International Journal of Scientific Engineering and Science
Volume 4, Issue 6, pp. 15-17, 2020.
ISSN (Online): 2456-7361

Analysis of YouTube Videos: Detecting Click bait on YouTube

Neha Reddy Vadde[1], Piyush Gupta[2], Prasham Mehta[3], Puneet Gupta[4], Vikranth BM[5]
[1, 2, 3, 4]4th Year student, Dept. of CSE, BMS College of Engineering, Bengaluru, Karnataka
[5]Asst. Professor, Dept. of CSE, BMS College of Engineering, Bengaluru, Karnataka
Email address: [1]1BM16CS057@bmsce.ac.in, [2]1BM16CS066@bmsce.ac.in, [3]1BM16CS070@bmsce.ac.in,
[4]1BM16CS072@bmsce.ac.in, vikranthbm.cse@bmsce.ac.in

Abstract— *Consumption of content from YouTube (Lanyu Shang, 2019) and other OTT (over-the-top) platforms is constantly increasing. YouTube (Lanyu Shang, 2019) being a source of education, entertainment and promotion, is a very lucrative platform. YouTubers tend to unethically attract viewers into clicking their video by manipulating their title and/or thumbnail. In this paper we present a method to train a model to classify a video as click bait (Lanyu Shang, 2019) video or non-click bait (Lanyu Shang, 2019) video.*

Keywords— *Clickbait, YouTube (Lanyu Shang, 2019) [1], Comments, Title, Thumbnail.*

I. Introduction

YouTube is becoming a major resource for sharing and consuming video content. It is gaining immense popularity and support from viewer community due to its comprehensive repository of videos. Also, it supports diversity by having different facets such as modals, languages, domains and cultures. For a YouTube (Lanyu Shang, 2019) content developer or a You Tuber with various notable channels, (Lanyu Shang, 2019) this is a profession with a lot of monetary potential. The younger generations are recently shifting to YouTube (Lanyu Shang, 2019) and other OTT platforms, away from the traditional television.

A YouTube (Lanyu Shang, 2019) video often consists of a title, thumbnail, video content along with other non-video features. Despite it being unethical, content developers deliberately manipulate the heading and the thumbnail so as to attract more audience and baiting them into viewing their content. There are quite a few instances when the content of the video mismatches with the heading of the video or the thumbnail of the video. This is known as a Clickbait (Lanyu Shang, 2019) Video. Our aim is to classify a video as to whether it is a Clickbait (Lanyu Shang, 2019) or not. This is critically important as a majority of people spend their time on YouTube (Lanyu Shang, 2019) and not getting what they search for is a waste of their precious time. We use sentiment analysis on viewer comments to identify a video as click bait or not.

II. DATASET

YouTube is visited by over 1.9 billion logged-in users each month and over a billion hours of video are watched daily. In view of the diversity and popularity of, we take YouTube as our data source to collect video information. We are only working with YouTube (Lanyu Shang, 2019) data that consists of viewer comments.

The data is collected with the help of YouTube (Lanyu Shang, 2019) API v3. We created a Google Developer account and obtained authorization credentials to extract all the details of a video in the form of a JSON file. This dataset contains all the details of the trending YouTube videos along with its likes, dislikes, comments, tags and views for each video for a particular year, which comprises a top-level comment and replies, if any exist, to that comment.

III. Methodology

We implement our project by dividing the process into 3 modules. They consist of: Comments Characteristics Analysis, Metadata Characteristics Analysis and Supervised Classification.

Comments Characteristics Analysis

YouTube (Lanyu Shang, 2019) comments have two level thread structure. It has a top-level comment node and another level is replies to that node. This module is designed to capture characteristics from the audience's comments of an online video.

We have done all steps from simple text cleaning by removing white spaces, punctuations, emoji’s and special characters up to more sophisticated normalization techniques such as tokenization, splitting a sentence into words. We also replaced contractions with their full forms, removed a word if it exists in the list of stop words provided by NLTK and reduced the derived words to their word stem i.e. root form.

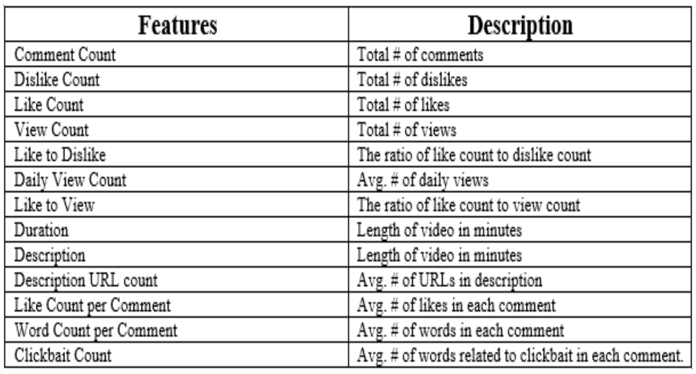
We implemented tokenizing of text in to number vectors with the help of Keras. This helped us in training the model better. Every comment is tokenized to words initially and then it is converted to vectors (numbers). After converting to vector it is transformed into a 2D array with the help of Numpy. It also pads the word to a max length of 140. Sequences that are shorter than 140 are padded with 0 in the beginning of each sequence until each sequence has the same length as the longest sequence. Sequences longer than 140 are truncated so that they fit the desired length.

We split our data into two subsets: training data and testing data and fit our model on the train data, using which we made predictions on the test data.

Metadata Characteristics Analysis

In this module we extract a few metadata features of the videos as specified below:

TABLE 1. Metadata Features



Features	Description
Comment Count	Total # of comments
Dislike Count	Total # of dislikes
Like Count	Total # of likes
View Count	Total # of views
Like to Dislike	The ratio of like count to dislike count
Daily View Count	Avg. # of daily views
Like to View	The ratio of like count to view count
Duration	Length of video in minutes
Description	Length of video in minutes
Description URL count	Avg. # of URLs in description
Like Count per Comment	Avg. # of likes in each comment
Word Count per Comment	Avg. # of words in each comment
Clickbait Count	Avg. # of words related to clickbait in each comment.

First we tokenize the title through a custom tokenizer by adding a space between numbers and letters, removing punctuation, repeated whitespaces, words shorter than 2 characters, and stop-words and returning a list of stems and, eventually, emoji’s. Then we embed the title into a vector representation by computing the mean vector representation of the title tokens from a Word2Vec model.

If the number of views, likes, dislikes and comments are known, we compute their logarithm because the logarithm of the numeric parameters is generally normally distributed. We then applied min-max scaling using a scaler previously trained on the train set (min-max-scaler) since our model SVM assumes that the data it works with is in a standard range.

This method is a metadata features-based click bait classification approach that detects the click bait video on YouTube based on the title, number of views, likes, dislikes and comments of the video. This data gives us information which cannot be extracted from comments characteristics.

Supervised Classification

This module integrates results of all the above modules and performs a binary classification on it. We compare different classification techniques such as Support Vector Machines, Long short Term Memory networks, Random Forest classifier. We select the best-performed one to be the one used in our scheme.

IV. Evaluation

We evaluate the accuracy and precision of all the methods against each other and also plot a Receiver Operating Characteristic (ROC) curve for all to evaluate the robustness of their performance.

The SVM model has been trained on more than 28 thousand samples and tested on more than 7 thousand samples hence it can detect click bait YouTube videos through their metadata, with a 96% F1 score.

When we trained the LSTM model with a batch size of 20 and a training and validation split of 80 to 20, it reached an accuracy of 93% for the training data.

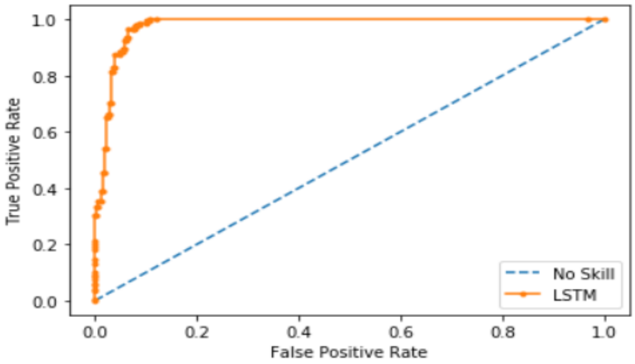


Fig. 1. ROC Curve for LSTM

A random forest classifier when trained with the number of estimators = 600, gave us accuracy as high as 97% which is pretty good compared to previous classifiers.

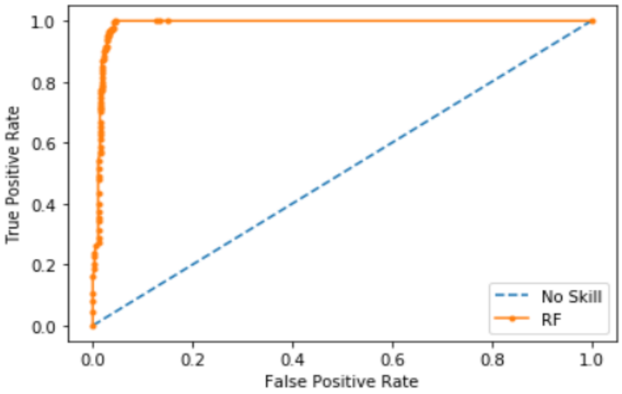


Fig. 2. ROC Curve for RF

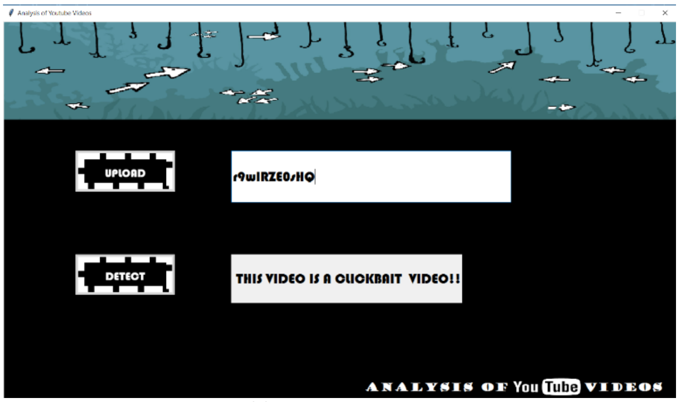
The performance of all compared schemes is summarized in Table 2. We observe that Random Forest classifier performs the best among all classifiers and thus is selected to be the default classifier in our scheme.

TABLE 2. Click bait Classification Performance for All Methods

Algorithms	Accuracy	Precision	Recall	F1-Score
SVM	0.9676	0.97	0.96	0.97
LSTM	0.9379	0.94	0.94	0.93
RF	0.9714	0.95	1.00	0.98

V. Testing

First the user enters video Id of the video he wants to find out as input. The script extracts the title of the video to analyze and, if known, the number of views, likes, dislikes, and comments as well. The script will print the model prediction based on the metadata: 1 if the video is probably click bait, 0 otherwise. Then the comments are preprocessed and random forest classifier model will predict whether the video is click bait or non – click bait.



r9wIRZE0sHQ

THIS VIDEO IS A CLICKBAIT VIDEO!!

Fig. 3. Integration of front end with machine learning models

VI. Conclusion and Future Work

In this paper, we developed a content-free theme to find click bait videos on on-line video sharing platform, you tube. Our theme leverages the comment and interaction between users who watched the video, and learns latent options from their unstructured and sophisticated comments.

We also outline a few future research directions that can be built on the work from this paper. First, click bait videos on YouTube are often customized by experienced content creators who know how to game with the YouTube recommendation system. To make our scheme more robust and generalizable across different video platforms, we plan to extend its compatibility with other video sharing platforms using different commentary structures. Examples of such platforms include both video-centric platforms (e.g., Vimeo) and social-based platforms (e.g., Twitter). Second, there exist many marketing vendors from which content creators can buy comments and high-retention views. In future work, we will further study how to identify and filter such fake or machine-generated comments.

Acknowledgement

We are highly indebted to Mr. Vikranth B M for his guidance and constant supervision as well as for providing necessary information regarding the project and also for his support in completing the project. The work reported in this paper is supported by college through the Technical Education Quality Improvement Program (TEQIP – III) of the Ministry of HRD, Government of India.

References

[1] Towards Reliable Online Clickbait Video Detection: A Content-Agnostic Approach: Lanyu Shang, Daniel Zhang, Michael Wang, Shuyue Lai, Dong Wang
[2] Peter Adelson1, Sho Arora2, and Jeff Hara3 - Clickbait; Didn't Read: Clickbait Detection using Parallel Neural Networks
[3] Kennedy Ogada, Waweru Mwangi, Wilson Cheruiyot - N-gram Based Text Categorization Method for Improved Data Mining
[4] Hanif Bhuiyan, Jinat Ara, Rajon Bardhan - Retrieving YouTube Video by Sentiment Analysis on User Comment
[5] Muhammad Zubair Asghar1, Shakeel Ahmad2, Afsana Marwat1, Fazal Masud Kundi1 - Sentiment Analysis on YouTube: A Brief Survey
[6] Tyler West - Going Viral: Factors That Lead Videos to Become Internet Phenomena
[7] M. M. U. Rony, N. Hassan, M. Yousuf, Diving deep into clickbaits: Who use them to what extent in which topics with what effects?
[8] M. Bartl, Youtube channels, uploads and views: A statistical analysis of the past 10 years.
[9] D. Wang, T. Abdelzaher, L. Kaplan, Social sensing: building reliable systems on unreliable data, Morgan Kaufmann, 2015 (2015).
[10] A. Agrawal, Clickbait detection using deep learning, in: 2016 2nd International Conference on Next Generation Computing Technologies (NGCT), IEEE, 2016.
[11] E. Alpaydin, Introduction to machine learning, MIT press, 2014