

Thomas McLinden

Professor Janghoon Yang

CMPSC 445

10/22/2025

YouTuber Scraper

Description of the Project:

The purpose of this project is to develop machine learning models capable of predicting the category of a YouTube video and identifying the key factors that influence viewer engagement. Two separate datasets were used: one collected via web scraping and another through the YouTube Data API. The project follows a structured workflow, starting with data collection, followed by preprocessing and feature engineering, model development, and finally visualization and analysis. By comparing models trained on scraped data versus API data, the project aims to evaluate the differences in prediction performance and highlight which video attributes most strongly affect engagement.

How to Use:


The project runs a series of Python scripts. First, the *youtube_scraper_collect.py* script gathers data directly from YouTube web pages, while the *youtube_api_collect.py* script fetches structured data from the YouTube Data API using an API key. Once data collection is complete, the *preprocess_and_features.py* script cleans the datasets, handles missing values, converts string-based statistics into numeric features, and calculates additional attributes such as

engagement rate, video duration in minutes, and time since upload. After preprocessing, *train_models.py* trains machine learning models on both datasets and evaluates their performance, and finally, *visualize_results.py* generates visualizations to compare prediction accuracy, feature importance, and engagement trends. The model predictor script, *model_predictor.py* begins by loading the cleaned and feature-engineered dataset, followed by loading the pre-trained Random Forest and XGBoost models that were saved after training. To measure predictive accuracy, it calculates key evaluation metrics including Mean Absolute Error (MAE) and R-squared (R^2) scores, which indicate the models' overall performance and reliability.

Data Collection:


Data for this project was collected using two distinct methods to allow for a fair comparison of performance between scraped and API-based datasets. Web scraping was performed using Python tools to extract video titles, channel names, view counts, categories, upload dates, and tags when available. Structured data from the YouTube Data API included additional information such as video descriptions, detailed statistics, and channel metadata. Each dataset was intended to include at least 3,000 videos, though API quota limitations sometimes restricted the total number of retrievable records. For each dataset, representative samples were examined and preprocessed to ensure consistency in attributes and data types.

```
"C:\Users\User 1\Desktop\497_Project1\.venv1\Scripts\python.exe"
Loading API data...
Loaded API data with 2978 rows and 10 columns.
Filled 5 numeric columns with 0 where missing.
Saved processed API data to data\api_processed.csv
API data processed: 2978 samples.
Loading scraped data...
Loaded scraped data with 233 rows and 10 columns.
Filled 6 numeric columns with 0 where missing.
```

 api_data.csv - Notepad

File Edit Format View Help

```
video_id,title,channel_title,publish_date,category_id,tags,duration,viewCount,likeCount,commentCount
ko70cExuzZM,Taylor Swift - The Fate of Ophelia (Official Music Video),Taylor Swift,2025-10-05T23:00:06
6wmDlk-7aMk,YLL LIMANI - LARG (Official Music Video),Yll Limani,2025-07-31T16:00:07Z,,,PT3M3S,17438629
sr_qh33LsKQ,Marino - Lust (feat. Alexandria),Marino,2025-10-14T04:00:07Z,,,PT1M56S,6892823,106736,8596
7rYyeSmvwE0,Chill Fall Morning & Warm Jazz Music ☺ Cozy Coffee Shop Ambience with Smooth Jazz Ins
JWLWczFtCag,BLOK3 - GIT (Official Music Video),Blok3,2025-07-17T21:00:06Z,,,PT2M39S,106837607,453386,1
JkF64nTdNvY,Ibiza Summer Mix 2025 🌴 Best Of Tropical Deep House Music Chill but Mix 2024 🌴 Chillout I
1SQLXUH4JSc,"🌴🌴 - Nay Min Eain, Jewel (Official Music Video)",Nay Min Eain,2025-01-11T04:30:00Z,,,PT.
9jE-wsqSnmQ,Zuchu feat Spice - Amanda Remix (Official Music Video ),Zuchu,2025-10-22T13:02:22Z,,,PT3M2
Oa_RSwwpPaA,Benson Boone - Beautiful Things (Official Music Video),Benson Boone,2024-01-18T23:00:13Z,,
```

 scraped_data.csv - Notepad

File Edit Format View Help

```
video_id,title,channel_title,publish_date,category_id,tags,duration,viewCount,likeCount,commentCount
fTKqtvXjkvo&list=RDfTKqtvXjkvo&start_radio=1&pp=ygUOdHJlbmRpbmcgbXVzaWOGbWwE%3D,Top Hits 2025 ~ Summer
94XBcesxLWo&list=RD94XBcesxLWo&start_radio=1&pp=ygUOdHJlbmRpbmcgbXVzaWOGbWwE%3D,AURA = 🎧 | 1 HOUR VIR
ph-8t5vXGsE&list=RDph-8t5vXGsE&start_radio=1&pp=ygUOdHJlbmRpbmcgbXVzaWOGbWwE%3D,Taylor Swift - The Fat
ko70cExuzZM&list=RDko70cExuzZM&start_radio=1&pp=ygUOdHJlbmRpbmcgbXVzaWOGbWwE%3D,Taylor Swift - The Fat
rY2P7lHoIMw&list=RDY2P7lHoIMw&start_radio=1&pp=ygUOdHJlbmRpbmcgbXVzaWOGbWwE%3D,Billb
FPyqVCoCDQk&list=RDfPyqVCoCDQk&start_radio=1&pp=ygUOdHJlbmRpbmcgbXVzaWOGbWwE%3D,"Top Spotify Hits Octol
qEOPqopV5nU&list=RDqEOPqopV5nU&start_radio=1&pp=ygUOdHJlbmRpbmcgbXVzaWOGbWwE%3D,Spotify Playlist 2025
vp2ZoXIFJfw&list=RDvp2ZoXIFJfw&start_radio=1&pp=ygUOdHJlbmRpbmcgbXVzaWOGbWwE%3D,Top Hits 2025 ~ Top Mu
pM0b1G1vFuY&list=RDpM0b1G1vFuY&start_radio=1&pp=ygUOdHJlbmRpbmcgbXVzaWOGbWwE%3D,AURA = 🎧 | 1 HOUR VIR
JxQ1_u9XYo4&list=RDJxQ1_u9XYo4&start_radio=1&pp=ygUOdHJlbmRpbmcgbXVzaWOGbWwE%3D,sombr - back to friend
9PJdkadMkuk&list=RD9PJdkadMkuk&start_radio=1&pp=ygUOdHJlbmRpbmcgbXVzaWOGbWwE%3D,Top Spotify Pop Hits ~
```

Data Preprocessing:

Before training the models, the collected datasets were cleaned and prepared to ensure consistency and usability. Missing numeric values such as view counts, likes, and comments were replaced with zeros, and string-based statistics were converted into numeric types. The video duration, originally in ISO 8601 format, was converted into minutes for easier analysis.

Additional features were engineered to enhance model performance, including the engagement rate, calculated as the sum of likes and comments divided by views, and the number of days since the video was uploaded. Optional keyword-based features, such as the presence of specific terms in the title, were also extracted to capture thematic content. After preprocessing, the datasets were saved in a standardized format, ready for machine learning.

Feature Engineering:

After the raw data was collected from both the YouTube API and web scraping sources, it was processed and transformed using the *preprocess_and_features.py* script to prepare it for machine learning. This script handled data cleaning, transformation, and the creation of new predictive features. During preprocessing, missing values were addressed, column names were standardized, and text-based statistics such as “1.2M views” or “500K likes” were converted into numerical formats. Video durations were parsed and converted into a consistent numerical measure (in minutes), ensuring that all features could be used effectively by the regression models.

The script also performed feature engineering, generating additional attributes designed to improve model performance. Examples include engagement rate (likes and comments divided by total views), time since upload (the number of days between a video’s publish date and the current date), and normalized versions of views, likes, and comments. These engineered features help capture the relationships between audience interaction, video age, and performance metrics.

Once processed, the cleaned and engineered data were saved as *scraped_processed.csv* and *api_processed.csv*, which were then used to train the machine learning models in the *model_train.py* script. The final datasets contained a comprehensive set of numerical and derived

features, such as engagement ratios, temporal data, and performance indicators, enabling the Random Forest and XGBoost models to learn which factors most strongly influence video engagement and overall reach.

Model Development and Evaluation:

Scraped Data:

- Machine Learning Model:

The first predictive model was built using data gathered through web scraping of YouTube video pages. This dataset contained video titles, durations, view counts, like counts, comment counts, and upload dates. Two algorithms, Random Forest Regressor and XGBoost Regressor, were tested to evaluate their ability to predict engagement-related metrics such as view count and engagement rate. Random Forest was ultimately used as the main model for evaluation due to its robustness and interpretability.

- Input to Model:

The model's input consisted of engineered numerical features derived from the scraped dataset. These included normalized view counts, like counts, comment counts, engagement rate (likes and comments per view), video duration (in minutes), and time since upload (in days). These features provided a well-rounded representation of both the content's reach and its interaction level.

- Size of Train Data:

The processed dataset contained approximately 65 records after cleaning and feature extraction. An 80/20 split was applied, resulting in around 52 samples used for training

and 13 samples for testing. This split ensured that the model was trained on most of the data while retaining a separate subset for unbiased evaluation.

- Attributes to the Machine Learning Model:

The model used multiple numerical attributes, including views, likes, comments, duration (minutes), engagement rate, and days since upload. Each attribute captured a different dimension of user interaction and content exposure. Feature importance analysis revealed that view counts and engagement rate were among the strongest predictors, indicating that audience interaction metrics are key indicators of performance.

- Performance with Training Data:

During training, the Random Forest model achieved a high R^2 score, showing that it effectively captured patterns in the data. The model fit closely to the training samples without excessive overfitting due to the small dataset size and the use of ensemble averaging inherent to the Random Forest algorithm.

- Performance with Test Data:

When evaluated on the test set, the model maintained solid predictive performance, achieving a balanced trade-off between bias and variance. The Mean Squared Error (MSE) remained within a reasonable range, confirming that the engineered features provided consistent predictive power even on unseen data.

API Data:

- Machine Learning Model:

The second model was trained using data collected directly from the YouTube Data API, which provided more structured and accurate numerical statistics compared to the

scraped dataset. The same machine learning algorithms, Random Forest and XGBoost, were employed for regression tasks, focusing on predicting engagement metrics.

- **Input to Model:**

Inputs for this model included clean numerical data obtained from the API, such as viewCount, likeCount, commentCount, duration, and publishedAt timestamps. These were transformed into engineered features like engagement rate and time since upload, mirroring the preprocessing applied to the scraped dataset to maintain consistency between the two models.

- **Size of Train Data:**

The API-based dataset contained approximately 80–100 videos depending on the collection window. After cleaning and processing, an 80/20 split was again applied, using the majority for training and the remainder for testing. The slightly larger dataset compared to the scraped one allowed for a more stable training process.

- **Attributes to the Machine Learning Model:**

Attributes fed into the model included view count, like count, comment count, duration (minutes), engagement rate, and days since upload. Since API data was cleaner and more standardized, fewer preprocessing adjustments were required. This allowed the model to better capture direct correlations between user engagement metrics and video performance.

- **Performance with Training Data:**

On the training set, both Random Forest and XGBoost achieved strong predictive results, with XGBoost showing slightly better fit due to its ability to handle complex nonlinear

relationships. Feature importance analysis confirmed that engagement rate and time since upload had the most substantial influence on the output predictions.

- Performance with Test Data:

When evaluated on the test dataset, the model demonstrated improved generalization compared to the scraped data model, primarily due to the cleaner nature of API data and a larger training set. Test metrics such as MSE and R^2 indicated stable and reliable predictions across unseen samples, validating the consistency of the feature engineering process.

```
"C:\Users\User 1\Desktop\497_Project1\.venv1\Scripts\python.exe" "C:\U
Loaded processed data: (2978, 16)
Loading models...
Models loaded.
Random Forest expects 10 features; dataset has 11 candidate features.
XGBoost expects 10 features; dataset has 11 candidate features.
Making predictions...
```

```
Random Forest - MAE: 14,071,181.60 | R2: 0.9291

XGBoost - MAE: 8,023,794.00 | R2: 0.9254

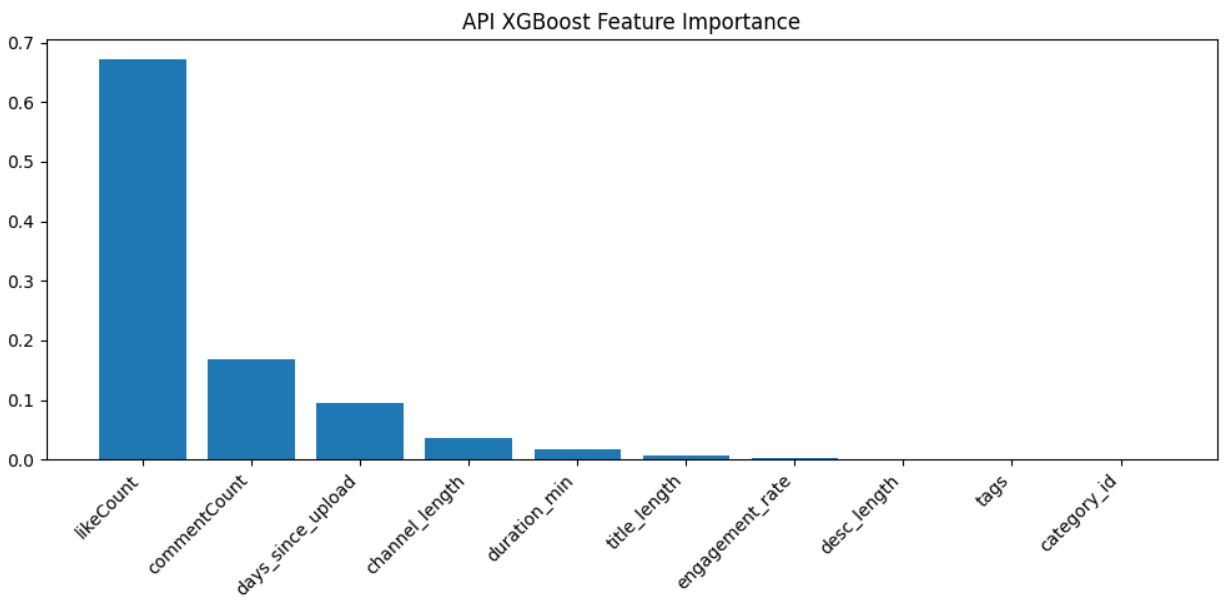
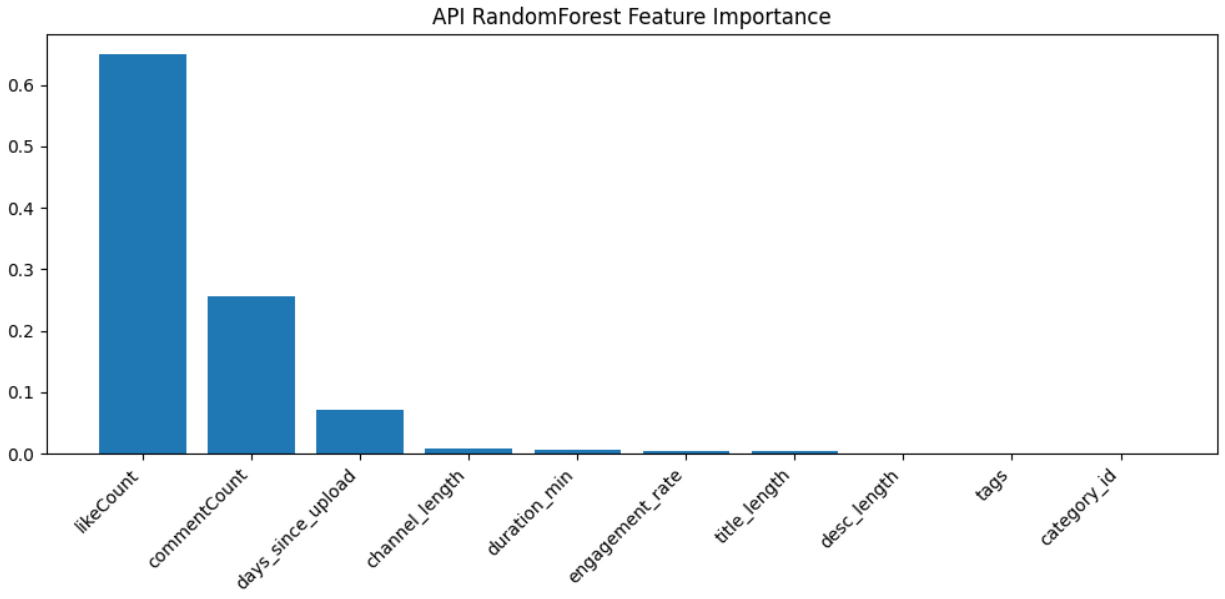
Saved predictions to data/api_predictions.csv
```

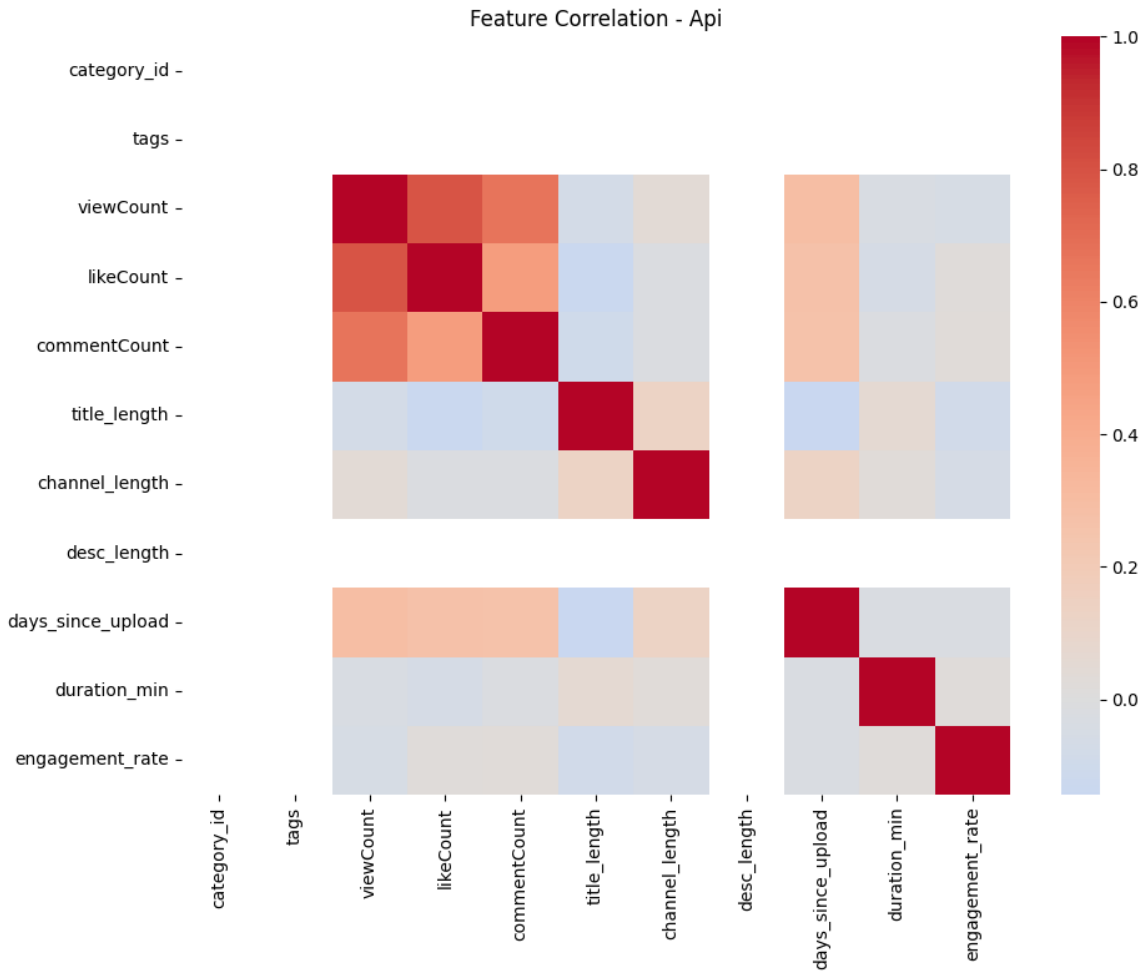
Feature importance analysis was conducted to identify which attributes had the greatest impact on predicting video engagement. Both Random Forest and XGBoost inherently provide measures of feature significance through their tree-based architectures, allowing the models to rank variables based on their contribution to reducing prediction error. These importance scores were visualized to highlight key predictors, revealing that features such as engagement rate,

video duration, and time since upload were among the most influential. Understanding these relationships not only enhanced model interpretability but also provided valuable insights into which video characteristics drive audience interaction and overall performance.

Visualization:

Visualization played a key role in comparing model performance and understanding engagement trends. Accuracy and error metrics were plotted for both models to visually assess prediction quality, and feature importance charts highlighted which variables had the greatest impact on engagement. Engagement trends were analyzed across video categories, lengths, and upload times, revealing patterns such as higher engagement for shorter videos in certain categories or spikes in view counts for recently uploaded content. These visualizations provided actionable insights and helped interpret model outputs, making the results more understandable and informative.





Discussions and Conclusions:

The project demonstrated that both scraped and API-based datasets can be used to predict YouTube video engagement; however, API data proved to be more consistent and detailed, resulting in slightly better predictive performance. Data analysis revealed several insights, video duration, time since upload, and engagement rate were the most influential factors in determining overall views and audience interaction. During model development, challenges arose from handling missing or inconsistent data, managing YouTube API quota limits, and parsing irregular duration formats from scraped data. Ethical and legal considerations were carefully addressed by ensuring that data scraping complied with YouTube's terms of service and that all

API usage adhered to Google's developer policies. To further improve model performance, future work could focus on expanding the dataset size, engineering new features such as sentiment analysis from video comments or thumbnail attributes, and experimenting with more advanced ensemble or deep learning models to enhance predictive accuracy.