3/8/2024

Varanasi Venkata Akshay

Intern career data analysit TAsk-1 internship project

YOUTUBE STREAMER ANALYSIS

**Prerequisites of model execution:**

**For installing “PANDAS”**

* pip install pandas

**For installing “MATPLOTLIB”**

* pip install matplotlib

**For installing “SEABORN”**

* pip install seaborn

**PYTHON CODE:**

import pandas as pd

import matplotlib.pyplot as plt

import seaborn as sns

**Task 1: Data Exploration**

**load dataset**

youtube\_data = pd.read\_csv("youtubers\_df.csv")

print(youtube\_data.info())

**Output:**

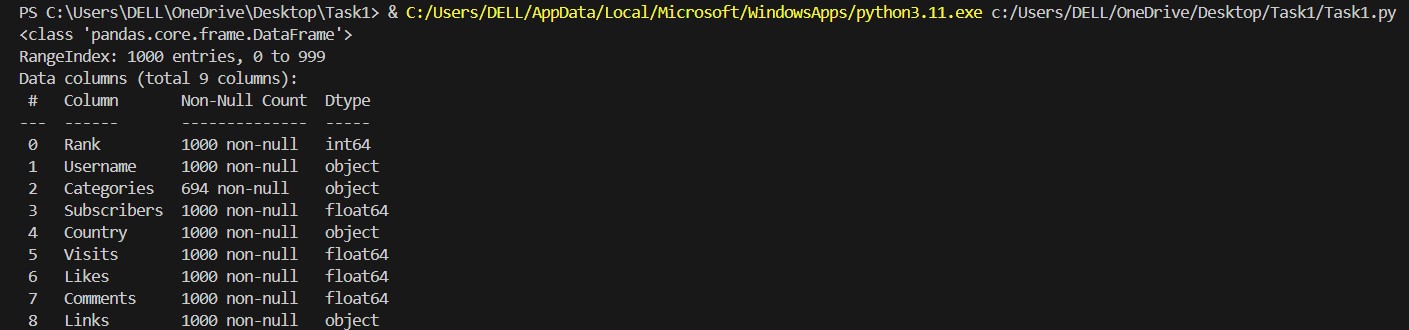


Fig.1 Loading the Data set

**Check for missing data and outliers**

print(youtube\_data.describe())

plt.boxplot(youtube\_data['Subscribers'])

plt.show()

**Output:**

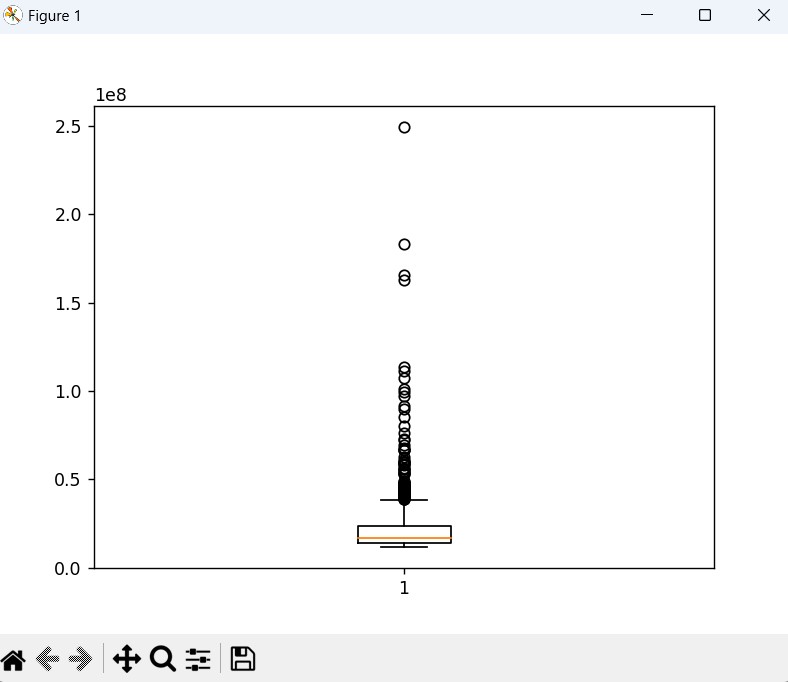


Fig.2 Checking of Missing Data and Outliers in the data

**Trend Analysis**

**Identify popular categories using bar plots or pie charts**

category\_counts = youtube\_data['Categories'].value\_counts()

category\_counts.plot(kind='bar')

plt.xlabel('Categories')

plt.ylabel('Count')

plt.title('Popular Categories')

plt.show()

**Output:**

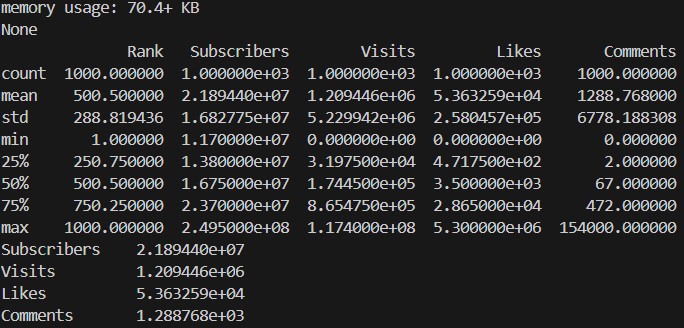


Fig.3 Count of Categories

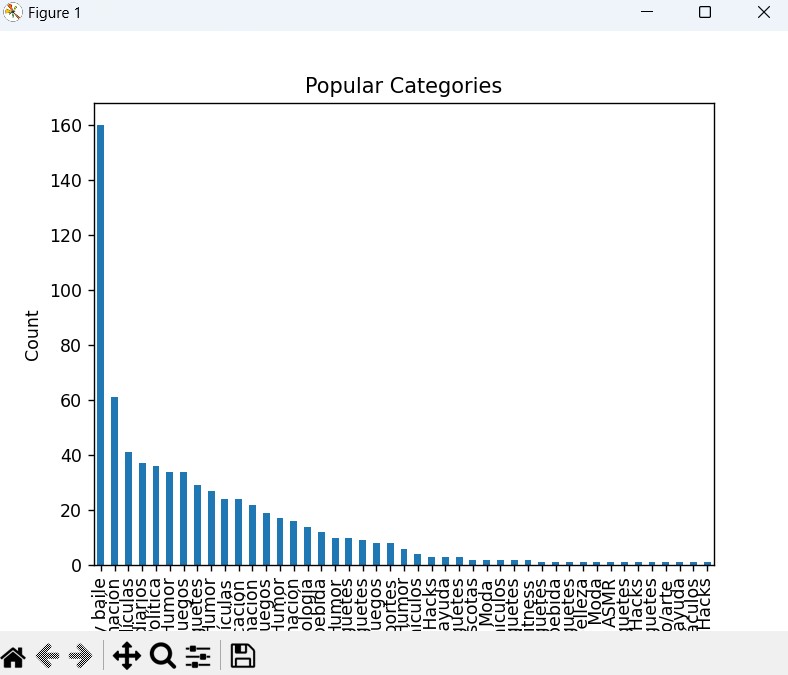


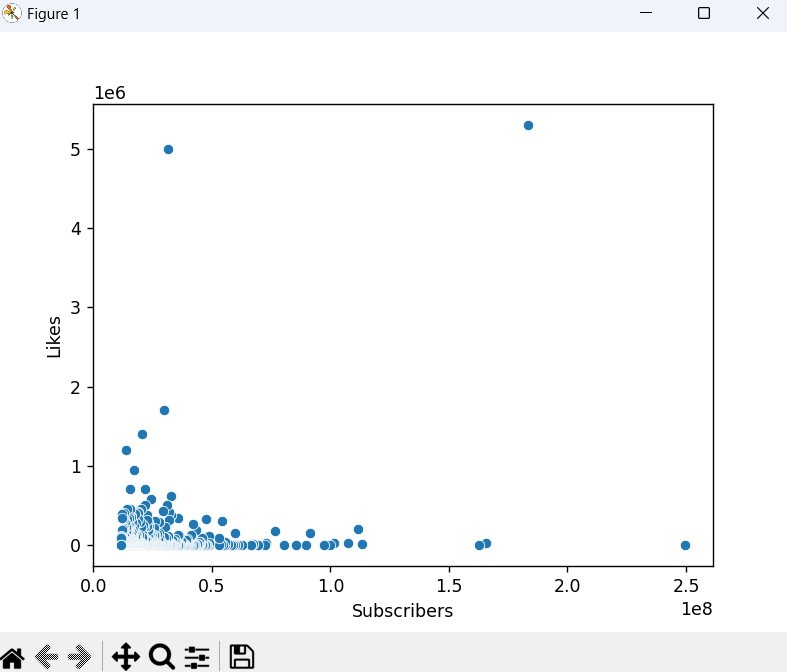
Fig.4 Trend Analysis Bar Graph

**Explore correlation between subscribers and likes/comments using scatter plots**

sns.scatterplot(x='Subscribers', y='Likes', data=youtube\_data)

plt.show()

**Output:**



Fig,5 Scatterplot of Corelation between subscribers and Comments

**Audience Study**

**Analyze audience distribution by country using bar plots or maps**

country\_counts = youtube\_data['Country'].value\_counts()

country\_counts.plot(kind='bar')

plt.xlabel('Country')

plt.ylabel('Count')

plt.title('Audience Distribution by Country')

plt.show()

**Output:**

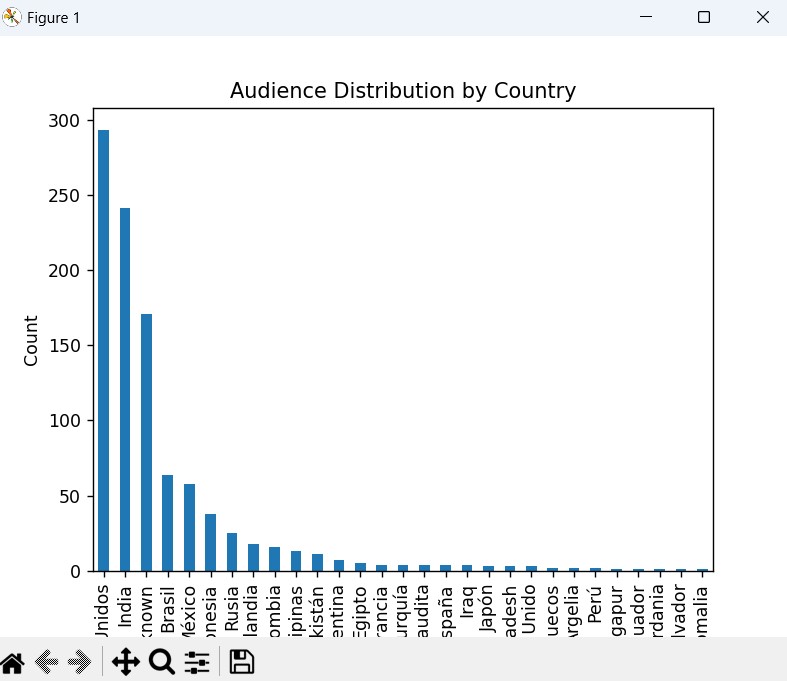


Fig.6 Audience Distribution by Country

**Performance Metrics**

**Calculate and visualize average metrics using summary statistics and boxplots**

avg\_metrics = youtube\_data[['Subscribers', 'Visits', 'Likes', 'Comments']].mean()

print(avg\_metrics)

sns.boxplot(data=youtube\_data[['Subscribers', 'Visits', 'Likes', 'Comments']])

plt.xticks(rotation=45)

plt.show()

**Output:**

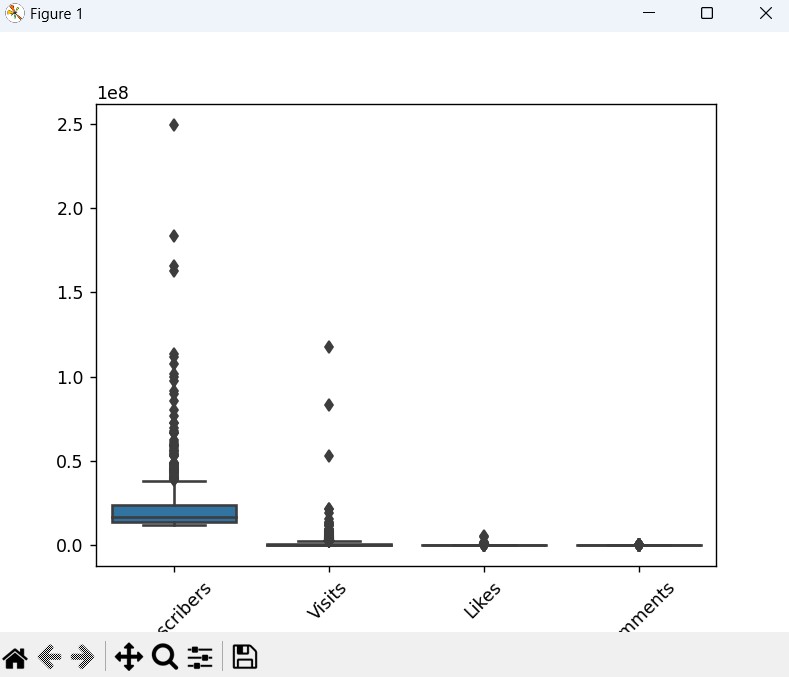


Fig.7 Average Metrics

**Content Categories**

**Explore distribution of content categories using bar plots or pie charts**

category\_counts = youtube\_data['Categories'].value\_counts()

category\_counts.plot(kind='bar')

plt.xlabel('Categories')

plt.ylabel('Count')

plt.title('Content Categories Distribution')

plt.show()

**Output:**

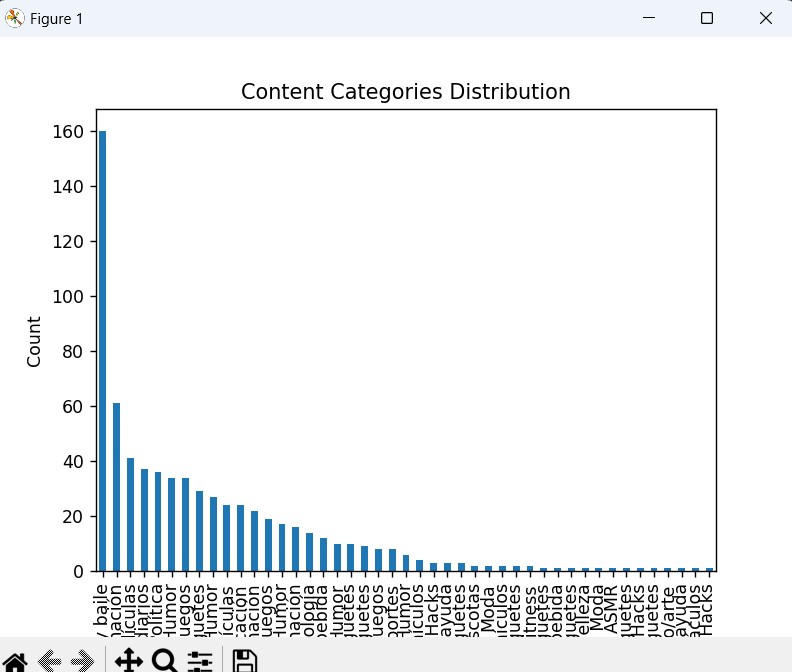


Fig.8 Content Categories Distribution

**Benchmarking**

**Identify top-performing content creators using ranking and comparison metrics**

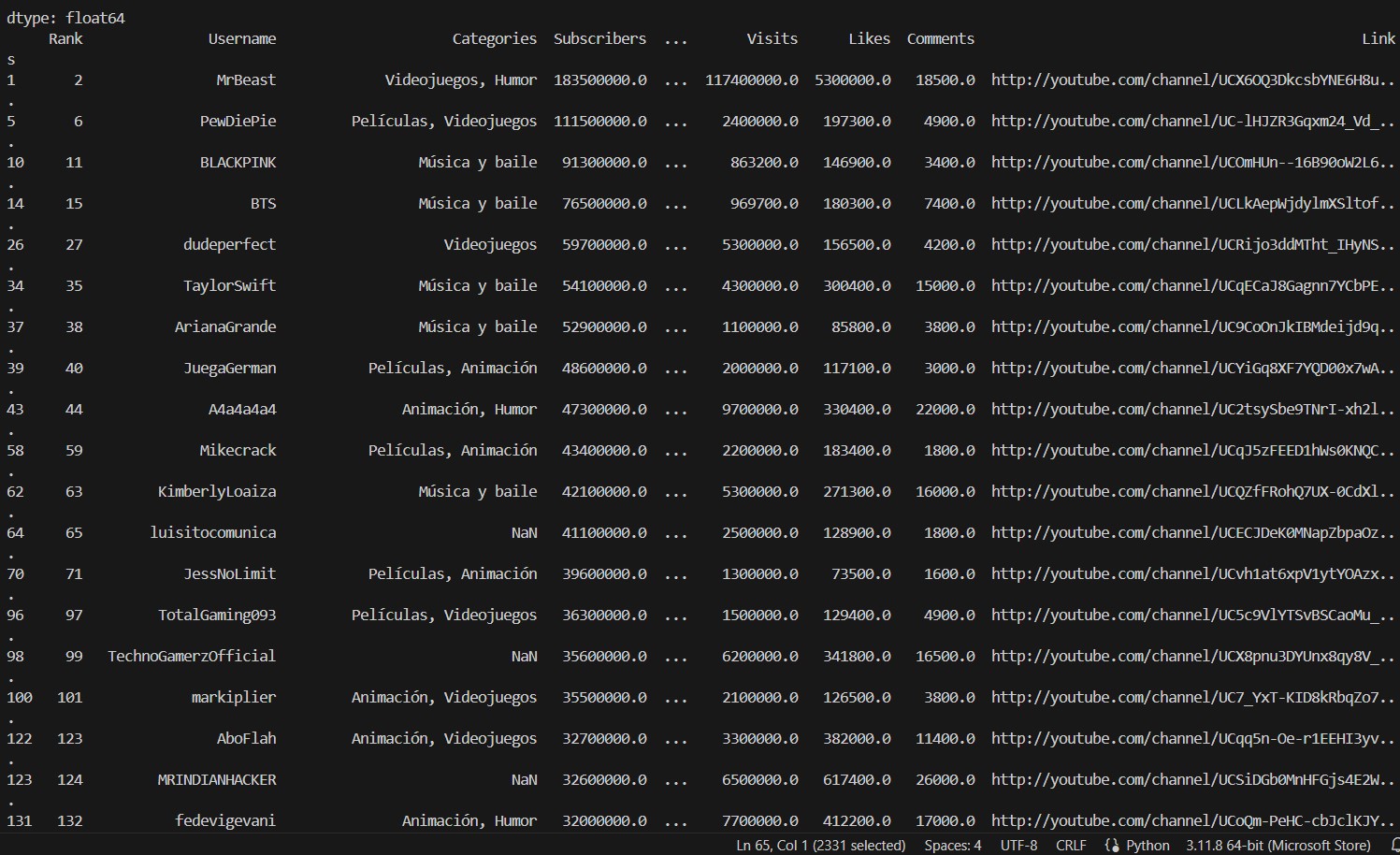
top\_creators = youtube\_data[(youtube\_data['Subscribers'] > youtube\_data['Subscribers'].mean()) &

                            (youtube\_data['Likes'] > youtube\_data['Likes'].mean()) &

                            (youtube\_data['Comments'] > youtube\_data['Comments'].mean())]

print(top\_creators)

**Output:**



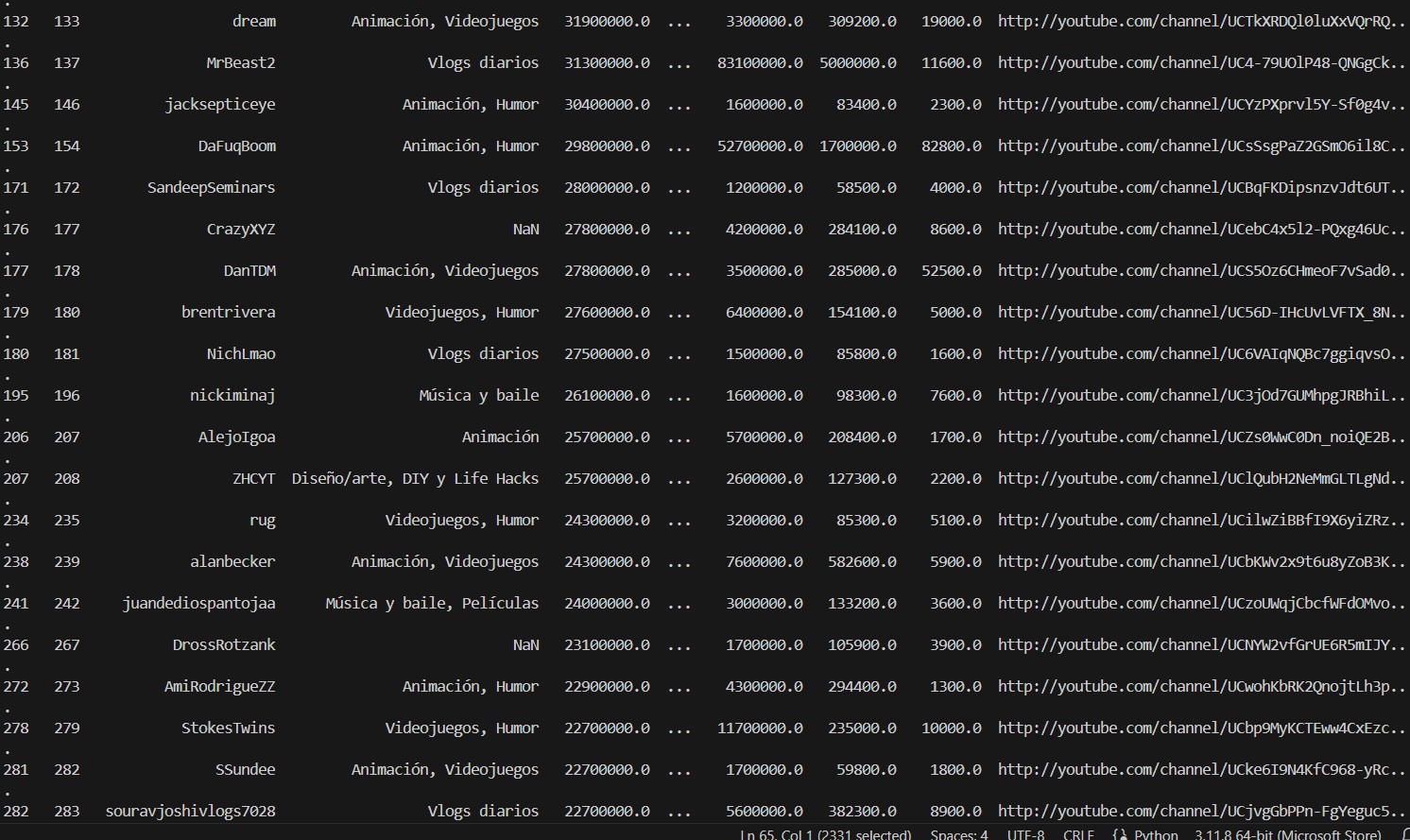


Fig.9 Top Performing Content Creators using Ranking and Comparison Metrics

**Propose a system for enhancing content recommendations to YouTube users:**

**Insights from Streamer Analysis:**

* **Popular Content Categories:** Identify the most popular content categories among top streamers (Task 5). This can inform YouTube about where user interest lies.
* **Category Performance:** Analyze which categories have streamers with exceptional performance metrics (subscribers, visits, likes, comments) (Task 5). This can highlight categories that might be under-recommended despite high potential for user engagement.
* **Regional Preferences (if data allows):** If your data includes streamer audience location data, analyze regional preferences for content categories (Task 3). This can be used to personalize recommendations based on user location.

**Content Recommendation System based on Streamers:**

Building on these insights, propose a system that leverages streamer data to improve content recommendations:

* **Streamer-based Filtering:** Incorporate streamer information into the recommendation algorithm. When a user watches a video from a particular category, recommend content from other streamers within the same category, especially those with high performance metrics (subscribers, engagement). This can help users discover new, successful creators within their preferred genre.
* **Content Diversification:** While promoting popular categories, ensure the system also recommends content from up-and-coming streamers or niche categories with high engagement metrics. This helps users discover diverse content beyond the most mainstream options.
* **Dynamic Recommendations:** Consider user watch history and preferences alongside streamer data. If a user frequently watches content from a particular streamer, even outside their usual category, recommend similar content from other streamers regardless of category. This personalizes recommendations based on user behavior.

**Additional Considerations:**

* **Content Quality:** While streamer performance metrics are valuable, prioritize recommending high-quality content. Integrate content quality assessment into the system alongside streamer data.
* **User Feedback:** Allow users to provide feedback on recommendations, including why they liked or disliked a suggestion. This feedback loop can help refine the system over time.

By incorporating streamer data and the proposed recommendations, YouTube can create a more dynamic and user-centric recommendation system that caters to diverse interests and helps users discover new, engaging content creators. Remember, this is a high-level proposal, and the specifics of the system would require further technical design and development.