Name: Xiyuan Liu, Xin Li
Course: DATA/MSML 602
Submission date: Dec 12th, 2022
Professor: Mohammad T. Hajiaghayi and Arefeh A. Nasri

Project final report

1 Research question:

Nowadays, As the total users of YouTube broke 2562 million, the influence of YouTube becomes more powerful than ever before. In order to understand why the number of YouTube users has such a growth, more and more researchers and YouTube content creators would wonder how much and how a YouTube video will influence the public. Among all videos, data from YouTube trending videos should be focused, since YouTube trending videos not only have a stronger influence on the public, but also reflect real time public interests. The prediction of the number of views in trending YouTube videos will be helpful to this research. In this project, our research team decided to develop statistical analysis of YouTube trending videos and complete a deep learning model in order to predict how many views will be based on current trending data.

2 Dataset:

2.1 Data set introduction:

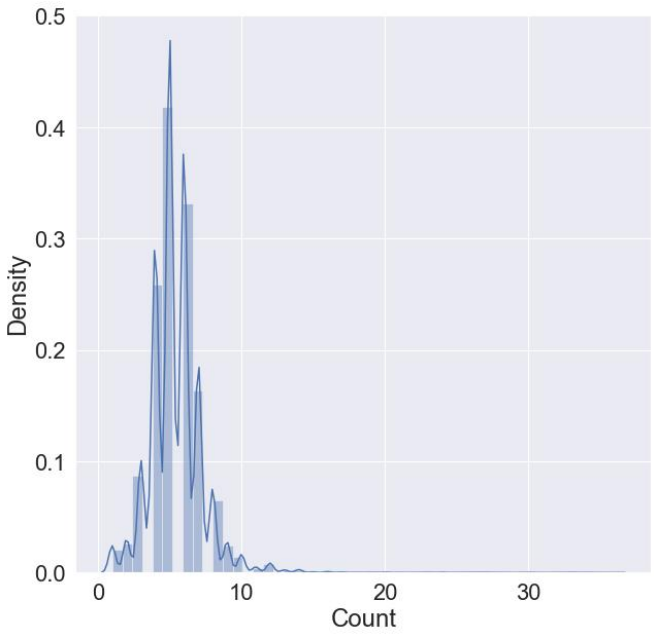
The dataset used in the project is from Kaggle, and all data came from YouTube Data API directly. This dataset includes 2 years (starting on Aug. 2020) of data on daily trending YouTube videos. The whole dataset includes India, the USA, Great Britain, Canada, France, Russia, Brazil, Mexico, South Korea, and Japan with up to 200 listed trending videos per day. Each region's data is in a separate file. Data includes the video title, channel title, publish time, tags, views, likes, dislikes, description, and comment count. And there is a JSON file to store the category name for each video. For simplicity, this project will only focus on US data.

The US trending data file contains different types of data, such as date data, numerical data, and text data. Among these data, text data may contain sentences and URL links, and it all appears on the same line. We first separate these text data by comma, slash, and quotation marks, then replace these symbols with a space to make the data more readable in the data frame. Since we are dealing with the US data, adding a country column seems a reasonable

	video_id	count	unique	top	freq
0	--14w5SOEUs	10	5	2021-06-11T00:00:00Z	2
1	--2O86Z0hsM	4	4	2022-03-11T00:00:00Z	1
2	--40TEbZ9Is	5	5	2021-09-21T00:00:00Z	1
3	--DKkzWVh-E	4	4	2021-12-08T00:00:00Z	1
4	--FmExEAsM8	4	4	2021-12-02T00:00:00Z	1
...	...	...	...	...	...
31187	zzCrFWjKPy8	5	5	2022-08-30T00:00:00Z	1
31188	zzd4ydafGR0	10	10	2021-02-12T00:00:00Z	1
31189	zziBybeSAtw	2	2	2021-01-17T00:00:00Z	1
31190	zzk09ESX7e0	14	7	2021-06-03T00:00:00Z	2
31191	zzsIqPVv2Q4	5	5	2022-05-01T00:00:00Z	1

step to do. After the process of data cleaning, there are 170,190 rows of data in total.

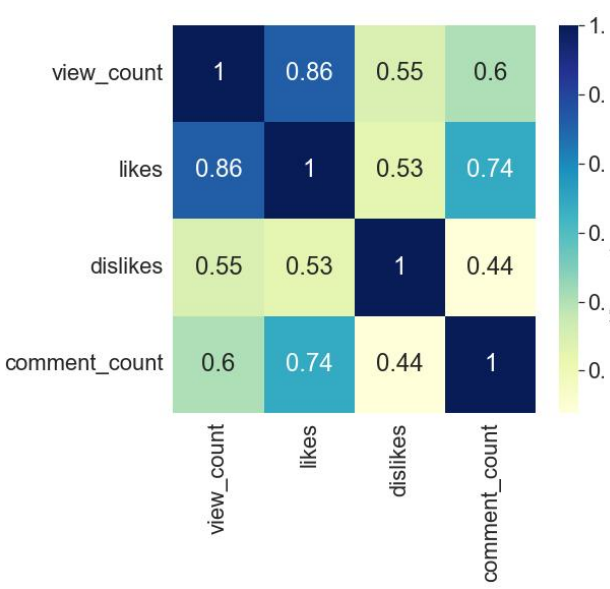
Each video on YouTube has a unique video id, Due to the fact that this dataset is updated daily through YouTube API, popular videos may repeatedly appear in this dataset. Since the data file contains other information such as channel title, video title, and descriptions, data file has been simplified to the unique video id and date to count how many times each video had appeared. The unique count represents how many days each video had been trending. Now we have the total trending days count data for every video, we calculate the mean of days count is 5.42, and the median is 5. These two numbers provide an important basis to filter the data.

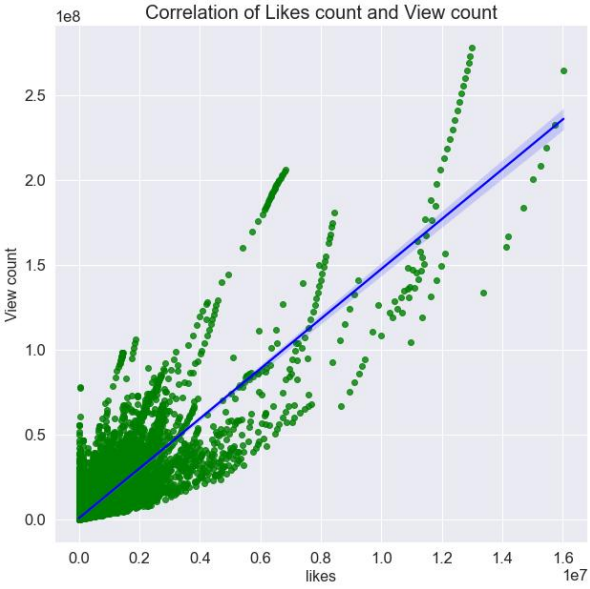


2.2 Data correlation analysis:

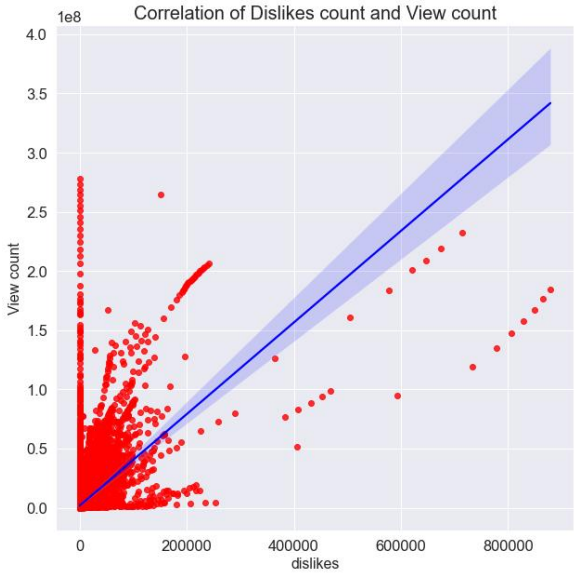
Compared to data of trending day counts of each video, view counts likes counts dislike counts and comment counts for each video become the most valuable dataset in the whole data. Therefore, analysis of the correlation of each of these data of all videos will be needed.

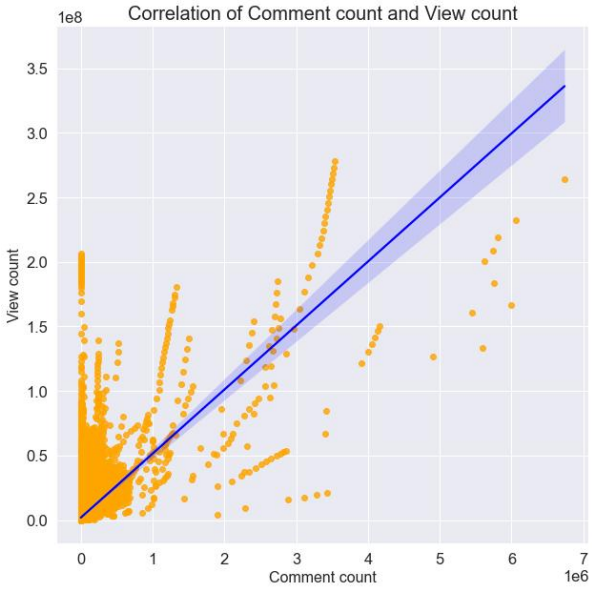
From the below heatmap, likes, dislikes, and comment count all have a positive correlation with respect to view count. Through plotting a regression chart for these three data with respect to the view count, a clear trend shows up that as likes, dislikes, and comment count increase, the view count increases as well. This trend appears strongest in the combination of likes and view count, this is also





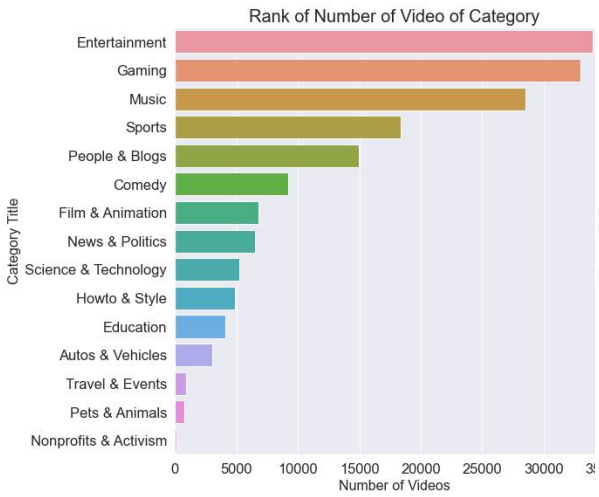
shown in the heatmap which has a correlation score of 0.86.

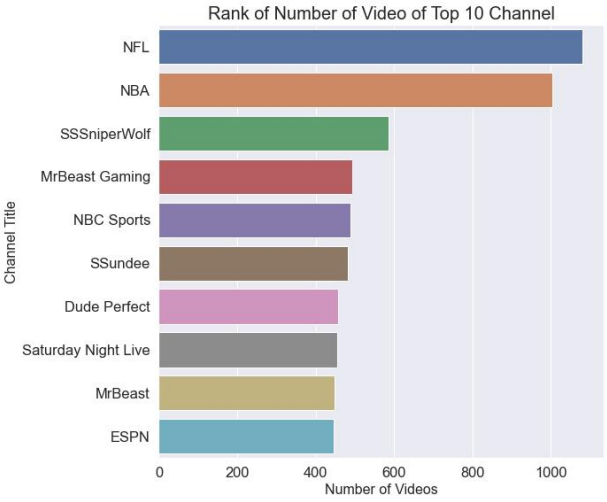




2.3 Data Category analysis:

After finding the correlation of numerical data, our project team also visualizes how category and time will demonstrate the view counts. Since the US trending video file only contains category id instead of category title, the category title has been extracted from the US JSON file as a data frame and merged with the US data frame together. Now we will be able to graph which channel, category, days and hours have the most videos and views.





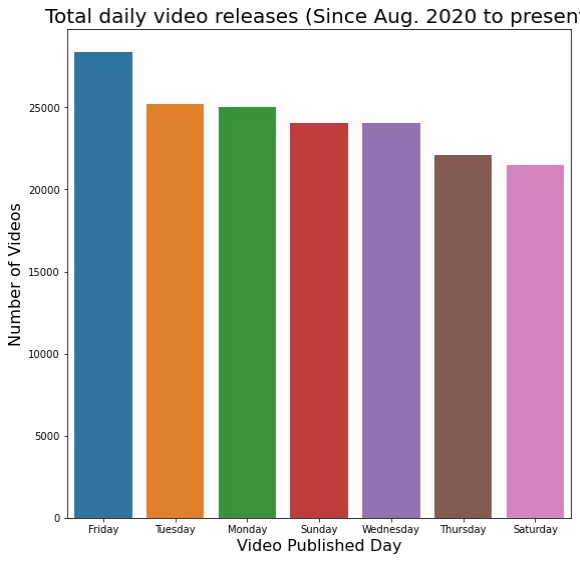
From above two charts demonstrate that Entertainment, Gaming, and Music are the most popular video category in the US. The total amount of these three types of videos accounts for almost half of the entire data set, or even more than half. However, the chart from the left side does not reflect the same trend. As can be seen from the channel names, among the top 10 channels with the

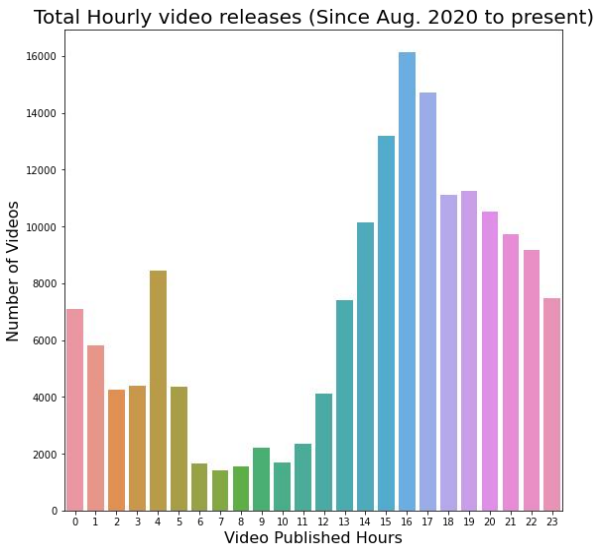
most video uploads, there are four sports-related channels, among which NFL and NBA rank first and second respectively. However, in the category chart, the total number of sports videos does not even rank in the top three. Such inconsistency makes the category a feature that is not very strong in prediction. Because the category and the number of trending videos do not show a strong relationship in the above two charts.

	category_title	Views in million	
		mean	count
1	Entertainment	3.09	33962
2	Gaming	1.82	33020
3	Music	4.44	28481
4	Sports	1.96	18346
5	People & Blogs	1.84	14968
6	Comedy	1.78	9187
7	Film & Animation	2.31	6791
8	News & Politics	2.29	6538
9	Science & Technology	2.64	5215
10	Howto & Style	1.30	4837
11	Education	1.82	4107
12	Autos & Vehicles	0.96	2999
13	Travel & Events	1.08	860
14	Pets & Animals	1.18	765
15	Nonprofits & Activism	2.45	114

In order to understand this data set better, it is useful and necessary to look at the data from another angle. The table on the left lists each category of videos per million views. count column represents the total view counts (in million), and the mean column represents the average view counts of each video over the total number of videos, the average view count which is also shown in million. The order of the category title as same as the Rank of Number of Video Category chart, but the mean column tells a different story. As mentioned before, Entertainment, Gaming, and Music are the top 3 category of total number of videos in this data set, but Music is the category which has the highest average value of view counts out of all categories. Even though Music is not the category that has the greatest number of videos in this data set but the third the greatest number of videos, and this category has biggest mean value out of all other categories which implies that music related video tend to have more view count than any other videos, and this category may be easier to trend compared to other videos.

2.4 Publish time analysis:





Despite the numbers of the videos and their category, the team also curious about what the distribution of publishing dates and times of all videos in this data set looks like. The top left graph shows that YouTube channels (or Youtuber) tend to upload their video on Friday, and this behavior may be due to videos getting more views at the beginning of the weekend when most people are free from work and school. The top right graph also supports this guess that majority of video are post in the afternoon, and the peak appears at 4 pm and 5 pm.

3 ML Methodology:

3.1 Data preprocessing:

In order to efficiently use resources, the model should be trained with the most valuable samples which means some unimportant samples should be dropped before loading into the model. Since the methodology of YouTube video trending is based on view count on unique video ID, it is reasonable to consider that those videos which stay on trending YouTube video table with more days have higher value than videos which stay on the table with less days. In order to do so, our team first regrouped the data frame by video id and calculated how long the videos stay on the table for each unique video ID. After the calculation, the mean of trending date length for all videos is 5.425 days and all video ID with trending date length less than 6 days has been dropped. After dropping the value, the size of the dataset dropped from 168590 rows to 93392 rows.

Before the data has been passed to the model, the value in each column has been standardized. Based on past experience, standardized data will improve the performance of the model and significantly reduce the computation time. For data splitting, the whole dataset has been split into 3 portions. 60% of the whole data has been used as training data. 20% of the whole dataset has been used as validation data. And the rest of 20% has been used as test data.

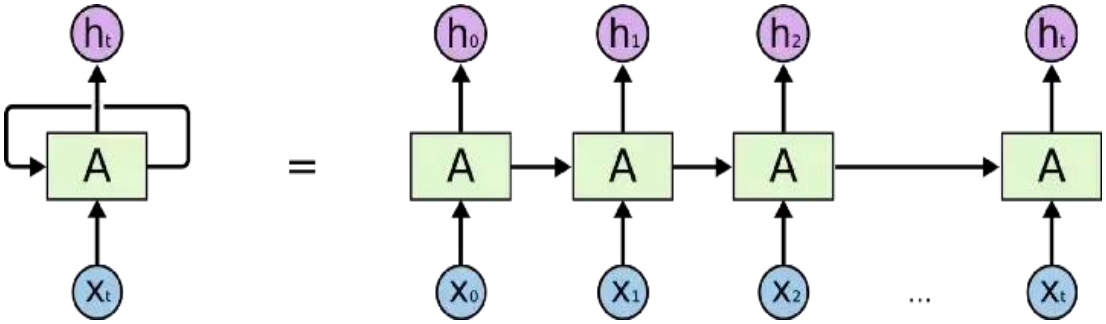
For each input, x has been constructed by the first 5 days data and y has been constructed by the 6th day. The objective of the regression model is to use the first 5 days data to predict the data for the 6th day. In order to observe the influence of different combinations of features, the input has been constructed in 2 forms to fit 2 models. For model 1, the input of x only contains count of view, and the size of input will be 5. For model 2, the input contains count of view, like, dislike, comment and the size of input will be 20.

size of input data	X	Y
model 1	5	1
model 2	20	1

In most cases, adding features will only improve the model performance. In order to see how the feature selection will influence the model, our team decided to use multiple features and single features at the same time and observe the performance.

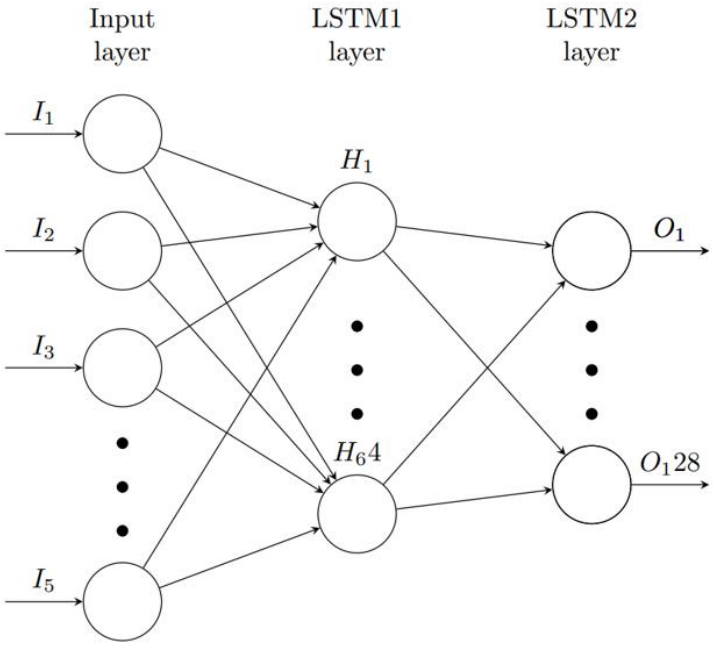
3.2 Model Selection:

Based on our study in the class, we are planning to do the project with a neural network. From our team perspective, among various neural networks models, CNN (Convolutional neural network) will be one of the best models to apply. With further learning about CNN, our team decided to use LSTM as a model. LSTM is a CNN model which has been specialized to solve the problem with gradient vanishing and gradient explosion. Since we are passing massive data from different video IDs, the model should be able to memorize the past data and use past data to predict the new data. Using LSTM will get a better outcome compared to CNN.



3.3 Model construction:

The LSTM model has been set to a sequential model which is appropriate for layers where each layer has exactly one input tensor and one output tensor. The base model also has been set with 2 LSTM hidden layers and 1 Dense hidden layer. The first hidden layer which is a LSTM layer will take the input (input size will be 5 for model 1 and 20 for model 2) and generate output in dimension (None, 5, 64). The second layer which also is a LSTM layer will take the outcomes from the first hidden layer as input and generate output in dimension (None, 128).



The third layer, which is a Dense layer, will take the outcome from the second layer and process the data with a linear activation function. Since the model is doing a regression problem, it is necessary that the outcome passes with the linear activation function. Since this is a big network, the model has been set to randomly select neurons to ignore during training in order to satisfy regularization purposes. For testing how different model structure influences the final outcome, the project will apply multiple different structures of model and evaluate the model performance. The architecture of different model will be different combination of hidden layers from the base model.

4 Results:

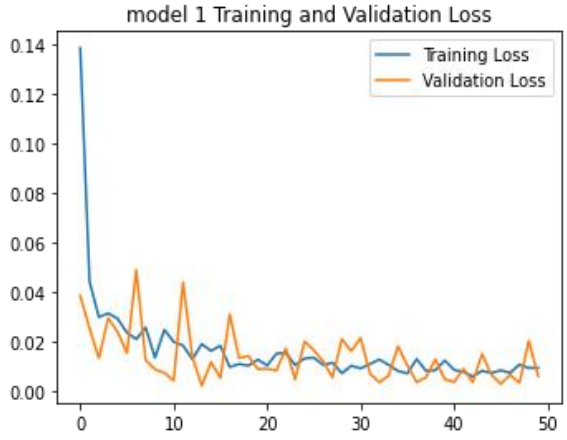
In the real world, neural networks have been successful on multi-task, and it is important to understand how the features and model structure influence the performance of a single network. As mentioned in section 3.1, in most cases, increase of the size of features will improve the performance of the model. This being said, it does not mean that increasing the feature size and model size will perform the best of all models. The goal of a network trained on YouTube trending video view prediction. The following table shows a collection of different architectures of LSTM and their corresponding loss.

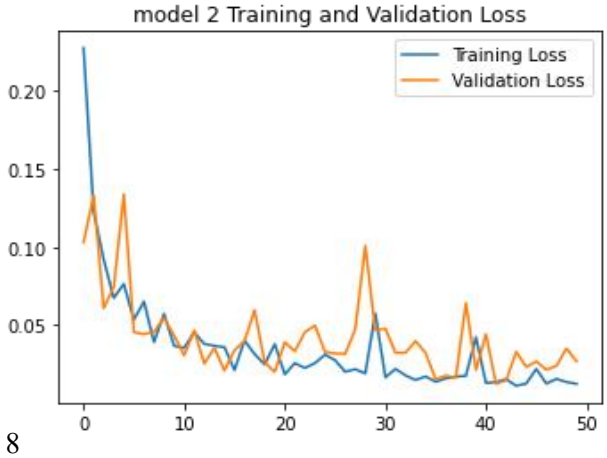
	feature	model construction	training score (RMSE)	validation score (RMSE)
model 1 (Base model)	view count	LSTM (64, 'tanh') + Dropout (0.2) + LSTM (128, 'tanh') + Dense (1, 'linear')	536762.75	527361.94
model 2	view count+ like count+ dislike count+ comment count	LSTM (64, 'tanh') + Dropout (0.2) + LSTM (128, 'tanh') + Dense (1, 'linear')	953128.67	1007219.69
model 3	view count	LSTM (64, 'tanh') + Dropout (0.2) + LSTM (128, 'tanh') + Dropout (0.2) + LSTM (128, 'tanh') + Dense (1, 'linear')	746502.79	996693.79

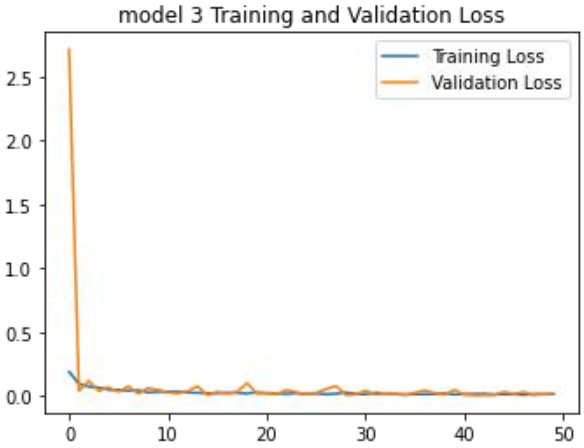
model 4	view count	LSTM (64, 'tanh') + LSTM (128, 'tanh') + LSTM (128, 'tanh') + Dense (1, 'linear')	551557.44	388200.11
model 5	view count	LSTM (64, 'tanh') + LSTM (128, 'tanh') + Dense (1, 'linear')	407390.63	314178.32
model 6	view count	LSTM (64, 'tanh') + Dense (1, 'linear')	347587.62	294136.83
model 7	view count+ like count+ dislike count+ comment count	LSTM (64, 'tanh') + LSTM (128, 'tanh') + Dense (1, 'linear')	1180579.60	909351.06

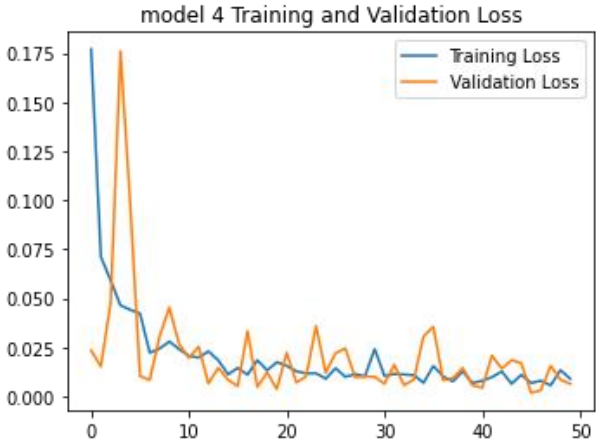
From the model performance table, model 6 has the best performance in all models. Since model 6 only has one hidden layer and one dense layer with linear activation function, model 6 performs similar to linear models. On the other hand, models with more layers have worse performance compared to model 1. In this case, adding complexity to the model does not improve the model performance.

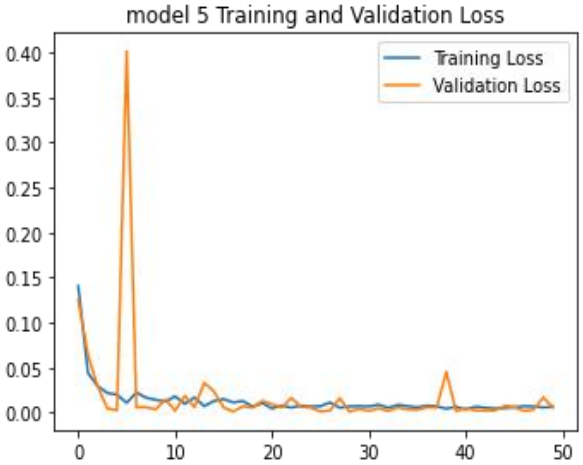
Generally speaking, the LSTM model is a model with low bias and high variance which means the model is more likely to cause overfitting problems compared to underfitting problems. In order to decrease the chance of the model getting an overfit problem, some models have dropouts which will randomly select neurons and ignore them during training. The parameter has been set to 0.2. From the model performance table, adding dropouts in model training is not helping the performance of the model. Compared with the same model with and without dropouts, for example, model 3 and model 4, models without dropouts have better performance than models with dropouts. In conclusion, the advantage of dropouts is less than the disadvantages of dropouts in this case.

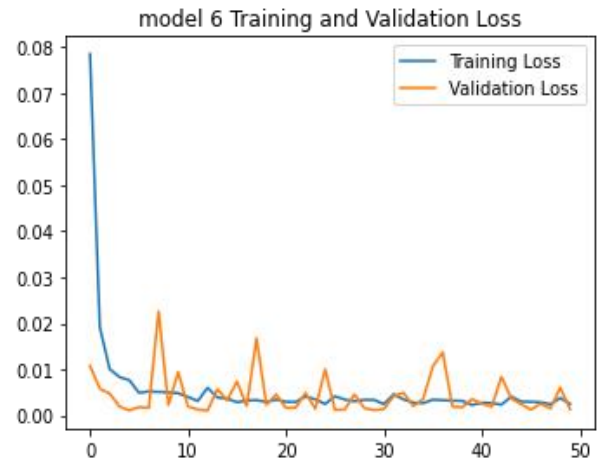


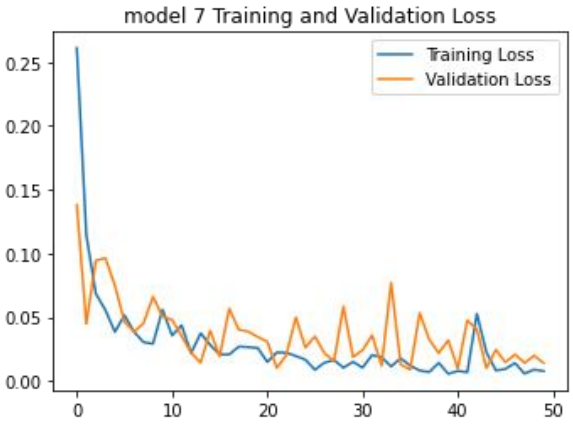












The training and validation loss for all models represented that most of the models demonstrated good fit learning curves and none of the models demonstrated underfit learning curves. Among all models, model 2 is showing overfitting symptoms. From the training score, model 2 has significantly increased on RMSE training and validation score than model 1. For the reason why model 2 is overfitting, model 2 compared to model 1, takes additional 4 features which is generating noise during the training. Since additional features do not improve the model performance, these features should be categorized as garbage values. At the same time, most of the models are showing the symptoms of processing unrepresentative validation dataset which means the validation dataset lacks sufficient information to evaluate the ability of the model to generalize.

5 Website construction:

One requirement for this project is to construct a user interface on a website. Due to the fact that members of the team do not have enough knowledge of HTML, a website developing tool called Anvil which allows using python code to construct a website had been used to construct website application. The Jupyter notebook which contain all codes and models works as server of interface website. In order to have a full functioning anvil web application, a fully ran Jupyter is necessary.

The website interface has multiple pages, the first page includes the project object information. the model page contains two models which are model 1 and model 2 respectively. In order to help the user to better understand how the input and output will look like, there are some brief information about input for both models. There also are the example of output for both models. The following

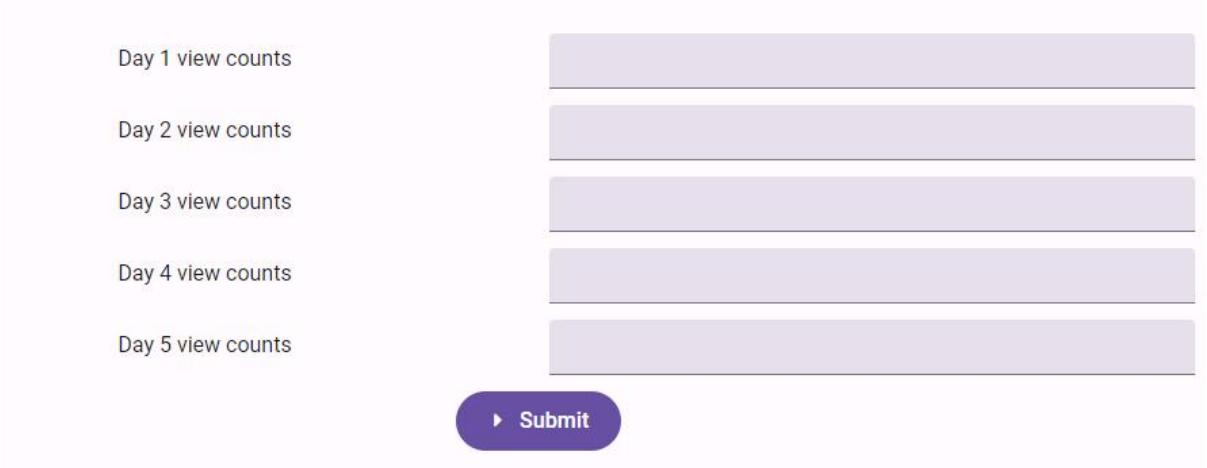


image is the input interface for model 1, it will ask the user to input the view counts of five consecutive days of a video in order to predict the sixth day's view count.

And the image on the left is an example of the output from the base model. As the example shows, it does not show one example output but ten example outputs which are the test data from the base model. In order to let the users have a general idea about the model, the example result includes a training score and a validation score which is the performance of how the test data performs. For each example, the result follows a specific format that first present model prediction value, actual value, and the input data set.

The link to the interface website clicks here.

6 Discussion:

The most interesting finding of the project is that model complexity is not always correlated to accuracy of model. In today's life, people are focusing more on learning complex models since complex models seem to be more capable of solving different problems. However, for the result of the project, a complex model does not always improve the accuracy. It seems like a paradox. Since a complex model with plenty of layers should be able to understand the data better than a simple model with single layers. It is against the intuition, but it is happening.

One possible explanation of this paradox is that if the dataset can be easily interpreted by linear relationship when the model has multiple layers, some layers may learn about the noise of the data which may cause additional loss to the model. Such a theory is possible, but hard to be proved. Since the model is too complex for humans, most of the time, humans cannot interpret how different features study on different layers which causes people cannot fine-tune the model based on

layers' performance. When the training process becomes a huge black box, it is hard to do any adjustment. In this case, model tuning becomes a possibility game in which people just keep building models and find out which model performs better. Or people build models based on their understanding of the data and assume how different layers will learn different features of data. Either way cannot be said to be the best approach to the problem.