Precog Round 2 Task Report
Predicting the Likes Count of A YouTube Video

Mayank Saxena

I. Introduction

The proliferation of Social Networking websites such as Facebook, Instagram, Twitter, YouTube, etc., has generated considerable interest in predictive analytics. Popularity of such sites has raised interest of individuals as well as business organizations to gather 'likes' for their uploaded content from as large a number of viewers/ subscribers as possible. In today's world, people base their social popularity or sometimes even their self-esteem on the number of *likes* their Facebook post or their Instagram pictures gather. From the commercial point of view, business organizations are interested in getting *likes* for their content for reasons ranging from financial benefits to market share among others. For example, the number of *likes* of a movie trailer might result in more footfalls at the theatre for the movie. In the recent US Election, both the candidates, Trump and Hillary, had their own social media teams tasked with generating social media content that would become popular, i.e., get the maximum *likes*.

This project contributes to the topic of popularity prediction by predicting the *like* count of YouTube videos based on features gathered from Youtube and other social networking sites. The ability to predict the *like* count of a YouTube video could be converted into knowledge of how to generate content so that it becomes more popular. Early identification of viral videos can gainfully be applied to network optimization through mobile data prefetching or by content delivery networks (CDNs) to design new popular content [1], [2]. The application prediction popularity of YouTube videos is not only important for advertising agencies to plan advertising campaigns, ad revenue and estimate costs, but also for companies to be able to quickly respond to rivals in viral marketing campaigns [3], [4].

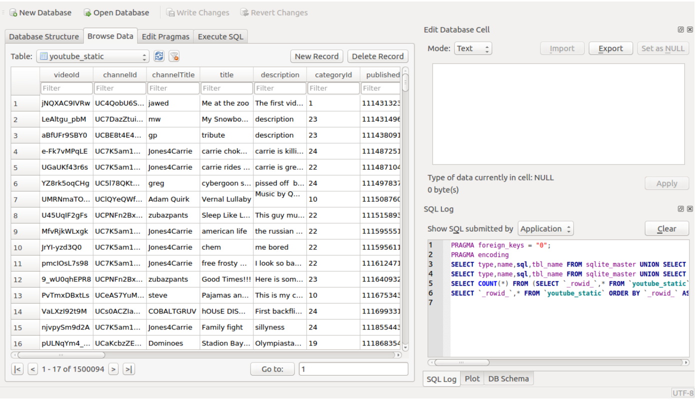
To achieve the goal of *like* prediction, I built a Stochastic Gradient Descent based Linear Regression Model using a dataset of about 1 million YouTube videos. Experimentation from an initial set of features that were directly collected from the YouTube API augmented with other features gathered from social network presence of the video resulted in a predictive model that shows good performance.

II. Method

A. Collecting Data

A Python script using the official libraries of the YouTube API was developed to collect data. Using the YouTube search API, I gathered data of videos which were uploaded between 1st July 2016 to 31st December 2016.

Initial attempts at storing the data as JSON or even CSV files resulted in failure due to frequent loss of Internet connectivity. The solution was to save the gathered data in a MySQLite databse with frequent *commits* at regular intervals (as suggested by [5]). In case of Internet failure, the script was restarted from the date till when the data was collected successfully. This resulted in a data set of attributes of 1.1 million YouTube videos.



	videoId	channelId	channelTitle	title	description	categoryId	published
	Filter	Filter	Filter	Filter	Filter	Filter	Filter
1	jNQXAC9IVRw	UC4QobU6S...	jawed	Me at the zoo	The first vid...	1	111431323
2	LeAltgu_pbM	UC7DazZtui...	mw	My Snowbo...	description	23	111431496
3	aBfUFr9SBY0	UCBE8t4E4...	gp	tribute	description	23	111438091
4	e-Fk7vMPqLE	UC7K5am1...	Jones4Carrie	carrie chok...	carrie is killi...	24	111487251
5	UGaUKf43r6s	UC7K5am1...	Jones4Carrie	carrie rides ...	carrie is gre...	22	111487104
6	YZ8rk5oqCHg	UC5l78QKt...	greg	cybergoon s...	pissed off b...	24	111497837
7	UMRNmaTO...	UClQYeQWf...	Adam Quirk	Vernal Lullaby	Music by Q...	10	111508760
8	U45UqIF2gFs	UCPNFn2Bx...	zubazpants	Sleep Like L...	This guy mu...	22	111515893
9	MfvRjkWLxgk	UC7K5am1...	Jones4Carrie	american life	the russian ...	22	111595551
10	JrYI-yzd3Q0	UC7K5am1...	Jones4Carrie	chem	me bored	22	111595611
11	pmclOsL7s98	UC7K5am1...	Jones4Carrie	free frosty ...	I look so ba...	22	111612471
12	9_wU0qhEPR8	UCPNFn2Bx...	zubazpants	Good Times!!!	Here is som...	23	111640932
13	PvTmxDBxtLs	UCeAS7YuM...	steve	Pajamas an...	This is my c...	10	111675343
14	VaLXzI92t9M	UCs0ACZIa...	COBALTGRUV	hOUsE DIS...	First backfli...	24	111699331
15	njvpySm9d2A	UC7K5am1...	Jones4Carrie	Family fight	sillyness	24	111855443
16	pULNqYm4_...	UCaKcbzZE...	Dominoes	Stadion Bay...	Olympiasta...	19	111868354

|< < 1 - 17 of 1500094 > >|

Go to: 1

Type of data currently in cell: NULL
0 byte(s)

```
PRAGMA foreign_keys = "0";
PRAGMA encoding
SELECT type,name,sql,tbl_name FROM sqlite_master UNION SELECT
SELECT type,name,sql,tbl_name FROM sqlite_master UNION SELECT
SELECT COUNT(*) FROM (SELECT `_rowid_`,* FROM `youtube_static`
SELECT `_rowid_`,* FROM `youtube_static` ORDER BY `_rowid_` AS
```

Fig. 1. Snapshot of selected YouTube Video data in DB Browser.

Figure 7 shows a snapshot of selected rows of YouTube video data in DB Browser.

B. Features Chosen

I started with collecting features of YouTube videos that would intuitively be correlated (positive or negative) with the video *like* count directly from the YouTube API. The YouTube API provided the following features:

- **View Count** - The number of times a video has been viewed.
- **Comment Count** - The number of comments the video has received.
- **Dislike Count** - The number of users who had indicated that they disliked the video.
- **Favorite Count** - The number of users who have added the video to their Favorites.
- **Definition** - Whether the video was available in high definition or standard definition.

- **Caption Availability** - Whether the video had captions or not.
- **Duration** - The length of the video in seconds.
- **Licensed Content** - Whether the video contained licensed content.
- **Dimension** - Whether the video was available in 2D or 3D.
- **Desciption** - The description of the video.
- **Publishing Date** - The date and time the video was published. From the publishing date, a new features , lifetime of video was calculated.
- **Title** - The video's title.
- **Category** - The video category associated with the video.
- **Embeddability** - Whether the video could be embedded on a third-party website.
- **Availability of Statistics** - Whether the video's statistics could be viewed by anyone.
- **Latitude** - The latitude coordinates of the video's origin in degrees.
- **Longitude** - The longitude coordinates of the video's origin in degrees.

Another set of features was gathered from characteristics of the channel via which the video was uploaded. Popularity of the uploader is a more important factor for a video to go viral than the upload time [3]. Intuitively, channels with a higher number of subscribers would definitely garner higher number of *likes* on average than a video which was posted by a channel with fewer number of subscribers. For example, any video uploaded by PewDiePie (the YouTube channel with the highest number of subscribers) would definitely garner a higher number of likes on any video than a video uploaded by a channel having single digit subscribers on its YouTube channel.

Following are the attributes that were gathered about the channel using the YouTube API -

- **Channel Subscribers Count** - The number of subscribers the publisher of the video has.
- **Channel Video Count** - The total number of videos which have been posted by the owner of the video.
- **Channel View Count** - The sum of all the views which have generated on all the videos published by the channel.
- **Channel Comment Count** - The sum of all the comments which have been generated on all the videos publish by the channel.

Next, I thought about the possible ways a YouTube video gets popular and hence more *liked* on social media. There exists a high correlation between the number of total views on a YouTube video and the total number of likes. This led me to find out the number of times the video had been shared on various social networking platforms such as Facebook, Google Plus, Linkedin and Pinterest.

- **Facebook Shares** - The total number of times the video has been shared on Facebook.
- **Google Plus Shares** - The total number of times the video has been shared on Google Plus.
- **Linked In Shares** - The total number of times the video has been shared on Linked In.
- **Pinterest Shares** - The total number of times the video has been shared on Pinterest.
- **Total Shares** - The number of times the video has been shared on all social media platforms.

On reading this article (http://www.wikihow.com/Get-More-Views-on-Your-YouTube-Videos) I discovered that more features can be extracted from the description of the video. Two such features I decided to incorporate in the model were -

- Whether the description of the video contained a link to any website.
- Whether the description of the video contained links to any of the social media platforms such as Facebook, Twitter, Instagram etc.
- Character count of the description and the title of the video.
- Word count of the description and the title of the video.

C. Post Processing of Data

Analysis of the values of frequency counts of YouTube videos, such as the *view count*, *like count*, *dislike count*, and *comment count* showed an extreme range of values from no likes to little more than 9 million *likes*. The least number of *likes* is 0, the maximum *likes* are $93, 68, 896$ and the average number of *likes* are 3315.91.

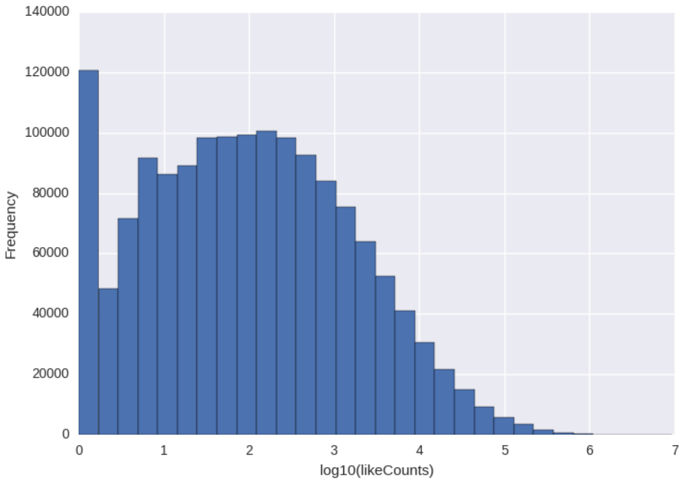


Fig. 2. Frequency Distribution of the log10 of *like* count of videos in the YouTube dataset.

Figure 2 shows the frequency distribution of the *like* counts in the YouTube dataset. The skewness of the frequency counts necessitated an data transformation on the frequency counts before using them as features in the regression model. The log10 transformation is the appropriate transformation for this situation because it deals with extreme values.

III. Analysis of Data

After extracting features from different sources and storing them in an SQLite database I found out the Karl Pearson's correlation coefficients between the features. From the plot below it is clear that there exists a high positive correlation between view count and the like count. Surprisingly, there also exists a positive correlation of 0.44 between dislike count and like count. This can be attributed to the fact that videos having a high number of views will have a higher count of both like count as well as dislike count. As expected there also existed a strong positive correlation between comment count and like count for a YouTube video.

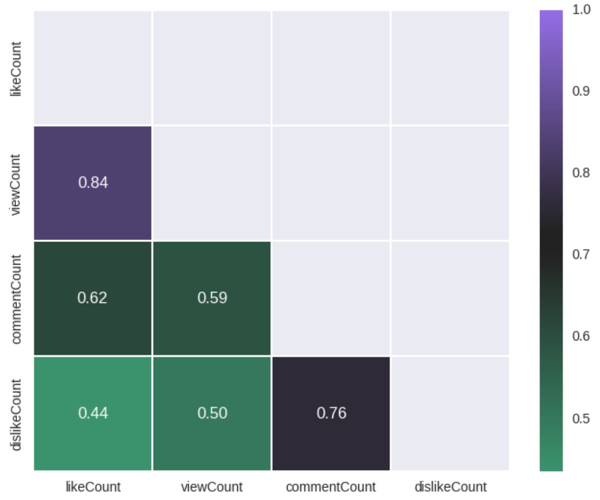


Fig. 3. Karl Pearson's Correlation coefficient plot between Like Count, View Count, Comment Count and Dislike Count.

Next I plotted the frequency distribution curves for View Count, Comment Count and Dislike Count. Since the data was highly skewed, barely anything could be understood from the plot at first. Hence, I applied a transformation of log10 on each value and then plotted the frequency distribution graphs. From the plots it is evident that majority of the videos had very few comments and dislikes and a very less number of videos had a high number of views, comments and dislikes.

The frequency distribution curve for the view count was plotted and is clear from the plot it is a bell curve i.e. it follows normal distribution.

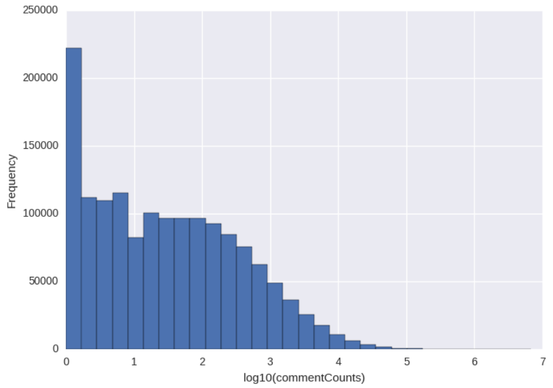


Fig. 4. Frequency distribution plot as a histogram for Comment Count.

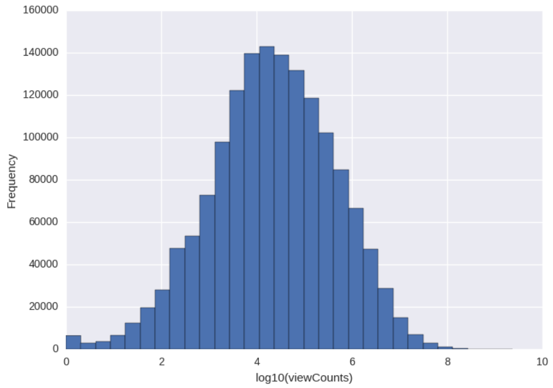


Fig. 5. Frequency distribution plot as a histogram for View Count.

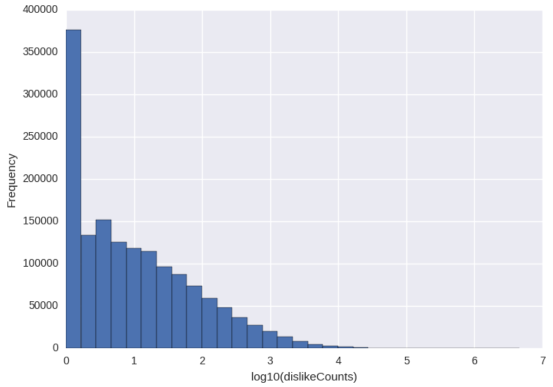


Fig. 6. Frequency distribution plot as a histogram for Dislike Count.

IV. Experiments

This section focuses on the experiments with Stochastic Gradient Descent (SGD) based Linear Regression models in predicting the number of likes for a given YouTube video. For this purpose, the machine learning library for Python called as ScikitLearn was used. The fact that the dependant variable (video like count) is known in advance in this study makes all the following experiments of the supervised learning type.

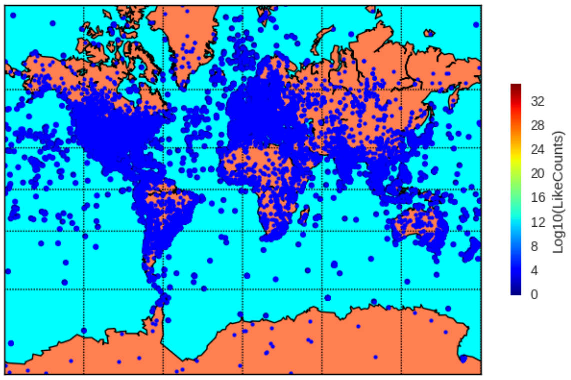


Fig. 7. Location wise plot of videos which were uploaded along with their Latitude and Longitude.

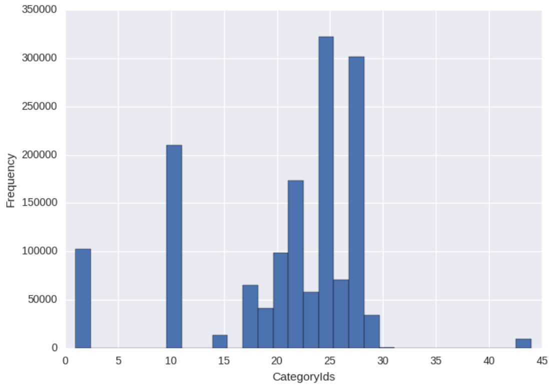


Fig. 8. Frequency distribution plot as a histogram for Category Id.

A. Different Data Sets

The data was collected as described in the method section, resulting in a data set of 1.5 million YouTube videos to be used in linear regression models. The data set was divided into a training set, a validation set and a testing set. The reason for this was that the data set was sufficiently large. Two aspects were important in splitting the data over different sets. First, the data set had to be randomly shuffled before dividing it into training, validation and testing sets because the entire data set was ordered by video age. This shuffling process was done with a pre-set seed value to ensure that for each experiment, as well as for the final regression models, the same training data were used to training with and the same validation and test data were predicted on. Secondly, the data set was split as follows :

- Training data set - 60
- Validation data set - 20
- Test data set - 20

TABLE I
CATEGORYID OF SOME CATEGORIES OF YOUTUBE VIDEOS

CategoryId	**Category**
1	Film & Animation
2	Autos & Vehicles
10	Music
15	Pets & Animals
17	Sports
18	Short Movies
19	Travel & Events
20	Gaming
21	Videoblogging
22	People & Blogs
23	Comedy
24	Entertainment
25	News & Politics
26	Howto & Style
27	Education
28	Science & Technology
29	Nonprofits & Activism
30	Movies
31	Anime/Animation
32	Action/Adventure
33	Classics
34	Comedy
35	Documentary
36	Drama
37	Family
38	Foreign
39	Horror
40	Sci-Fi/Fantasy
41	Thriller
42	Shorts
43	Shows

B. Linear Regression with Stochastic Gradient Descent

Multivariate linear regression model is used to predict the textitlike count of YouTube videos. The model is built from samples in the training data set (from which the model learns). Since linear regression is not scalable, I used the Stochastic Gradient Descent (SGD) optimization algorithm that improves performance in large datasets, where N $\geq$ 1,000,000. Therefore, in this project, SGD based linear regrsession was used to predict *like* count with the features that were listed in the previous section. Many combinations of the parameters of the function (alpha, loss function, penalty) were tried to find the best model.

The best fit model was pickled to make predictions on new data. The Standard Scaler and the One Hot Encoder were pickled and saved so that they could be used to encode the data for the YouTube video for which the number of likes have to be predicted.

Initially, I attempted to use the Support Vector Regression model(SVR) from sklearn to fit the data. However, I realized that the SVR may not be scalable,its training algorithm takes between quadratic and cubic time.

V. Conclusion and Future Work

The first conclusion which can be drawn from this research is that the popularity of a video in terms of its like count can be predicted fairly well from features such as the view count, comment count, dislike count, channel subscriber count, channel video count, total shares across social media platforms along with features such as finding out whether the description of the video contains links to any third party website or any other social media platforms such as Facebook, Twitter, Instagram etc. Future research should also look into deriving features from image thumbnails. This will require retrieving, storing and extracting features from a large number of images. Extraction of values for red, blue, green, hue, saturation, brightness could be considered but such values might be too generic. More information rich features (eg - object detection and identification) are more computationally heavy but will definitely improve the accuracy of the regression model. Intuition says that perhaps the presence of peoples faces in video thumbnails results in more likes than when it is not. Another area where this model can be improved is by considering the content of the video into account. For example, if the person in the video asks the viewer to like the video then the viewers are more likely to like the video. Similarly, supposing a YouTuber announces a prize give away competition and asks the people to like the video to get enrolled in the competition, there will definitely be a higher number of likes on that video.

References

[1] P. Jain, J. Manweiler, A. Acharya, and R. R. Choudhury, "Scalable social analytics for live viral event prediction," in *Proceedings of the Eighth International Conference on Weblogs and Social Media, ICWSM 2014, Ann Arbor, Michigan, USA, June 1-4, 2014.*, 2014.

[2] G. Fontanini, M. Bertini, and A. Del Bimbo, "Web video popularity prediction using sentiment and content visual features," in *Proceedings of the 2016 ACM on International Conference on Multimedia Retrieval*, ICMR '16, (New York, NY, USA), pp. 289–292, ACM, 2016.

[3] L. Jiang, Y. Miao, Y. Yang, Z. Lan, and A. G. Hauptmann, "Viral video style: A closer look at viral videos on youtube," in *Proceedings of International Conference on Multimedia Retrieval*, ICMR '14, (New York, NY, USA), pp. 193:193–193:200, ACM, 2014.

[4] H. Pinto, J. M. Almeida, and M. A. Gonçalves, "Using early view patterns to predict the popularity of youtube videos," in *Proceedings of the Sixth ACM International Conference on Web Search and Data Mining*, WSDM '13, (New York, NY, USA), pp. 365–374, ACM, 2013.

[5] D. C. S. Jordy Snijders, Dr. G.A. Chrupala, "YouTube Video Popularity." http://msc.jordysnijders.com/msc-thesis_jordy-snijders.pdf, 2015.