## **Project 9: YouTube Channel Insights — Mining Video Trends from a Tech Influencer**

You’ve just been hired by **VidIntel**, a media analytics startup focused on helping creators and tech companies understand how their video content is performing and how audiences are responding over time.

Your first assignment is to analyze the video content and audience engagement of a **single YouTube channel** that covers emerging technology topics — like **AI**, **Blockchain**, and **Quantum Computing**. While traditional services like Social Blade give surface-level stats, your boss wants a custom analysis pipeline using direct data from YouTube’s own free **Data API v3**.

You’ll collect video metadata and comments from this channel’s public videos, analyze video publishing trends, audience sentiment, and video engagement patterns, and deliver insights that could help the channel owner improve content strategy.

By the way, it might be helpful to read a little about YouTube’s engagement metrics (views, likes, comments, CTR) and how video content strategies evolve over time.

For this project you will do the following:

1. Connect to YouTube Data API v3 and collect video and comment data for one channel
2. Clean and process the data into a usable format
3. Exploratory Data Analysis (EDA)
4. Basic sentiment analysis on comments
5. Data visualization and reporting

## **Part 1: Data Collection and Preparation**

### **Step 1: Connect to YouTube’s API**

Use the google-api-python-client library to connect to the YouTube Data API v3.  
 You’ll need to:

* Register a project in Google Cloud Console
* Enable the YouTube Data API v3
* Obtain an **API key**

**Example target:** A tech YouTube channel (e.g., *“ColdFusion”*, *“Two Minute Papers”*, *“Lex Fridman”*, or another emerging tech channel — you can substitute in the channel ID of your choice.)

**You will:**

* Retrieve the **Channel ID** (via API or manually)
* Fetch **all videos published in the past 2 years**
* For each video, collect:
  + videoId
  + title
  + publishedAt
  + viewCount
  + likeCount
  + commentCount
  + tags (if available)
  + description

Then, for each video:

* Fetch the **top 50 comments**
  + authorDisplayName
  + textDisplay
  + likeCount
  + publishedAt

### **Step 2: Tidy the Video Metadata**

Prepare your video dataset:

* Convert publishedAt to datetime objects
* Convert numeric fields (viewCount, likeCount, commentCount) to appropriate numeric types
* Clean text fields (title, description) to remove problematic characters and whitespace

### **Step 3: Tidy the Comment Data**

Similarly, process your comment dataset:

* Remove comments from “[deleted]” or if textDisplay is empty
* Convert publishedAt to datetime objects
* Clean and tokenize comment text (remove URLs, special characters)
* Use NLTK or spaCy to remove stopwords

## **Part 2: Analysis**

With your data ready, it's time to explore its distributions and relationships. You are free to take this in any direction you deem fit. Here are possible things you can try:

### **Question 1: Publishing Trend Analysis**

Plot the **number of videos published per month** over the past two years.  
 Are they posting more or fewer videos over time?  
 Create a line chart to show these trends.

### **Question 2: Video Popularity Distribution**

Create histograms of **view counts**, **like counts**, and **comment counts**.  
 Are there outlier viral videos? Is engagement evenly distributed or highly skewed?

Also, compute mean, median, and standard deviation for each metric.

### **Question 3: Sentiment Analysis of Comments**

Use TextBlob or VADER to compute sentiment polarity for each comment:

* Positive (polarity > 0.1)
* Neutral (between -0.1 and 0.1)
* Negative (polarity < -0.1)

Create:

* A **bar chart** showing the proportion of positive, neutral, and negative comments overall
* A **time series chart** of average comment sentiment per month

### **Question 4: Word Frequency and Tag Analysis**

Use your cleaned comment text and video tags:

* Create a **word cloud** of the most frequent words in video comments
* Identify and visualize the most common tags associated with the channel’s most successful videos (by views or engagement)

## **Deliverables:**

* Cleaned datasets (CSV files)
* Jupyter notebook with code and visualizations
* Final summary/report of insights and recommendations (Markdown or PDF)

## **Ethics Note:**

* Use YouTube’s API responsibly and respect rate limits
* Only collect publicly available video metadata and comments
* Avoid collecting personal identifying information beyond publicly displayed usernames