Clusters in Large Spatial Databases", truy cập tại: https://www.researchgate.net/publication/44250717_A_Density_Based_Algorithm_for_Discovering_Density_Varied_Clusters_in_Large_Spatial_Databases