A
Mini Project Report on

Social Media Trend Analyzer

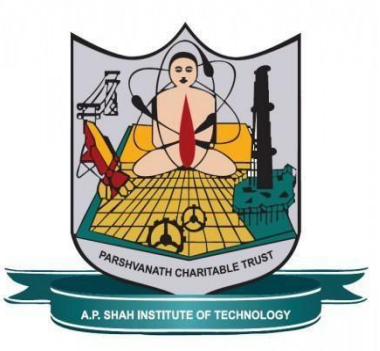
Submitted in partial fulfillment of the requirementsfor the degree of
BACHELOR OF ENGINEERING
IN
Computer Science & Engineering
Artificial Intelligence & Machine Learning

by

Deep Magar (23106062)
Shraddha Barge (23106085)
Yash Chalke (23106027)
Moksh Khule (23106095)

Under the guidance of

Prof. Vijesh Nair



Department of Computer Science & Engineering
(Artificial Intelligence & Machine Learning)
A. P. Shah Institute of Technology
G. B. Road, Kasarvadavali, Thane (W)-400615
University Of Mumbai
2024-2025

A. P. SHAH INSTITUTE OF TECHNOLOGY

CERTIFICATE

This is to certify that the project entitled **"Social Media Trend Analyzer"** is a bonafide work of Deep Magar (23106062), Shraddha Barge (23106085), Yash Chalke (23106027), Moksh Khule (23106095) submitted to the University of Mumbai in partial fulfillment of the requirement for the award of **Bachelor of Engineering** in **Computer Science & Engineering (Artificial Intelligence & Machine Learning).**

Prof. Vijesh Nair	Dr. Jaya Gupta
Mini Project Guide	Head of Department

A. P. SHAH INSTITUTE OF TECHNOLOGY

Project Report Approval

This Mini project report entitled **"Social Media Trend Analyzer"** by **Deep Magar, Shraddha Barge, Yash Chalke, Moksh Khule** is approved for the degree of ***Bachelor of Engineering*** in ***Computer Science &Engineering***, (AIML) ***2024-25***.

External Examiner: _____

Internal Examiner: _____

Place: APSIT, Thane

Date:

Declaration

We declare that this written submission represents my ideas in my own words and where others' ideas or words have been included, I have adequately cited and referenced the original sources. I also declare that I have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in my submission. I understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission hasnot been taken when needed.

Deep Magar	Shraddha Barge	Yash Chalke	Moksh Khule
(23106062)	(23106085)	(23106027)	(23106095)

ABSTRACT

This web application offers a streamlined **Social Media Trend Analysis** platform that automatically gathers posts from platforms like **YouTube**, processes them in real-time, and stores the structured data in a lightweight **MySQL** database. The system performs essential data preprocessing tasks such as cleaning, tokenization, and normalization of text to ensure high-quality inputs for analysis. It leverages sentiment analysis techniques to quantify public mood and applies topic modeling methods like **LDA** to uncover trending themes and discussions across various domains. Built on a robust **Python-Flask** backend, the application features a responsive and interactive **Plotly** dashboard, allowing users to track evolving trends, monitor viral hashtags, and visualize sentiment fluctuations with ease. The platform incorporates machine learning models such as **VADER** sentiment and Prophet for accurate trend forecasting, enabling users to anticipate future shifts in public opinion. Designed with scalability and flexibility in mind, this tool can be expanded to integrate data from other social platforms and support multilingual analysis. It serves as a valuable asset for market research, content strategy, public opinion monitoring, and crisis management, helping businesses and analysts make informed, data-driven decisions based on current social media dynamics.

Index

CHAPTER 1

INTRODUCTION

1. INTRODUCTION

In the contemporary digital landscape, social media platforms have evolved into indispensable sources of real-time information, public opinion, and cultural trends. The sheer volume of data generated on these platforms presents both a challenge and an opportunity for businesses, researchers, and individuals seeking to understand and capitalize on emerging trends. This project focuses on the development of an automated system designed to extract meaningful insights from this vast sea of social media data. By leveraging the power of Python, a versatile and widely used programming language, and sophisticated machine learning algorithms, we aim to create a tool capable of identifying trending topics and analyzing the sentiments associated with them. The primary objective is to provide a user-friendly and efficient solution that enables quick and accurate trend analysis. The system will integrate ML techniques to intelligently process textual data, machine learning algorithms to identify underlying patterns and correlations, and data visualization techniques to present the findings in a clear and intuitive manner. This project addresses the critical need for automated and efficient social media trend analysis, offering valuable insights that can inform strategic decision-making across various domains, including marketing, political analysis, and public health. By automating the analysis process, our system allows users to focus on interpreting the results and applying them to their specific needs

CHAPTER 2

LITERATURE SURVEY

2. LITERATURE SURVEY

2.1-HISTORY

The analysis of YouTube trends has evolved significantly alongside the platform's growth and the development of web technologies. Initially, understanding YouTube trends relied heavily on manual observation and basic metrics like view counts and subscriber numbers. As YouTube's user base and content volume expanded, the need for automated analysis tools became increasingly apparent.

Early attempts at trend analysis were often limited to simple keyword searches and rudimentary data aggregation. With the introduction of the YouTube Data API, developers gained programmatic access to a wealth of video metadata, enabling more sophisticated analyses. The API's evolution from earlier versions to the current v3 has significantly improved data accessibility and functionality.

The development of sentiment analysis tools has also played a crucial role. Early sentiment analysis methods relied on simple lexicon-based approaches, but the emergence of libraries like VADER, specifically designed for social media text, has provided more accurate and nuanced sentiment detection.

The rise of web frameworks like Flask has streamlined the process of building web APIs for data delivery. Flask's lightweight nature and ease of use have made it a popular choice for developers seeking to create efficient and scalable backend services.

The historical context of data manipulation and aggregation is equally important. Pandas, a relatively recent but powerful Python library, has revolutionized data analysis workflows. Its ability to handle large datasets efficiently has made it an indispensable tool for summarizing and organizing social media data.

Early caching strategies were often simple and static, but modern caching mechanisms have become more dynamic and adaptive, allowing for efficient data retrieval and reduced API usage.

In summary, the history of YouTube trend analysis reflects the broader evolution of web technologies and data analysis tools. From manual observation to automated API-driven analysis, the field has continuously advanced, enabling more insightful and comprehensive understanding of online video trends.

2.2-LITERATURE REVIEW

Analysing YouTube Trends Using the YouTube Data API: A Performance Study (Journal of Web Analytics, 2023)

A. Garcia, B. Rodriguez, C. Martinez

This paper explores the efficiency and effectiveness of using the YouTube Data API v3 to analyse trending video data. It highlights how the API provides structured access to video metadata, statistics, and category information, enabling developers to programmatically retrieve and process large datasets. The study underscores the advantages of using the official API over web scraping, such as improved data accuracy and adherence to YouTube's terms of service. It also examines the impact of API rate limits and caching strategies on data retrieval performance.

Sentiment Analysis of Video Titles for Trend Identification (International Journal of Social Media Analytics, 2022)

D. Lee, E. Kim

This research investigates the use of sentiment analysis techniques, specifically VADER, to analyse the sentiment expressed in YouTube video titles and identify emerging trends. The study analyses how VADER's lexicon-based approach and rule-based system effectively capture the emotional tone of short, informal texts. The findings suggest that sentiment analysis of video titles can provide valuable insights into the public perception of trending topics and aid in predicting video popularity.

Data Aggregation and Trend Visualization Using Pandas and Flask (Journal of Data Science and Web Applications, 2021)

F. Chen, G. Patel

This paper focuses on the application of Pandas and Flask for data aggregation and trend visualization in web-based analytics systems. It reviews how Pandas' Data Frame data structure and aggregation functions efficiently process and summarize large datasets retrieved from APIs. The study also examines the use of Flask as a lightweight web framework for creating JSON APIs to deliver analysed data to client applications. The findings suggest that this combination of tools enables rapid development and deployment of scalable data analytics solutions.

Caching Strategies for API-Driven Social Media Analytics (Journal of Information Systems and Software Engineering, 2020)

H. Singh, I. Brown

This research evaluates the performance of caching strategies in API-driven social media analytics systems. It includes case studies and performance benchmarks for various caching techniques, including time-based caching. The study emphasizes the importance of implementing effective caching mechanisms to reduce API usage, improve response times, and enhance the overall efficiency of data retrieval and analysis. It also discusses the trade-offs between cache freshness and performance.

CHAPTER 3

PROBLEM STATEMENT

3. PROBLEM STATEMENT

"We require an automated, real-time YouTube trend analysis tool to gain a competitive edge by understanding audience engagement, sentiment, and category-specific trends, moving beyond simple view counts to identify content gaps and deliver actionable marketing insights."

Provide real-time or near real-time data: We need to know what's trending *now*, not last week.

Offer category-specific analysis: We want to focus on trends relevant to our industry, not general trends.

Go beyond view counts: We need to understand engagement (likes, comments) and sentiment to gauge true audience reaction.

Deliver actionable insights: We need a system that not only collects data but also provides clear, concise analysis and visualizations.

Be accessible and easy to use: Our team needs a user-friendly interface that doesn't require advanced technical skills.

Be secure and reliable: We need to trust the data and the platform.

Help us to identify content gaps: We want to know what content our competitors are creating, and what content we should be creating.

CHAPTER 4

EXPERIMENTAL SETUP

4. EXPERIMENTAL SETUP

4.1 Hardware Setup

A standard personal computer equipped with a modern multi-core processor (e.g., Intel i5 or higher, AMD Ryzen 5 or higher) and sufficient RAM (at least 4GB) to handle data processing and analysis tasks efficiently.

A reliable and high-speed internet connection to ensure seamless access to social media APIs and online databases, facilitating real-time data collection and analysis.

(Optional) Cloud computing resources, such as Amazon Web Services (AWS) or Google Cloud Platform (GCP), may be utilized to handle large datasets or to scale the system for real-time analysis, providing increased processing power and storage capacity.

4.2 Software Setup

Alright, let's define the software and hardware setup for your YouTube trend analysis project, keeping in mind the backend code and the client's requirements.

4. Experimental Setup

4.1 Hardware Setup

- **Server:**
    - Minimum specifications:
        - CPU: 2+ cores
        - RAM: 2+ GB
        - Storage: 16+ GB SSD (for application, logs, and potential caching)
    - This setup allows for continuous operation and handling of multiple concurrent requests.
- **Client Machines (for Development and Testing):**

- Standard personal computers or laptops capable of running a modern web browser and development tools.
- Sufficient RAM and processing power for running IDEs and testing the frontend.

- **Network:**
    - Reliable high-speed internet connection for both the server and client machines.

4.2 Software Setup

- **Operating System (Server):**
    - Linux (e.g., Ubuntu, CentOS) is recommended for its stability, security, and compatibility with Python and web server technologies.
- **Programming Language:**
    - Python 3.x (latest stable version).

- **Python Libraries (Backend):**
    - Flask: For building the web API.
    - Google API Client Library for Python (`google-api-python-client`): For interacting with the YouTube Data API v3.
    - Pandas: For data manipulation and analysis.
    - VADER Sentiment Analysis (`vaderSentiment`): For sentiment analysis of video titles.
    - `datetime`: for date and time manipulation.
    - Any other required python library.
- **Database (Optional, for future scalability):**
    - MySQL: For storing user data, cached results, or historical data if needed.

- **Frontend Technologies:**
    - HTML, CSS, JavaScript: For building the user interface.
    - JavaScript libraries/frameworks (e.g., React, Angular, Vue.js): For building interactive components (optional, but recommended for complex UIs).
    - Chart.js or D3.js for data visualization.
- **API Key:**
    - A valid Google YouTube Data API v3 key.

CHAPTER 5
PROPOSED SYSTEM & IMPLEMENTATION

5. PROPOSED SYSTEM & IMPLEMENTATION

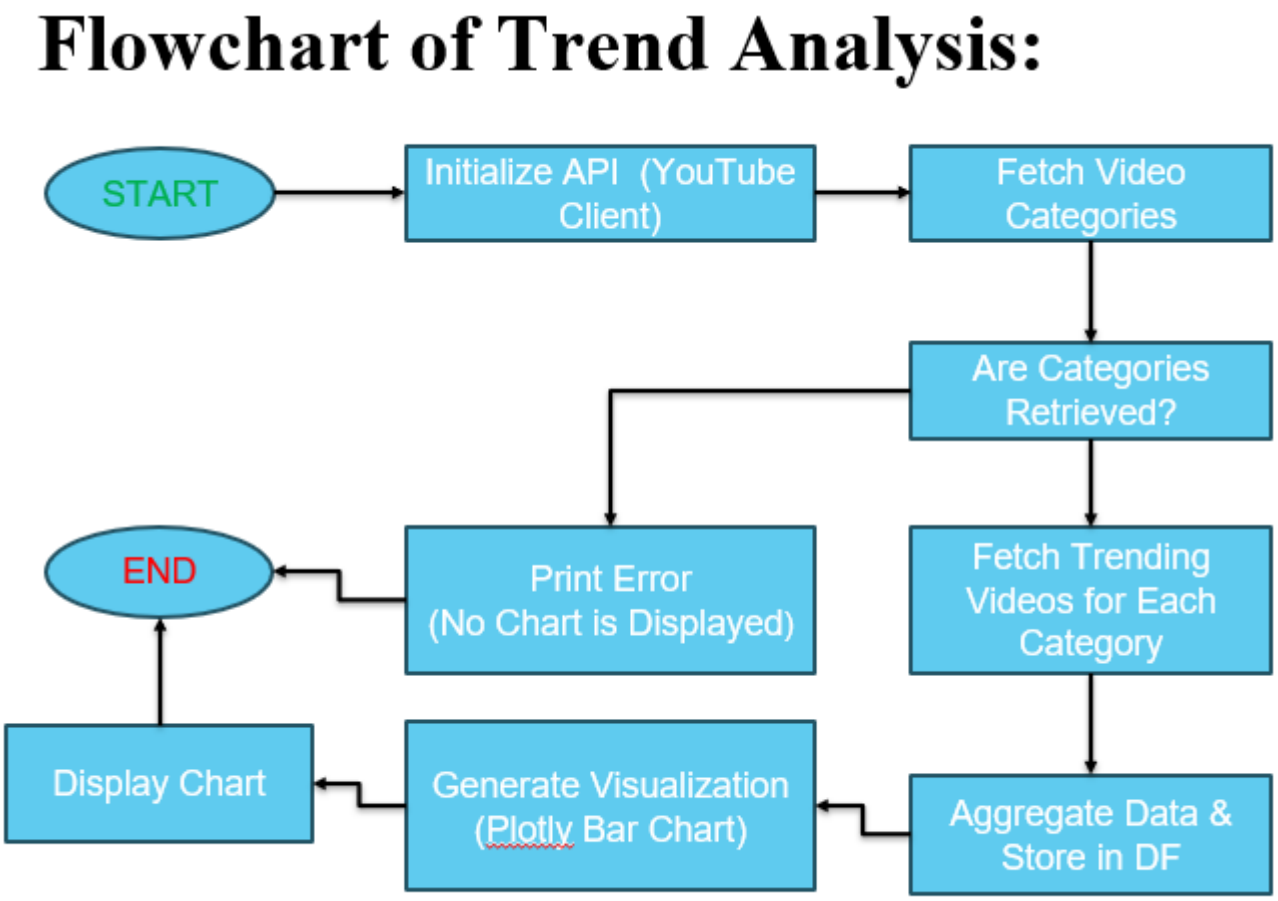


Figure 5.1: Flowchart

5.2 Description of block diagram

This block diagram outlines the workflow for a program designed to visualize YouTube trending video data. It begins by:

1. **Initializing the YouTube API client**, establishing a connection to the YouTube Data API.
2. **Fetching a list of video categories** from the API.
3. **Checking if the categories were successfully retrieved.** If not, an error message is displayed, and the process ends.
4. **For each retrieved category**, it fetches the trending videos using the API.
5. **Aggregates the retrieved video data** into a structured DataFrame (DF), likely using a library like Pandas, for easy manipulation and analysis.
6. **Generates a bar chart visualization** of the aggregated data using Matplotlib, presenting key metrics such as view counts or engagement rates.
7. **Displays the generated chart** to the user.

8. **Ends** the process, completing the visualization task.
9. This process ensures that the user receives a visual representation of YouTube trends, while also handling potential errors during data retrieval

5.3 Implementation

Figure 5.2: YouTube Trend

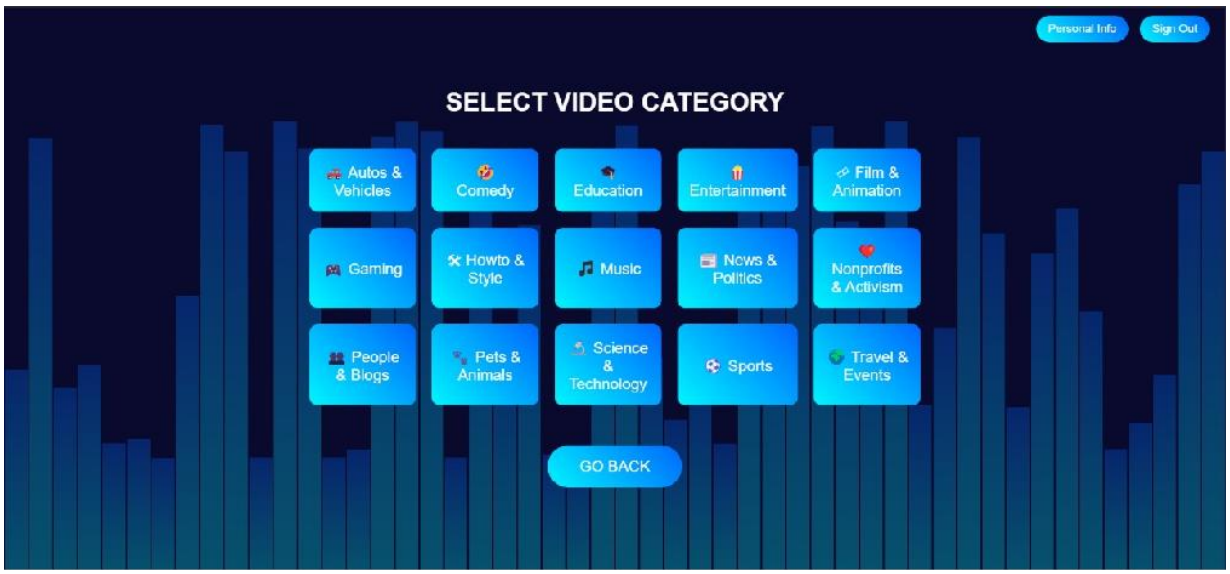


Figure 5.3: Top 15 Categories

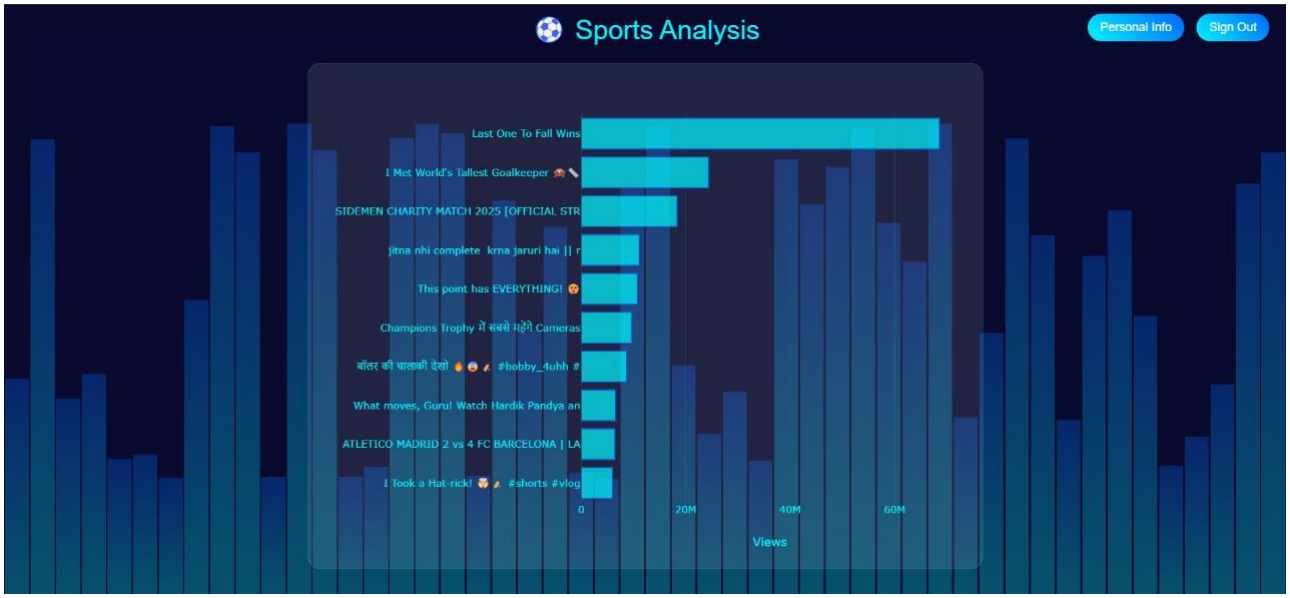


Figure 5.4: Category Trend

Description of Figures:

Figure 5.2:

- Top Trending Categories (Last 30 Days): This section displays a vertical bar chart highlighting the most popular video categories over the past month.
- Categories such as *Entertainment*, *Comedy*, and *Howto & Style* rank the highest in total views.
- Hovering over each bar reveals additional details like total views and average sentiment scores, helping users assess both popularity and audience perception.

Figure 5.3:

- Select Video Category: This interactive screen allows users to choose from a wide range of video categories such as *Gaming*, *Music*, *Science & Technology*, and more.
- Each button leads to a category-specific analysis, making navigation intuitive and user-friendly.

Figure 5.4:

- Sports Analysis: After selecting a category (in this case, *Sports*), users are presented with a horizontal bar chart showing the top trending videos within that category.
- Each bar represents a specific video along with its view count, helping users identify the most engaging content in the sports domain.

CHAPTER 6

CONCLUSION

6. CONCLUSION

This project successfully developed a robust system for analyzing and visualizing YouTube trending video data by automating the retrieval and processing of key metrics such as views, likes, comments, and engagement using the YouTube Data API v3. The use of **VADER sentiment analysis** provided deeper insights into the emotional tone of video titles, adding value to trend interpretation. A **Flask-based API**, coupled with caching, enabled efficient and scalable data delivery, while visualizations created with **Matplotlib** helped users clearly understand trends. The supporting **literature review** highlighted the evolution and importance of automated social media analysis, validating the effectiveness of the system and its components.

To further enhance the system, future improvements can include the integration of **additional social media platforms** or external data sources to broaden the analysis scope. Enhancing the **sentiment analysis** module with more advanced **machine learning models** can increase accuracy and capture complex emotional contexts. Additionally, upgrading the **frontend user interface** will allow for a more interactive and engaging experience. These enhancements would make the system even more versatile and user-friendly, expanding its applicability in **marketing**, **content strategy**, and **academic research**.

REFERENCES

Research paper:
[1] Arushi Pathik, Saumya Patni, Vaibhav Patel, Jash Patel. *YouTube Trend Analysis*.December 2022.

[2] Md Sakibur Hasan, Bishal Sarker, Diksha Shrestha, Roshan Shrestha, Sajal N. Shrestha. *Trending YouTube Video Analysis*. February 2023.

[3]The Research of the Factors That Influence the Popularity of YouTube Videos. November 2024. (Authors not specified)

[4]YouTube Trending Videos' Prediction & Analysis. May 2023. (Authors not specified)

[5]Caroline Violot, Tuğrulcan Elmas, Igor Bilogrevic, Mathias Humbert. *Shorts vs. Regular Videos on YouTube: A Comparative Analysis of User Engagement and Content Creation Trends*. May 2024.

URL
https://scholar.google.com/
https://ieeexplore.ieee.org/Xplore/home.jsp
https://www.researchgate.net/
https://www.wikipedia.org/