PAID PROMOTERS RECOMMENDATION SYSTEM

A PROJECT REPORT

Submitted by

AKASH A (180701017)

CHRIS GAVIN B (180701056)

in partial fulfillment for the award of the degree

of

BACHELOR OF ENGINEERING

in

COMPUTER SCIENCE AND ENGINEERING

RAJALAKSHMI ENGINEERING COLLEGE

ANNA UNIVERSITY, CHENNAI

MAY 2022

RAJALAKSHMI ENGINEERING COLLEGE, CHENNAI

BONAFIDE CERTIFICATE

Certified that this project titled "**PAID PROMOTERS RECOMMENDATION SYSTEM**" is the bonafide work of "**AKASH A (180701017), CHRIS GAVIN B (180701056)**" who carried out the work under my supervision. Certified further that to the best of my knowledge the work reported herein does not form part of any other project report or dissertation based on which a degree or award was conferred on an earlier occasion on this or any other candidate.

SIGNATURE

Dr. Revathy P, M.E., Ph.D.,

HEAD OF THE DEPARTMENT

Department of Computer Science and Engineering

Rajalakshmi Engineering College

Chennai - 602 105

SIGNATURE

Mrs. Ponmani S, M.E.,

SUPERVISOR

Assistant Professor (SG)

Department of Computer Science and Engineering

Rajalakshmi Engineering College

Chennai - 602 105

Submitted to Project Viva-Voce Examination held on _____

Internal Examiner **External Examiner**

ABSTRACT

Advertisements are one of the key revenue in today's digital world, ads are approaching users in multiple ways and has been evolving from generation to generation, previously we could only see ads on tv, radio, and newspaper but nowadays in the modern world ads reach us mostly through online platforms such as Youtube, Facebook, Instagram, etc. In today's digital platform YouTube is one of the immense benefiter from ads by running video ads on start and in between a content video. In the first quarter of 2021 Youtube's income was about 8.6 billion U.S. dollars from advertisements alone. One step ahead, apart from the YouTube-generated ads, content creators who make and publish videos on the YouTube platform and personally promote advertisements are called Paid Promoters. The impact of advertisement when delivered by paid promoters will have more impact on the viewers than auto-generated youtube ads. So, advertisers contact these paid promoters to advertise their products on youtube. But these advertisements have very less impact on the viewers. Our project idea is to make a recommendation system that recommends content creators relevant to ads of advertisers with a possibility of viewership. Using this the advertisers can reach their relevant audience and make their investment in ads in a guaranteed way.

ACKNOWLEDGEMENT

First, we thank the almighty god for the successful completion of the project. Our sincere thanks to our Chairman **Mr. S. Meganathan, B.E., F.I.E.,** for his sincere endeavours in educating us in his premier institution. We would like to express our deep gratitude to our beloved Chairperson **Dr. Thangam Meganathan, Ph.D.,** for her enthusiastic motivation which inspired us a lot in completing this project. We also thank our Vice Chairman **Mr. Abhay Shankar Meganathan, B.E., M.S.,** for providing us with the requisite infrastructure.

We also express our sincere gratitude to our beloved college Principal, **Dr. S. N. Murugesan M.E., PhD.,** and **Dr. P. Revathy M.E., Ph.D., Professor and Head, Department of Computer Science and Engineering** and our project guide **Mrs. S. Ponmani, M.E.,** for her guidance throughout our project. We thank our project coordinator **Dr. S. Vinodh Kumar, M.Tech., Ph.D.,** for his encouragement and guidance towards successful completion of this project and to our parents, friends, all faculty members and supporting staff for their direct and indirect involvement in successful completion of the project for their encouragement and support.

AKASH A

CHRIS GAVIN B

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

Youtube uses various methods in advertising. One is youtube auto-generated advertisement in which youtube recognizes the top trending videos in a particular region and then it randomly assigns advertisements to the particular video. The next type of advertisement is the promoter's advertisement. Promoters are none other than YouTubers who promote advertisements in their videos. They may promote the advertisement at any duration within the videos. These promoters receive advertisements from advertisers directly. The advertisers give their ads to the YouTuber who are successful and are familiar with them at the time. But these types of advertisements are not effective. They have very little impact on the public. Youtube has a very low advertisement conversion ratio. This is because the advertisements are not promoted by the right YouTubers. When a food Vlogger does an ad on a gaming application, the impact of the advertisement will be less. This is because the subscribers would have subscribed to his youtube channel only to know about food. When a YouTuber tells something beyond his content or is irrelevant to his/her content, the public tends to be less interested in it. This loss of interest will affect both the YouTuber as well as the advertiser. When youtube advertises an irrelevant advertisement the viewers and subscribers tend to skip the advertisement as they don't connect to it. But when the same food vlogger gives an advertisement about a hotel people tend to visit the hotel as the food vlogger will have an influence on their subscribers and as he is a food vlogger who does daily videos on food the viewers will also have more trust in the advertisement this will increase the advertisement conversion ratio. So the basic idea is to connect advertisers with the promoters whose content is relevant to the content of the advertisement. When advertisers choose the promoters relevant to the content of their advertisement the impact of the advertisement will be more. This will in turn increase the advertisement conversion ratio and will increase the profit of the advertiser.

1.1 Text Classification with Machine learning

The role of automated text classification is to classify documents into predetermined categories, typically applying machine learning algorithms. Generally speaking, organizing and using the huge amounts of information, which exist in unstructured text format, is one of the most important techniques. Classification of text is a widely studied field of language processing and text mining study. A document is represented in traditional text classification as a bag of words in which the world's terms are cut from their finer context, that is, their location in a sentence or a document. Only the wider document context is utilized in the vector space with some type of term frequency information. Hence, the semantics of words, which can be derived in a sentence from the finer sense of the world's location and its relationships with neighboring words, are generally ignored. Nonetheless, the meaning of words and semantic relations between words and documents are essential as methods, which capture semantics, generally achieve better performance in classification. Due to the wide range of sources that generate enormous amounts of data, such as social networks, blogs/forums, websites, e-mails, and digital libraries that publish research papers, text mining studies have become increasingly important in recent years. With new technological advances, such as speech-to-text engines and digital assistants or smart personal assistants, the growth of electronic textual information will no doubt keep increasing. A fundamental problem is the automatic processing, organization, and handling of this textual data. Text mining has several important applications such as classification, filtering of documents, summarization, and sentiment analysis/opinion classification. Machine learning and natural language processing (NLP) techniques work together to detect and automatically classify patterns from different types of documents.

1.2 Region recognition

There are more than 121 different languages spoken in different states around India. Our project focuses on four regions such as Tamil, Telugu, Hindi, and Malayalam. The technology of region recognition aims to search and extract the state of origin as well as language belongs too for efficient classification. A region recognition system is

implemented to recognize the target audiences who only watch those region videos. So by recognizing, advertisers can publish their advertisements according to their regions.

1.3 Stochastic Gradient Descent

Stochastic Gradient Descent (SGD) is a simple yet very efficient approach to fitting linear classifiers and regressors under convex loss functions such as (linear) Support Vector Machines and Logistic Regression. Even though SGD has been around in the machine learning community for a long time, it has received a considerable amount of attention just recently in the context of large-scale learning. SGD has been successfully applied to large-scale and sparse machine learning problems often encountered in text classification and natural language processing. Given that the data is sparse, the classifiers in this module easily scale to problems with more than 10^5 training examples and more than 10^5 features.

1.3.1 Stochastic Gradient Descent Classification

The class SGD Classifier implements a plain stochastic gradient descent learning routine that supports different loss functions and penalties for classification. Below is the decision boundary of an SGD Classifier trained with the hinge loss, equivalent to a linear SVM. SGD Classifier supports multi-class classification by combining multiple binary classifiers in a “one versus all” (OVA) scheme. For each of the K classes, a binary classifier is learned that discriminates between that and all other K - 1 classes. At testing time, we compute the confidence score (i.e. the signed distances to the hyperplane) for each classifier and choose the class with the highest confidence. The class SGD Classifier implements a first-order SGD learning routine. The algorithm iterates over the training examples and for each example updates the model parameters according to the update rule given by $\omega \leftarrow \omega - \eta \left[\alpha \frac{\partial R(\omega)}{\partial \omega} + \frac{\partial R(\omega^T x_i + b, y_i)}{\partial \omega}\right]$ where n is the learning rate that controls the step size in the parameter space. The intercept is updated similarly but without regularization (and with additional decay for sparse matrices). The learning rate can be either constant or gradually decaying. For classification, the default learning rate schedule is given by $\eta^{(t)} = \frac{1}{\alpha(t+t_0)}$ where t is the time step (there are a total of n_samples * n_iter time steps), t0 is determined based

on a heuristic proposed by Léon Bottou such that the expected initial updates are comparable with the expected size of the weights (this assuming that the norm of the training samples is approx. The exact definition can be found in _init_t in Base SGD. For regression, the default learning rate schedule is inverse scaling, given by $\eta^{(t)} = \frac{eta_0}{t^{-t}}$ where eta0 and power_t are hyperparameters chosen by the user via eta0 and power_t, resp. For a constant learning rate use learning_rate='constant' and use eta0 to specify the learning rate. For an adaptively decreasing learning rate, use learning_rate='adaptive' and use eta0 to specify the starting learning rate. When the stopping criterion is reached, the learning rate is divided by 5, and the algorithm does not stop. The algorithm stops when the learning rate goes below 1e-6. The model parameters can be accessed through the coef_ and intercept_ attributes: coef_ holds the weights w and intercept_ holds b. When using Averaged SGD (with the average parameter), coef_ is set to the average weight across all updates: coef_= $\frac{1}{T}\sum_{t=0}^{T-1}\omega^{(t)}$, where T is the total number of updates, found in the t_ attribute.

1.4 TF-IDF Vectorization

Term frequency-inverse document frequency (TF-IDF) is a text vectorizer that transforms the text into a usable vector. It combines 2 concepts, Term Frequency (TF) and Document Frequency (DF). The term frequency is the number of occurrences of a specific term in a document. Term frequency indicates how important a specific term is in a document. Term frequency represents every text from the data as a matrix whose rows are the number of documents and columns are the number of distinct terms throughout all documents. Document frequency is the number of documents containing a specific term. Document frequency indicates how common the term is. Inverse document frequency (IDF) is the weight of a term, it aims to reduce the weight of a term if the term's occurrences are scattered throughout all the documents. IDF can be calculated as $idf_i = \log(\frac{n}{df_i})$. Where idf_i is the IDF score for term i, df_i is the number of documents containing term i, and n is the total number of documents. The higher the DF of a term, the lower the IDF for the term. When the number of DF is equal to n which means that the term appears in all documents,

the IDF will be zero, since log(1) is zero, when in doubt just put this term in the stopword list because it doesn't provide much information. The TF-IDF score as the name suggests is just a multiplication of the term frequency matrix with its IDF, it can be calculated as $w_{i,j} = tf_{i,j} \times idf_i$. Where w_{ij} is the TF-IDF score for term i in document j, tf_{ij} is the term frequency for term i in document j, and idf_i is the IDF score for term i.

1.5 Objective

Paid promoter's recommendation system must be able to predict the category of the youtube channel content and the region with the help of text classification so that it will be very helpful to advertisers by choosing the promoter to promote their advertisement to the right audiences. And this system provides its user to select a target in comparison mode and is also region-based.

CHAPTER 2

LITERATURE SURVEY

[2.1] SURVEY

[1] A STUDY OF "FORCED-AD RESISTANCE" LEADING TO "SKIP AD" ON YOUTUBE

Author: Abdelkader OA

Publication Year: 2021

This paper presents two concepts aiming at understanding YouTube users' behavior and maximizing ad effectiveness by exploring; "forced-ad resistance" (FAR) and "ad-characteristics resistance" (ACR) factors. The current study aims to discuss to what extent ad-skipping behavior is affected by the influence of each one of these two factors and to explore the significant impact of four demographic characteristics: age, gender, education, and income.

ADVANTAGES

Current study provides a major contribution by introducing the two terms of FAR and ACR and examining their impact on Youtube Ad skipping, which is expected to lead to a better understanding of ad-skipping on the YouTube platform.

DISADVANTAGES

This paper just focuses on YouTube ad-skipping and does not discuss all other influences on users' behavior.

[2] APPLYING AHP TO COMPREHEND THE FACTORS INFLUENCING CONSUMER ATTENTION TO SPONSORED ADS BY YOUTUBERS

Author: Chang JY

Year : 2018

This paper has presented a two-stage integrated hierarchical structure of evaluation for analyzing the evaluation factors of user attention to a sponsored ad by a YouTuber.

This model uses the AHP method to analyze the key factors that influence the success of sponsored ads by YouTubers in attracting users.

ADVANTAGES

This model has used a two-stage integrated hierarchical structure of evaluation for analyzing the evaluation factors of user attention to a sponsored ad by a YouTuber.

DISADVANTAGES

This study is solely based on the analytic hierarchy process method.

[3] EVALUATING THE INFLUENCE OF YOUTUBE ADVERTISING FOR ATTRACTION OF YOUNG CUSTOMERS

Author: Dehghani M, Niaki MK, Ramezani I, and Sali R

Publication Year: 2016

This paper identifies four dimensions of YouTube advertising (i.e., entertainment, Informativeness, Customization, and irritation) that may be affected by advertising value as well as brand awareness, and accordingly on purchase intention of consumers. The conceptual model hypotheses that ad value strategies are positively associated with brand awareness, which in turn influences the perceived usefulness of YouTube and continued purchase behavior.

ADVANTAGES

The effectiveness of advertising on YouTube in ad value has been evaluated with four hypotheses.

DISADVANTAGES:

This research was only based on a small population.

[4] How YouTube Developed into a Successful Platform for User-Generated Content

Author: Holland M

Publication Year: 2016

This study analyzed elements within the videos of three prominent YouTubers and the structure of their channel. They were selected based on Lavaveshkul's (2012) study, which analyzed the top 10 most subscribed to YouTube channels of 2012. These 10

channels could be divided into three categories of gaming, comedy, or how-to. The current study selected one channel from each category based on their popularity on YouTube. The three YouTubers were Felix Kjellberg (gaming), Grace Helbig (comedy), and Zoe Sugg (how to).

ADVANTAGES

The wide variety of content makes YouTube a place where just about anyone can find a video that interests them.

DISADVANTAGES

This section focused on dealing only with the three research questions.

[5] EMOTIONS AS PREDICTOR FOR CONSUMER ENGAGEMENT IN YOUTUBE ADVERTISEMENT

Author: Kujur F, Singh S

Publication Year: 2018

In this paper aim at measuring the ripple effect of the emotional appeals on ads and also try to compare the impact between positive and negative emotional appeals so that it becomes easy for the marketers to determine the context in which it can be applied.

ADVANTAGES

The study compares the effectiveness of both positive and negative emotions on consumer engagement.

DISADVANTAGES

The present study only focuses on the emotional aspects of the advertisement.

[6] A Framework for Collecting YouTube Meta-Data

Author: Malik H, Tian Z

Publication Year: 2017

This paper presents a methodology to fill the gap, i.e., systematically and continuously mine and store the YouTube data. The methodology has two modules, a video discovery and a video meta-data collection.

ADVANTAGES

All the previous research, studies, and analysis so far, are only conducted on a very small volume of YouTube video data. Our model evaluated on large volume.

DISADVANTAGES

Since the proposed framework relies on the YouTube/Google proprietor API for mining the data, any change in their API will necessitate modifying our code to parse the Json Object.

[7] YOUTUBE VLOGGERS AS BRAND INFLUENCERS ON CONSUMER PURCHASE BEHAVIOUR

Author: Rybaczewska M, Jebet Chesire B, Sparks L

Publication Year: 2020

This study involves a mixed-method approach (often connected with netnography) that incorporated non-participant observation of vloggers' activities and vloggers-viewers interactions within selected popular vlogs, supported by an online survey with both vloggers and viewers.

ADVANTAGES

This study shows how vloggers influence the modern business market.

DISADVANTAGES

This study analyzes only very few vlogs. Viewer-viewer interaction on the vlogs platforms was underexplored.

[8] ADVERTISEMENT RECOMMENDATION ENGINE IMPROVING YOUTUBE ADVERTISEMENT SERVICES

Author: Shanmuga Skandh Vinayak E, Venkatanath A G S, Shahina A, Nayeemulla Khan A

Publication Year: 2020

This paper recommends engines are developed and compared with each other, to determine the efficiency and the user specificity of each engine. From the experiments and user-based testing conducted, it is observed that the engine that recommends

advertisements utilizing the objects and the texts recognized, along with the video watch history, performs the best, by recommending the most relevant advertisements in 90% of the testing scenario.

ADVANTAGES

This paper proposes an improved solution to the conventional YouTube advertisement recommendation by utilizing the features of the videos watched by the user such as the objects, texts present in the video frames, and the description data of the user's watch history.

DISADVANTAGES

The YOLO algorithm utilizes sophisticated hardware components such as high-end Graphics Processing Units (GPU) and computing components such as Central Processing Units (CPU) and memory units that support such GPUs with high-end configurations. These types of systems are highly inefficient in an enterprise-level solution that provides video streaming services to a significantly large user base

[9] YouTube in Teaching Activities

Author: Shizhen Jia

Publication Year: 2019

This paper conducts a literature review of journal articles and conference proceedings to understand the benefits and limitations of using YouTube in teaching and how educators apply this technology in their teaching activities. A search of various databases including ACM Digital Library, Springer and ScienceDirect identified 30 unique articles on this topic. We analyzed these articles using a taxonomy for teaching. This literature review should be of interest to educators who want to apply YouTube in their teaching activity.

ADVANTAGES

This paper proposes an improved solution to the articles and conferences recommendation by utilizing the features of literature survey on each document.

DISADVANTAGES

Limits the paper context compared to conferences and articles.

[10] YouTube across the Disciplines: A Review of the Literature [10]

Author: Snelson C

Publication Year: 2011

This paper presents about YouTube in online education is that it provides online access to vast quantities of free public video on a broad spectrum of topics. It is a simple matter to link to or embed YouTube videos in online course content or discussion forums. Content management is also a benefit. Online educators can establish YouTube channels to collect, organize, host, and distribute video.

ADVANTAGES

This model is promoting various ways to educate on youtube across regions.

DISADVANTAGES

Included regions are only available to explore.

[11] FIXATION AND RECALL OF YOUTUBE AD BANNERS: AN EYE-TRACKING STUDY

Author: Tangmanee C

Publication Year: 2016

This study investigates whether YouTube visitors fixate on ad banners, what the correlations between fixation duration on banners and overall fixation counts are, and the extent to which site visitors can recall details of ad banners and the clip viewed. Using a Miramatrix eye-tracker to record YouTube viewers' eye movements, this study showed that nearly all fixated at least once on an ad banner in a clip.

ADVANTAGES

The findings of this study have both theoretical and practical implications.

DISADVANTAGES

Context of Internet use is constantly changing. Data was collected in a captive environment.

[12] Analysis of YouTube of Videos

Author: Vadde, Neha Reddy, Piyush Gupta, Prasham Mehta, Puneet Gupta, Vikranth BM

Publication Year: 2020

This paper presents a method to train a model to classify a video as Clickbait video or non-Clickbait video. Consumption of content from YouTube and other OTT(over-the-top) platforms is constantly increasing. YouTube 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.

ADVANTAGES

Uses sentiment analysis on viewer comments to identify a video as clickbait or not.

DISADVANTAGES

Works only with YouTube data that consists of viewer comments

[13] MULTIMODAL CONTENT ANALYSIS FOR EFFECTIVE ADVERTISEMENTS ON YOUTUBE

Author: Vedula N, Sun W, Lee H, Gupta H, Ogihara M, Johnson J, Ren G, and Parthasarathy S

Publication Year: 2017

This paper has implemented a computational framework for the predictive analysis of the content-based features extracted from advertisement video files and various effectiveness metrics to aid the design and production processes of commercial advertisements. The proposed predictive analysis framework extracts multi-dimensional temporal patterns from the content of advertisement videos using multimedia signal processing and natural language processing tools.

ADVANTAGES

Compared to our other models, the multimodal LSTM model achieved the best accuracy and an F1 score greater than 0.8, and the difference in accuracy is significant.

DISADVANTAGES

The finding that the temporal location of brand mention is irrelevant is not supported by marketing literature.

[14] TO STUDY THE IMPACT OF YOUTUBE TECH INFLUENCERS ON THE CONSUMER BUYING BEHAVIOR OF ELECTRONIC GADGETS

Author: Veluchamy R, Sans RK, Rajagopal P

Publication Year: 2021

This study uses a quantitative online survey research methodology to better evaluate the effects of YouTube tech influencers on customer purchase intent. Five quantitative scales made up the survey instrument.

ADVANTAGES

This study provides a better understanding of how YouTubers influence consumer buying behavior.

DISADVANTAGES

This study was only based on a small population.

[15] THE IMPACT OF VIEWERS' BEHAVIOR AND YOUTUBERS' CREDIBILITY IN ADVERTAINMENT ON BUILDING CONSUMER TRUST

Author: Wickramasinghe S.P, Welgama S.D, Rajapakse RP, Jayasuriya N, and Munasinghe AA

Publication Year: 2021

In this paper the proposed framework consists of two variables in YouTuber's advertisement to measure the impact on building consumer trust. These two variables of YouTuber's advertisement have been identified as having a direct relationship to consumer trust.

ADVANTAGES

This study provides an understanding of how the factors of advertisement have facilitated and influenced building consumer trust.

DISADVANTAGES

This study was only based on a small population.

2.2 EXISTING SYSTEM

The existing system was only a youtube auto-generated recommendation system. These advertisements were irrelevant and had very less impact. These advertisements were assigned randomly irrelevant to the content of the YouTuber. The process in which youtube's auto-generated recommendation system works is, youtube will generate top trending videos daily in a particular region. After the generation of top trending, it will assign advertisements randomly. These advertisements are assigned randomly and do not connect well with the audience. The advertisement conversion ratio gets affected by this which in turn is a loss for the advertisers.

2.2.1 DRAWBACKS OF EXISTING SYSTEM

- Relevancy - The Advertisement which was auto-generated by youtube will not be relevant to the content of a particular YouTuber. This will have very little impact on the viewers which will, in turn, affect the advertisement conversion ratio.

- Less Advertisement Conversion Ratio - The content of the Advertisement will not be similar to the content of the particular YouTuber, this will impact the advertisement conversion ratio.

2.3 PROPOSED SYSTEM

The proposed system is based on our text classification trained model which classifies the category of the youtube channel. Then the dataset is made out of these classifications and categorized by Indian regions. This dataset contains a group of id's categorized based on various advertisement genres. These datasets are used to fetch the channel details from youtube API and store it in the cloud. The system is capable of monitoring the status of the channel every 24 hours to keep the data up to date.

2.3.1 ADVANTAGES OF THE PROPOSED SYSTEM

- Our model can identify 4 regions.
- Model has an accuracy of 86%.
- Model takes an average of 5 seconds to recognize the category and region.
- Current status updated every 24 hours

CHAPTER 3

SYSTEM DESIGN

3.1 GENERAL

Create a dataset that contains youtube channel IDs categorized by their content genre. Then we save this dataset in CSV file format. We are accessing this dataset using python for uploading data to the cloud(Firebase). Then we use web scraping(because youtube API did not provide the required information for the project such as contact, statistics, etc) to get contact information from all youtube channels by requesting channels' URLs to obtain HTML content. We parse all the data in the HTML content with meta and description keywords. Then we use regex to check any mail id in the particular content and store it in the cloud(Firebase). The channel id which didn't find any mail id after web scraping will be considered as a channel that is not interested in ad collaborations. So, these data are stored in separate columns in a cloud. Now we use the youtube ids stored in the cloud to request data(subscriber count, latest video view count, total videos posted, last active date, total views of the channel) from youtube API which fetches all the above details from all the channels. The channel details will be updated every 24 hours so that the youtube channel's current status will be updated to provide accurate information. We'll display all the information in the cloud to the advertisers using a web portal that is developed in react js. Here the information is sorted in an orderly manner on the basis of the approximate view count rate which will collect the latest 15 videos of every youtube channel. After collecting the latest 15 videos we will remove the first 5 videos as these videos only would have been released recently and will only have a very less view count. Then we'll consider the next 10 videos, these videos though may not have been watched by all youtube viewers they will be watched by the subscribers. Now, we will take the average view count of these 10 videos and consider them as approximate view count rates. We'll display a linear graph that shows the latest video's view count.

3.2 SYSTEM ARCHITECTURE DIAGRAM

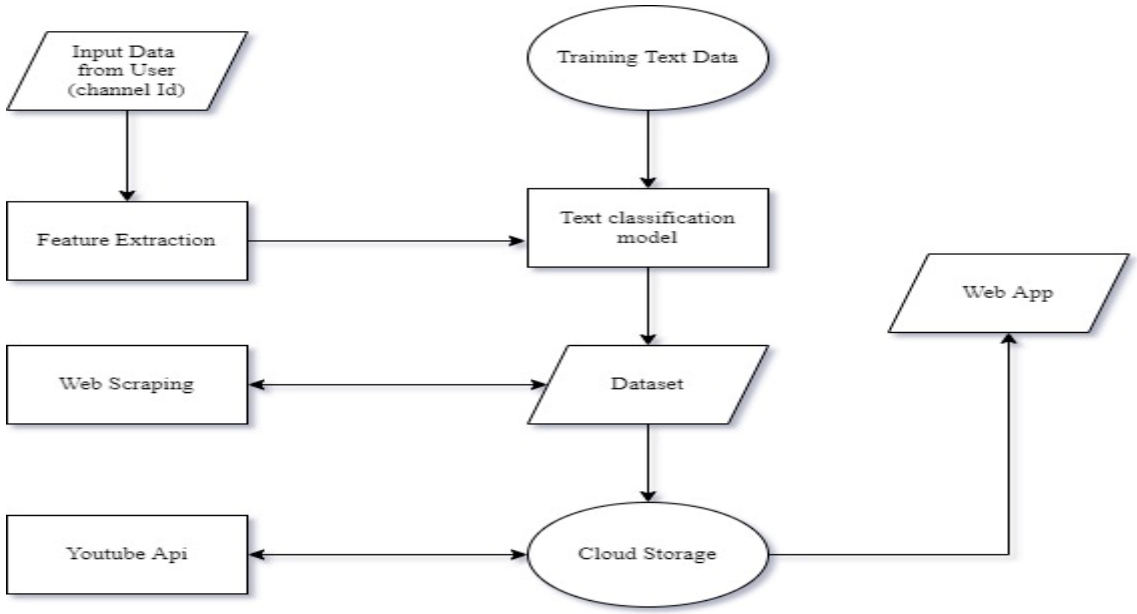


Fig 3.1 System Architecture

The system architecture diagram as shown in Fig 3.1 deals with the flow of the dataset from uploading it to the cloud where channel details are fetched through youtube API every 24 hours and displayed through the web application.

3.3 DEVELOPMENT ENVIRONMENT

3.3.1 HARDWARE REQUIREMENTS

The hardware requirements may serve as the basis for a contract for the implementation of the system and should therefore be a complete and consistent specification of the whole system. They are used by software engineers as the starting point for the system design. It shows what the system does and not how it should be implemented.

Table 3.1 Hardware Requirements

COMPONENT	SPECIFICATION
PROCESSOR	Intel Core i5
RAM	8 GB DDR4 RAM
GPU	NVIDIA MX 130
MONITOR	15” COLOR
HARD DISK	10 GB
PROCESSOR SPEED	MINIMUM 500MHZ

3.3.2 SOFTWARE REQUIREMENTS

The software requirements document is the specification of the system. It should include both a definition and a specification of requirements. It is a set of what the system should do rather than how it should do it. The software requirements provide a basis for creating the software requirements specification. It is useful in estimating the cost, planning team activities, performing tasks, tracking the team, and tracking the team’s progress throughout the development activity.

- React JS
- Python 3.8
- JavaScript
- Cloud Firebase
- Youtube API

3.4 DESIGN OF THE ENTIRE SYSTEM

3.4.1 USE CASE DIAGRAM

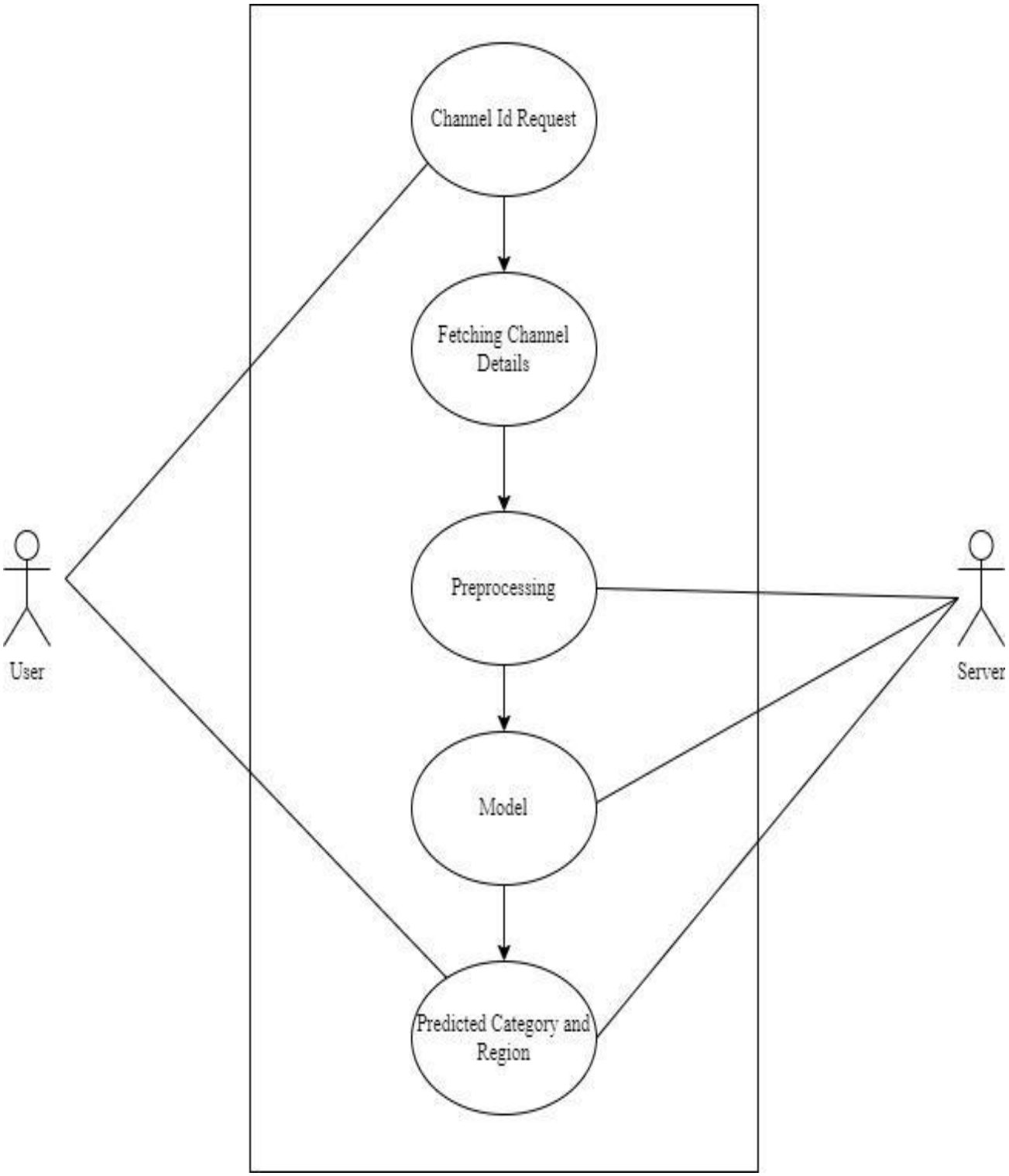


Fig 3.2 Use Case Diagram

The client/user feeds the channel Id into the system and the request is sent to the server after the pre-processing steps. The preprocessed text is then sent to the model and it recognizes and predicts the category and region value which is then displayed in the client's application as shown in Fig 3.2.

3.4.2 ACTIVITY DIAGRAM

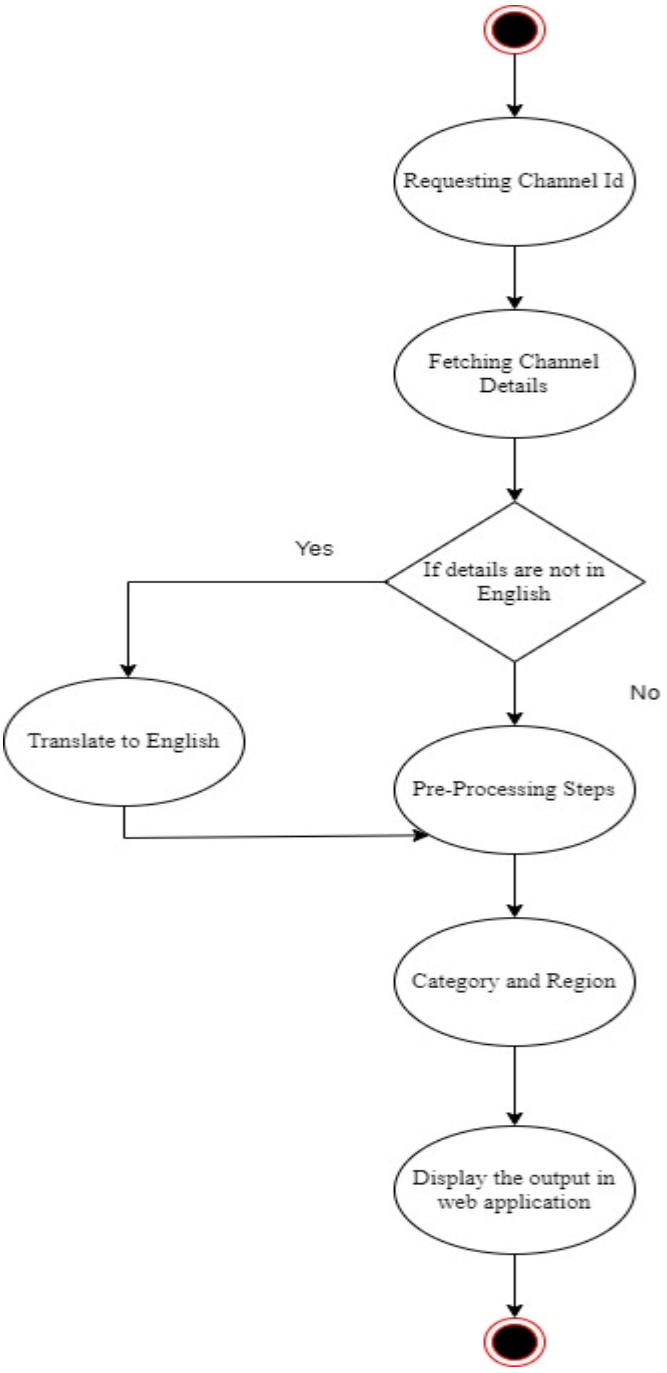


Fig 3.3 Activity Diagram

The flow of activity starts from requesting the youtube channel Id in the request form of the web application. The requested youtube channel Id is then uploaded to the firebase cloud server and extracted to the admin system where the preprocessing steps takes place. With the extracted youtube channel Id requests, the system performs web scraping to gain the information of all particular channel details such as contacts, view count, etc... Fetched channel details are then given to the model with prerequisite cleaning to predict the category and the region of the requested youtube channel Id, which is then arranged by region and uploaded to cloud by the admin and displayed in the web application as shown Fig 3.3.

CHAPTER 4

PROJECT DESCRIPTION

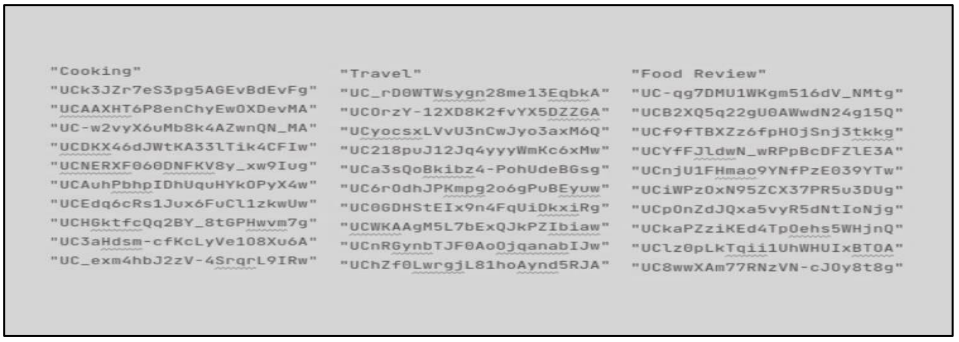
4.1 MODULES

- Dataset Creation
- Web Scraping
- Fetching youtube channel details
- Cloud Storage
- Web Application

4.2 MODULES DESCRIPTIONS

4.2.1 DATASET CREATION

4.2.1.1 Gathering Data:



"Cooking"	"Travel"	"Food Review"
"UCk3JZr7eS3pg5AGEvBdEvFg"	"UC_rD0WTWsygn28me13EqbkA"	"UC-qg7DMU1WKgm516dV_NMtg"
"UCAAXHT6P8enChyEwOXDevMA"	"UCOrzY-12XD8K2fvYX5DZZGA"	"UCB2XQ5q22gU0AWwdN24g15Q"
"UC-w2vyX6uMb8k4AZwnQN_MA"	"UCyocsxLVvU3nCwJyo3axM6Q"	"UCf9fTBXZz6fpHOjSnj3tkkg"
"UCDKX46dJWtKA33lTik4CFIw"	"UC218puJ12Jq4yyyWmKc6xMw"	"UCYfFJldwN_wRPpBcDFZlE3A"
"UCNERXF060DNFKV8y_xw9Iug"	"UCa3sQoBkibz4-PohUdeBGsg"	"UCnjU1FHmao9YNfPzE039YTw"
"UCAuhPbhpIDhUquHYkOPyX4w"	"UC6rOdhJPKmpg2o6gPuBEyuw"	"UCiWPzOxN95ZCX37PR5u3DUg"
"UCEdq6cRs1Jux6FuCl1zkwUw"	"UC0GDHStEIx9n4FqUiDkxiRg"	"UCpOnZdJQxa5vyR5dNtIoNjg"
"UCHGktfcQq2BY_8tGPHwvm7g"	"UCWKAAgM5L7bExQJkPZIbiaw"	"UCkaPZziKEd4TpOehs5WHjnQ"
"UC3aHdsm-cfKcLyVe1O8Xu6A"	"UCnRGynbTJF0AoOjqanabIJw"	"UClz0pLkTqii1UhWHUIxBTOA"
"UC_exm4hbJ2zV-4SrqrL9IRw"	"UChZf0LwrgjL81hoAynd5RJA"	"UC8wwXAm77RNzVN-cJOy8t8g"

Fig 4.1 Snapshot of Dataset

A Comma separated values(CSV) file that contains a list of youtube channel IDs from various regions and their respective categories is created manually for training purposes as shown in Fig 4.1. It is used to fetch information such as channel title and description through web scraping. After fetching data, we use the CSV format to store that information.

4.2.1.2 Data Cleaning and Preprocessing:

```
,Channel_Id,Clean_text,Category
0,UCk3JZr7eS3pg5AGEvBdEvFg,village cooking village grandpa cooking
traditional village food country foods tasty recipes foodies
children villagers poor people village cooking entertains cooking
sharing foods,Cooking
1,UCAAXHT6P8enChyEwOXDevMA,chef deena kitchen chef deena dhayalan
famous adupankarai show jaya tv also anjaraipetti zee tv chef deena
kitchen cdk cooking traditional foods visiting traditional places
chef deena kitchen cdk cooking chef deena b link
chefdeenakitchen,Cooking
2,UC-w2vyX6uMb8k4AZwnQN_MA,amma samayal meenakshi home maker cooking
journey started years old since interest cooking increased cooking
years interest developed years cooking also cleaning making healthy
home remedies would recipes vlogs way maintain family experiences
thought mind presenting visit website ammasamayals com cubecreationz
com order amma samayal products wtsapp featured ammaveetusamayal mom
cooking mini chef available social media,Cooking
```

Fig 4.2 Data Cleaning and Preprocessing

Title and Description are unprocessed raw texts. Therefore, to filter out the noisiness, we'll follow certain approaches for cleaning the text:

- Converting to Lowercase: This step is performed because capitalization does not make a difference in the semantic importance of the word.
- Removing numerical values and punctuations: Numerical values and special characters used in punctuations($,! etc.).
- Removing extra white spaces: Such that each word is separated by a single white space, else there might be problems during tokenization.
- Tokenizing into words: This refers to splitting a text string into a list of 'tokens', where each token is a word. Removing non-alphabetical words and 'Stop words': 'Stop words' refer to words like and, the, is, etc, which are important words when learning how to construct sentences, but of no use to us for predictive analytics.
- Stemming: Stemming is a technique used to extract the base form of the words by removing affixes from them. It is just like cutting down the branches of a tree to its stems. For example, the words 'traveling' and 'traveler' will both be converted into their simplest meaning 'travel' as shown in Fig 4.2.
- Label Encoding: The 'Category' column in the dataset is an output variable, Thus we need to encode each class as a numerically based feature.

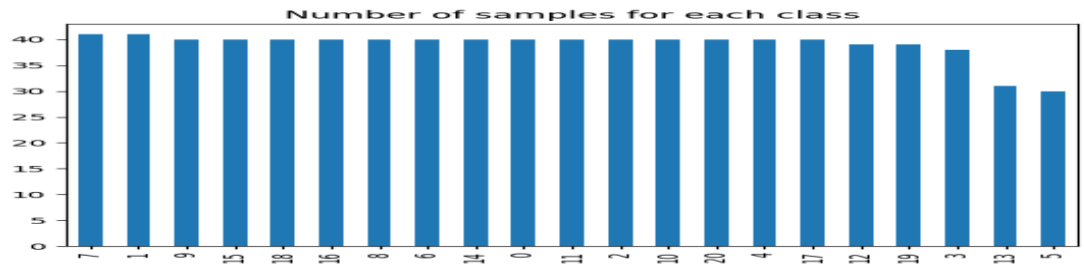


Fig 4.3 Number of samples for each class

• TF-IDF Vectorizer: To extract data from the above-processed text as features and represent them in a numerical format, a common approach is to vectorize them. TF IDF(Term Frequency-Inverse Document Frequency) calculates the frequency of each word inside and across multiple documents to identify the importance of each word as shown in Fig 4.3.

4.2.1.3 Modeling and Training:

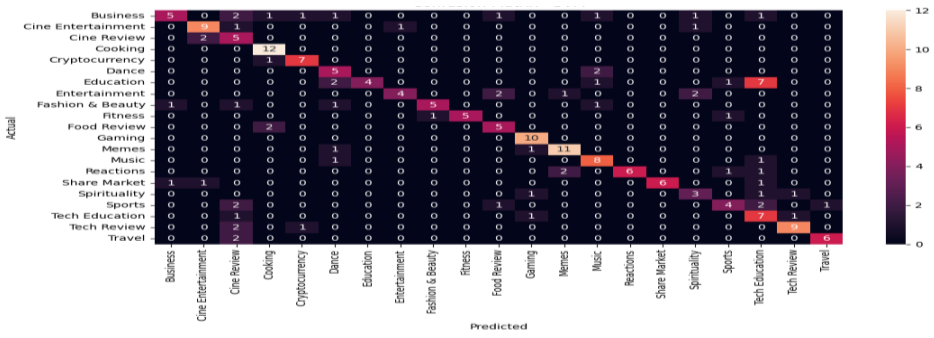


Fig 4.4 Confusion Matrix

The dataset is split into Train and Test sets with a split ratio of 8:2. Features for Title and Description are concatenated to construct a final feature matrix as shown in Fig 4.4. The Stochastic gradient descent(SGD) is an optimization algorithm to find the model parameters that correspond to the best fit between predicted and actual outputs and a highly robust classifier used to train the model.

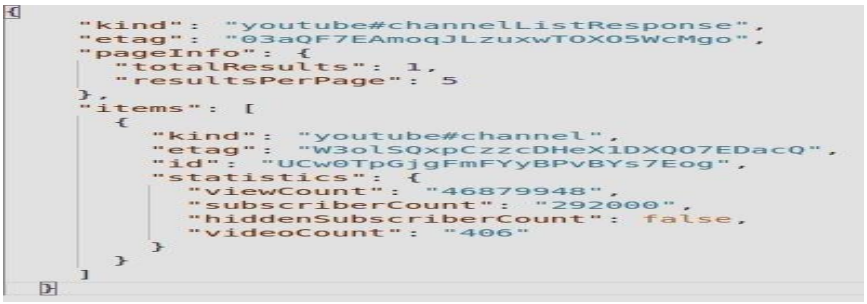
4.2.2 WEB DATA EXTRACTION

```
{
  "Cooking": [
    "UC-w2vyX6uMb8k4AZwnQN_MA | cubecreationz@gmail.com",
    "UCNERXF060DNFKV8y_xw9Iug | gomathichannel@gmail.com",
    "UCAuhPbhpIDhUquHYkOPyX4w | kavithasamayalarai.official@gmail.com"
  ],
  "Tech Reviewers": [
    "UCe_-TsRz3GH8UVjN0ApzXJQ | shan@techshan.com",
    "UCvyZS6W6zMJCZBVzF-Ei6sw | collab@a2dmediagroup.com",
    "UCZj7TPWYy-bxVoFukc3V2zQ | directrixreviews@gmail.com",
    "UCON_gtDQSNnWS1gwBcqgRKg | technologysatire@gmail.com",
    "UCkpgytHFxc69qJzxkukYyOw | loudolitech@gmail.com"
  ],
  "Travel": [
    "UC_rD0WTWsygn28me13EqbkA | vijaypc46@gmail.com",
    "UCOrzY-12XD8K2fvYX5DZZGA | explorewithbavin@gmail.com",
    "UC218puJ12Jq4yyyWmKc6xMw | chennaivloggerofficial@gmail.com",
    "UCnRGynbTJF0AoOjqanabIJw | way2gomadhavan@gmail.com"
  ]
}
```


Fig 4.5 Web scraped Contact Data

Web scraping is the process of extracting content and data from a website, web scraping extracts underlying HTML code. We use this technique with Python to extract contact details from their channel if the channel owner published one. An example of this is given in Fig 4.5.

4.2.3 FETCHING YOUTUBE CHANNEL DETAILS



```
{
    "kind": "youtube#channelListResponse",
    "etag": "03aQF7EAmoqJLzuxwTOXO5WcMgo",
    "pageInfo": {
      "totalResults": 1,
      "resultsPerPage": 5
    },
    "items": [
      {
        "kind": "youtube#channel",
        "etag": "W3olSQxpCzzcDHeX1DXQO7EDacQ",
        "id": "UCw0TpGjgFmFYyBPvBYs7Eog",
        "statistics": {
          "viewCount": "46879948",
          "subscriberCount": "292000",
          "hiddenSubscriberCount": false,
          "videoCount": "406"
        }
      }
    ]
}
```


Fig 4.6 Youtube API requested data

With the YouTube Data API, we request Advertisers focused information and receive responses in JSON format as given in Fig 4.6. Such as subscribers count, total view count of the channel, Total videos posted in a channel, Latest video uploaded, and View count for latest videos.

4.2.4 CLOUD DATABASE

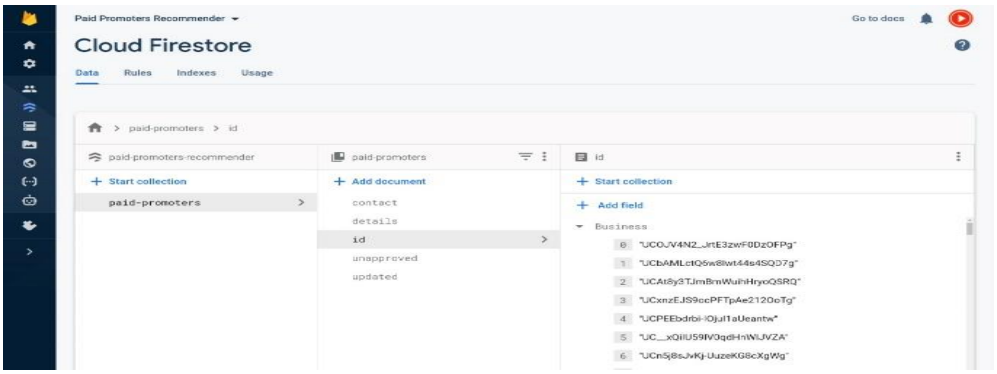


Fig 4.7 Snapshot of Cloud Storage

Firebase is a Backend-as-a-Service (Baas) from Google. The Firebase firestore database allows secure access to the database directly from client-side code. We use firebase to store our data such as Youtube Channel Ids, Channel details, and Contact details as shown in Fig 4.7.

4.2.5 WEB APPLICATION

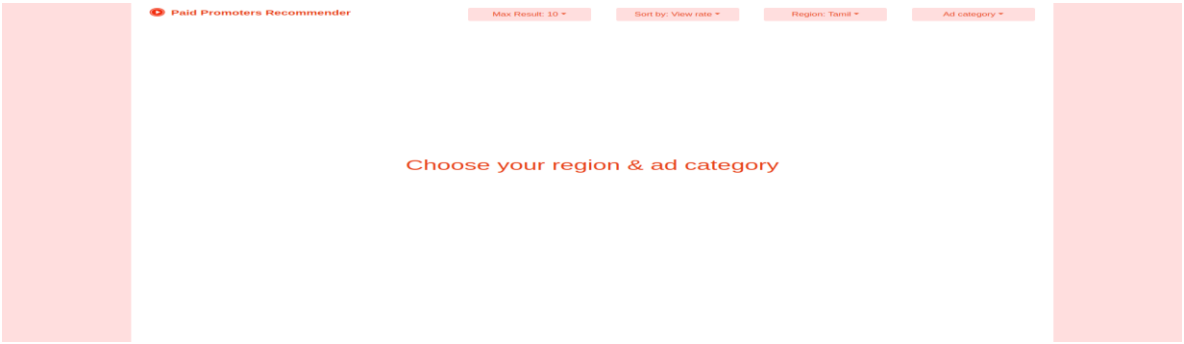


Fig 4.8 Snapshot of developed Web application

React-js is an open-source JavaScript library from meta that is used for building user interfaces specifically for single-page applications. We used reactjs to build our website to showcase the youtube channels and new YouTubers can also make a request to add their channel to our application as shown in Fig 4.8.

CHAPTER 5

RESULTS AND DISCUSSIONS

5.1 DATASET

To train a machine learning model, we need more data for a better performing model, hence we have to perform some of the vectorizing techniques such as TF-IDF Vectorizer. TF-IDF(Term Frequency-Inverse Document Frequency) calculates the frequency of each word inside and across multiple documents to identify the importance of each word. After implementing the above process, to check if the features extracted using TF-IDF vectorization made any sense, we decided to plot the most correlated unigrams for each class using both the Titles and the Description features.

Table 5.1 Most correlated unigrams for each class

Category	Unigrams	Category	Unigrams
Education	. Studies . Physics . Academy	Fitness	. Body . Weight . diet
Cooking	. traditional . recipe . kitchen	Spirituality	. god . spirituality . speeches
Cryptocurrency	. blockchain . coins . bitcoin	Tech Education	. doubts . programming . computer
Business	. money . machinery . small	Gaming	. streaming . games . gaming
Fashion & Beauty	. mens . grooming . lifestyle	Tech Review	. technology . unboxing . gadgets

Music	. piano . lessons . guitar	Travel	. countries . traveler . traveling
Cine Entertainment	. gossips . kollywood . celebrity	Sports	. players . coach . cricket
Entertainment	. digital . sketches . divo	Cine Review	. movie . hollywood . review
Dance	. studio . managed . dancing	Food Review	. eat . foodie . exploring

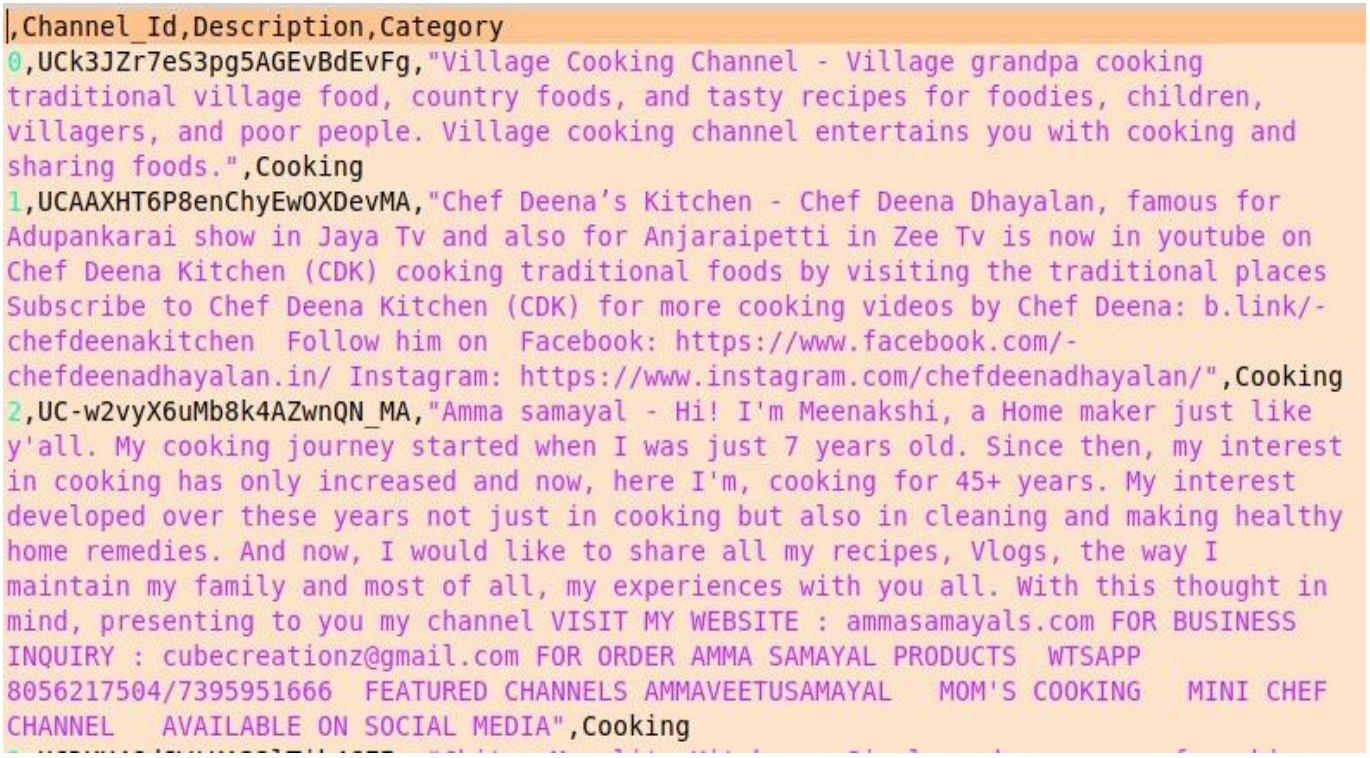
The dataset is split into Train and Test sets with a split ratio of 8:2. Features for Title and Description are concatenated to construct a final feature matrix. The Stochastic gradient descent(SGD) is being used to find the model parameters that correspond to the best fit between predicted and actual outputs.

Table 5.2 Precision and Recall summary

Category	Precision	Recall	Support
Business	0.81	0.66	14
Cine Entertainment	0.85	0.85	12
Cine Review	0.61	0.81	7
Cryptocurrency	0.85	1.00	2
Dance	0.88	0.88	8
Education	0.65	0.81	7
Entertainment	1.00	0.57	15
Fashion and Beauty	0.80	0.64	9
Fitness	0.83	0.56	9
Food Review	1.00	0.81	7
Gaming	0.56	0.81	7
Memes	0.87	1.00	10
Music	0.79	0.85	13

Reactions	0.62	0.80	10
Share Market	1.00	0.70	10
Spirituality	1.00	0.77	9
Sports	0.63	0.70	6
Tech Education	0.57	0.60	10
Tech Review	0.63	0.80	10
Travel	0.82	0.85	12
Macro Avg	0 .86	0.85	8
Weighted Avg	0.82	0.87	205

5.2 WEB SCRAPING



```
,Channel_Id,Description,Category
0,UCk3JZr7eS3pg5AGEvBdEvFg,"Village Cooking Channel - Village grandpa cooking
traditional village food, country foods, and tasty recipes for foodies, children,
villagers, and poor people. Village cooking channel entertains you with cooking and
sharing foods.",Cooking
1,UCAAXHT6P8enChyEwOXDevMA,"Chef Deena's Kitchen - Chef Deena Dhayalan, famous for
Adupankarai show in Jaya Tv and also for Anjaraipetti in Zee Tv is now in youtube on
Chef Deena Kitchen (CDK) cooking traditional foods by visiting the traditional places
Subscribe to Chef Deena Kitchen (CDK) for more cooking videos by Chef Deena: b.link/-
chefdeenakitchen  Follow him on  Facebook: https://www.facebook.com/-
chefdeenadhayalan.in/ Instagram: https://www.instagram.com/chefdeenadhayalan/",Cooking
2,UC-w2vyX6uMb8k4AZwnQN_MA,"Amma samayal - Hi! I'm Meenakshi, a Home maker just like
y'all. My cooking journey started when I was just 7 years old. Since then, my interest
in cooking has only increased and now, here I'm, cooking for 45+ years. My interest
developed over these years not just in cooking but also in cleaning and making healthy
home remedies. And now, I would like to share all my recipes, Vlogs, the way I
maintain my family and most of all, my experiences with you all. With this thought in
mind, presenting to you my channel VISIT MY WEBSITE : ammasamayals.com FOR BUSINESS
INQUIRY : cubecreationz@gmail.com FOR ORDER AMMA SAMAYAL PRODUCTS  WTSAPP
8056217504/7395951666  FEATURED CHANNELS AMMAVEETUSAMAYAL   MOM'S COOKING   MINI CHEF
CHANNEL   AVAILABLE ON SOCIAL MEDIA",Cooking
```

Fig 5.1 Web scraped Data

This technique is used with Python to extract title, description, and contact details from their channel if the channel owner published one as shown in Fig 5.1.

5.3 CLOUD STORAGE

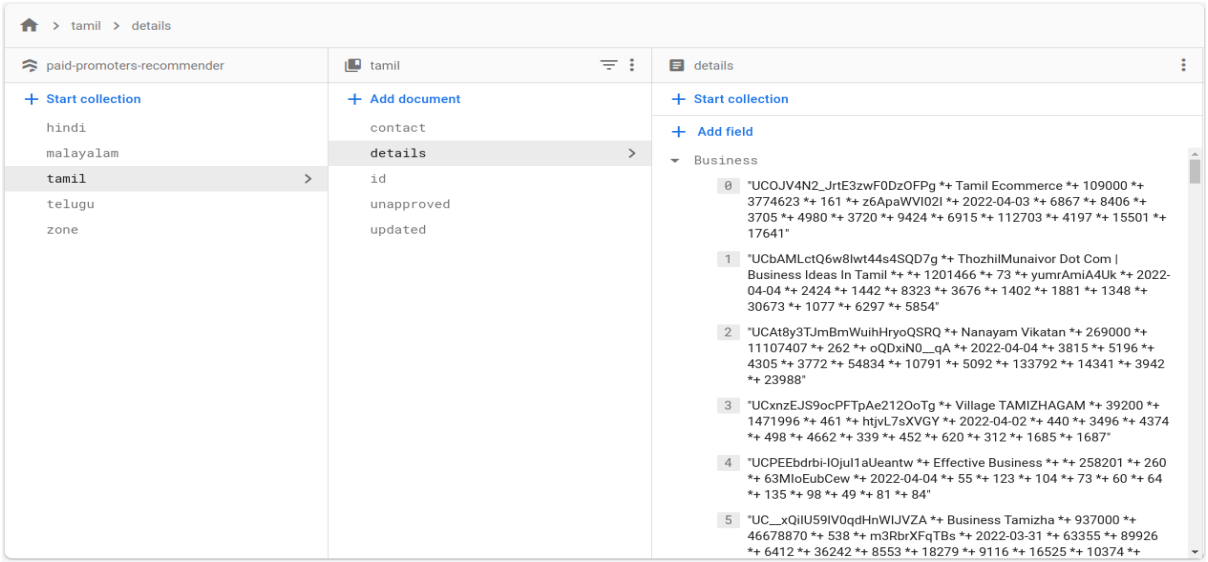


Fig 5.2 Data stored in the firebase

We used Cloud storage (firebase) to store and retrieve our data such as Youtube Channel Ids, Channel details, and Contact details, and also store the new channel request data as shown in Fig 5.2.

5.4 FINAL OUTPUT

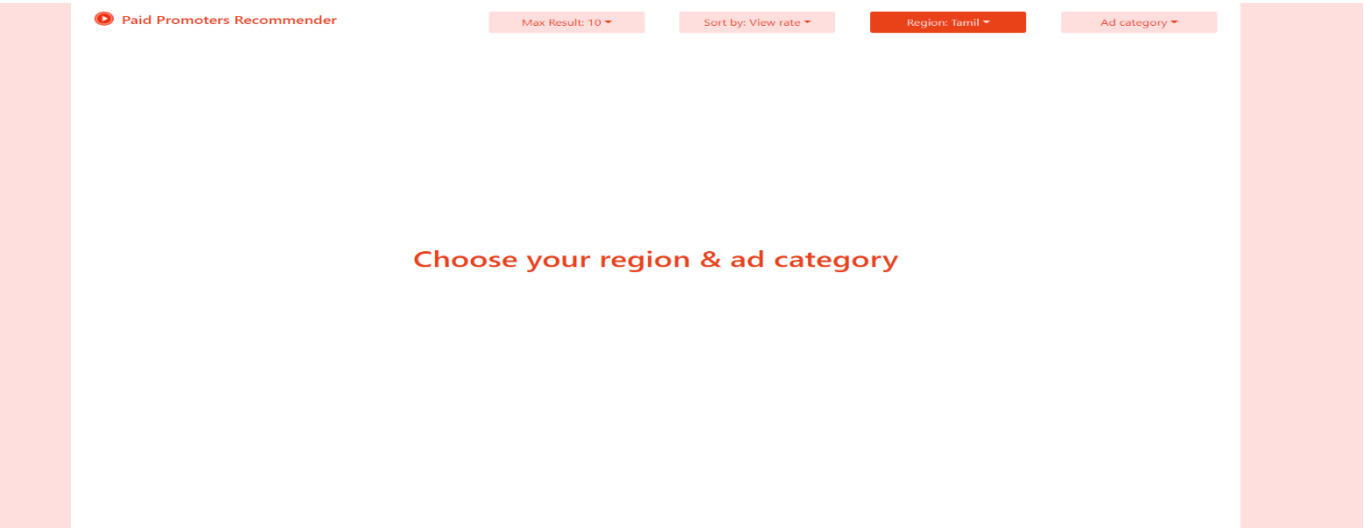


Fig 5.3 Output Image of the developed web application- Region

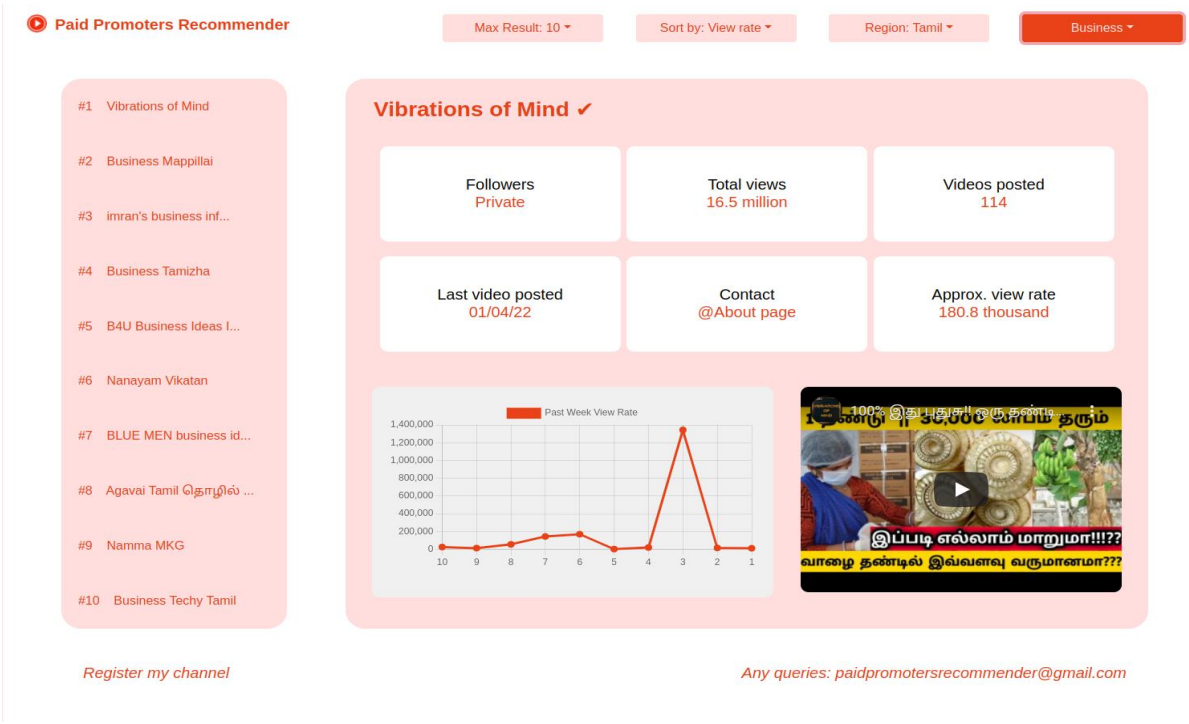


Fig 5.4 Output Image of the developed web application-Home

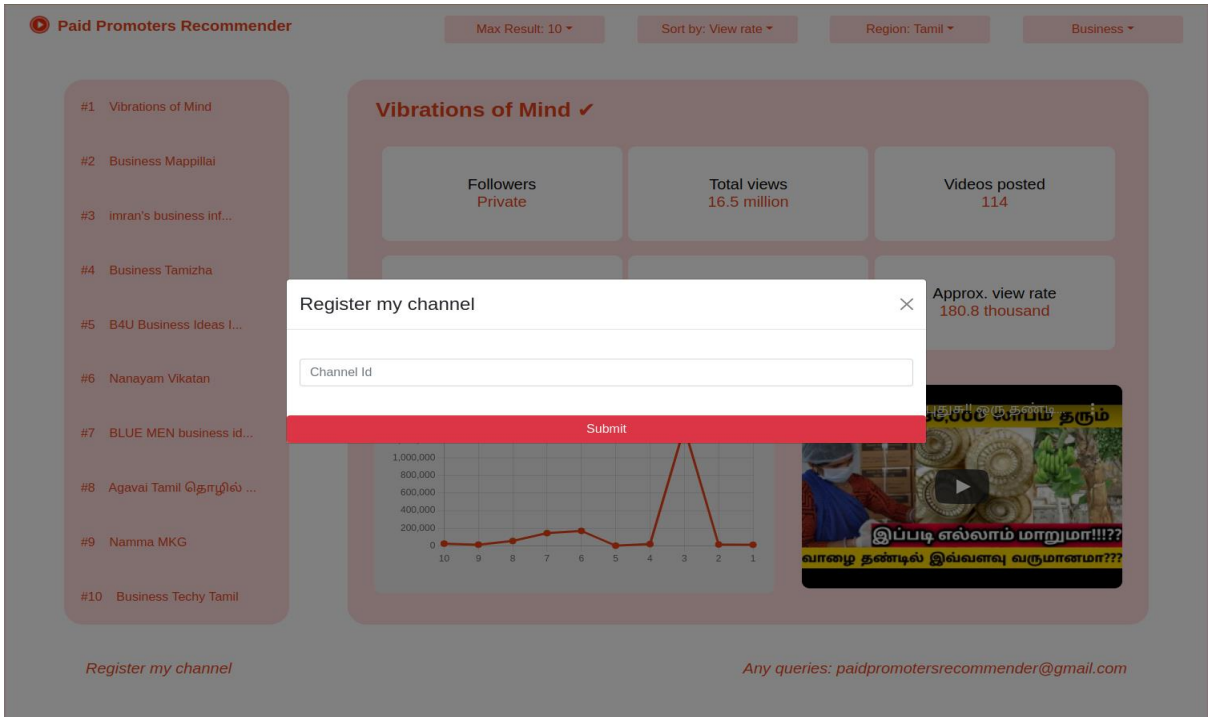


Fig 5.5 Output Image of the developed web application- Registration

The advertiser can choose the region in which the advertisement needs to be promoted as shown in Fig 5.3. Then the advertiser can choose the promoter according to the category of the advertisement as shown in Fig 5.4. After selecting the category of the advertisement, the list of promoters available will be displayed in an orderly manner. The contact information of every promoter will be displayed accordingly. There will be a linear graph that displays the latest view count rate of every youtube channel. Promoters can register their channel in the portal as shown in Fig 5.5. Based on the interest of the advertisers one can choose the promoter.

CHAPTER 6

CONCLUSION AND SCOPE FOR FUTURE WORK

6.1 CONCLUSION

This paper presents a framework for a recommendation system for advertisers in the field of youtube paid promoters. Web applications are developed for easy interaction with the current status of all youtube channels. This interactive portal will allow advertisers to compare and choose the best relevant paid promoter for their advertisements. A prototype system is implemented and various test scenarios were considered to validate the proposed system

6.2 FUTURE WORK

The current proposed system is available in 4 regions, so the future work lies in enlarging the dataset so that the website will be available in 22 regions recogonized by the constitution of India. The web portal needs to be improved so that the end-user will get a great user experience.

APPENDIX I

Main.py

```
from google.cloud import firestore
from bs4 import BeautifulSoup
import os, csv, requests, re, pandas as pd, matplotlib.pyplot as plt, pickle, numpy as np, scikitplot as skplt, seaborn as sns
from googletrans import Translator
from nltk.corpus import stopwords
from nltk.stem.porter import PorterStemmer
from sklearn.preprocessing import LabelEncoder
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.feature_selection import chi2
from sklearn.model_selection import train_test_split
from sklearn import linear_model, metrics
from sklearn.metrics import confusion_matrix
stop = stopwords.words('english')
translator= Translator()
os.environ['GOOGLE_APPLICATION_CREDENTIALS'] = "paid-promoters-recommender-firebase-adminsdk-vpjhv-c4c614db64.json"
db = firestore.Client()
fn = int(input("PPR developer side:\n1. Add new data\n2. Update & upload to cloud\n3. Create dataset model\n4. Extract new entry requests from cloud\nEnter a number to do an operation: "))
if fn == 1:
    add_new_id()
elif fn == 2:
    update_region()
elif fn == 3:
    add_new_id()
elif fn == 4:
    new_entry_cloud()
else:
    print("Please, Select a right operation!")
```

Dataset Creation

```
def create_dataset():
file = open(f'channel_id.csv')
csv_reader = csv.reader(file)
datas = {}
```

```
temp_data = []
classes_ex_id = []
classes_ex_cleantext = []
classes_ex_category = []
classes_id = []
classes_cleantext = []
classes_category = []
for i in csv_reader:
if len(i) == 0:
print(f"{temp_data[0]} phase completed --- Next phase on-process ->")
datas[temp_data[0]] = temp_data[1:]
temp_data = []
else:
try:
r = requests.get(f"https://www.youtube.com/channel/{i[0]}/about")
soup = BeautifulSoup(r.content, 'html.parser')
table = soup.find("meta", itemprop="description")["content"]
tt = soup.find("meta", itemprop="name")["content"] + " - " + table.replace("\n", " ")
classes_ex_id.append(i[0])
classes_ex_cleantext.append(tt)
classes_ex_category.append(temp_data[0])
translation = translator.translate(tt, dest="en")
text = translation.text.lower()
text = re.sub(r"(@\[A-Za-z0-9]+)|([^0-9A-Za-z \t])|(\w+:\/\/\S+)|^rt|http.+?", " ", text)
text = " ".join([word for word in text.split() if word not in (stop)])
stemmer = PorterStemmer()
text = "".join([stemmer.stem(word) for word in text])
common_words = ["subscribe", "share", "like", "follow", … ]
querywords = text.split()
resultwords = [word for word in querywords if word not in common_words]
classes_id.append(i[0])
classes_cleantext.append(' '.join(resultwords))
classes_category.append(temp_data[0])
except:
pass
temp_data.append(i[0])

data = pd.DataFrame({'Channel_Id': classes_ex_id, 'Description': classes_ex_cleantext,
'Category': classes_ex_category})
data.to_csv('channel_id_extraction.csv')
data = pd.DataFrame({'Channel_Id': classes_id, 'Clean_text': classes_cleantext, 'Category':
classes_category})
data.to_csv('channel_id_cleantext.csv')
le = LabelEncoder()
le.fit(data.Category)
```

```
data.Category = le.transform(data.Category)
tfidf_title = TfidfVectorizer(sublinear_tf=True, min_df=5, norm='l2', encoding='latin-1',
ngram_range=(1, 2), stop_words='english')
labels = data.Category
features_title = tfidf_title.fit_transform(data.Clean_text).toarray()
# print('Title Features Shape: ' + str(features_title.shape))
data['Category'].value_counts().sort_values(ascending=False).plot(kind='bar', y='Number
of Samples', title='Number of samples for each class')
plt.show()
for current_class in list(le.classes_):
current_class_id = le.transform([current_class])[0]
features_chi2 = chi2(features_title, labels == current_class_id)
indices = np.argsort(features_chi2[0])
feature_names = np.array(tfidf_title.get_feature_names_out())[indices]
unigrams = [v for v in feature_names if len(v.split(' ')) == 1]
print("# '{}':".format(current_class))
print("Most correlated unigrams:")
print('-' * 30)
print('. {}'.format('\n. '.join(unigrams[-5:])))
print("\n")
X_train, X_test, y_train, y_test = train_test_split(data.iloc[:, 1:2], data['Category'],
random_state=0)
X_train_title_features = tfidf_title.transform(X_train['Clean_text']).toarray()
pickle.dump(tfidf_title, open("transform.pickle", "wb"))
features = np.concatenate([X_train_title_features], axis=1)
svm = linear_model.SGDClassifier(loss='modified_huber', max_iter=1000, tol=1e-
3).fit(features, y_train)
pickle.dump(svm, open('final_model.sav', 'wb'))
X_test_title_features = tfidf_title.transform(X_test['Clean_text']).toarray()
test_features = np.concatenate([X_test_title_features], axis=1)
y_pred = svm.predict(test_features)
y_probas = svm.predict_proba(test_features)
print(metrics.classification_report(y_test, y_pred, target_names=list(le.classes_)))
conf_mat = confusion_matrix(y_test, y_pred)
fig, ax = plt.subplots(figsize=(10, 10))
sns.heatmap(conf_mat, annot=True, fmt='d', xticklabels=list(le.classes_),
yticklabels=list(le.classes_))
plt.ylabel('Actual')
plt.xlabel('Predicted')
plt.title('Confusion Matrix’)
plt.show()
skplt.metrics.plot_precision_recall_curve(y_test, y_probas)
plt.title('Precision-Recall Curve - SVM')
plt.show()
```

Model

```
def add_new_id():
file = open(f'new_entry.csv')
csv_reader = csv.reader(file)
combined_csv = pd.concat([pd.read_csv(f) for f in ["tamil.csv", "telugu.csv",
"hindi.csv", "malayalam.csv"]])
combined_csv.to_csv("combined_region.csv", index=False, encoding='utf-8-sig')
combined_csv_reader = [o[0] if len(o)!=0 else " " for o in
csv.reader(open("combined_region.csv"))]
lang_dict = {"ta": "tamil", "te": "telugu", "hi": "hindi", "ml": "malayalam"}
already = 0
for i in csv_reader:
if i[0] in combined_csv_reader:
already = 1
#print(f"This Id({i[0]}) is already registered.")
r = requests.get(f"https://www.youtube.com/channel/{i[0]}/about")
soup = BeautifulSoup(r.content, 'html.parser')
table = soup.find("meta", itemprop="description")["content"]
tt = soup.find("meta", itemprop="name")["content"] + " - " + table.replace("\n", " ")
translation = translator.translate(tt, dest="en")
text = translation.text.lower()
text = re.sub(r"(@\[A-Za-z0-9]+)|([^0-9A-Za-z \t])|(\w+:\/\/\S+)|^rt|http.+?", " ", text)
text = " ".join([word for word in text.split() if word not in (stop)])
stemmer = PorterStemmer()
text = "".join([stemmer.stem(word) for word in text])
common_words = ["subscribe", "share", "like", "follow",… ]
querywords = text.split()
resultwords = [word for word in querywords if word not in common_words]
data = pd.DataFrame({'text': [" ".join(resultwords)]})
loaded_model = pickle.load(open('final_model.sav', 'rb'))
loaded_trans = pickle.load(open('transform.pickle', 'rb'))
mapping = {'Business': 0, 'Cine Entertainment': 1, 'Cine Review': 2,
'Cooking': 3, 'Cryptocurrency': 4, 'Dance': 5,
'Education': 6, 'Entertainment': 7, 'Fashion & Beauty': 8,  'Fitness': 9, 'Food Review': 10,
'Gaming': 11, 'Memes': 12, 'Music': 13, 'Reactions': 14,  'Share Market': 15,
'Spirituality': 16, 'Sports': 17, 'Tech Education': 18, 'Tech Review': 19, 'Travel': 20}
X_test_title_features = loaded_trans.transform(data['text']).toarray()
test_features = np.concatenate([X_test_title_features], axis=1)
predicted_category = loaded_model.predict(test_features)
category = [k for k, v in mapping.items() if v == predicted_category[0]]
region = 'unknown'
for j in ['tamil', 'telugu', "hindi", "malayalam"]:
for k in text.split():
```

```
if j in k.lower():
region = j
if region == 'unknown':
for j in tt.split():
lang = str(translator.detect(j)).split()[0][-3:-1]
if lang in ["ta", "te", "hi", "ml"]:
region = lang_dict[lang]
if region == 'unknown':
r = requests.get(f"https://www.googleapis.com/youtube/v3/search?key=AIzaSyBB8-ie5_GgpC3bejsBz35PV-mvAwNjdmg&channelId={i[0]}&part=id&order=date&maxResults=1")
vid = r.json()['items'][0]['id']['videoId']
r=requests.get(f'https://www.googleapis.com/youtube/v3/videos?part=snippet&id={vid}&key=AIzaSyBB8-ie5_GgpC3bejsBz35PV-mvAwNjdmg')
try:
lang = r.json()['items'][0]["snippet"]['defaultAudioLanguage']
if lang in ['ta', 'te', 'hi', 'ml']:
region = lang_dict[lang]
else:
raise Exception()
except:
title_temp = r.json()['items'][0]["snippet"]['title']
description_temp = r.json()['items'][0]["snippet"]['description']
for j2 in ['tamil', 'telugu', "hindi", "malayalam"]:
for k2 in (title_temp + description_temp).split():
if j2 in k2.lower():
region = j2
if region == 'unknown':
for j2 in (title_temp + description_temp).split():
lang = str(translator.detect(j2)).split()[0][-3:-1]
if lang in ["ta", "te", "hi", "ml"]:
region = lang_dict[lang]
if region != 'unknown':
if already == 0:
file = open(f'{region}.csv')
csv_reader = [i[0] if len(i) != 0 else " " for i in csv.reader(file)]
for k, j in enumerate(csv_reader):
 if j == category[0]:
csv_reader.insert(k + 1, i[0])
with open(f'{region}.csv', 'w') as ff:
writer = csv.writer(ff)
writer.writerows([[k] for k in csv_reader])
file.close()
print(f"Registered: {i[0]} - {region} - {category[0]}")
else: print("Can't find the region of this Id!")
```

Cloud Updation

```
def update_region():
regions = ["tamil", "telugu", "hindi", "malayalam"]
region = int(input("REGIONS\n1. Tamil\n2. Telugu\n3. Hindi\n4. Malayalam\nEnter a
number to select a region: "))
file = open(f'{regions[region - 1]}.csv')
csv_reader = csv.reader(file)
data = {}
data_contact = {}
data_unapproved = {}
temp_data = []
temp_data_contact = []
temp_data_unapproved = []
for i in csv_reader:
if len(i) == 0:
print(f"{temp_data[0]} phase completed --- Next phase on-process ->")
data[temp_data[0]] = temp_data[1:]
temp_data = []
data_contact[temp_data_contact[0]] = temp_data_contact[1:]
temp_data_contact = []
data_unapproved[temp_data_unapproved[0]] = temp_data_unapproved[1:]
temp_data_unapproved = []
else:
try:
r = requests.get(f"https://www.youtube.com/channel/{i[0]}/about")
soup = BeautifulSoup(r.content, 'html.parser')
table = soup.find("meta", itemprop="description")["content"]
mail_id = re.findall(r"[a-z0-9\.\-+_]+@[a-z0-9\.\-+_]+\.[a-z]+", table)
if ((r.text).find("For business enquiries") == -1) and (len(mail_id) == 0):
temp_data_unapproved.append(i[0])
if len(mail_id) != 0:
temp_data_contact.append(i[0] + ' | ' + mail_id[0])
except:
temp_data_unapproved.append(i[0])
temp_data_contact.append(i[0])
temp_data.append(i[0])
db.collection(regions[region - 1]).document("unapproved").set(data_unapproved)
db.collection(regions[region - 1]).document("id").set(data)
db.collection(regions[region - 1]).document("contact").set(data_contact)
```

Request Extraction From Cloud

```
def new_entry_cloud():
entry = db.collection("zone").document("new_entry").get({'id'}).to_dict()
entry = list(set(entry['id'].split(' / ')))
entry.remove('start')
with open(f'new_entry.csv', 'w') as ff:
writer = csv.writer(ff)
writer.writerows([[k] for k in entry])
ff.close()
db.collection("zone").document("new_entry").set({"id": "start"})
```

Web app

```
<div className="App">
<div className="Bg001">
<div className='Header001'>
<img className='Logo001'
src="https://lh3.googleusercontent.com/ogw/ADea4I58SlVwNZskpULql2McX5H7oxME
mRKs2PJzoV1s=s83-c-mo" alt="PPR Logo" />
<h1 className='Title001'>Paid Promoters Recommender</h1>
<Dropdown style={{ marginTop: "15px", marginLeft: "auto", marginRight: "-150px" }}>
<Dropdown.Toggle variant="danger" className='Dd_btn' id="dropdown-basic">
Max Result: {category_input_result}
</Dropdown.Toggle>
<Dropdown.Menu style={{ backgroundColor: "rgb(255, 222, 222)" }}>
{
[5, 10, 15, 20].map((option, id) => (<Dropdown.Item className='Dd_options' key={id}
onClick={(e) => { Setcategory_input_result(option) }}>{option}</Dropdown.Item>))
}
</Dropdown.Menu>
</Dropdown>
<Dropdown style={{ marginTop: "15px", marginLeft: "auto", marginRight: "-150px" }}>
<Dropdown.Toggle variant="danger" className='Dd_btn' id="dropdown-basic">
```

```
Sort by: {sort}
</Dropdown.Toggle>
<Dropdown.Menu style={{ backgroundColor: "rgb(255, 222, 222)" }}>
{
["Followers", "View rate"].map((option, id) => (<Dropdown.Item
className='Dd_options' key={id} onClick={(e) => { Setsort(option)
}}>{option}</Dropdown.Item>))
}
</Dropdown.Menu>
</Dropdown>
<Dropdown style={{ marginTop: "15px", marginLeft: "auto", marginRight: "-150px" }}>
<Dropdown.Toggle variant="danger" className='Dd_btn' id="dropdown-basic">
Region: {region.charAt(0).toUpperCase() + region.slice(1)}
</Dropdown.Toggle>
<Dropdown.Menu style={{ backgroundColor: "rgb(255, 222, 222)" }}>
{
region_options.map((option, id) => (<Dropdown.Item className='Dd_options' key={id}
onClick={(e) => { Setregion(option) }}>{option.charAt(0).toUpperCase() +
option.slice(1)}</Dropdown.Item>))
}
</Dropdown.Menu>
</Dropdown>
<Dropdown style={{ marginTop: "15px", marginLeft: "auto", marginRight: "30px" }}>
<Dropdown.Toggle variant="danger" className='Dd_btn' id="dropdown-basic">
{category_input}
</Dropdown.Toggle>
<Dropdown.Menu style={{ backgroundColor: "rgb(255, 222, 222)" }}>
{
ad_category.map((option, id) => (<Dropdown.Item className='Dd_options' key={id}
onClick={(e) => { Setcategory_input(option) }}>{option}</Dropdown.Item>))
```

```
}
</Dropdown.Menu>
</Dropdown>
</div>
<div>
{category_input === "Ad category" ? <>
<h1 style={{ color: "rgb(233, 66, 24)", marginTop: "40vh" }}>Choose your region & ad category</h1>
</> : <div className="Ranking">
<ButtonGroup vertical>
{
(category_details.slice(0, category_input_result)).map((option, id) => (<Button variant="danger" onClick={() => { Setchannel_details(option); is_contact_available(option[0]) }} className="Ranking_buttons" key={id}>#{id + 1}    {textEllipsis(option[1])}</Button>))
}
</ButtonGroup>
<div className="Details">
{channel_details && <>
<h3 className="Details_title" onClick={() => window.open(`https://www.youtube.com/channel/${channel_details[0]}`)}>{channel_details[1]} {!category_details_unapproved.includes(channel_details[0]) && <>&#10004;</>}</h3>
<div className="Details_basic">
<h5 className='Info_tag'><span style={{ color: "black" }}>Followers<br /></span> { channel_details[2] === "" ? "Private" : numWords(channel_details[2])
}</h5>
<h5 className='Info_tag'><span style={{ color: "black" }}>Total views<br /></span> {numWords(channel_details[3])}</h5>
```

```
<h5 className='Info_tag'><span style={{ color: "black" }}>Videos posted<br /></span>
{numWords(channel_details[4])}</h5>
<h5 className='Info_tag'><span style={{ color: "black" }}>Last video posted<br
/></span> {channel_details[6].slice(8, 10) + '/' + channel_details[6].slice(5, 7) + '/' +
channel_details[6].slice(2, 4)}</h5>
{
channel_details_contact === "contact_unavailable" ?
<h5 className='Info_tag' style={{ cursor: "pointer" }} onClick={() =>
window.open(`https://www.youtube.com/channel/${channel_details[0]}/about`)}><span
style={{ color: "black" }}>Contact<br /></span> @About page</h5> :
<h5 className='Info_tag'><span style={{ color: "black" }}>Contact<br /></span>
{String(channel_details_contact).split("@")[0]}<br
/>@{String(channel_details_contact).split("@")[1]}</h5>
}
<h5 className='Info_tag'><span style={{ color: "black" }}>Approx. view rate<br
/></span> {numWords(channel_details[17])}</h5>
</div>
<div className="Fillups">
<div className='Line_chart'><Line data={line_chart_data()} /></div>
<YouTube className="Utube_vdo" videoId={channel_details[5]} />
</div>
</>}
</div>
</div>}
<br />
<br />
{category_input !== "Ad category" ?
<div className='Footer'>
<h5 className="Register" onClick={() => setShow(true)}>
Register my channel
```

```
</h5>
<Modal
show={show}
onHide={() => setShow(false)}
size="lg"
aria-labelledby="contained-modal-title-vcenter"
centered
>
<Modal.Header closeButton>
<Modal.Title id="contained-modal-title-vcenter">
Register my channel
</Modal.Title>
</Modal.Header>
<br/>
<Modal.Body>
<Form.Group>
<Form.Control type="text" required placeholder='Channel Id' value={cid} onChange={e
=> (Setcid(e.target.value), e.preventDefault())} />
</Form.Group>
</Modal.Body>
<br/>
<Button variant='danger' type='submit' onClick={() =>
cid_submission(cid)}>Submit</Button>
</Modal>
<h5 className='Queries'>Any queries: paidpromotersrecommender@gmail.com</h5>
</div> : <></>}
<br />
</div>);
```

APPENDIX II

CO-PO Mapping

PROJECT WORK COURSE OUTCOME (COs):

CO1: On completion the students capable of execute the proposed plan and become aware of and overcome the bottlenecks throughout every stage.

CO2: On completion of the project work students could be in a role to take in any difficult sensible issues and locate answer through formulating right methodology.

CO3: Students will attain a hands-on revel in in changing a small novel idea / method right into an operating model / prototype related to multi-disciplinary abilities and / or understanding and operating in at team.

CO4: Students will be able to interpret the outcome of their project. Students will take on the challenges of teamwork, prepare a presentation in a professional manner, and document all aspects of design work.

CO5: Students will be able to publish or release the project to society.

PROGRAM OUTCOMES (POs):

PO1: Engineering knowledge: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.

PO2: Problem analysis: Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.

PO3: Design/development of solutions: Design solutions for complexengineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.

PO4: Conduct investigations of complex problems: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

PO5: Modern tool usage: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.

PO6: The engineer and society: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

PO7: Environment and sustainability: Understand the impact of theprofessional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.

PO8: Ethics: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

PO9: Individual and team work: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.

PO10: Communication: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

PO11: Project management and finance: Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

PO12: Life-long learning: Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

PROGRAM SPECIFIC OUTCOMES (PSOs):

PSO1: Foundation Skills: Ability to understand, analyze and develop computer programs in the areas related to algorithms, system software, web design, machine learning, data analytics, and networking for efficient design of computer-based systems of varying complexity. Familiarity and practical competence with a broad range of programming language and open-source platforms.

PSO2: Problem-Solving Skills: Ability to apply mathematical methodologies to solve computational task, model real world problem using appropriate data structure and suitable algorithm. To understand the Standard practices and strategies in software project development using open-ended programming environments to deliver a quality product.

PSO3: Successful Progression: Ability to apply knowledge in various domains to identify research gaps and to provide solution to new ideas, inculcatepassion towards higher studies, creating innovative career paths to be an entrepreneur and evolve as an ethically social responsible computer science professional.

PO/PSO CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO 1	3	3	2	1	3	2	2	2	3	3	3	3	3	1	3
CO 2	3	2	2	2	3	2	2	2	3	3	2	3	2	2	3
CO 3	3	3	2	2	3	1	2	2	3	2	2	1	3	2	3
CO 4	3	3	2	3	3	2	1	2	3	3	3	3	2	3	3
CO 5	3	-	3	-	2	2	1	1	3	3	3	-	3	-	2
Average	3	2.75	2.2	2	2.8	1.8	2	1.8	3	2.8	2.6	2.5	2.6	2	2.8

REFERENCES

[1] Abdelkader OA "**A study of forced-ad resistance leading to Skip Ad on YouTube**" Turkish Journal of Computer and Mathematics Education (TURCOMAT). 2021 Apr 28;12(10):7263-71.

[2] Chang JY "**Applying AHP to Comprehend the Factors Influencing Consumer Attention to Sponsored Ads by YouTubers**" 18th International Conference on Electronic Busines 2018 June 12.

[3] Dehghani M, Niaki MK, Ramezani I, Sali R "**Evaluating the influence of YouTube advertising for attraction of young customers**" Computers in human behavior. 2016 Jun 1;59:165-72.

[4] Holland M "**How YouTube developed into a successful platform for user-generated content**" Elon journal of undergraduate research in communications. 2016;7(1).

[5] Kujur F, Singh S "**Emotions as predictor for consumer engagement in YouTube advertisement**" Journal of Advances in Management Research. 2018 May 14.

[6] Malik H, Tian Z "**A framework for collecting youtube meta-data**" Procedia computer science. 2017 Jan 1;113:194-201.

[7] Rybaczewska M, Jebet Chesire B, Sparks L. "**YouTube vloggers as brand influencers on consumer purchase behaviour**" Journal of Intercultural Management.2020;12(3):117-40.

[8] Shanmuga Skandh Vinayak E, Venkatanath A G S, Shahina A, Nayeemulla Khan A. "**Advertisement Recommendation Engine - Improving YouTube Advertisement Services**" International Journal of Recent Technology and Engineering (IJRTE) ISSN: 2277-3878 (Online), Volume-9 Issue-4 November 2020.

[9] Shizhen Jia "**Literature review of YouTube in teaching activities"** PACIS 2019 Proceedings.

[10] Snelson C "**YouTube across the disciplines: A review of the literature**" MERLOT Journal of Online learning and teaching. 2011.

[11] Tangmanee C"**Fixation and recall of YouTube ad banners: An eye-tracking study**" International Journal of Electronic Commerce Studies. 2016 May 10;7(1):49-76.

[12] Vadde, Neha Reddy, Piyush Gupta, Prasham Mehta, Puneet Gupta, and Vikranth BM. "**Analysis of YouTube Videos: Detecting Click bait on YouTube**" ISSN (Online): 2456-7361 volume 4, Issue 6, pp. 15-17, 2020.

[13] Vedula N, Sun W, Lee H, Gupta H, Ogihara M, Johnson J, Ren G, Parthasarathy S"**Multimodal content analysis for effective advertisements on youtube**" 2017 IEEE international conference on data mining (ICDM) 2017 Nov 18 (pp. 1123-1128). IEEE.

[14] Veluchamy R, Sans RK, Rajagopal P"**To study the impact of youtube tech influencers on the consumer buying behavior of electronic gadget**" ISSN: 2249-6661 Vol-44 No.-01(XIII) : 2021.

[15] Wickramasinghe SP, Welgama SD, Rajapakse RP, Jayasuriya N, Munasinghe AA "**The Impact of Viewers' Behavior and YouTubers' Credibility in Advertainment on Building Consumer Trust**" AJEBA, 21(9): 87-97, 2021; Article no.AJEBA.70504