



# YouTube Advertisement Putting

Yuqing Wu

# Content



**01** Introduction

**02** EDA

**03** Model

**04** Result

**05** Q&A

# 01 Introduction: Business Objective

## Objective:


Analyze YouTube video clicks to determine which videos to advertise for, thereby increasing potential users and sales, and increasing revenue



# 01 Introduction: Two Main Questions

**Q1:** When should we put the advertisement?

**Q2:** What kind of video(s) should we put the advertise?



# 02 EDA: About Data

## About Data:

- This csv dataset includes data for different types of YouTube videos in the United States region
- The json file includes category title corresponding to category ID

## Data Types:

csv file: 16 columns of data in total

## 02 EDA: *About Data*

**Video ID**

**Trending Date**

**Title**

**Channel Title**

**Category ID**

**Publish Time**

**Tags**

**Views**

**Likes**

**Dislikes**

**Comment Count**

**Thumbnail Link**

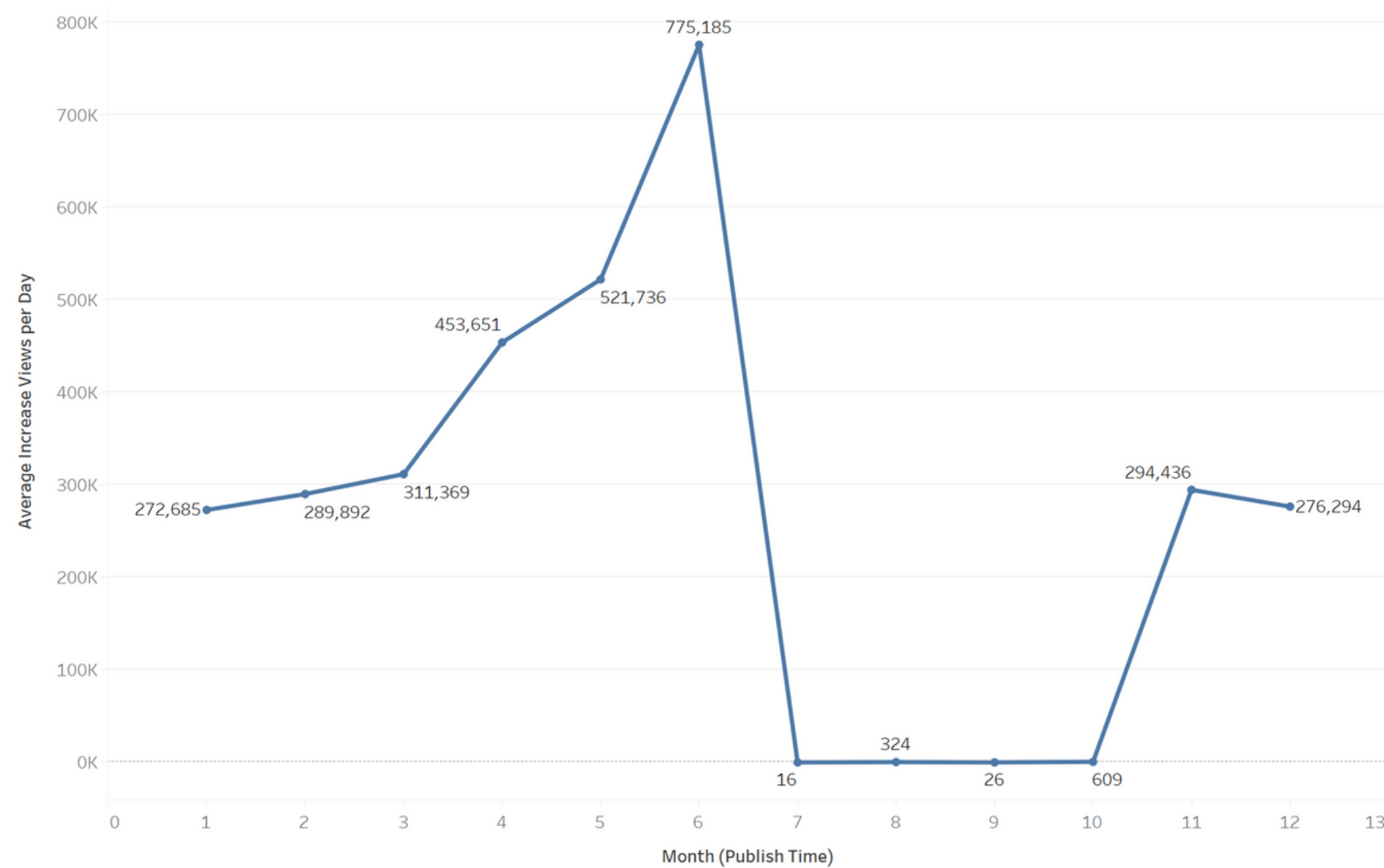
**Comments Disabled**

**Rating Disabled**

**Video Error or  
Removed**

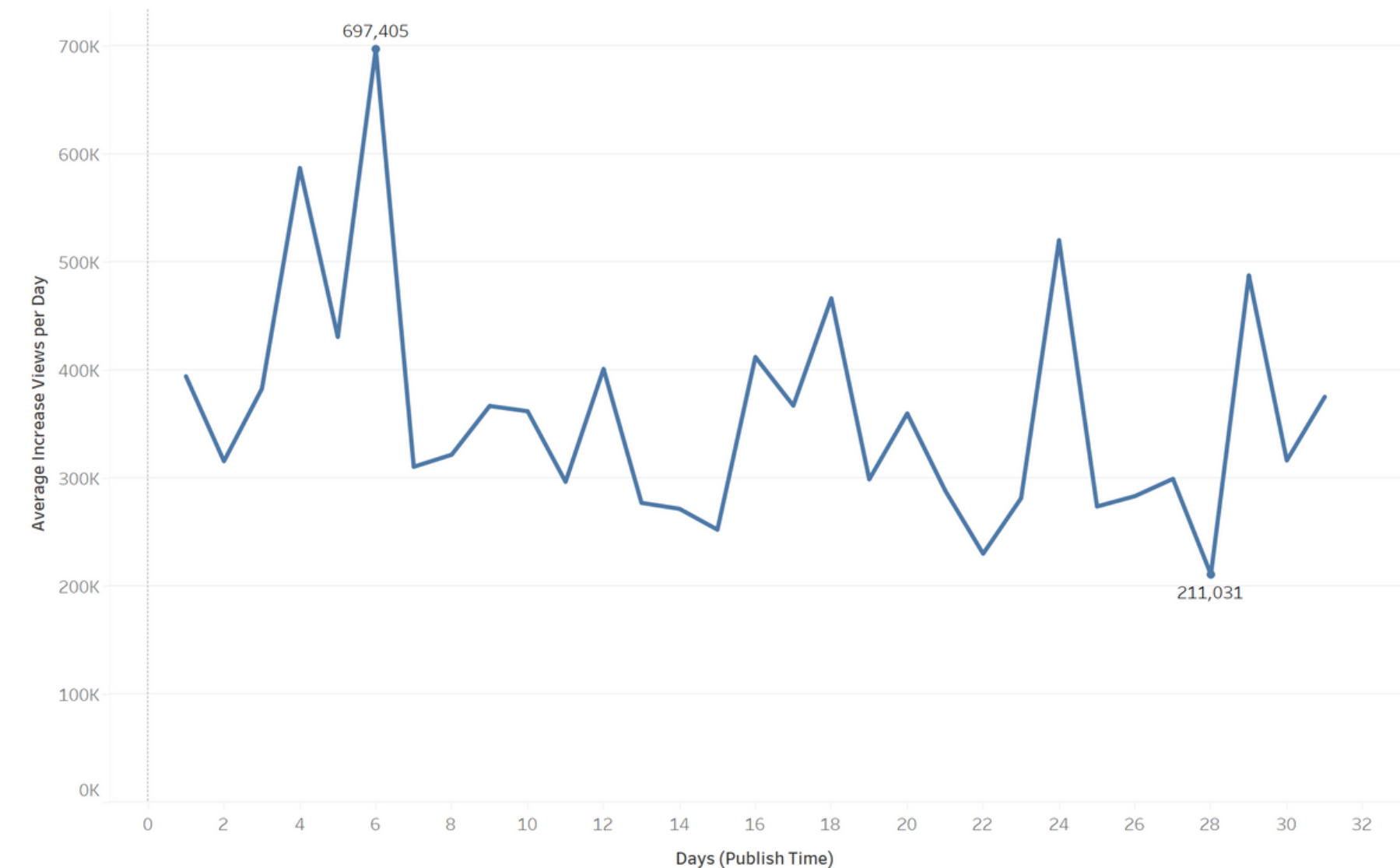
**Description**

# 02 EDA: Line chart on average daily & monthly growth views



The trend of average of Increase/Days for Publish Time Month.

Monthly

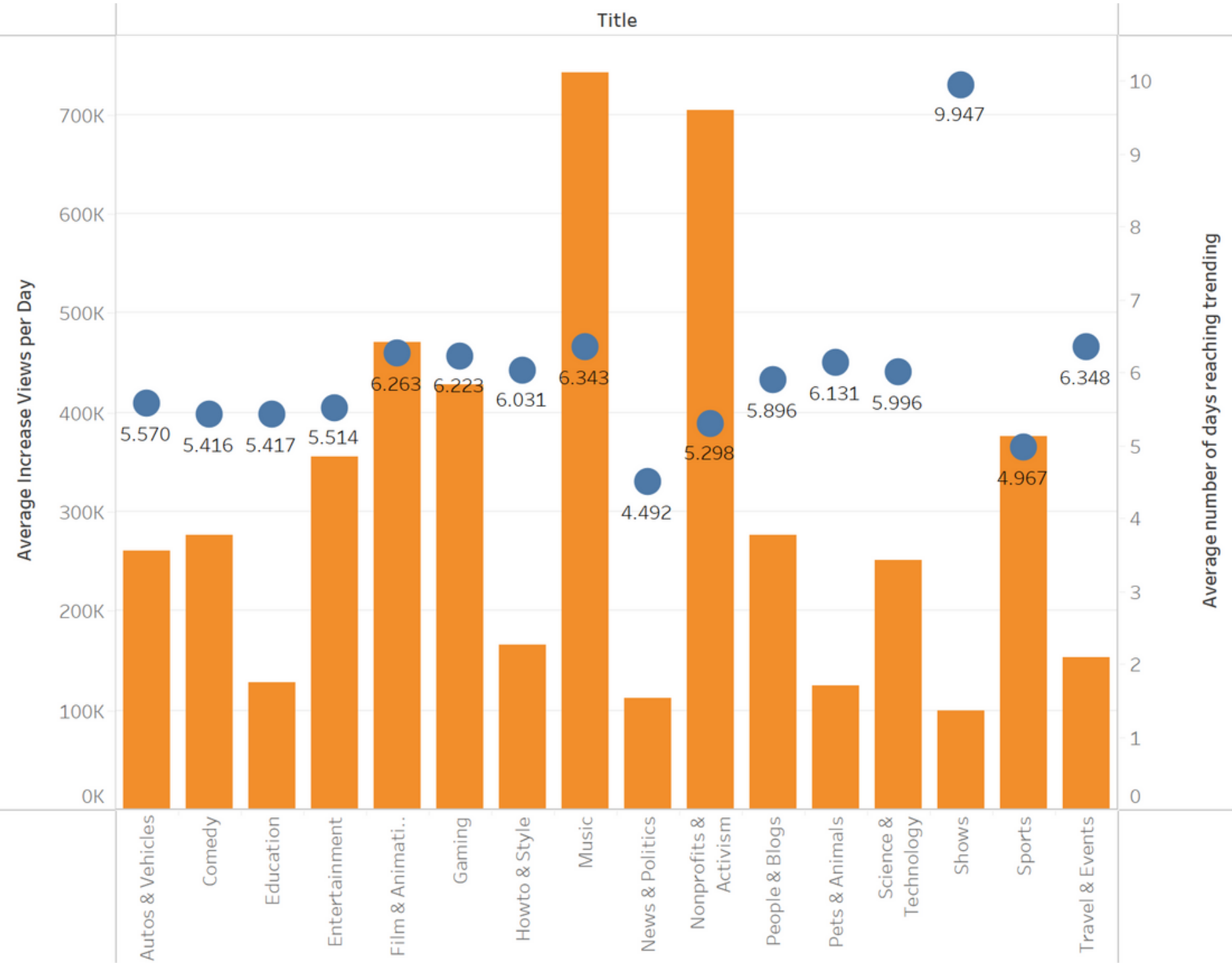


The trend of average of Increase/Days for Publish Time Day.

Daily

Q1

# 02 EDA: Bar charts and tables for video classification



Avg. Increase/Days and Median Increase Day Diff for each Title. Color shows details about Avg. Increase/Days and Median Increase Day Diff.

	category_id	views	title
0	1	7284156721	Film & Animation
1	2	520690717	Autos & Vehicles
2	10	40132892190	Music
3	15	764651989	Pets & Animals
4	17	4404456673	Sports
5	19	343557084	Travel & Events
6	20	2141218625	Gaming
7	22	4917191726	People & Blogs
8	23	5117426208	Comedy
9	24	20604388195	Entertainment
10	25	1473765704	News & Politics
11	26	4078545064	Howto & Style
12	27	1180629990	Education
13	28	3487756816	Science & Technology
14	29	168941392	Nonprofits & Activism
15	43	51501058	Shows



## **02 EDA:** Strategy – focus on product relevance

### **Travel & Events need to be included because:**

- Related to our product -> Travel App
- Total views not too high -> Cost may not high
- Not too low average daily traffic growth
- Fast trending speed of videos

**In addition, we can also advertise other video categories.....**

## 02 EDA: Strategy – focus on short-term

### A Music + Nonprofit & Activism

#### Pros:

- Average daily viewing volume of increased significantly
- Medium number of days a video reaches trending is relatively low

#### Cons:

- The total viewing of music videos is the highest, the cost may be higher
- The total viewing of nonprofit videos is relatively low, may have an impact on the advertising effect

## 02 EDA: Strategy – focus on long-term

**B** Gaming + Entertainment /  
Film & Animation+ Sport

### Pros:

- Not low total views, but not too high, so costs can be controlled
- The average daily traffic growth and medium trending duration are both in the upper middle range

### Cons:

- People who watching gaming and entertainment may not necessarily have a strong interest in tourism

# 03 Data Preprocessing: Check Fraud Data

## What is fraud data?

Too few likes, dislikes, and comments in the same level of views

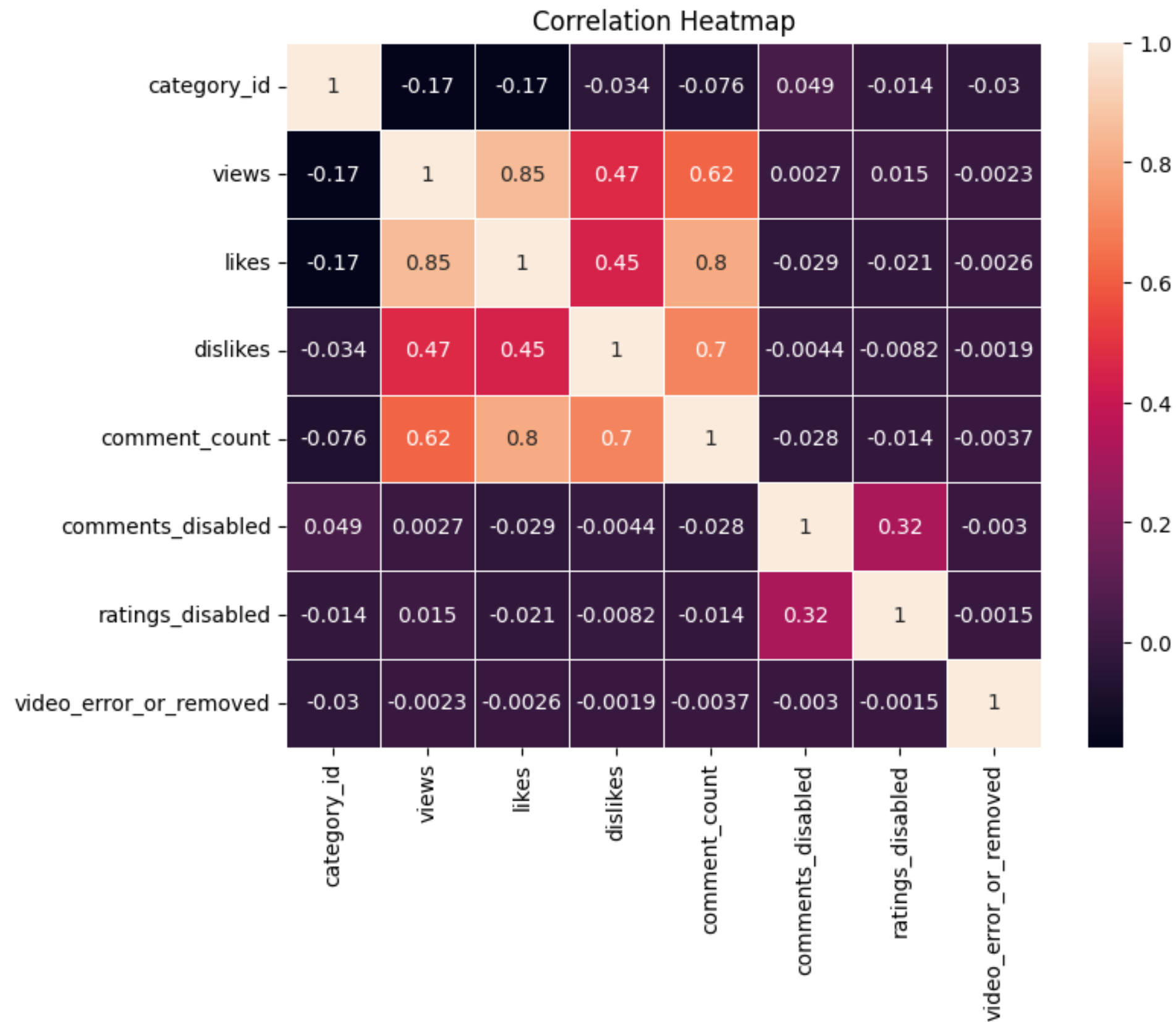
## Why need to check fraud data?

- Malicious browsing can affect our judgment on video placement choices
- Disrupting the training results of models

## How we check fraud data?

Discovering the relationship between views and the number of likes, dislikes, and comments through the model, and setting threshold values to achieve filtering

# 03 Data Preprocessing: Correlation Map



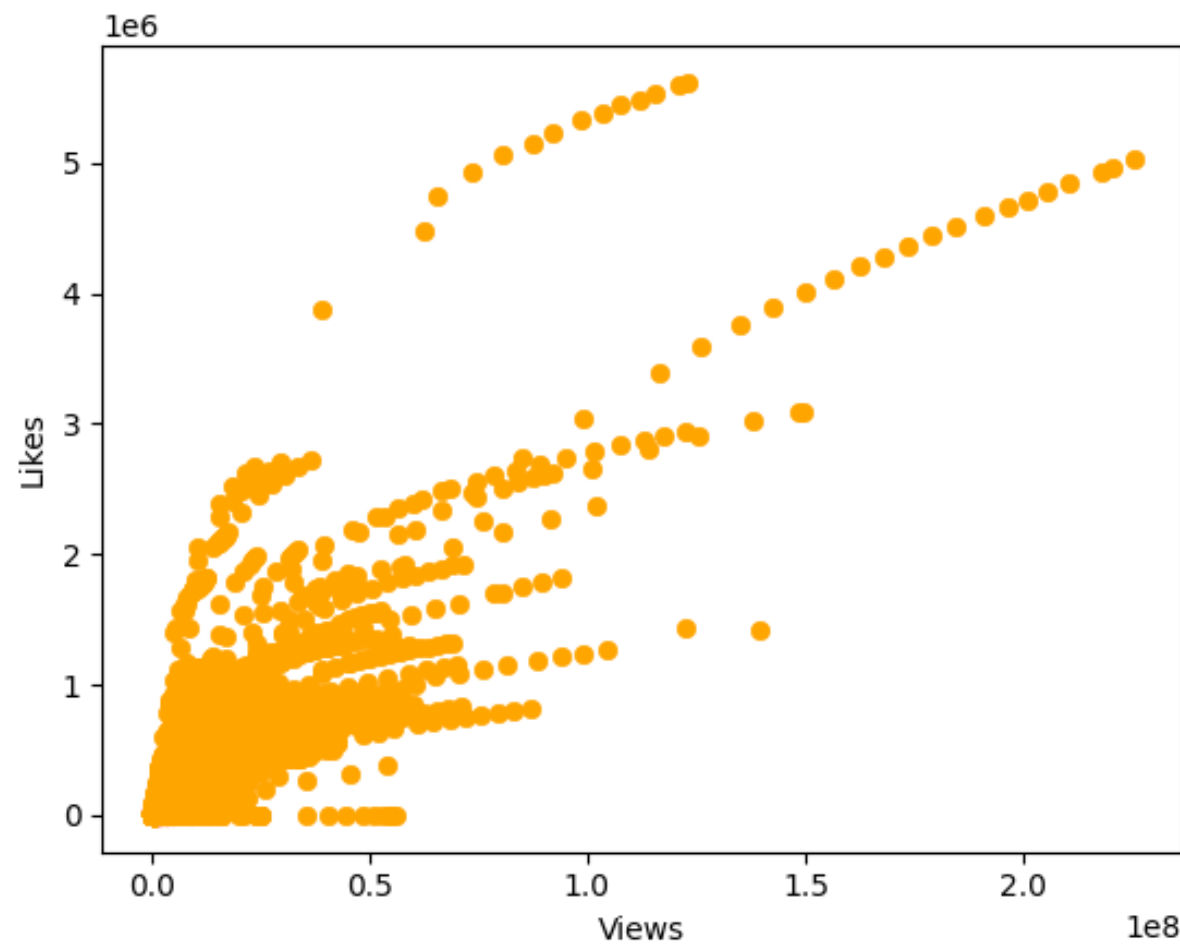
Views vs Likes: **0.85**

Views vs Dislikes: **0.47**

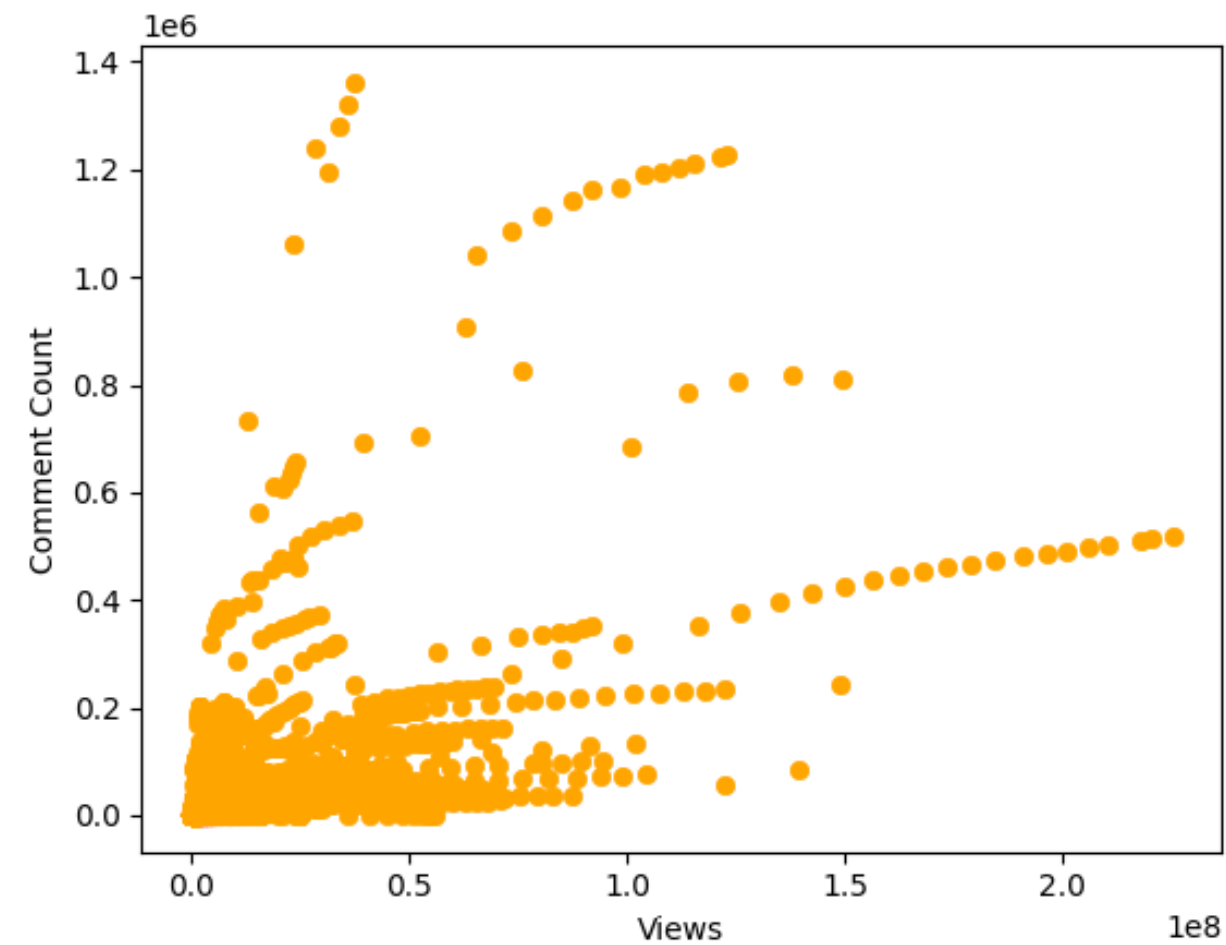
Views vs Comment\_count: **0.62**

-> Positively and nearly high related

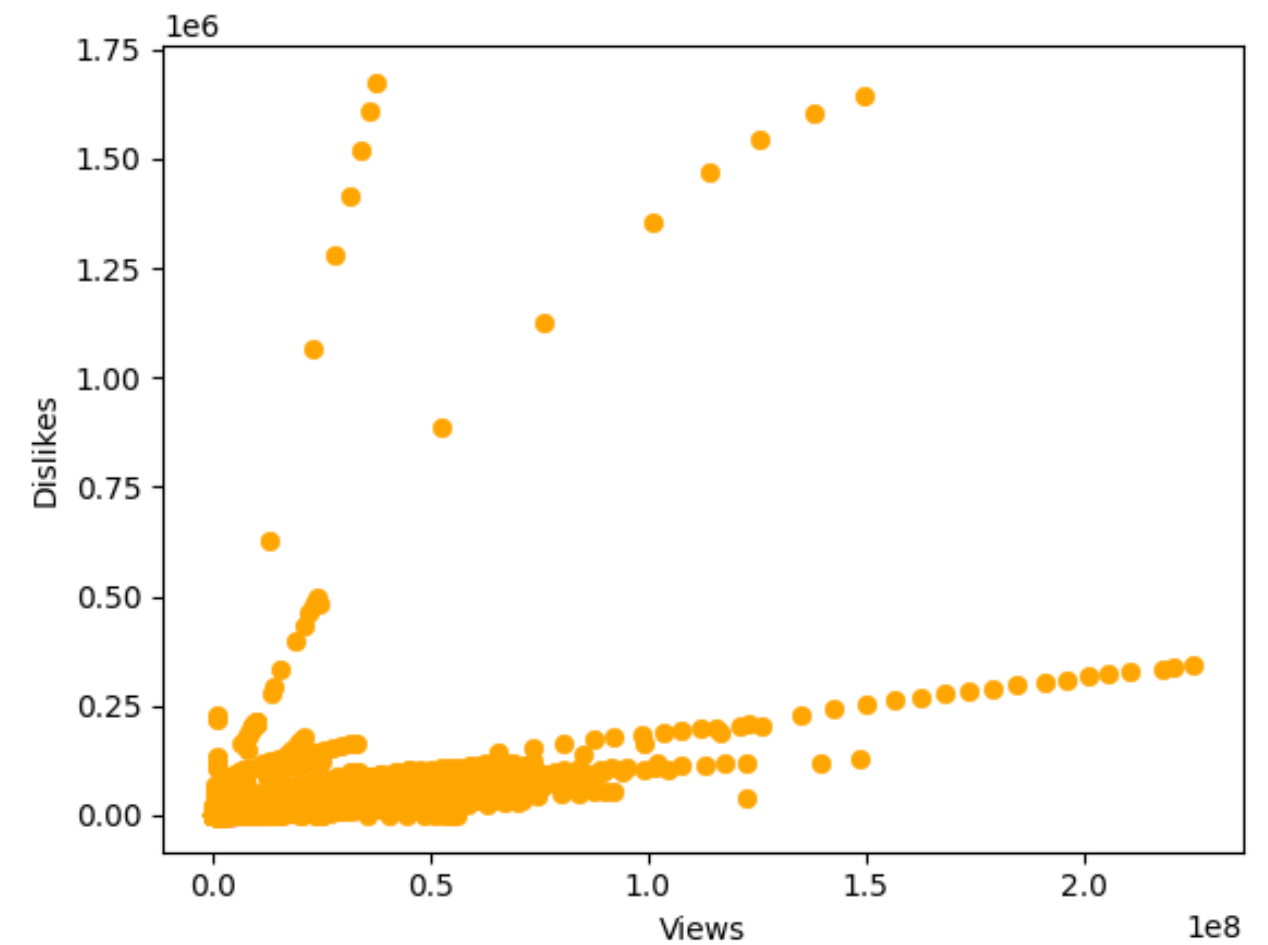
# 03 Data Preprocessing: Correlation Map



Views vs Likes



Views vs Comment\_count



Views vs Dislikes

**Linear Relationship**

# 03 Data Preprocessing: Check Fraud Data

## Two Model Select

Linear Regression

Decision Tree

### Process:

- Train models separately based on video categories
- Calculate mse and compare sizes
- Among the 16 categories, **93.75%** have smaller mse under decision tree model

**-> Select Decision Tree to do the fraud check**

# 03 Data Preprocessing: Choose Threshold

Threshold	Rows Detected
2	524 rows
3	251 Rows
4	145 Rows
5	102 Rows



# 03 Data Preprocessing: Choose Threshold = 2

## Threshold = 2: 524 Rows

Being too precise resulted in some data not belonging to fraud being excluded as well

Example: category\_id = 1

Row id	Views	Likes	Dislikes	Comment	Detected
4345	88657	593	27	52	Fraud
4129	88257	590	26	54	Real
4572	89002	595	26	52	Real
4799	89311	599	26	52	Real

# 03 Data Preprocessing: Choose Threshold = 3

**Threshold = 3: 251 Rows**

Being too precise resulted in some data not belonging to fraud being excluded as well

Example: category\_id = 2

Row id	Views	Likes	Dislikes	Comment	Detected
27795	1783926	1852	172	245	Fraud
27582	1713302	1165	85	163	Real
28005	1808781	2122	234	273	Real

# 03 Data Preprocessing: Choose Threshold = 4

**Threshold = 4: 145 Rows**

Detected the fraud data well

Example: category\_id = 10

Row id	Views	Likes	Dislikes	Comment	Detected
5331	190152	926	40	20	Fraud
99	195685	14338	171	1070	Real
126	205869	11198	120	446	Real
347	204298	9321	386	993	Real

# 03 Data Preprocessing: Choose Threshold = 5

**Threshold = 5: 102 Rows**

Detected the fraud data too narrow which missed some fraud data

Example: category\_id = 10

Row id	Views	Likes	Dislikes	Comment	Detected
5331	190152	926	40	20	Real
14073	112310	612	7	95	Real
109	125962	5048	139	369	Real
222	122426	6310	298	1624	Real
39975	116841	6767	75	403	Real

# 03 Data Preprocessing: Remove Fraud Data

## Threshold = 4:

Views corresponding to likes, dislikes, and comments of the same level are normal between two to three times, so select 4 as the threshold for filtering

## Process:

- Use the corresponding model for preview for each video category
- Compare the preview results with the original views and delete those that are less than four times the size

-> Remove 145 rows of the fraud data

# 03 Data Preprocessing: Remove Fraud Data

## Fraud data with 0 likes/dislikes/comment:

- Remove data has views but no likes/dislikes/comments without disabled comment/rating

-> Remove 145 rows of the fraud data

-> Remove 106 rows of the data has views but no likes/dislikes/comment

# 03 Data Preprocessing: Predict Future Views Trends

## Linear Regression:

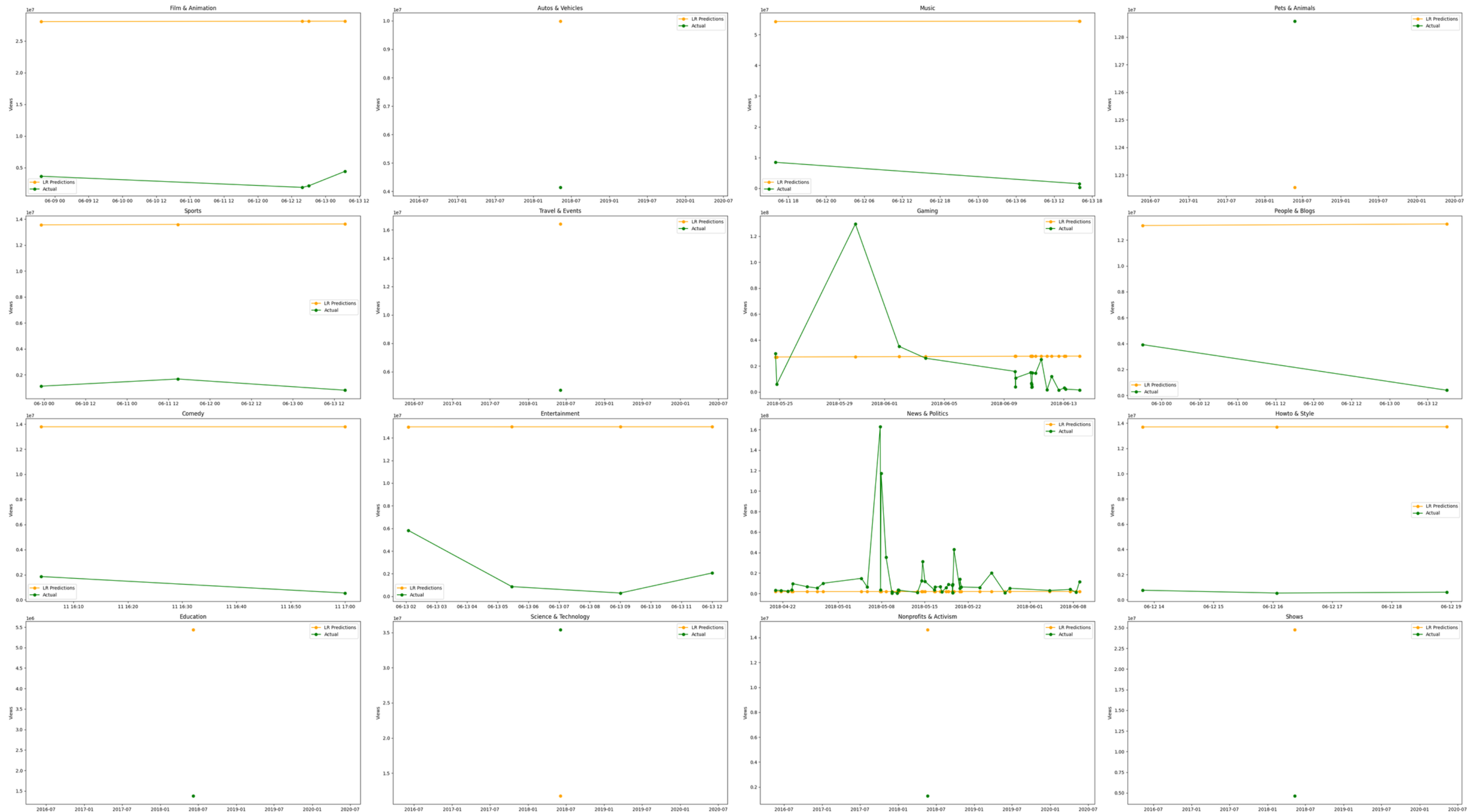
Train linear regression with views based on the difference in days between trending date and publishing date

## ARIMA:

Train ARIMA models with trending date and publishing date and views respectively

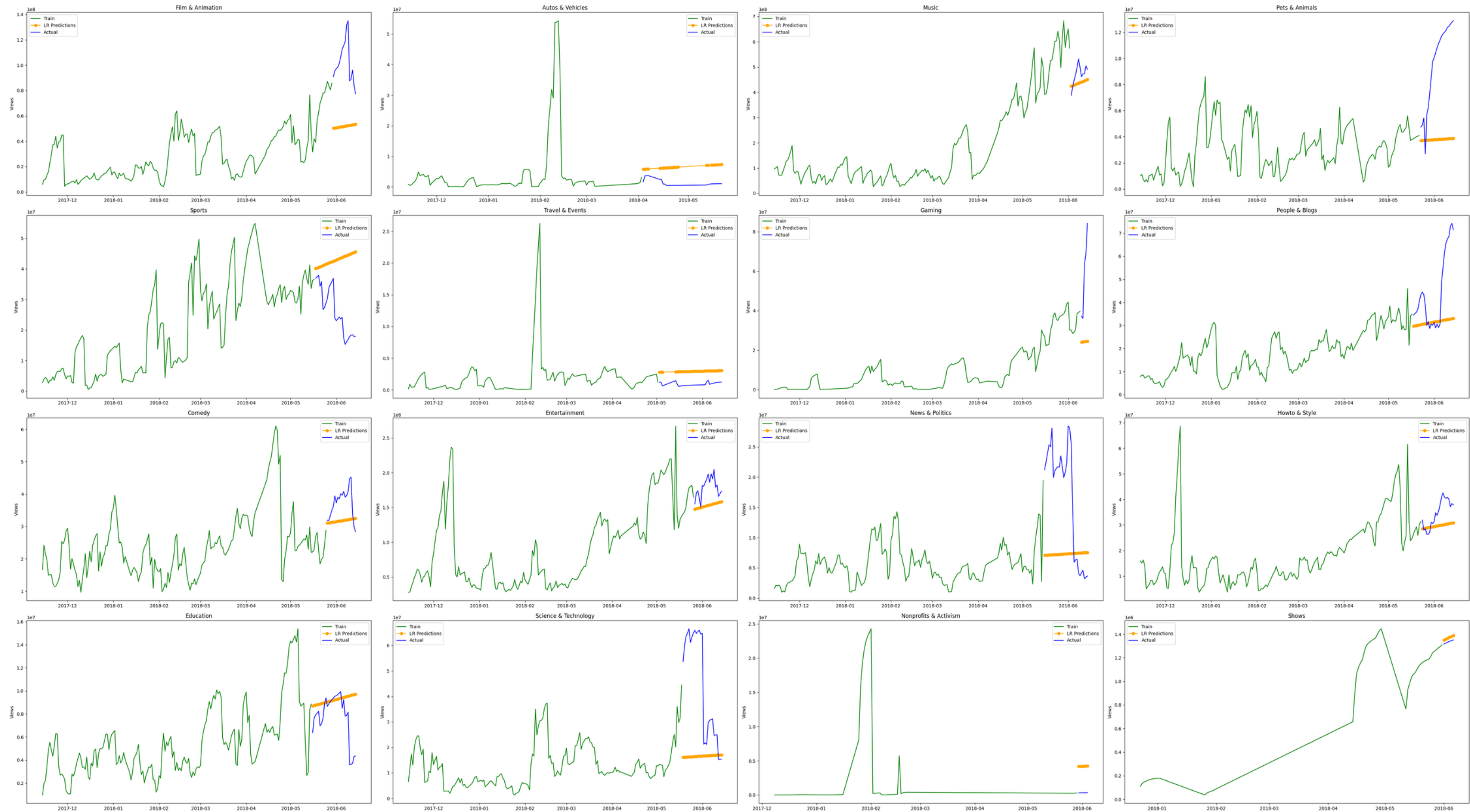
**Both models predict views trends for the last 10% of the dataset**

# 03 Data Preprocessing: Linear Regression – Publish Date

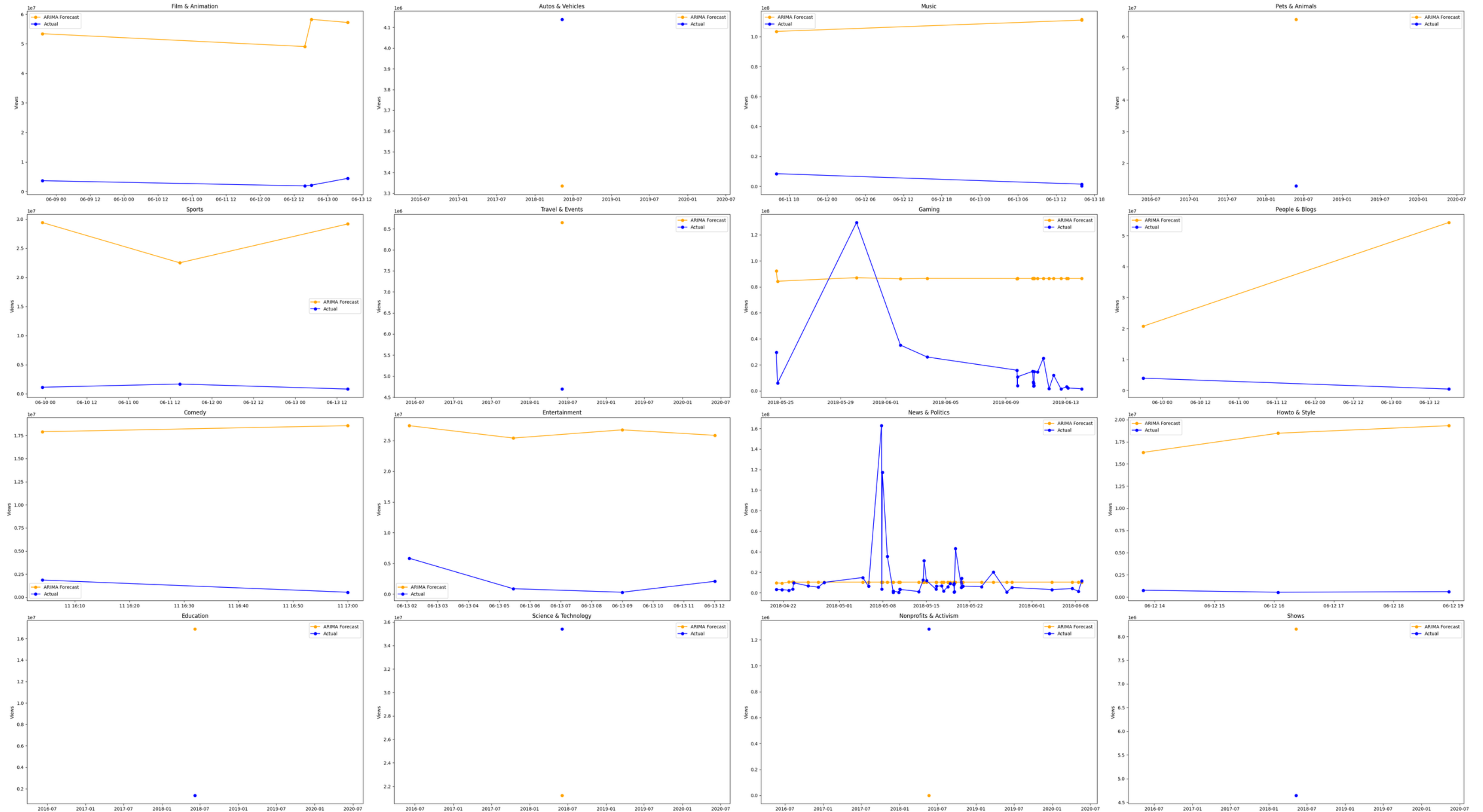




# 03 Data Preprocessing: Linear Regression – Trend Date



# 03 Data Preprocessing: ARIMA – Publish Date



# 03 Data Preprocessing: ARIMA – Trending Date



# 03 Data Preprocessing: Compare Model

**Trending date vs Views:**

Model	Same Slope Trend
Linear Regression	$10/16 = 0.625$
ARIMA	$8/16 = 0.5$

**Publish date vs Views:**

# 04 Classification: Word2Vec

## Title, Channel\_title, Tags, Description:

- Remove all special characters (e.g. &, !, -), keep only English letters and transfer into lower case
- Use nltk library to remove English stopwords
- Train the Word2Vec model by using the cleaned text data - use CBOW
- Get the word vector by using the trained Word2Vec model

-> Select Logistic Regression, SVM, Random Forest, to do the classification

# 04 Classification: Logistic Regression

	precision	recall	f1-score	support
1	0.78	0.66	0.72	486
2	0.59	0.56	0.57	77
10	0.92	0.92	0.92	1262
15	0.83	0.91	0.87	189
17	0.87	0.87	0.87	429
19	0.79	0.74	0.77	66
20	0.85	0.87	0.86	149
22	0.59	0.49	0.54	669
23	0.82	0.80	0.81	674
24	0.75	0.79	0.77	1954
25	0.83	0.87	0.85	479
26	0.81	0.84	0.82	841
27	0.78	0.80	0.79	330
28	0.77	0.77	0.77	513
29	0.45	0.50	0.48	10
43	0.89	0.62	0.73	13
accuracy			0.80	8141
macro avg	0.77	0.75	0.76	8141
weighted avg	0.79	0.80	0.79	8141

# 04 Classification: SVM

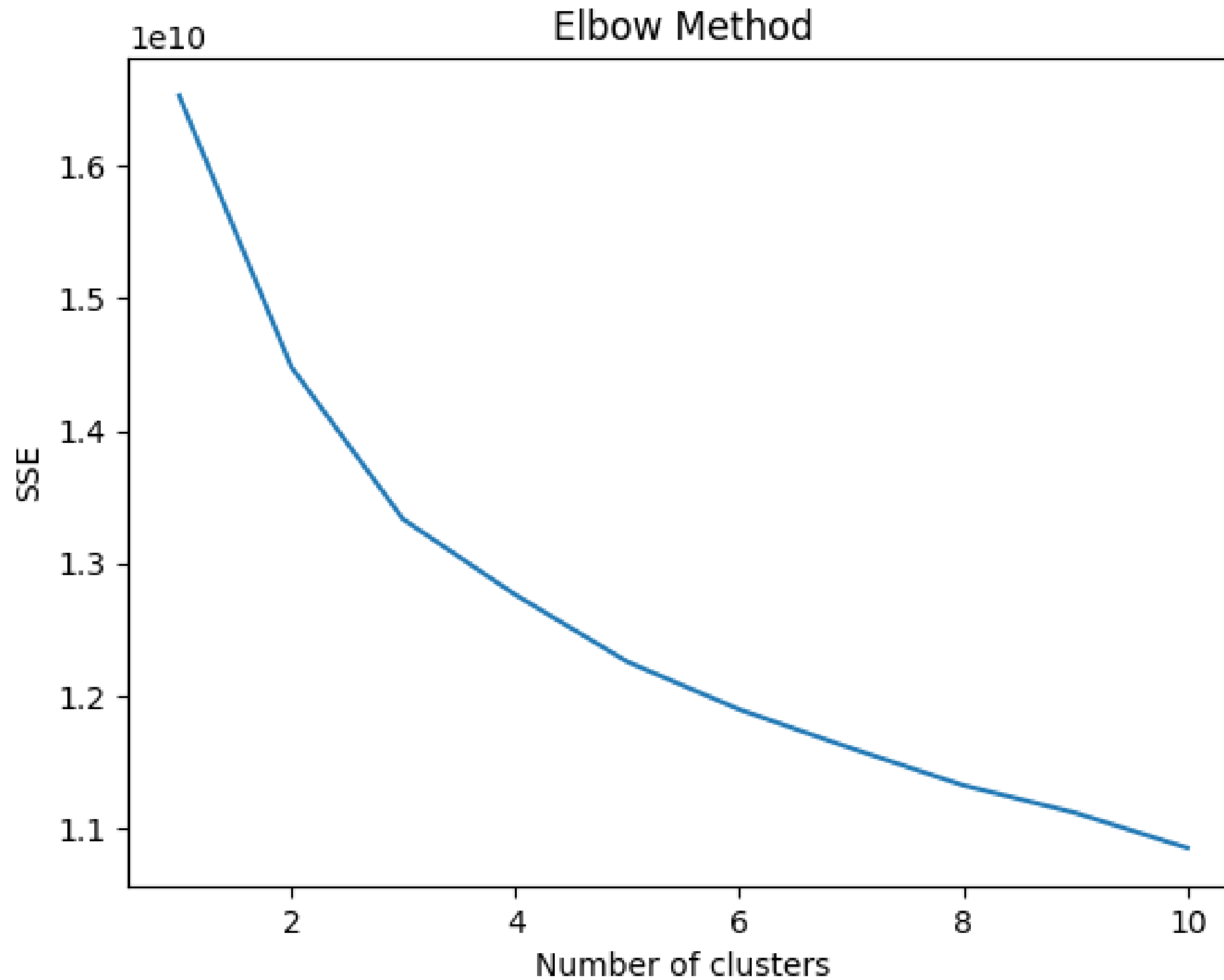
	precision	recall	f1-score	support
1	0.99	0.76	0.86	486
2	0.83	0.49	0.62	77
10	0.94	0.95	0.94	1262
15	0.91	0.89	0.90	189
17	0.94	0.92	0.93	429
19	0.98	0.67	0.79	66
20	0.97	0.77	0.86	149
22	0.60	0.75	0.67	669
23	0.96	0.80	0.87	674
24	0.82	0.91	0.86	1954
25	0.91	0.87	0.89	479
26	0.91	0.91	0.91	841
27	0.95	0.82	0.88	330
28	0.84	0.85	0.84	513
29	0.00	0.00	0.00	10
43	1.00	0.69	0.82	13
accuracy			0.87	8141
macro avg	0.85	0.75	0.79	8141
weighted avg	0.88	0.87	0.87	8141

# 04 Classification: Random Forest

	precision	recall	f1-score	support
1	1.00	0.72	0.84	486
2	1.00	0.16	0.27	77
10	0.98	0.97	0.98	1262
15	0.99	0.74	0.85	189
17	0.99	0.93	0.96	429
19	1.00	0.47	0.64	66
20	1.00	0.62	0.76	149
22	0.89	0.81	0.85	669
23	1.00	0.89	0.94	674
24	0.75	0.98	0.85	1954
25	0.95	0.90	0.92	479
26	0.96	0.95	0.96	841
27	1.00	0.74	0.85	330
28	0.86	0.91	0.89	513
29	0.00	0.00	0.00	10
43	1.00	0.54	0.70	13
accuracy			0.89	8141
macro avg	0.90	0.71	0.77	8141
weighted avg	0.91	0.89	0.89	8141



# 05 Clustering: Kmeans



**K = 3 should be chosen**

# 05 Clustering: Kmeans

# 06 LangChain: