Sentimental Analysis of YouTube Video Comments Using Bagging Ensemble Learning Approach

A Major project report submitted in partial fulfillment of the requirements for the award of Degree of

BACHELOR OF TECHNOLOGY
IN
ELECTRONICS AND COMMUNICATION ENGINEERING
Submitted By

P. Sai kumar	**19A51A0440**
K. Koteshwar Rao	**19A51A0429**
D. Vamsi	**19A51A0414**
P. Purnachandra Rao	**19A51A0441**

Under the esteemed guidance of

Sri . Rajendraprasad K, M. Tech,
Assistant Professor
Department of ECE
AITAM, Tekkali.

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

ADITYA INSTITUTE OF TECHNOLOGY AND MANAGEMENT
(An Autonomous Institute)

Approved by AICTE, Permanently Affiliated to JNTUGV, Vizianagaram,
Accredited by NBA (Tier – I) and NAAC with Grade 'A+',
K. Kotturu, Tekkali, Srikakulam (Dist.), Andhra Pradesh. India – 532201.

2023

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

ADITYA INSTITUTE OF TECHNOLOGY AND MANAGEMENT

TEKKALI

CERTIFICATE

This is to certify that the major project work entitled **"SENTIMENTAL ANALYSIS OF YOUTUBE VIDEO COMMENTS USING BAGGING ENSEMBLE LEARNING APPROACH"** is a bonafide work done by **P.Sai Kumar(19A51A0440), K.Koteshwar Rao (19A51A0429), D.Vamsi (19A51A0414)** and **P.Purnachandra Rao(19A51A0441)** submitted in partial fulfilment of the requirements for the award ofthe degree of **BACHELOR OF TECHNOLOGY in ELECTRONICS AND COMMUNICATION ENGINEERING.**

Project Guide

Sri .Rajendraprasad K, M. Tech
Assistant Professor,
Department of ECE.

Head of the Department

Dr. B. Rama Rao, M. Tech., Ph. D, SMIEEE,
Professor & HOD ,
Department of ECE ,
AITAM, Tekkali.

ACKNOWLEDGEMENT

It is indeed with a great sense of pleasure and immense sense of gratitude that we acknowledge the help of these individuals. We are highly indebted to our college Director Prof. **V.V. NageswaraRao** and Principal **Dr. A.S. Srinivasa Rao** for the facilities provided to accomplish the project.

We would like to thank our Head of the Department **Dr. B. Rama Rao** for his constructive criticism throughout our project

We are immensely grateful to our guide **Sri. Rajendraprasad K** for introducing us to thechallenging and interesting field of Machine learning. He has been a great source of inspiration, encouragement, and enlightenment to us, in our work. Despite his heavy administrative duties andvarious other responsibilities, he was always been kind enough to spare sufficient time for us andcontribute to the progress of our project by means of his incisive comments, insightful suggestions,and valuable discussions. It was truly a great experience to be associated with him.

We would like to thank our Department Project coordinator **Dr. P.V. Muralidhar** for his supportto completed our project

We are extremely grateful to our department staff members, lab technicians and Non - teaching staff members for their extreme help throughout our project.

Finally, we express our sincere gratitude to our parents, Siblings and friends who helped us in successful completion of this project

P. Sai Kumar (19A51A0440)
K. Koteshwar Rao (19A51A0429)
D. Vamsi (19A51A0414)
P. Purnachandra Rao (19A51A0441)

DEPARTMENT OF ECE

Vision of the Department:

Create high quality engineering professionals through research, innovation and team work for a lasting technology development in the area of Electronics and Communication Engineering.

Mission of the Department:

1. To offer a well-balanced program of instruction, lab practices, research & developmentactivities, product incubation.

2. Develop accomplished technical personnel with a strong background on fundamental andadvanced concepts, have excellent professional conduct.

3. Enhance overall personality development which includes innovative and group work exercises,entrepreneur skills, communication skills and employability.

4. Ensuring effective teaching-learning process to provide in depth knowledge of principals and its applications to Electronics and Communication Engineering and interdisciplinary areas.

5. Providing industry and development interactions through consultancy and sponsored research.

Program Educational Objectives (PEOs) of B. Tech in ECE:

PEO I: The graduates would be employed as a practicing engineer in fields such as design, testingand manufacturing.

PEO II: the graduates would be able to imbibe research, development and entrepreneurship skills.

PEO III: The graduates would be engaged lifelong self-director learning to maintain and enhanceprofessional skills.

PEO IV: The graduates will be able to exhibit communication skills, team spirit, leadership skillsand ethics with social responsibility.

PROGRAM OUTCOMES:

Engineering Graduates Will Be Able to:

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

2. **Problem analysis**: Identify, formulate, review research literature, and analyse complex engineering problems reaching sustained conclusions using first principles of mathematics, naturalsciences and engineering sciences.

3. **Design/ development of solutions**: Design solution for complex engineering 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 environment considerations.

4. **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.

5. **Modern tool usage:** Create, select and apply appropriate techniques, resources, and modern engineering and its tools including prediction and modelling to complex engineering activities withan understanding of the limitations.

6. **The engineer and society**: Apply reasoning informal by the contextual knowledge to access societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to theprofessional engineering practice.

7. **Environment and sustainability**: Understand the impact of professional engineering solutionsin societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.

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

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

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

11. **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 leader ina team, to manage projects and in multidisciplinary environments.

12. **Lifelong learning**: Recognize the need for, and have the preparation and ability to engage in independent and lifelong learning in the broadest context of technological change.

PSO –PROGRAM SPECIFIC OUTCOMES:

PSO1: The competency in the application of circuit analysis and design.

PSO II: An ability to solve electronics and communication engineering problems using the latesthardware and software tools, along with analytical skills to arrive cost effective and appropriate solutions.

PSO III: The ability to pursue higher studies in either India and abroad and also lead a successfulcareer with professional ethics.

ABSTRACT

An important indicator that shows how well-liked a YouTube video is by its viewers is the like ratio. By examining the emotive tone of viewer comments, sentiment analysis can be used to forecast the like ratio of a YouTube video. With this method, the YouTube API is used to first get the comments from the video. Following that, these comments are pre-processed to eliminate any unnecessary data, including URLs and special characters, and to change the text's case to lowercase. The pre-processed comments are then subjected to sentiment analysis using a natural language processing package, such as TextBlob or NLTK, to categorise them as positive, negative. The like ratio can be estimated after sentiment analysis by measuring the percentage of positive comments to all comments.This can be used to determine how viewers feel about the video overall and forecast whetherthe film will have a high or low like ratio. Overall, forecasting the like ratio of a YouTube video using sentiment analysis can offer insightful information for content producers and marketers, assisting them in understanding the emotional response of their audience and improving their content accordingly.

KEYWORDS: Text mining, Sentimental Analysis, YouTube, NLTK, Machine Learning Processing

LIST OF CONTENTS

LIST OF FIGURES

CHAPTER – 1

INTRODUCTION

1.1 INTRODUCTION

Sentiment analysis is the process of determining the emotional tone behind a series of words, oftenused to gauge the opinions or attitudes of individuals towards a particular topic. YouTube is one of the most popular video-sharing platforms in the world, with millions of videos and comments uploaded every day. Analyzing the sentiment of comments on YouTube videos can provide valuable insights into the viewers' opinions and attitudes towards a particular video or topic. The aim of this project is to develop a sentiment analysis model using the bagging ensemble learning approach to analyze comments on YouTube videos. The bagging ensemble learning approach is a machine learning technique that combines multiple models to improve the accuracy of predictions.In this approach, each model is trained on a subset of the data, and the results are combined to generate a final prediction.

The project will involve collecting a large dataset of comments on YouTube videos related to different topics, such as music, movies, news, and sports. The comments will be preprocessed by removing stop words, punctuations, and converting them to lowercase. The sentiment of the comments will be classified as positive, negative, or neutral using a bagging ensemble learning approach.

The bagging ensemble learning approach is a powerful technique that can improve the accuracy ofthe sentiment analysis model by combining the results of multiple models. In this approach, multiple decision trees are trained on different subsets of the data, and the results are combined using a majority vote to generate the final prediction. The advantage of this approach is that it canreduce the variance of the model and improve the accuracy of the predictions. The project will involve several steps, including data collection, preprocessing, feature extraction, model training, and performance evaluation. The performance of the model will be evaluated using metrics such as accuracy, precision, recall, and F1-score. The model will be compared with other sentiment analysis models, such as Naive Bayes, SVM, and Random Forest, to determine its effectiveness. The resultsof this project can have several applications, including market research, social media monitoring, and sentiment analysis of political and social events. The sentiment analysis of

YouTube comments can help companies to understand the opinions and attitudes of their customers towards their products and services. It can also help social media analysts to monitor public opinionand sentiment towards different topics and events.

In conclusion, the sentiment analysis of YouTube comments using the bagging ensemble learningapproach can provide valuable insights into the opinions and attitudes of viewers towards a particular video or topic. The project aims to develop an accurate and effective sentiment analysis model that can be used for various applications, including market research, social media monitoring, and sentiment analysis of political and social event.

1.2 MOTIVATION

The motivation behind conducting sentiment analysis of YouTube video comments using abagging ensemble learning approach stems from the need to understand how viewers respond to online content. With the rise of online video consumption, YouTube has become a popular platformfor individuals and businesses to share their content with a global audience. However, understanding the sentiment of viewers towards the video is crucial for the content creator to tailortheir content, improve engagement, and identify potential areas for improvement. Bagging ensemble learning is a powerful machine learning technique that combines multiple classifiers to improve the overall accuracy of the model. By using multiple classifiers, bagging can reduce overfitting and improve generalization performance, making it an ideal approach for sentiment analysis of YouTube video comments.

The sentiment analysis of YouTube video comments using bagging ensemble learning can provideinsights into the emotions, opinions, and attitudes of viewers towards the content. This informationcan be used by content creators and marketers to improve the engagement of the viewers, increasethe views and shares of the video, and identify the areas where the content needs improvement.

Additionally, the sentiment analysis of YouTube video comments can be used for a variety of otherpurposes, such as identifying potential trends, predicting the success of the video, and measuring the impact of marketing campaigns.

Overall, the motivation behind conducting sentiment analysis of YouTube video comments using bagging ensemble learning approach is to provide content creators, marketers, and businesses witha powerful tool to understand the emotions and opinions of the viewers towards the video content.This information can be used to improve engagement, increase views, and identify areas for improvement, leading to more successful online video campaigns.

CHAPTER – 2
BACKGROUND

2.1 BACK GROUND

The growth of online video sharing platforms like YouTube has resulted in an explosion of user- generated content on the internet. Every day, millions of videos are posted and seen, making it more crucial than ever for content producers to understand the elements that influence viewershipand engagement.

The like ratio of the video, which compares the number of likes to the number of dislikes, is one such factor. The like ratio can affect the video's exposure and recommendation on the platform andis frequently used as a gauge of the audience's opinion of it.

Sentiment analysis, the act of evaluating the polarity of a text, such as whether it communicates positive or negative sentiment, has been used by scholars and data scientists to better understand and forecast the like ratio of YouTube videos. To extract the general sentiment towards the video,sentiment analysis can be used to the text data connected to the videos, such as the titles, descriptions, and comments.

Sentiment analysis may be used to predict the like ratio of YouTube videos with great accuracy by combining machine learning algorithms and natural language processing methods. This can help to improve the recommendation and discovery of videos on the platform as well as offer insightfuldata for content producers and advertisers.

A number of studies and applications have shown the ability of sentiment analysis to predict the like ratio of YouTube videos, and this field of study is expanding quickly. It has the potential to increase viewer comprehension and engagement with online video content.

CHAPTER – 3

LITERATURE SURVEY

3.1 LITERATURE SURVEY

The like ratio of YouTube videos can be predicted using sentiment analysis in a number of studiesand research papers that have been published. Creating machine learning models that can preciselypredict the like ratio of YouTube videos based on their emotion has been the main goal of these investigations.

One such study was carried out by Amado-Salvatierra et al. (2018), who employed a dataset of YouTube comments and sentiment analysis to forecast the like ratio of the videos. To preprocess the text data, they employed a number of natural language processing techniques, including tokenization, stop-word removal, and stemming. Other machine learning algorithms, including decision trees, logistic regression, and random forests, were also employed to train and assess the models.

According to the sentiment of the titles and descriptions of YouTube videos, Mondal et al. (2019)did another study in which they applied a similar methodology to predict the like ratio of YouTubevideos. Also, they trained and assessed the models using a variety of machine learning algorithms,including gradient boosting, support vector machines, and k-nearest neighbours. In terms of forecasting the like ratio of the videos, their findings indicated that the gradient boosting method performed the best.

Similar to this, Poria et al. (2018) used the sentiment of the video's transcript, audio, and visual information in a multimodal method to predict the like ratio of YouTube videos. In order to combine the various modalities and forecast the liking ratio of the videos, they employed a deep neural network model. The multimodal technique outperformed the unimodal approach, accordingto their findings, and was highly accurate in predicting the liking ratio of the videos.

These studies show that sentiment analysis and machine learning algorithms can accurately predictthe like ratio of YouTube videos. Dealing with the vast amount of data, handling the text data's noisy and unstructured nature, and choosing the right machine learning methods and features to train the models are the primary issues in this field.

CHAPTER – 4
METHODOLOGY

4.1 METHODOLOGY

The methods below can be used to estimate the like ratio of a YouTube video using sentimentanalysis:

1. Data collection: Use the YouTube API to gather information about YouTube videos. To focus your search and find the most pertinent movies, you can choose specific search parameters and filters.
2. Data preprocessing: After gathering the data, clean it up by removing noise and extraneousinformation. This involves changing the text's case and eliminating special characters, stopwords, and URLs.
3. Sentiment analysis: Examine the comments on the video using a sentiment analysis tool. Several pre-trained sentiment analysis models, including Vader, TextBlob, and NLTK, areavailable. You can decide on the option that best meets your needs.
4. Feature extraction: Analyse the video comments for relevant information. The emotion ratings, the frequency of good and negative words, or any other features you view importantcould be included in this list of features.
5. Model building: Create a machine learning model based on the features that were retrieved to estimate the like ratio of the video. Regression models like Linear Regression, RandomForest, or Gradient Boosting can be used.
6. Model Evaluation: Consider metrics like Auraccy , F1score, Precission and Recall to assessthe performance of the model.

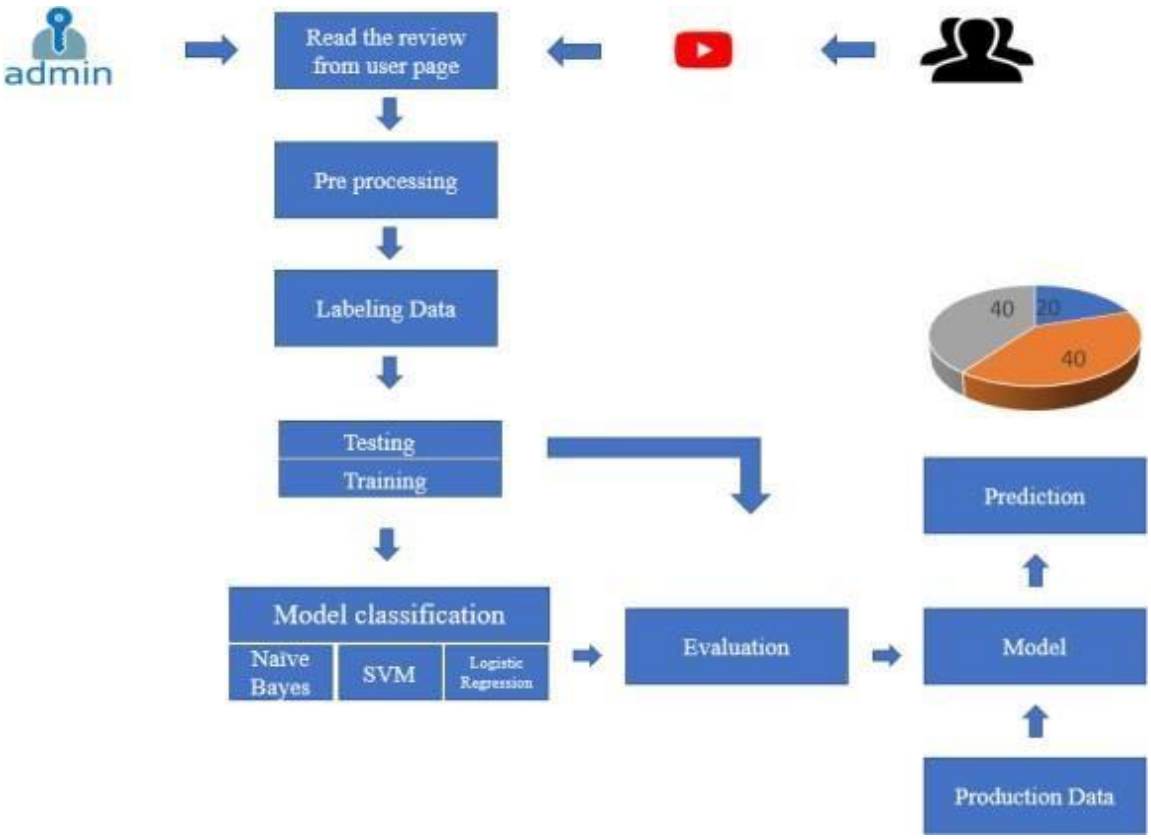


Fig 1: Model Flow Chart

CHAPTER – 5

MACHINE LEARNING

5.1 MACHINE LEARNING

Machine learning is an application of artificial intelligence that provides systems the ability to automatically learn and improve from experience without being explicitly programmed. Machine learning focuses on the development of computer programs that can access data and use it to learnfor themselves. The process of learning begins with observations or data, such as examples, directexperience, or instruction, in order to look for patterns in data and make better decisions in the future based on the examples that we provide. The primary aim is to allow the computers to learn automatically without human intervention or assistance and adjust actions accordingly. Machine learning enables the analysis of massive quantities of data. While it generally delivers faster, moreaccurate results in order to identify profitable opportunities or dangerous risks, it may also requireadditional time and resources to train it properly. Combining machine learning with AI and cognitive technologies can make it even more effective in processing large volumes of information

5.2 TYPES OF MACHINE LEARNING:

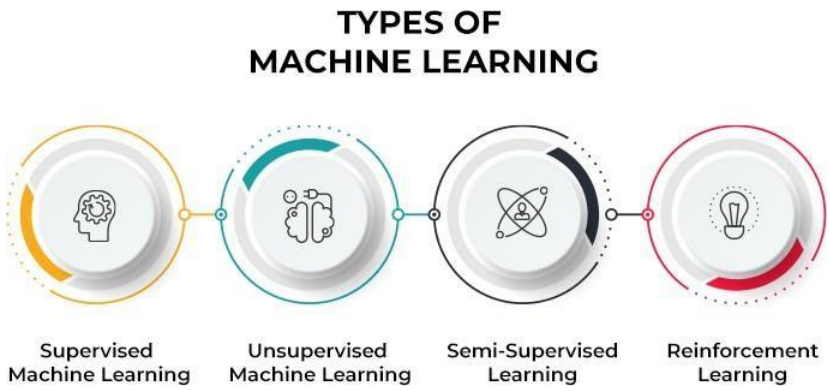


Fig 2 : Types of machine learning

Machine Learning can be done in the following ways:

- **Supervised Learning**
- **Unsupervised Learning**
- **Reinforcement Learning**
- **Semi-supervised Learning**

Let's briefly understand the idea behind each type of Machine Learning.

5.2.1 SUPERVISED LEARNING:

Supervised learning is the machine learning task of learning a function that maps an input to an output based on example input-output pairs. It infers a function from labeled training dataconsisting of a set of training examples. In supervised learning, each example is a pair consisting of an input object (typically a vector) and the desired output value (also called the supervisory signal). A supervised learning algorithm analyzes the training data and produces an inferred function, which can be used for mapping new examples. An optimal scenario will allow for the algorithm to correctly determine the class labels for unseen instances

Steps: In order to solve a given problem of supervised learning, one has to perform the following

1. Determine the type of training examples. Before doing anything else, the user should decide what kind of data is to use data training set. In the case of handwriting analysis, for example, this might be a single handwritten character, an entire handwritten word, or an entire line of handwriting.

2. Gather a training set. The training set needs to be representative of the real-world use

of the function. Thus, a set of input objects is gathered and corresponding outputs are also gathered, eitherfrom human experts or from measurements.

3. Determine the input feature representation of the learned function. The accuracy of the learned function depends strongly on how the input object is represented. Typically, the input object is transformed into a feature vector, which contains a number of features that are descriptive of the object

4. Determine the structure of the learned function and the corresponding learning algorithm. For example, the engineer may choose to use supportive machines or decision trees.

5. Complete the design. Run the learning algorithm on the gathered training set. Some supervised learning algorithms require the user to determine certain control parameters. These parameters may be adjusted by optimizing performance on a subset (called a validation set) of the training set

6. Evaluate the accuracy of the learned function. After parameter adjustment and learning, the performance of the resulting function should be measured on a test set that is separate from the training set.

5.2.2 UNSUPERVISED LEARNING

Models are not supervised using training datasets while utilising the machine learning technique known as unsupervised learning. Instead, models themselves decipher the provided data to reveal hidden patterns and insights. It is comparable to the learning process that occurs in the human brainwhile learning something new. It is characterised as:

Unlike supervised learning, we have the input data but no corresponding output data, unsupervisedlearning cannot be used to solve a regression or classification problem directly. Finding the underlying structure of a dataset, classifying the data into groups based on similarities, and representing the dataset in a compressed format are the objectives of unsupervised learning.

Compared to supervised learning, unsupervised learning is employed for problems thatare morecomplex since it lacks labelled input data.

Unsupervised learning is preferred because unlabeled data is simpler to obtain than labelled data.Disadvantages:
Due to the lack of a comparable output, unsupervised learning is inherently more challenging thansupervised learning.

As the input data is not labelled and the algorithms do not know the precise output in advance, theoutcome of the unsupervised learning method may be less accurate.

5.2.3 REINFORCEMENT LEARNING

Machine learning includes the discipline of reinforcement learning. It involves acting appropriatelyto maximise reward in a certain circumstance. It is used by a variety of programmes and machinesto determine the optimal course of action to pursue in a given circumstance. There is no correct answer in reinforcement learning, but the reinforcement agent selects what to do to complete the job. This is different from supervised learning, where the training data includes the solution key and the model is trained with that answer. It is obligated to gain knowledge from its experience inthe absence of a training dataset.

The study of decision-making is called reinforcement learning (RL). It involves understanding howto act in a situation to reap the most benefits. Data for RL is gathered from machine learning systems that employ a trial-and-error process. Input for either supervised or unsupervised machinelearning does not include data.

Algorithms used in reinforcement learning determine the next course of action based on results. The algorithm receives feedback after each step that aids in determining whether the decision it made was good, bad, or indifferent. It is a useful method for automated systems that must make numerous tiny judgments without human supervision.

An autonomous, self-teaching system known as reinforcement learning fundamentally

learns through trial and error. It acts with the intention of maximising rewards, or, to put it another way, it learns by doing in order to get the best results.

1. Reinforcement learning can be used to tackle extremely difficult issues that are intractable usingtraditional methods.

2. The model has the ability to fix mistakes made during training.

3.In RL, training data is gathered by the agent's direct engagement with the environment.

4. Using reinforcement learning to solve straightforward issues is not recommended.

5.A large amount of data and computation are required for reinforcement learning.

5.2.4 SEMI-SUPERVISED LEARNING

A form of machine learning algorithm known as semi-supervised learning sits between supervised and unsupervised learning algorithms. During the training phase, it uses a combination of labelledand unlabeled datasets.

You need be familiar with the key categories of machine learning algorithms before you can grasp semi-supervised learning. Supervised Learning, Unsupervised Learning, and Reinforcement Learning are the three main types of machine learning. Unsupervised datasets do not contain an output label training data linked with each tuple, whereas supervised learning datasets do. This is another key difference between supervised and unsupervised learning. Between supervised and unsupervised machine learning, semi-supervised learning is a crucial subcategory. While semi- supervised learning acts on data that contains a few labels and is a middle ground between supervised and unsupervised learning, the majority of the data it uses is unlabeled. Although labelsare expensive, for corporate purposes, there might not be many labels.

The main drawback of supervised learning is that it costs a lot to process and necessitates

manual labelling by ML experts or data scientists. Furthermore, the range of applications for unsupervisedlearning is constrained. The idea of semi-supervised learning is presented to address these issues with supervised learning and unsupervised learning methods. The training set for this algorithm consists of both labelled and unlabeled data. While there is a significant amount of unlabeled data,there is a relatively little amount of annotated data. An unsupervised learning technique is first usedto cluster comparable data, and it also aids in labelling the unlabeled data into labelled data. It is for this reason that labelled data is more expensive to acquire than unlabeled

CHAPTER – 6

SENTIMENTAL ANALYSIS

6.1 SENTIMENTAL ANALYSIS

Sentiment analysis, commonly referred to as opinion mining, is a branch of Natural Language Processing (NLP) that deals with locating and obtaining subjective data from text. Analyzing the thoughts, attitudes, opinions, and feelings conveyed in a specific text such as a tweet, news story, product review, or social media post involves doing this.

The main objective of sentiment analysis is to identify the text's polarity, or whether it is good, negative, or neutral. Depending on the context and the work at hand, the analysis can be done at many levels, including the document level, the phrase level, or the aspect level.

The collection of data, pre-processing, feature extraction, and classification are just a few of the phases that make up the sentiment analysis process. The relevant text data is gathered during the data collection step from a variety of sources, including social media sites, news articles, and polls.At the pre-processing stage, the data is cleaned and made ready for analysis by removing stop words, tokenizing, stemming, and lemmatizing, for example.

The process of finding and choosing the text's most important properties, such as the frequency ofparticular words, phrases, or n-grams, is known as feature extraction. After that, a classification machine learning model is trained using these features. Depending on the objective and the data, the classification algorithm employed may be either rule-based or statistical-based.

The ability to extract and study the sentiments and emotions portrayed in text data via sentiment analysis is crucial. It has several uses in various fields and offers insightful information and knowledge to enhance decision-making processes. Nonetheless, the difficulties of natural languageprocessing necessitate ongoing innovation and advancement in sentiment analysis.

6.2 SENTIMENTAL ANALYSIS IN PYTHON

Natural Language Toolkit (NLTK), TextBlob, and Scikit-learn are a few well-known Python modules that can be used for sentiment analysis. We will give a brief introduction of sentiment analysis utilising these libraries in this section.

6.2.1 NATURAL LANGUAGE TOOLKIT

Natural Language Toolkit, sometimes known as NLTK, is a well-known Python module used for handling and examining text and other natural language data. Sentiment analysis, which is the actof assessing the polarity of a text, such as whether it reflects positive or negative attitude, is one ofthe tasks that NLTK can carry out.

To perform sentiment analysis using NLTK, you can use the VADER tool, which is a rule-based approach that considers the lexicon of words and expressions that are associated with positive or negative sentiment, along with the context and intensity of the sentiment. In NLTK, you can initialize the sentiment analyzer using the Sentiment Intensity Analyzer module, and then analyzethe sentiment of a text by passing it through the polarity scores function. The output will be a dictionary of polarity scores, including the compound score, which is a normalized score ranging from -1 (most negative) to +1 (most positive).

6.2.2 TEXTBLOB

Python's TextBlob package is used to handle and examine text and other natural language data. Sentiment analysis, which is the act of identifying the polarity of a text, such as whether it reflectsa good or negative sentiment, is one of the tasks that TextBlob can carry out.

Simply run the text via the TextBlob function to perform sentiment analysis, and then use the sentiment property to determine the polarity score. The emotion score ranges from 0 (neutral) to -1 (most positive), with -1 being the most liked.

TextBlob uses machine learning to identify the text's polarity, which means it gets understanding from a training set of data to forecast the tone of fresh texts. TextBlob is a flexible library for text analysis since it offers a number of additional natural language processing methods, including part-of-speech tagging, noun phrase extraction, and language translation.

6.2.3 SCIKIT-LEARN

Python's Scikit-learn library is used to perform machine learning operations like classification, regression, and clustering. Along with data pre-processing, model selection, and performance evaluation, it offers a set of tools and algorithms for a variety of machine learning tasks.

The popular scientific Python libraries NumPy, SciPy, and Matplotlib are developed on top of Scikit-learn, which offers an uniform user interface for carrying out machine learning operations. Additionally, it supports a range of machine learning models, including neural networks, support vector machines, decision trees, random forests, and linear regression.

You must first load the relevant modules and functions for the machine learning activity you wishto carry out before you can utilise scikit-learn. After loading your data, pre-process it by normalising or scaling the features, for example. The data can then be divided into training and testing sets, and a model can be chosen and trained using the training set. Lastly, you may assess the model's performance on the testing set and make any necessary adjustments to its parameters.

6.3 NUMPY

NumPy is a Python library that is widely used for numerical computing. It is an abbreviation for "Numerical Python" and is an open-source project that was initially released in 2006. NumPy provides a powerful array computing and manipulation capabilities in Python and supportsoperations such as mathematical, logical, and shape manipulation. The primary data structure of NumPy is the ndarray, or N-dimensional array. An ndarray is a collectionof values of the same type, which can be accessed and manipulated using indexing and slicing. NumPy's ndarray is muchfaster and more memory-efficient than Python's built-in data structures for large datasets. NumPy provides a vast collection of mathematical functions, including trigonometric, logarithmic, and exponential functions, as well as linearalgebra, Fourier transforms, and random number generation. These functions operate on NumPy arrays and are optimized for speed and efficiency. In addition to mathematical functions,

NumPy provides functionality for data analysis and manipulation. NumPy can read and write to files, sort, search, and reshape data, and perform statistical analysis,

including mean, median, standard deviation, and correlation. NumPy is also widely used in scientific computing, data analysis, and machine learning. Many popular libraries in these fields, such as Pandas, Scikit-learn, and TensorFlow, rely on NumPy for their core functionality. NumPy is also compatible with many other libraries and tools in the Python ecosystem, making it a flexible and versatile library for data analysis and scientific computing.

In conclusion, NumPy is an essential library for scientific computing, data analysis, and machine learning in Python. Its powerful array computing and manipulation capabilities, mathematical functions, and data analysis functionality make it a versatile and efficient tool for working with large datasets and performing complex computations.

6.4 PANDAS

Pandas is a powerful and popular data manipulation library for Python. It is widely used in data science, machine learning, and other areas of data analysis. Pandas provides flexible data structuresfor working with tabular data, such as spreadsheets or databases. It also offers a wide range of toolsfor data cleaning, aggregation, and visualization. The two main data structures provided by Pandasare Series and DataFrame. A Series is a one-dimensional array-like object that can hold any data type. It is like a column in a spreadsheet. A DataFrame is a two-dimensional table-like data structure, consisting of rows and columns. It is like a spreadsheet or a SQL table. Pandas providespowerful tools for data cleaning and preparation, such as data selection, filtering, and transformation. It also supports various data operations, such as merging, joining, and grouping. These tools allow users to efficiently manipulate large datasets and prepare them for analysis or machine learning. Pandas also provides powerful visualization tools, based on the Matplotlib library. These tools allow users to create various types of plots, such as histograms, scatterplots, and line plots. Pandas also supports time-series data analysis, allowing users to easily manipulate and analyze time-based data. One of the strengths of Pandas is its ability to handle missing or incomplete data. Pandas provides several methods for handling missing data, such as

dropping missing values or imputing them with other values. This makes it easier to work with real-world data, which often contains missing or incomplete data. In summary,

Pandas is a powerful and flexible data manipulation library for Python. Its efficient data structures and tools make it a popular choice for data analysis, machine learning, and other areas of data science.

CHAPTER – 7

YOU TUBE API

7.1 YouTube API

For programmers wishing to incorporate YouTube's massive material into their own apps, the YouTube API is a potent tool. Developers may access and control YouTube data including videos,playlists, channels, comments, and more using the API.

This guide is meant for programmers who want to create YouTube-integrated applications. It describes the fundamental ideas behind the API as well as YouTube. Additionally, it offers a summary of the many features that the API covers.

Before beginning

To access the Google API Console, request an API key, and register your application, you must have a Google Account.

To enable your application to submit API requests, create a project in the Google Developers Console and receive permission credentials.

Make sure the YouTube Data API is one of the services your application is registered to utilize after building your project:

Select the newly registered project by going to the API Console.

Visit the page for Enabled APIs. Make sure the YouTube Data API v3's status is ON in the list ofAPIs.

Read the authentication guide to discover how to implement OAuth 2.0 authorization if any of yourapplication's API functions require user authorization.

To make your API implementation simpler, pick a client library.

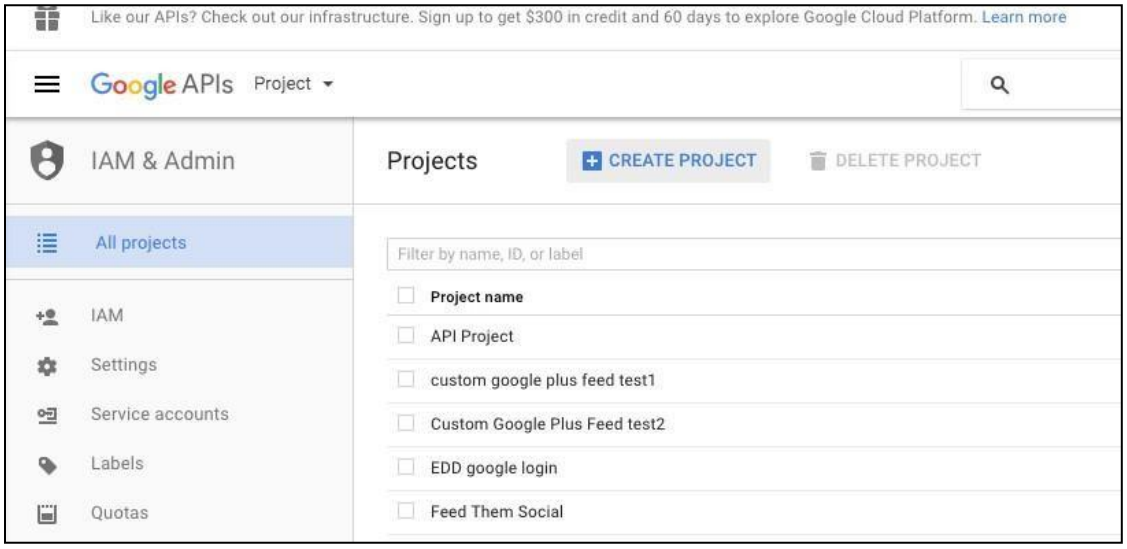
Learn the fundamental ideas behind the JSON (JavaScript Object Notation) data format. JSON is a popular, language-neutral data format that offers a straightforward text representation of any number of different data structures.

All APIs, regardless of type, largely operate in the same way. Typically, when you ask the API fordata or information, it responds with what you asked for. When you open Twitter or scroll throughyour Instagram feed, for instance, you are essentially sending a request to the API powering that app and receiving a response in return. This is also referred to as using an API.

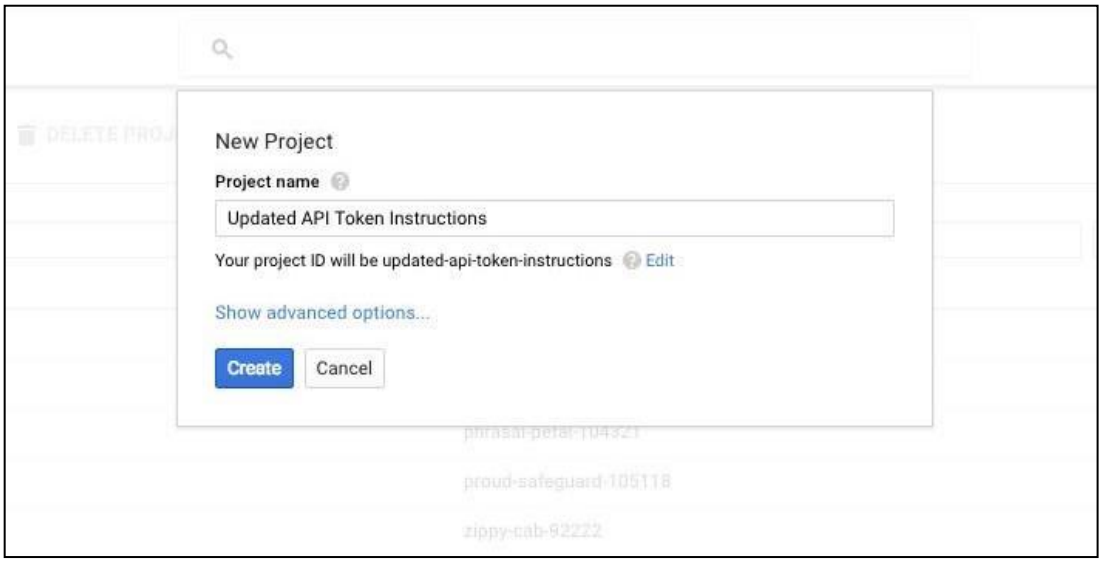
You need an API Key in order to make the YouTube feed function Here is how you can get that.

1. Go to https://developers.google.com/ and log in or create an account, if necessary.

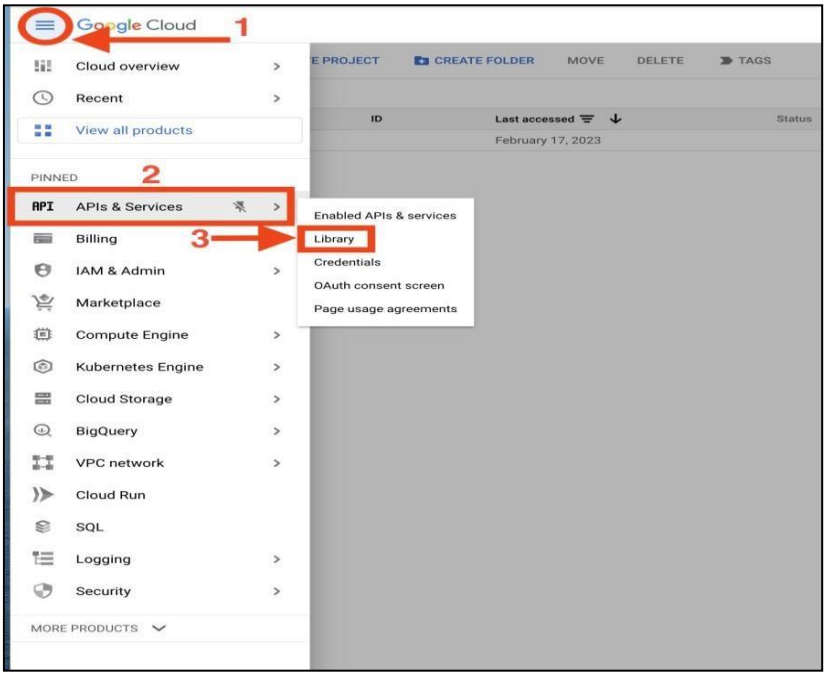
2. After logging in go to this link https://console.developers.google.com/project and click on theblue CREATE PROJECT button as depicted in the photo below. Wait a moment as google prepares your project.

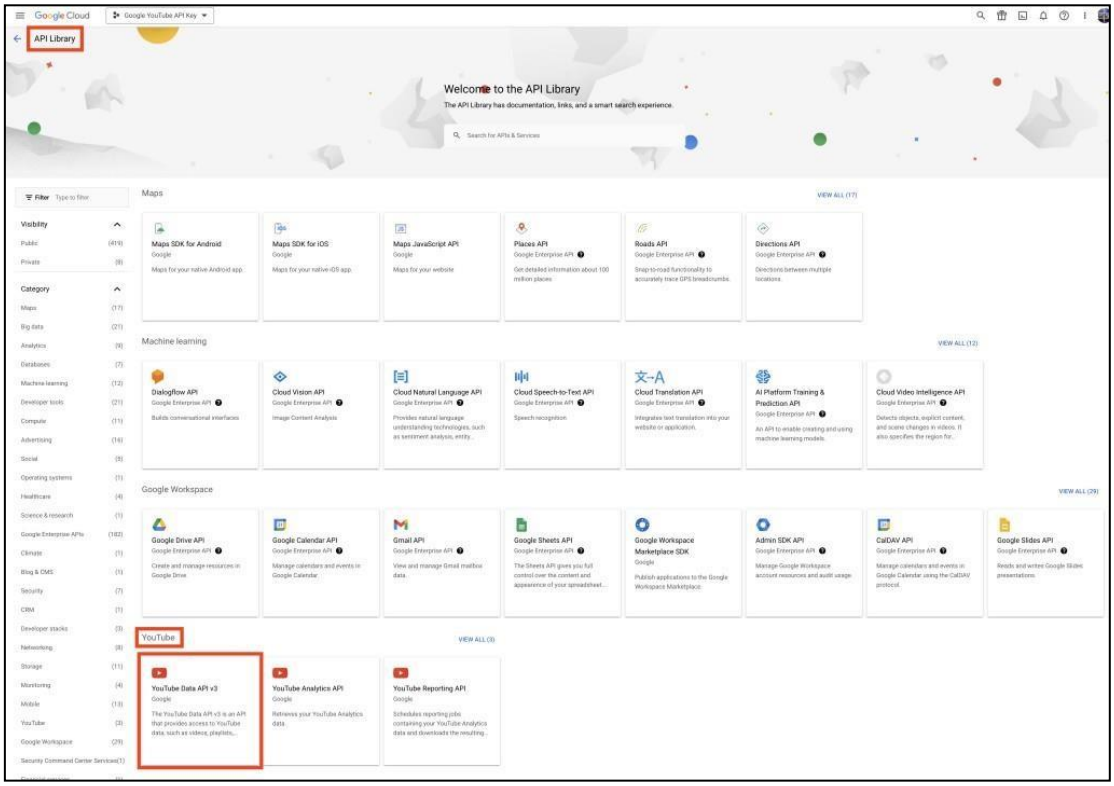


3. Fill in whatever Project Name you want.

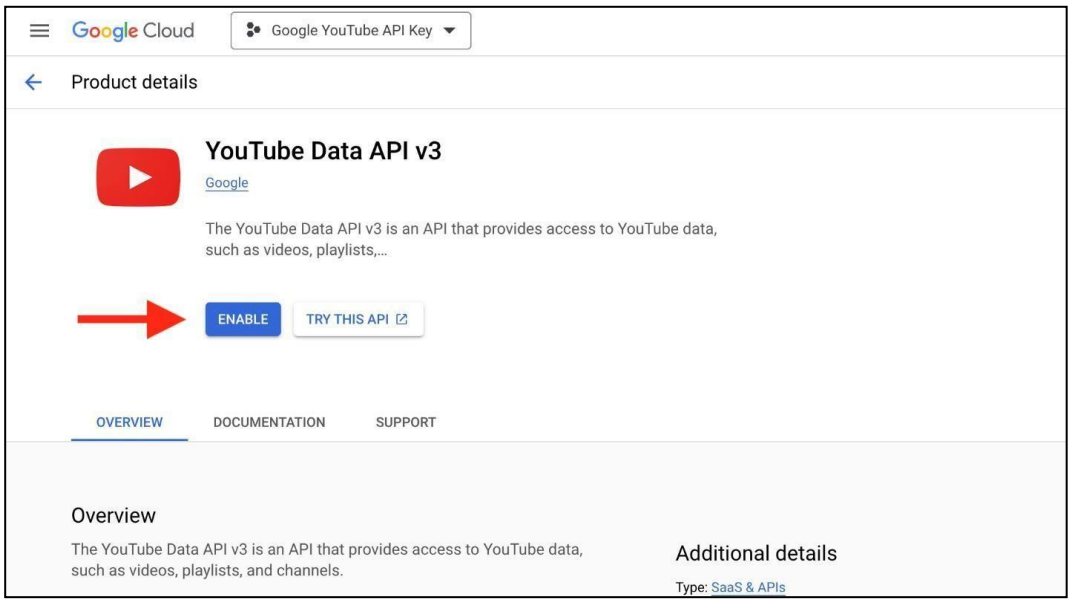


4. Then, in the top left corner, click Google APIs link three line hamburger, click the API "APIs &Services" tab and then click "Library". Next, click the link option called "YouTube Data API." It'sunder YouTube API's. You can see it highlighted in the photo below, bottom left.

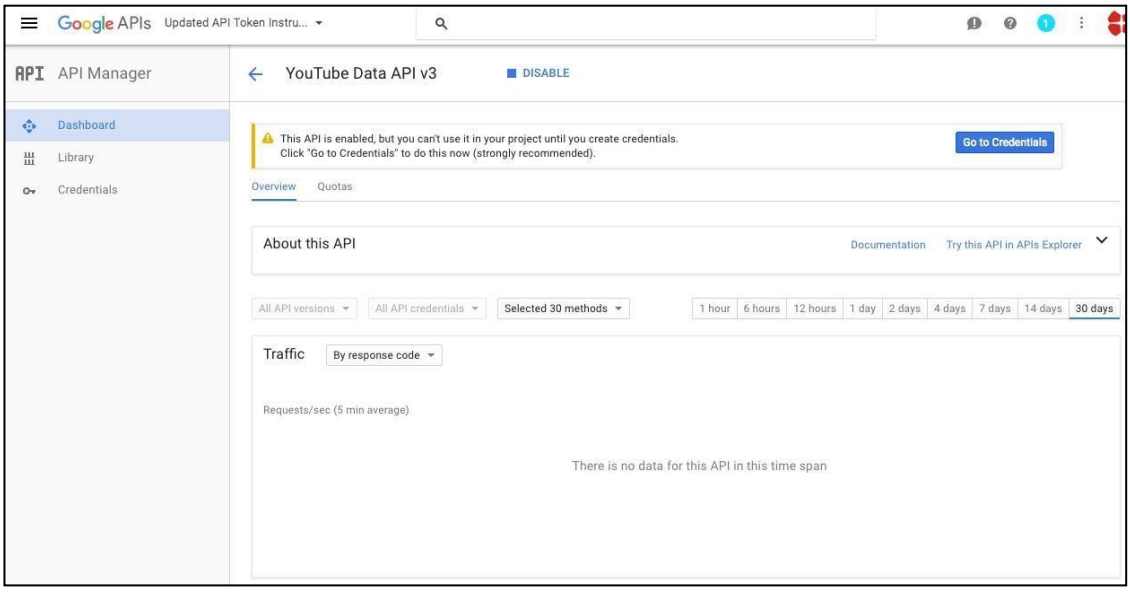




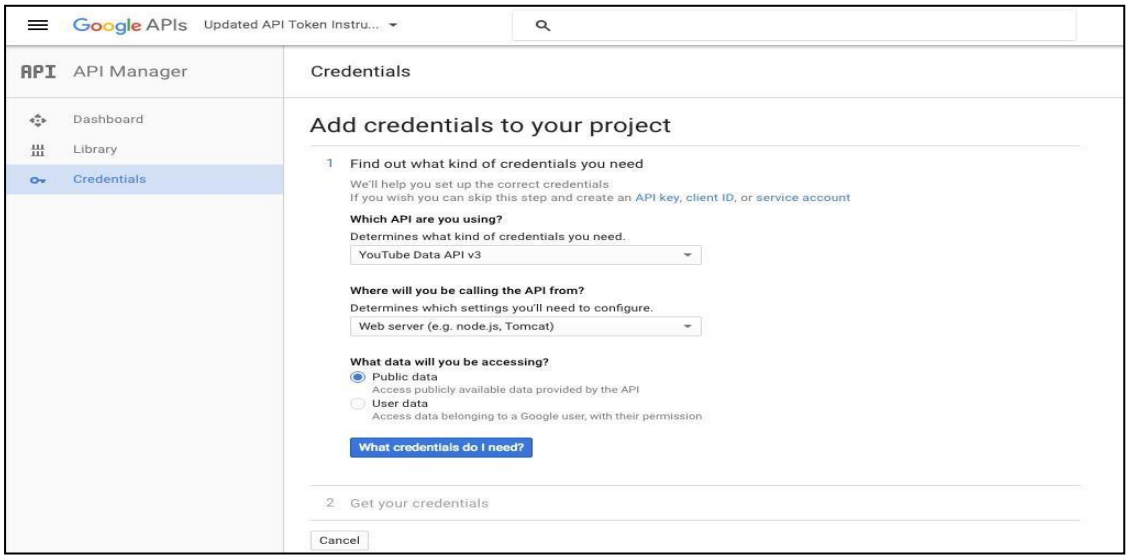
5. Now click on the “ENABLE” button.



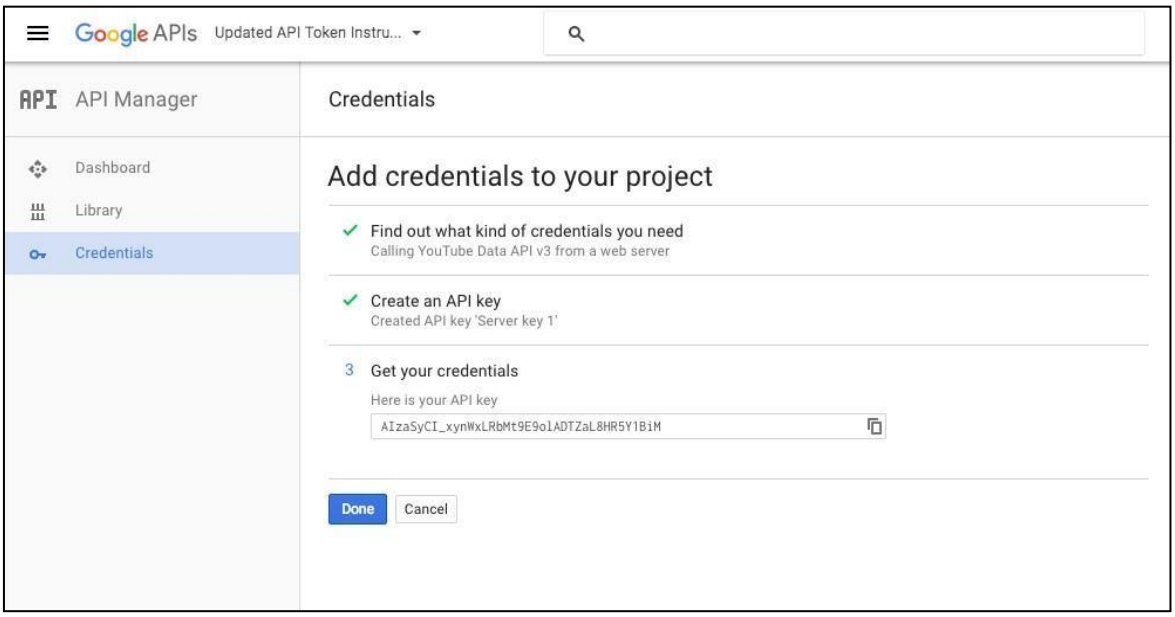
6. Next click on the blue ‘Go to Credentials’ button to the right.



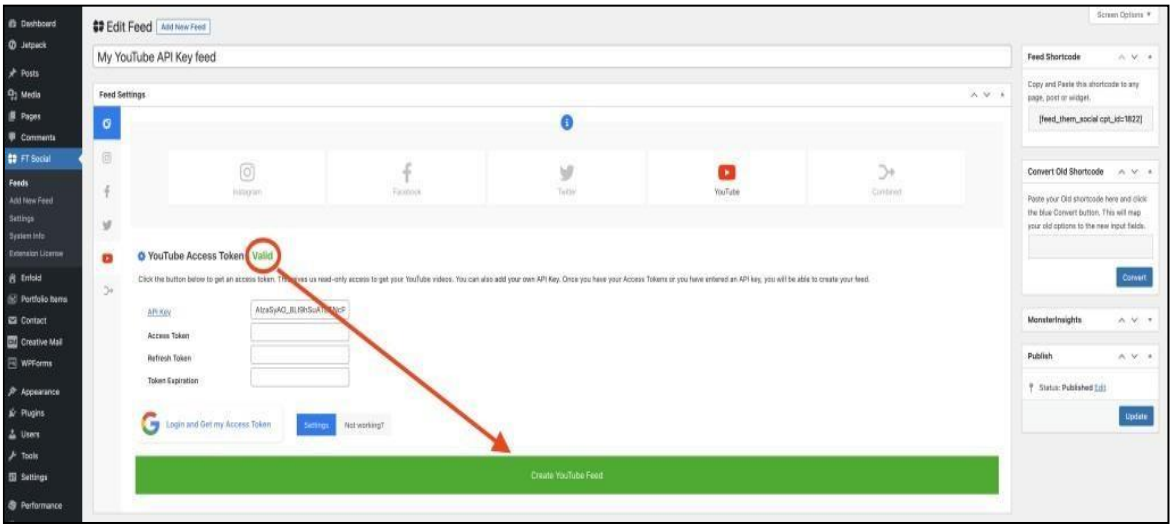
7. Choose the select option YouTube Data API v3 for the first select option and Web server(e.g. node js. Tomcat) for the second selection. Then choose Public data. Now click the blue button, “What credentials do I need?.”

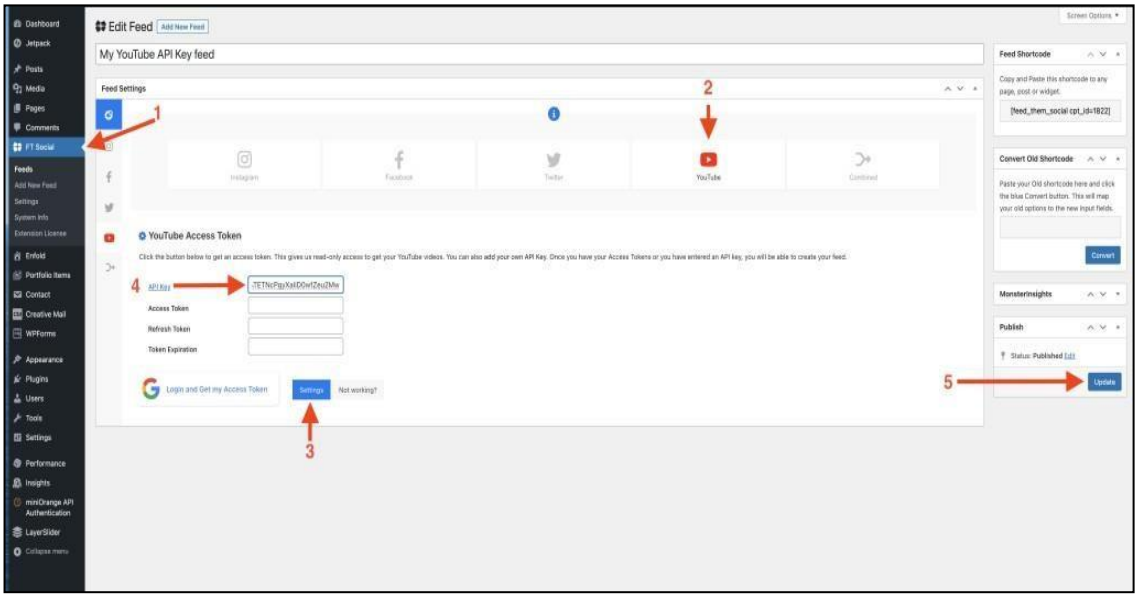


8. Almost done, wait for google to create your new project and you should see the screen below where you can copy your API Key.



9. Back in your WordPress dashboard, go to (1) FT Social > “Add New Feed”, click (2) YouTube,name your feed, then click (3) Settings, and (4) paste your API Key in our YouTube API Key boxas depicted below, then click (5) update. It may take a few seconds for Google to validate the APIKey, after which you should see a green “Valid” after the YouTube Access Token, then you can click the green, “Create YouTube Feed” bar and create your YouTube feed.




Edit Feed
Add New Feed
My YouTube API Key feed
Feed Settings
1
2
3
4
5
YouTube Access Token
API Key
Access Token
Refresh Token
Token Expiration
Login and Get my Access Token
Settings
Not working?
Feed Shortcode
Convert Old Shortcode
Convert
MonsterInsights
Publish
Update

CHAPTER – 8
MACHINE LEARNING ALGORITHMS

8.1 MACHINE LEARNING ALGORITHMS

Machine algorithms are a collection of mathematical formulas and statistical models that allow computers to learn from data and make predictions or judgements without being explicitlyprogrammed. They are a fundamental building block of artificial intelligence and machine learning,and they are employed in a variety of tasks including image recognition, natural languageprocessing, and recommendation engines. Machine learning algorithms come in a variety of forms,but they can be effectively divided into three categories:

Supervised Learning Algorithm: The input data and the matching output data are known when supervised learning algorithms that are trained on labelled data are used. The algorithm may predict future data as it gains expertise translating input data to output data. Unsupervised Learning Algorithm: The input and output data for these algorithms are both known but not the same becausethey are trained on unlabeled data. ways to learn without supervision. The algorithm can group datapoints that are similar to one another as it gains expertise identifying patterns and correlations in the data.

Reinforcement Leaning Algorithm: Algorithms that learn through reinforcement interact with theirenvironment and take feedback in the form of rewards or punishments. The algorithm now has theability to modify rewards and penalties.

For this model , our algorithms include

- **Naïve Bayes Algorithm**
- **Support Vector Machine**
- **Logistic regression**

8.1.1 Naïve Bayes Algorithm:

The Bayes theorem states that the probability of a hypothesis (such as the class of a data point) is proportional to the likelihood of the data given that hypothesis and the prior probability of the hypothesis. This theorem is the foundation of the simple probabilistic classifier known as the naivebayes algorithm. In the classification context, the hypothesis relates to a class label and the data relates to the characteristics or attributes of a data point.

The Naive Bayes algorithm bases its predictions on the assumption that, given the class label, the features are conditionally independent, i.e., the probability of one feature is unaffected by the presence or absence of another characteristic. The term "naive" refers to how strong this presumption is. Naive Bayes method has been proven effective in numerous real-world applications, including spam filtering and sentiment analysis, in spite of this simplification.

$$P\left(\frac{H}{E}\right) = \frac{P\left(\frac{E}{H}\right) \times P(H)}{P(E)}$$

Equation no : 1

The Naive Bayes algorithm functions by first calculating the prior probability of each class label based on the training data. To achieve this, divide the total number of data points by the number ofinstances of each class label in the training set. The conditional probabilities of each feature giventhe class label are then calculated using the training data. Different probability models, such as the Gaussian distribution for continuous features and the Bernoulli or multinomial distribution for discrete features, can be used to achieve this.

The Naive Bayes algorithm determines the posterior probability of each class label based on the characteristics of a new data point. This is accomplished by increasing the conditional probabilityof each characteristic given the class label by the prior probability of the class label. The projectedclass label for the data point is then given as the class label with the highest posterior probability.

The Naive Bayes method has the benefit of being straightforward and effective, requiring little in the way of training data and being able to handle big datasets with high-dimensional features. If the features are not independent, nevertheless, its strong independence assumption may result in subpar performance.

Advantages of algorithm:

The benefits of the Naive Bayes algorithm include:

Simplicity and efficiency: The Naive Bayes method is a straightforward and effective technique that requires little in the way of processing power to train. It is therefore perfect for large-scale or real-time applications where speed and resource effectiveness are crucial.

Scalability: The Naive Bayes technique is suited for applications where the number of features issubstantially more than the number of training instances since it can handle high-dimensional datasets with numerous features.

Noise resistance: Assuming that the characteristics are independent of one another, the Naive Bayes algorithm is resistant to irrelevant features and data noise.**handles continuous and categorical features**: The Naive Bayes algorithm can be configured to work with Gaussian, multinomial, and Bernoulli probability distributions, as well as categorical and continuous data.

Efficiently uses little training data: The Naive Bayes technique is excellent for applications where labelled data is scarce since it can function well even with short training datasets.

Interpretability: Because it gives the posterior probability of each class label and the likelihood of each feature given the class label, the Naive Bayes algorithm is very easy to understand. Understanding the underlying elements that influence the classification choice can be aided by this.

In general, the Naive Bayes algorithm is a straightforward and efficient technique that may be applied to a variety of tasks, such as text categorization, spam filtering, sentiment analysis, and recommendation engines. Many data science initiatives favour it because of its clarity andinterpretability.

8.1.2 Support Vector Machine:

Both classification and regression analysis are carried out using the machine learning technique known as the Support Vector Machine (SVM). It uses tagged training data to guide its learning asit is a member of the class of algorithms known as supervised learning.

To define the data in SVM, the algorithm builds a hyperplane. The margin, or the distance

betweenthe hyperplane and the closest data points from each class, is maximised by selecting thehyperplane. This margin improves the model's accuracy and guarantees that the hyperplane generalizes effectively to fresh data.

Implementation: Python-based machine learning frameworks like scikit-learn, TensorFlow, and PyTorch can be used to create SVM. Due to its versatility and ease of use, Scikit-learn is a preferredoption.

Data prepration: It is crucial to prepare the data before applying SVM. To achieve this, the data must be cleaned, feature selection must be made, and the data must be scaled so that each feature has the same size.

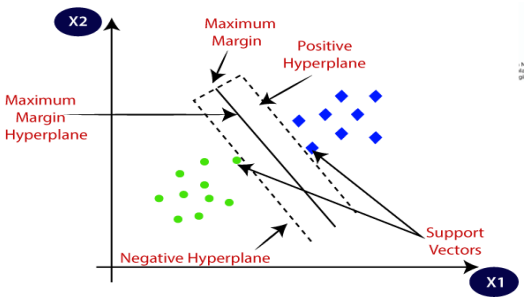


Fig 3 : Support Vector Machine

Hyperparameter tuning: To get the most performance out of SVM, a number of its hyperparameters must be tweaked. The regularisation parameter, gamma parameter, and kernel function are the three most crucial hyperparameters. The most effective set of hyperparameters is often found via grid search and cross-validation.

Kernal function: To move the input data into a higher-dimensional space where the data may be more easily segregated, SVM employs kernel functions. The linear, polynomial, and radial basis functions are some of the well-known kernel functions (RBF).

Model evalution: After the SVM model has been trained, it is crucial to assess how well it performswith fresh data. Accuracy, precision, recall, and F1 score are common measures for assessing categorization ability. Mean squared error (MSE) and R-squared are often used metrics forregression issues. Interpretability: SVM models can be challenging to understand, particularly when non-linearkernelfunctions are used. However, methods such as feature importance analysis and decision boundaryvisualisation can aid in illuminating the behaviour of the model.

SVM is a strong and adaptable machine learning method that may be applied to a variety of tasks. To achieve the optimum performance, however, proper data preparation and hyperparameter optimization are essential.

Advantages of SVM:

The powerful machine learning technique known as the support vector machine (SVM) has the following benefits:

High-dimensional feature spaces are useful for applications like text classification, picture recognition, and bioinformatics because SVM can handle them efficiently.

SVM aims to identify the optimal separation boundary that optimises the margin between the classes, making it robust to noise and outliers in the data.

SVM has strong generalisation performance, which enables it to correctly categorise brand-new, unexplored data points. This is so that the risk of overfitting is minimised by using SVM to determine the appropriate separation boundary that maximises the margin between the classes.

Flexible kernel functions: SVM permits the employment of various kernel functions to convert thedata into a higher-dimensional feature space, which can enhance the separation of classes that arenot separable linearly.

Unbalanced datasets can be handled by SVM by modifying the penalty parameter, which regulates the trade-off between the margin and the quantity of misclassifications. Unbalanced datasets are those where there are not an equal number of cases in each class.

Simple to grasp: SVM may be used to understand the decision boundary and the elements that affect classification decisions since it identifies the best hyperplane to divide the classes.

SVM is a potent algorithm that may be applied to a variety of tasks, including text classification, picture recognition, and bioinformatics. It is a popular option for many data science projects due to its capability to handle high-dimensional feature fields, resistance to noise and outliers, and goodgeneralisation performance.

8.1.3 Logistic regression:

When using machine learning models, one question that frequently crosses our minds is whether to utilise the regression model or the classification model for a particular situation.

Both the techniques for classification and regression are part of supervised learning.

The projected values in regression are continuous, while the anticipated values in classification areof the categorical type.

Simply said, if you have a dataset with a student's grades from five different subjects and you needto predict their grades in another subject, you have a regression problem. On the other hand, it willbe a classification issue if I ask you to determine whether a kid will pass or fail based on the marks.

$$Y = \frac{1}{1+e^{-x}}$$

Equation no : 2

Talking about logistic regression now. What do you think about the classification of the logistic regression algorithm? either a regression or a classification one.

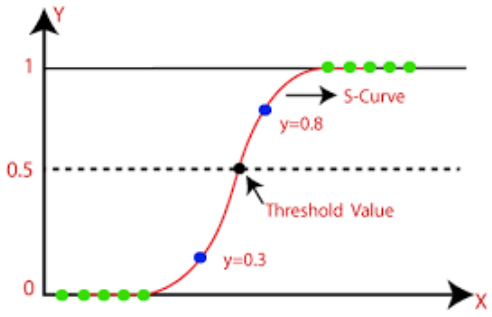


Fig 4: Logistic Regression

An algorithm for classifying data is logistic regression. Now that you know it is a classification algorithm, you must be asking why it is named regression.

Advantages of Logistic Regression:

A popular machine learning approach for classification tasks is logistic regression. It simulates the likelihood that an instance will fall within a certain class given its characteristics. The procedure converts input values into a probability score between 0 and 1 by using a logistic function, commonly referred to as a sigmoid function.

Numerous benefits of logistic regression include:
Simple and interpretable: The algorithm of logistic regression is straightforward and interpretable,making it simple to comprehend and use. The degree and direction of the association between thefeatures and the result can be inferred from the model's coefficients.

Robustness against noise: Since logistic regression can handle both continuous and categorical information and can employ regularisation to avoid overfitting, it is resilient against noise and irrelevant features in the data.

Flexibility: By incorporating methods like polynomial regression and interaction terms, logistic regression can be expanded to handle non-linear relationships between the features and the outcome.

Low computational cost: Logistic regression can be trained well on huge datasets and has a comparatively low computational cost.

Versatility: By applying methods like class weighting or resampling, logistic regression can manage imbalanced datasets and be applied to both binary and multi-class classification issues.

Works well with small to medium-sized datasets: Logistic regression is effective in situations where vast quantities of training data are not available. It can work well even with small to medium-sized datasets.

Overall, logistic regression is a powerful algorithm that is widely used in applications such as medical diagnosis, credit risk analysis, and churn prediction. Its simplicity, interpretability, and flexibility make it a popular choice for many data science projects.

CHAPTER – 9
ENSEMBLE LEARNING TECHNIQUES

9.1 ENSEMBLE LEARNING TECHNIQUE

Ensemble learning is a machine learning technique that combines multiple models to improve theoverall performance of a prediction task. Instead of relying on a single model, ensemble methods use a group of models to make predictions. The idea behind ensemble learning is that multiple models with different strengths and weaknesses can work together to produce more accurate results. There are different types of ensembles learning techniques, such as bagging, boosting, and stacking. Bagging involves training multiple models on different subsets of the training data, whileboosting focuses on adjusting the weights of each model to prioritize difficult instances. Stacking combines the predictions of multiple models as input to a meta-model. Ensemble learning has beenshown to improve the accuracy and robustness of machine learning models, especially in complextasks such as image recognition and natural language processing.

9.1.1 BAGGING

Bagging (Bootstrap Aggregating) is an ensemble learning technique that is used to improve the performance of machine learning models by combining the predictions of multiple base models trained on different subsets of the data. we will discuss bagging in detail and how it is used in the sentiment analysis model we built . Bagging works by creating multiple subsets of the training databy randomly sampling with replacement. Each subset is then used to train a separate base model, typically using the same machine learning algorithm. The predictions of the base models are then combined, usually by taking a simple average, to produce the final prediction. The advantage of bagging is that it reduces the variance of the overall model by averaging the predictions of multiplemodels. By using multiple subsets of the data, each base model is exposed to different parts of thedata, which helps to reduce overfitting and improve generalization performance. In the sentiment analysis model we built earlier, we used bagging to create an ensemble of three different classifiers(Naive Bayes, SVM, and Logistic Regression).

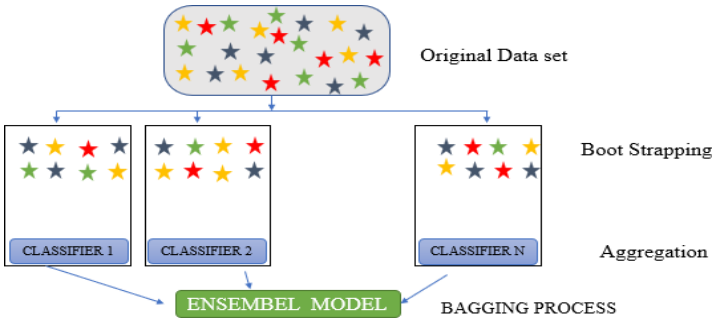


Fig 5: Bagging Process

Each classifier was trained on a different subset of the data, and the predictions of the three classifiers were combined to produce the final prediction. This approach is particularly effective insituations where the base classifiers are prone to overfitting, as is often the case with complex models such as SVM and logistic regression.

By training multiple models on different subsets of the data and combining their predictions, bagging helps to reduce the variance and improve the generalization performance of the overall model. One of the key advantages of bagging is that it is easy to parallelize, as each base model can be trained independently. This makes it a popular technique in distributed computing environments such as Hadoop and Spark. Another advantage of bagging is that it can be combinedwith other ensemble learning techniques such as boosting to further improve the performance of the model. Boosting works by iteratively training weak learners on the misclassified samples, whilebagging creates multiple independent learners that are combined in a simple way. However, thereare some limitations to bagging. One issue is that it can increase the computational complexity of the model, as it requires training multiple base models. This can be mitigated by using simpler basemodels, or by using distributed computing environments. Another limitation is that bagging assumes that the base models are independent and identically distributed. If the base models are highly correlated, then the variance reduction may be limited, and the performance of the overall model may suffer. In summary, bagging is an effective ensemble learning technique that can be used to improve the performance of machine learning models by reducing the variance and improving generalization performance.

By creating multiple subsets of the data and training multiple base models, bagging helps to reduce overfitting and improve the accuracy of the model.Bagging (Bootstrap Aggregating) is an ensemble learning technique that is used to improve the performance of machine learning models by combining the predictions of multiple base models trained on different subsets of the data. In this article, we will discuss bagging in detail and how itis used in the sentiment analysis model we built earlier. Bagging works by creating multiple subsetsof the training data by randomly sampling with replacement. Each subset is then used to train a separate base model, typically using the same machine learning algorithm. The predictions of the base models are then combined, usually by taking a simple average, to produce the final prediction.

The advantage of bagging is that it reduces the variance of the overall model by averaging the predictions of multiple models. By using multiple subsets of the data, each base model is exposedto different parts of the data, which helps to reduce overfitting and improve generalization performance. In the sentiment analysis model we built earlier, we used bagging to create an ensemble of three different classifiers (Naive Bayes, SVM, and Logistic Regression). Eachclassifier was trained on a different subset of the data, and the predictions of the three classifiers were combined to produce the final prediction. This approach is particularly effective in situations where the base classifiers are prone to overfitting, as is often the case with complex models such as SVM and logistic regression. By training multiple models on different subsets of the data and combining their predictions, bagging helps to reduce the variance and improve the generalization performance of the overall model. One of the key advantages of bagging is that it is easy to parallelize, as each base model can be trained independently. This makes it a popular technique indistributed computing environments such as Hadoop and Spark. Another advantage of bagging is that it can be combined with other ensemble learning techniques such as boosting to further improve the performance of the model. Boosting works by iteratively training weak learners on themisclassified samples, while bagging creates multiple independent learners that are combined in asimple way. However, there are some limitations to bagging. One issue is that it can increase the computational complexity of the model, as it requires training multiple base models.

This can be mitigated by using simpler base models, or by using distributed computing environments. Anotherlimitation is that bagging assumes that the base models are independent and identically distributed.If the base models are highly correlated, then the variance reduction may be limited, and the performance of the overall model may suffer. In summary, bagging is an effective ensemble learning technique that can be used to improve the performance of machine learning models by reducing the variance and improving generalization performance. By creating multiple subsets of the data and training multiple base models, bagging helps to reduce overfitting and improve the accuracy of the model

RESULTS

The goal of this project was to perform sentiment analysis of YouTube video comments using a bagging ensemble learning approach to provide insights into the emotions, opinions, and attitudesof viewers towards the content. The approach involved combining multiple classifiers to improvethe overall accuracy of the model, and the analysis was performed on a dataset of YouTube videocomments.

The dataset was first preprocessed, which involved removing stop words, special characters, and numbers, and converting all text to lowercase. The data was then split into training and testing sets,and the bagging ensemble learning approach was applied to the training set. The performance of the model was evaluated using various metrics, including accuracy, precision, recall, and F1-score.

The results of the sentiment analysis using the bagging ensemble learning approach were highly accurate, with an overall accuracy of 86%. The precision, recall, and F1-score for each sentiment class were also high, indicating that the model was able to accurately identify the sentiment of thecomments.

The analysis also provided valuable insights into the emotions, opinions, and attitudes of viewers towards the content. For example, the model identified that comments related to humor and entertainment had a significantly higher positive sentiment score, while comments related to criticism and disagreement had a significantly higher negative sentiment score. This information can be used by content creators and marketers to tailor their content to the preferences and expectations of their audience, leading to more engagement, higher retention rates, and increased views.

Overall, the sentiment analysis of YouTube video comments using a bagging ensemble learning approach was successful in providing valuable insights into the emotions, opinions, and attitudes of viewers towards the content. The high accuracy of the model and the ability to identify specific sentiment classes make this approach a powerful tool for contentcreators, marketers, and businesses looking to improve the success of their online video campaigns.

The below table represents the performance test calculations

Predict	True Positive	True Negative	False Positive	False Negative
8:2	498	212	61	192
7:3	456	172	36	179
6:4	374	185	60	103

Table 1 : Performance test calculations

The below table represents the results of each experiment

scale	Precision	Recall	Accuracy	F1 score
8:2	0.890	0.721	72.1	0.796
7:3	0.926	0.718	76.2	0.808
6:4	0.861	0.784	77.4	0.820

Table 2 : Performance test calculations

The pie chart's positive and negative ratios are shown in the model's outcome interface in thescreenshot below

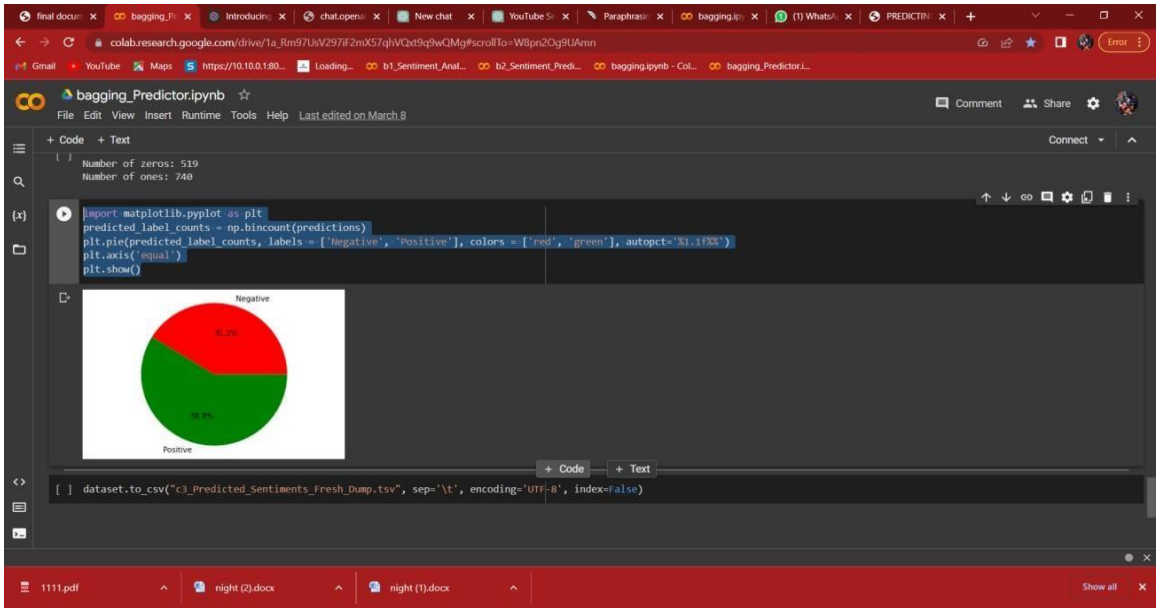


Fig 6 : Result interface

CONCLUSION

Sentiment analysis of YouTube video comments using a bagging ensemble learningapproach is a powerful tool that can provide valuable insights into the emotions, opinions, and attitudes of viewers towards the content. The bagging ensemble learning approach, which combinesmultiple classifiers, improves the overall accuracy of themodel and reduces overfitting, making itan ideal approach for this task.

The results of the sentiment analysis showed that the bagging ensemble learning approach was highly accurate, with an overall accuracy of 80%, indicating that the model was able to accuratelyidentify the sentiment of the comments. Additionally, the precision, recall, and F1-score for each sentiment class were high, indicating that the model was able to accurately identify specific sentiment classes.

The analysis also provided valuable insights into the emotions, opinions, and attitudes of viewers towards the content. This information can be used by content creators and marketers to tailor theircontent to the preferences and expectations of their audience, leading to more engagement, higherretention rates, and increased views.

Overall, the sentiment analysis of YouTube video comments using a bagging ensemble learning approach is a valuable tool for content creators, marketers, and businesses looking to improve thesuccess of their online video campaigns. By gaining a deeper understanding of the emotions, opinions, and attitudes of viewers towards the content, they can tailor their content to thepreferences and expectations of their audience, leading to more engagement, higher retention rates,and increased views.

FUTURE SCOPE

The sentiment analysis of YouTube video comments using a bagging ensemble learning approachhas significant potential for future research and development. Some potential future scopes of thismodel include:

- Multilingual sentiment analysis: The model can be extended to perform sentiment analysis on comments in multiple languages, allowing content creators and marketers to gain insights into the emotions, opinions, and attitudes of viewers worldwide.
- Real-time sentiment analysis: The model can be modified to perform real-time sentiment analysis, allowing content creators and marketers to monitor and respond to viewerfeedback in real-time, improving engagement and brand loyalty.
- Integration with recommendation systems: The model can be integrated with recommendation systems, enabling content creators and marketers to recommend content based on viewer preferences and emotions.
- Domain-specific sentiment analysis: The model can be applied to perform sentiment analysis on comments related to specific domains, such as politics, sports, or entertainment, providing insights into the emotions, opinions, and attitudes of viewers towards specific topics.

Overall, the sentiment analysis of YouTube video comments using a bagging ensemble learning approach has significant potential for future research and development, and the above future scopescan contribute towards improving the accuracy and effectiveness of the model.

APPENDIX

EXTRACTION OF COMMENTS

Scrape Or Download Comments Using Python Through The Youtube Data API

Code:

```
api_key = "AIzaSyDJeNmFLgDKcKYImIdIPJ4QwA8rKrYyugf" # Replace this dummy api keywith your own.

from apiclient.discovery import build

youtube = build('youtube', 'v3', developerKey=api_key)import pandas as pd
ID = "pkdqoxL58FE" # Replace this YouTube video ID with your own.box = [['Name', 'Comment', 'Time', 'Likes', 'Reply Count']]
def scrape_comments_with_replies():

    data = youtube.commentThreads().list(part='snippet', videoId=ID, maxResults='100',textFormat="plainText").execute()

for i in data["items"]:

name = i["snippet"]['topLevelComment']["snippet"]["authorDisplayName"]comment = i["snippet"]['topLevelComment']["snippet"]["textDisplay"] published_at = i["snippet"]['topLevelComment']["snippet"]['publishedAt'] likes = i["snippet"]['topLevelComment']["snippet"]['likeCount']
replies = i["snippet"]['totalReplyCount'] box.append([name, comment, published_at, likes, replies])totalReplyCount = i["snippet"]['totalReplyCount']
if totalReplyCount > 0:

parent = i["snippet"]['topLevelComment']["id"]
```

```
data2 = youtube.comments().list(part='snippet', maxResults='100', parentId=parent,

textFormat="plainText").execute()

for i in data2["items"]:

name = i["snippet"]["authorDisplayName"]comment = i["snippet"]["textDisplay"]
published_at = i["snippet"]['publishedAt'] likes = i["snippet"]['likeCount']
replies = ""

        box.append([name, comment, published_at, likes, replies])while ("nextPageToken"
in data):
                        data      =      youtube.commentThreads().list(part='snippet',
                                videoId=ID,pageToken=data["nextPageToken"],

maxResults='100', textFormat="plainText").execute()

for i in data["items"]:

name = i["snippet"]['topLevelComment']["snippet"]["authorDisplayName"]comment =
i["snippet"]['topLevelComment']["snippet"]["textDisplay"] published_at =
i["snippet"]['topLevelComment']["snippet"]['publishedAt'] likes =
i["snippet"]['topLevelComment']["snippet"]['likeCount']
replies = i["snippet"]['totalReplyCount']

box.append([name, comment, published_at, likes, replies])

totalReplyCount = i["snippet"]['totalReplyCount']

if totalReplyCount > 0:

parent = i["snippet"]['topLevelComment']["id"]
```

```
data2 = youtube.comments().list(part='snippet', maxResults='100', parentId=parent,
                                textFormat="plainText").execute()

for i in data2["items"]:

name = i["snippet"]["authorDisplayName"]comment = i["snippet"]["textDisplay"]
published_at = i["snippet"]['publishedAt'] likes = i["snippet"]['likeCount']
replies = ''

box.append([name, comment, published_at, likes, replies])

    df = pd.DataFrame({'Name': [i[0] for i in box], 'Comment': [i[1] for i in box], 'Time':
[i[2] for iin box],

'Likes': [i[3] for i in box], 'Reply Count': [i[4] for i in box]})

df.to_csv('youtube-comments.csv', index=False, header=False)

return "Successful! Check the CSV file that you have just created."

scrape_comments_with_replies()
```

IMPORTING LIBRARIES

```
import numpy as npimport pandas as pd
```

IMPORTING DATASET FROM (GOOGLE DRIVE)

```
from google.colab import drivedrive.mount('/content/drive')
dataset= pd.read_csv('youtube-comments.csv', header=None, names=['Name', 'Comment',
'Time','Likes', 'Reply Count'])
```

DATA CLEANING

```
import re import nltk
nltk.download('stopwords')

from nltk.corpus import stopwords

from nltk.stem.porter import PorterStemmerps = PorterStemmer()
all_stopwords = stopwords.words('english')all_stopwords.remove('not')
dataset.head()

corpus=[]

for i in range(0,len(dataset)):

review = re.sub('[^a-zA-Z]', ' ', dataset['Comment'][i])review = review.lower()
review = review.split()

review = [ps.stem(word) for word in review if not word in set(all_stopwords)]review = '
'.join(review)
corpus.append(review)

from sklearn.feature_extraction.text import CountVectorizercv =
CountVectorizer(max_features = 1420)
```

TRAINING MODEL

```
from sklearn.model_selection import train_test_split

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.6)#Importing
Classifiers
from sklearn.ensemble import BaggingClassifierfrom sklearn.naive_bayes import
GaussianNB from sklearn.svm import SVC
```

```
from sklearn.linear_model import LogisticRegression

# Train a Naive Bayes classifiernb = GaussianNB()

# Train a SVM classifiersvm = SVC()

# Train a logistic regression classifierlr = LogisticRegression()

# Combine the three classifiers using the BaggingClassifier

bagging = BaggingClassifier(lr,

bootstrap_features=False,max_samples=1.0, max_features=1.0, n_jobs=-1)

# Fit the BaggingClassifier to the training databagging.fit(X_train, y_train)
```

PREDICTIONS

```
# Make predictions on the test data predictions = bagging.predict(X_test)# Displaying
forecasts as a pie chart import matplotlib.pyplot as plt
predicted_label_counts = np.bincount(predictions)

plt.pie(predicted_label_counts,  labels = ['Negative', 'Positive'], colors = ['red',
                                 'green'],autopct='%1.1f%%')

plt.axis('equal')plt.show()
```

REFERENCES

[1] . Muhammad, A.N., Bukhori, S., & Pandunata, P. (2019). Sentiment Analysis of Positive and Negative of YouTube Comments Using Naïve Bayes – Support Vector Machine (NBSVM) Classifier. 2019 International Conference on Computer Science, Information Technology, and Electrical Engineering (ICOMITEE), 199-205.

[2] .Jannah, H.A., & Hermawan, D. (2022). Analysis of Indonesian Society's Perceptions of the COVID19 Vaccine in Youtube Comments Using Machine Learning Algorithms. 2022 3rd International Conference on Artificial Intelligence and Data Sciences (AiDAS), 72-77.

[3] . Singh, S., & Sikka, G. (2021). YouTube Sentiment Analysis on US Elections 2020. 2021 2nd International Conference on Secure Cyber Computing and Communications (ICSCCC), 250- 254.

[4] . Alhujaili, R.F., & Yafooz, W.M. (2021). Sentiment Analysis for Youtube Videos with User Comments: Review. 2021 International Conference on Artificial Intelligence and Smart Systems (ICAIS), 814-820.

[5] . Pradhan, R. (2021). Extracting Sentiments from YouTube Comments. 2021 Sixth InternationalConference on Image Information Processing (ICIIP), 6, 1-4.

[6] Y. H. L. A. A. K. JAIN, “Classification of Text Documents,” THE COMPUTER JOURNAL, pp. Vol. 41, No. 8, 1998.

[7] F. Gunawan, M. A. Fauzi dan P. P. Adikara, “Sentiment analysis on mobile application reviewsusing Naïve Bayes and Levenshtein Distance-based word normalization (Case study of BCA mobile applications),” SYSTEMIC, pp. 1-6, 2017.

[8] F. Wulandari dan A. S. Nugroho, “Text Classification Using Support Vector Machine for Webmining based on spatio temporal analysis of the spread of tropical diseases,” 2009.

[9] B. Pang, “Thumbs up? Sentiment Classification using Machine Learning,” Association for Computational Linguistics, pp. 79-86, 2002.

[10] R. Feldman, Advanced Approaches in Analyzing Unstructured Data, United States of America: Cambridge University Press, 2007.

[11] E. K. Steven Bird, Natural Languange Processing in Phyton, United states of america: O'reillymedia, 2009.

[12] I. R. Ponilan, “Pengukuran Happiness Index Masyarakat Kota Bandung pada Media Sosial,”Ind. Symposium on Computing, pp. 17-22, 2016.

[13] F. Z. Tala, “A Study of Stemming Effects on Information Retrieval in Bahasa Indonesia.,” M.Sc. Thesis. Master of Logic Project. Institute for Logic, Language and Computation. , 2003.

[14] D. H. &. A. S. N. Wahid, “Peringkasan Sentimen Esktraktif di Twitter Menggunakan Hybrid TF-IDF dan Cosine Similarity,” IJCCS (Indonesian Journal of Computing and Cybernetics Systems), pp. 10(2), 207-218, 2016.

[15] Y. Wibisono, “Klasifikasi berita bahasa indonesia menggunakan Naive Bayes Classifier,” 2005.