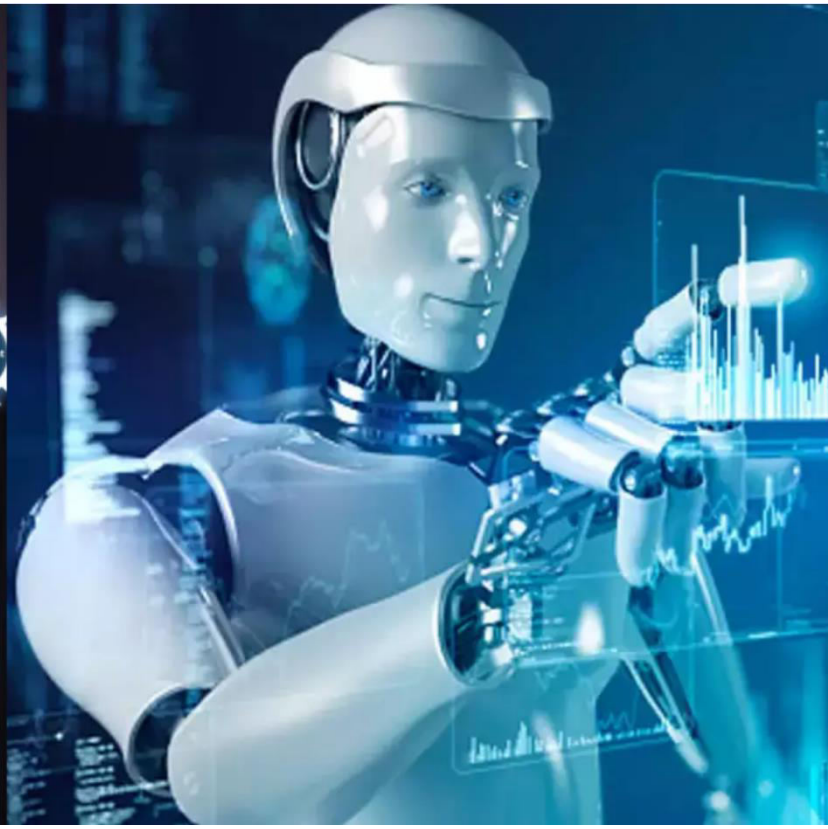
IJIRSET
International Journal of Innovative Research in Science | Engineering | Technology

ISSN(O): 2319-8753
ISSN(P): 2347-6710

International Journal of Innovative Research in Science Engineering and Technology (IJIRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)



Impact Factor: 8.699

Volume 14, Issue 4, April 2025

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 14, Issue 4, April 2025

|DOI: 10.15680/IJIRSET.2025.1404389|

Understanding Public Sentiments through YouTube Comments: A Machine Learning Approach

Byravarapu Prashant [1], Merugumala Vamsi Krishna Eswar [2], Sonti Gangaratnam [3], Vasa Deepthi [4], Shaik Bayazid [5]

Associate Professor& HoD of CSE-Data Science, Eluru College of Engineering & Technology, Eluru, India[1]

B.Tech Student, Department of CSE, Eluru College of Engineering & Technology, Eluru, India[2,3,4,5]

ABSTRACT: With the exponential increase in textual data, research in machine learning (ML) and natural language processing (NLP) has expanded significantly. Sentiment analysis of YouTube comments has become a particularly interesting topic due to the large volume of user interactions. However, extracting meaningful trends from these comments remains a challenge due to inconsistencies and varying quality. This study performs sentiment analysis on YouTube comments related to popular topics using various ML algorithms. By analyzing sentiment trends, seasonality, and forecasts, we demonstrate how real-world events influence public sentiments. Results indicate a strong correlation between user sentiment trends and associated events. This research aims to assist scholars in identifying quality sentiment analysis studies. Utilizing an annotated corpus of 1500 citation sentences, noise was removed through data normalization. Six ML algorithms—Naïve Bayes (NB), Support Vector Machine (SVM), Logistic Regression (LR), Decision Tree (DT), K-Nearest Neighbor (KNN), and Random Forest (RF)—were employed to classify the dataset. The system's accuracy was evaluated using F-score and accuracy metrics.

KEYWORDS: Machine Learning, Natural Language Processing, Support Vector Machine and Logistic Regression.

I. INTRODUCTION

With the rise of digital platforms, YouTube has become one of the most influential sources of information, entertainment, and public discourse. Understanding public sentiment through YouTube comments provides valuable insights into audience opinions, trends, and engagement levels.

This study explores a machine learning approach to analyze YouTube comments for sentiment classification. By leveraging natural language processing (NLP) techniques, the study aims to classify comments into positive, negative, or neutral sentiments. Various machine learning models, such as Naïve Bayes, Support Vector Machines (SVM), and deep learning-based models, are evaluated for accuracy and efficiency.

The research highlights the importance of sentiment analysis in areas like brand perception, political discourse, and customer feedback. By automating sentiment detection, businesses, policymakers, and content creators can make data-driven decisions based on public opinion.

1.1 MOTIVATION

Understanding public sentiment through YouTube comments using machine learning is a powerful way to analyze audience opinions at scale. With millions of users expressing thoughts on videos daily, sentiment analysis can reveal trends, public perception, and engagement patterns. Machine learning techniques, particularly **Natural Language Processing (NLP)**, enable automated classification of sentiments—positive, negative, or neutral—offering valuable insights for businesses, content creators, and researchers. This approach helps in **real-time decision-making**, improving content strategies, brand reputation management, and understanding societal trends more effectively.

1.2 PROBLEM DEFINITION

Understanding public sentiment through YouTube comments is crucial for analyzing opinions on various topics. However, the large volume, unstructured nature, and presence of slang, sarcasm, and multilingual content make manual analysis difficult. This study aims to develop a machine learning-based sentiment analysis model to automatically classify comments as positive, negative, or neutral. Using Natural Language Processing (NLP) techniques, the model

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 14, Issue 4, April 2025

|DOI: 10.15680/IJIRSET.2025.1404389|

will handle data preprocessing, sentiment detection, and spam filtering. The insights gained can benefit content creators, businesses, and policymakers by providing a clearer understanding of public opinions and trends.

1.3 OBJECTIVE OF THE PROJECT

The objective of this project is to analyze public sentiments expressed in YouTube comments using machine learning techniques. By collecting and processing comment data, the project aims to classify sentiments into categories such as positive, negative, and neutral. Using natural language processing (NLP) and various machine learning models, it seeks to identify trends, patterns, and audience reactions over time. The insights gained can help content creators, businesses, and policymakers understand public opinion, improve engagement strategies, and make data-driven decisions.

II. LITERATURE SURVEY

Literature Survey on Understanding Public Sentiments Through YouTube Comments.

Authors: Asghar.

Sentiment analysis of YouTube comments has gained significant attention in recent years due to the platform's vast user-generated content. Early studies relied on lexicon-based and traditional machine learning models like Naïve Bayes and Support Vector Machines (SVM) for sentiment classification. However, with advancements in natural language processing (NLP), deep learning models such as Long Short-Term Memory (LSTM) networks and transformers like BERT have shown improved accuracy in sentiment detection. Despite these advancements, analyzing YouTube comments presents challenges such as noisy data, slang, sarcasm, and multilingual content. Researchers have explored various preprocessing techniques to handle these complexities, but detecting nuanced emotions and sarcasm remains difficult. Applications of YouTube sentiment analysis include content recommendation, brand perception analysis, and understanding public opinion on social and political issues. While deep learning continues to enhance sentiment classification, future research should focus on hybrid approaches and real-time sentiment monitoring to address existing challenges effectively.

Predicting ability of machine learning methods for massive results.

Machine learning (ML) methods have demonstrated remarkable predictive abilities when dealing with massive datasets, making them essential for tasks such as sentiment analysis, recommendation systems, and pattern recognition. Traditional models like Naïve Bayes and Support Vector Machines (SVM) perform well for structured datasets but often struggle with scalability and handling complex relationships in large-scale data. In contrast, deep learning techniques such as Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), and transformer-based models like BERT and GPT have significantly improved prediction accuracy by capturing contextual relationships and handling high-dimensional data efficiently. The predictive ability of ML methods is further enhanced by feature engineering, hyperparameter tuning, and the use of pretrained embeddings such as Word2Vec or GloVe. However, challenges such as computational cost, overfitting, and interpretability remain key concerns. With advancements in cloud computing and distributed frameworks like TensorFlow and PyTorch, machine learning models are becoming more capable of processing massive datasets in real time, improving their predictive performance across various domains.

Sentiment Analysis of YouTube Comments Using Machine Learning Techniques.

Authors: V. Umarani.

Sentiment analysis of YouTube comments using machine learning techniques helps in understanding public opinions, user engagement, and content reception. The process involves collecting and preprocessing comments to remove noise, slang, and emojis before applying machine learning models for sentiment classification. Traditional models like Naïve Bayes and Support Vector Machines (SVM) rely on handcrafted features, while deep learning approaches such as Long Short-Term Memory (LSTM) networks and transformer-based models like BERT offer improved accuracy by capturing contextual relationships in text. However, challenges such as noisy data, sarcasm detection, multilingual content, and class imbalance affect classification performance. Despite these challenges, sentiment analysis has significant applications in content recommendation, brand perception, and public opinion tracking. Future advancements in hybrid models, real-time analysis, and multimodal approaches integrating text, audio, and video can further enhance the accuracy and effectiveness of sentiment analysis on YouTube comments.

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 14, Issue 4, April 2025

|DOI: 10.15680/IJIRSET.2025.1404389|

Applications of Machine Learning Techniques in Sentiment Analysis

Authors: Oussama El Azzouzy.

Machine learning techniques play a crucial role in sentiment analysis across various industries, enabling automated and accurate interpretation of emotions in textual data. In social media monitoring, companies analyze user sentiments on platforms like YouTube and Twitter to track brand reputation and audience engagement. Businesses leverage sentiment analysis on customer reviews and feedback to enhance products and services. In politics and social research, analyzing public sentiment helps governments and organizations understand opinions on policies and elections. Financial institutions use sentiment analysis on news and social media to predict market trends, while streaming platforms improve content recommendations based on audience emotions. Additionally, healthcare applications monitor patient sentiment for better mental health support, and AI-driven models detect fake news, hate speech, and online abuse. By providing valuable insights, sentiment analysis powered by machine learning enhances decision-making across industries, from marketing and finance to public safety and entertainment.

A Model for Understanding Public Sentiments Through YouTube Comments

Authors: Ms. Julanta Leela Rachel.

A machine learning-based model for sentiment analysis of YouTube comments follows a structured approach, starting with data collection using the YouTube Data API or web scraping. The collected comments undergo preprocessing, where noise, special characters, and stopwords are removed, and text is tokenized and lemmatized for standardization. Feature extraction is performed using methods like TF-IDF, Word2Vec, or BERT embeddings to convert text into numerical representations. For sentiment classification, traditional models like Naïve Bayes and Support Vector Machines (SVM) provide baseline results, while deep learning models such as LSTM, CNN, and transformer-based architectures like BERT enhance accuracy by capturing contextual nuances. The model is evaluated using metrics like accuracy, precision, recall, and F1-score, with hyperparameter tuning for optimization. Finally, visualization tools like sentiment trend graphs and word clouds present insights, helping businesses, content creators, and policymakers understand audience sentiments. Future advancements could integrate multilingual support, sarcasm detection, and multimodal analysis (text, audio, video) to improve sentiment understanding.

Understanding Public Sentiments Through YouTube Comments Using ML Algorithms

YouTube has become a major platform for expressing opinions on various topics, making sentiment analysis of its comments essential for understanding public sentiment. Machine learning (ML) algorithms play a key role in automating this process by analyzing large volumes of comments and classifying them into categories such as positive, negative, or neutral. The process begins with data collection using YouTube APIs, followed by text preprocessing techniques like tokenization, stopword removal, and lemmatization to clean the data. Traditional ML models such as Naïve Bayes, Support Vector Machines (SVM), and Random Forest are effective for basic sentiment classification, but deep learning models like Long Short-Term Memory (LSTM) networks and transformer-based models like BERT provide higher accuracy by capturing contextual nuances in text. Despite challenges such as noisy data, sarcasm detection, and multilingual content, ML-based sentiment analysis helps content creators, businesses, and policymakers gain valuable insights into audience opinions. Applications include improving content recommendations, monitoring brand perception, and understanding public reactions to social and political events. With advancements in deep learning and real-time sentiment analysis, ML algorithms continue to enhance the accuracy and effectiveness of sentiment analysis on YouTube comments.

Predictive Model Construction for Understanding Public Sentiments Through YouTube Comments Using Machine Learning Algorithms

Constructing a predictive model for understanding public sentiments through YouTube comments using machine learning involves several key steps, starting with data collection through APIs or web scraping. The collected comments undergo preprocessing, including text cleaning, tokenization, and lemmatization, to remove noise and standardize the text. Feature extraction techniques like TF-IDF, Word2Vec, or BERT embeddings transform text into numerical representations for sentiment classification. Traditional machine learning models such as Naïve Bayes, Support Vector Machines (SVM), and Random Forest serve as baselines, while deep learning models like LSTM and transformer-based architectures like BERT improve accuracy by capturing contextual meaning. The model is trained, evaluated using performance metrics like accuracy and F1-score, and optimized through hyperparameter tuning. Once deployed, the model can provide real-time sentiment analysis, offering insights into public opinion trends, brand perception, and user engagement. Continuous learning mechanisms further enhance accuracy by retraining the model

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 14, Issue 4, April 2025

|DOI: 10.15680/IJIRSET.2025.1404389|

with new data, ensuring it adapts to evolving language patterns and sentiment expressions.

Recommendation system to improve Public Sentiments through YouTube Comments.

An ensemble technique can improve sentiment analysis accuracy by combining multiple models like Random Forest, XGBoost, and deep learning approaches (BERT or LSTMs). A stacking ensemble can integrate weak and strong classifiers, ensuring robust predictions, while bagging (e.g., Random Forest) reduces variance and boosting (e.g., XGBoost) enhances performance. Hybridizing aspect-based sentiment analysis with ensemble methods can refine sentiment detection. Additionally, using ensemble learning for recommendation—combining collaborative filtering, content-based filtering, and deep learning embeddings—can improve comment recommendations, ensuring both accuracy and diversity in sentiment-driven insights.

Analysis of public sentiments through you-tube comments using ML algorithms.

Analyzing public sentiments through YouTube comments using machine learning algorithms involves data collection via the YouTube API, followed by preprocessing techniques like text cleaning, tokenization, and word embedding (TF-IDF, Word2Vec, or BERT). Traditional ML models such as Naïve Bayes, SVM, and Random Forest provide baseline sentiment classification, while advanced approaches like LSTM, BiLSTM, and transformer-based models (BERT, RoBERTa) enhance contextual understanding. To improve accuracy, ensemble techniques like bagging (Random Forest), boosting (XGBoost), and stacking (combining SVM, LSTM, and BERT) can be used. A sentiment-based recommendation system integrates collaborative filtering and content-based filtering, leveraging embeddings to personalize recommendations. Finally, models are evaluated using F1-score, AUC-ROC, and precision-recall metrics and deployed via cloud platforms for scalable, real-time sentiment analysis.

III. SYSTEM ANALYSIS

3.1 EXISTING SYSTEM

The existing system for sentiment analysis of YouTube comments typically relies on traditional NLP techniques and machine learning models. Comments are collected using the YouTube API or web scraping tools, then preprocessed through text cleaning, tokenization, and stopword removal. Sentiment classification is commonly performed using machine learning algorithms like Naïve Bayes, Support Vector Machines (SVM), or Random Forest, as well as deep learning models like LSTMs and BERT. Many existing systems use lexicon-based approaches (e.g., VADER, TextBlob) for basic sentiment scoring. However, these systems face challenges such as handling sarcasm, multilingual comments, spam filtering, and real-time processing.

3.1.1: DIS-ADVANTAGES OF EXISTING SYSTEM

Inaccuracy in Context Understanding – Traditional models struggle with sarcasm, slang, and ambiguous language, leading to incorrect sentiment classification.

Limited Multilingual Support – Many systems primarily focus on English, making it difficult to analyze comments in different languages or mixed-language texts.

High Computational Cost – Deep learning models like BERT and LSTMs require significant computational power, making real-time analysis expensive.

Spam and Bot Comments – Existing systems may fail to filter out irrelevant or automated comments, affecting sentiment accuracy.

Data Imbalance Issues – Sentiment categories (positive, negative, neutral) may be imbalanced, leading to biased model predictions.

Scalability Challenges – Processing large volumes of YouTube comments in real-time is challenging due to resource limitations.

Lack of Real-time Insights – Many systems analyze sentiment in batches rather than providing real-time monitoring of public opinion.

3.2 PROPOSED SYSTEM

The proposed system aims to enhance sentiment analysis of YouTube comments by integrating advanced machine learning and deep learning techniques for improved accuracy, scalability, and real-time processing. It will use the YouTube API for real-time comment extraction and apply NLP-based preprocessing (tokenization, stemming, stopword removal, and slang interpretation). Feature extraction will be optimized using word embeddings (Word2Vec, BERT, or Transformer-based models) to capture contextual meaning, improving sentiment classification accuracy.

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 14, Issue 4, April 2025

|DOI: 10.15680/IJIRSET.2025.1404389|

Hybrid models combining deep learning (LSTMs, CNNs) and traditional ML (SVM, Random Forest) will be used to handle sarcasm and ambiguous text. The system will include spam filtering, multilingual support, and real-time sentiment tracking with visualization dashboards. By addressing the limitations of existing systems, this approach ensures faster, more reliable, and scalable sentiment analysis for better public opinion insights.

3.2.1: ADVANTAGES OF PROPOSED SYSTEM

Improved Accuracy – Advanced deep learning models (BERT, LSTMs, CNNs) capture contextual meanings, sarcasm, and nuanced sentiments more effectively.
Multilingual Support – The system can analyze comments in multiple languages using **multilingual NLP models**, making it more inclusive.
Real-Time Sentiment Analysis – Unlike batch-processing systems, the proposed approach enables real-time tracking of public sentiment.
Better Spam and Bot Filtering – Incorporates AI-driven spam detection to remove irrelevant or automated comments, ensuring cleaner data.
Scalability and Efficiency – Uses optimized ML pipelines and cloud computing to handle large volumes of YouTube comments efficiently.
Contextual Understanding – Advanced NLP techniques like word embeddings (Word2Vec, BERT) help in understanding slang, sarcasm, and emojis better.
Comprehensive Visualization – Provides interactive dashboards to display sentiment trends, helping businesses and researchers analyze public opinions easily.
Adaptive and Self-Learning – The system can improve over time by continuously learning from new data, enhancing accuracy and relevance.

3.3 MODULES

Data Collection Module

- Uses the YouTube Data API v3 or web scraping (BeautifulSoup, Selenium) to fetch comments.
- Extracts metadata such as comment text, user ID, timestamp, and likes/dislikes.
- Stores raw data in a database (MySQL, MongoDB, Firebase, or CSV format) for processing.

Data Preprocessing Module

- Cleans the comments by removing special characters, URLs, emojis, and stopwords.
- Uses tokenization, stemming, and lemmatization for text normalization.
- Handles language detection and translation for multilingual comments.

Feature Extraction Module

- Converts text into numerical features using TF-IDF, Word2Vec, BERT embeddings, or FastText.
- Identifies key topics, entities, and sentiment-related words.
- Applies dimensionality reduction (PCA, LDA) to improve model efficiency.

Sentiment Classification Module

- Implements machine learning models (Naïve Bayes, SVM, Random Forest) and deep learning models (LSTM, CNN, BERT) to classify sentiments as positive, negative, or neutral.
- Uses supervised learning with labeled datasets for training and evaluation.
- Fine-tunes models for better accuracy and context understanding.

Spam & Irrelevant Data Filtering Module

- Detects and removes spam comments, bot-generated text, and repetitive messages.
- Uses rule-based filtering, machine learning classifiers, and deep learning (RNN, Transformer-based models) to identify fake or irrelevant content.

Visualization & Reporting Module

- Displays sentiment trends through graphs, word clouds, and statistical reports.
- Uses visualization tools like Matplotlib, Seaborn, Power BI, or Tableau.

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 14, Issue 4, April 2025

|DOI: 10.15680/IJIRSET.2025.1404389|

- Provides real-time sentiment tracking dashboards for businesses, researchers, and analysts.

Real-time Processing & API Integration Module

- Enables real-time sentiment monitoring for live videos and trending topics.
- Integrates with external APIs or social media monitoring tools for broader analysis.
- Supports deployment in cloud environments (Google Cloud, AWS, or Azure) for scalability.

IV. SYSTEM DESIGN

4.1 SYSTEM ARCHITECTURE

1. **Data Layer (Backend & Storage)**
   - Collects YouTube comments using the YouTube API v3 or web scraping tools (BeautifulSoup, Selenium).
   - Stores raw and processed data in a database (MySQL, MongoDB, Firebase, or CSV format).
2. **Processing Layer (ML/NLP Engine)**
   - Preprocessing Module: Cleans text using tokenization, stopword removal, lemmatization, and stemming.
   - Feature Extraction Module: Converts text into numerical features using TF-IDF, Word2Vec, or BERT embeddings.
   - Sentiment Classification Module: Uses ML models (Naïve Bayes, SVM, Random Forest) and DL models (LSTM, CNN, BERT) to categorize sentiment.
   - Spam Filtering Module: Identifies and removes bot-generated, irrelevant, or spam comments.
3. **Presentation Layer (User Interface & Visualization)**
   - Web Dashboard: Displays sentiment trends using charts, word clouds, and statistical reports.
   - Real-Time Monitoring: Allows users to track sentiment fluctuations dynamically.
   - Reports & Insights: Provides downloadable analytics reports for businesses and researchers.

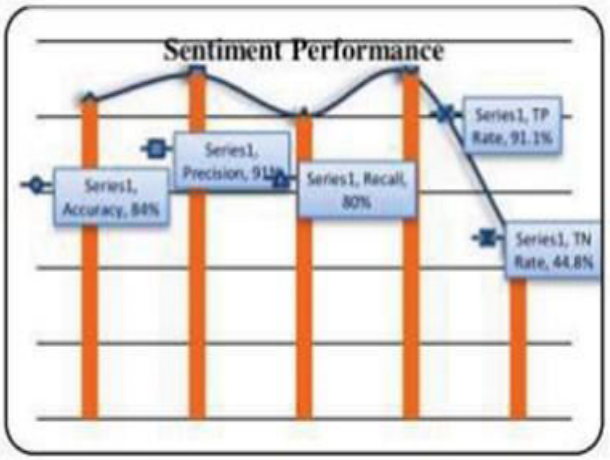


Fig 1: System Architecture

4.2 UML DIAGRAMS

UML stands for Unified Modeling Language. UML is a standardized general- purpose modeling language in the field of object-oriented software engineering. The standard is managed, and was created by, the Object Management Group. The goal is for UML to become a common language for creating models of object oriented computer software. In its current form UML is comprised of two major components: a Meta-model and a notation. In the future, some form of method or process may also be added to or associated with, UML. The Unified Modeling Language is a standard language for specifying, Visualization, Constructing and documenting the artifacts of software system, as well as for business modeling and other non-software systems.

UML was created as a result of the chaos revolving around software development and documentation. In the 1990s, there were several different ways to represent and document software systems.

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 14, Issue 4, April 2025

|DOI: 10.15680/IJIRSET.2025.1404389|

The UML represents a collection of best engineering practices that have proven successful in the modeling of large and complex systems. The UML is a very important part of developing objects oriented software and the software development process. The UML uses mostly graphical notations to express the design of software projects.

4.2a GOALS:

The Primary goals in the design of the UML are as follows:

1. Provide users a ready-to-use, expressive visual modeling Language so that they can develop and exchange meaningful models.
2. Provide extendibility and specialization mechanisms to extend the core concepts.
3. Be independent of particular programming languages and development process.
4. Provide a formal basis for understanding the modeling language.
5. Encourage the growth of object oriented tools market.
6. Support higher level development concepts such as collaborations, frameworks, patterns and components.
7. Integrate best practices.

V. RESULTS

5.1 ALGORITHMS

5.1.1 NAIVE BAYES ALGORITHM

The Naïve Bayes (NB) algorithm is a probabilistic classifier based on Bayes' theorem, which assumes that features are conditionally independent given the class label. Despite this strong assumption, NB performs well in many real-world applications, especially in text classification tasks like sentiment analysis. The algorithm calculates the probability of a data point belonging to a particular class by multiplying the prior probability of the class with the likelihood of observing the given features. There are different variations of Naïve Bayes, including Multinomial NB (suitable for text data with word frequencies), Gaussian NB (for continuous data following a normal distribution), and Bernoulli NB (for binary features). Due to its simplicity, efficiency, and ability to handle large datasets, NB is widely used in spam detection, sentiment analysis, and recommendation systems. However, it can struggle with feature dependence and small datasets, which may impact its accuracy.

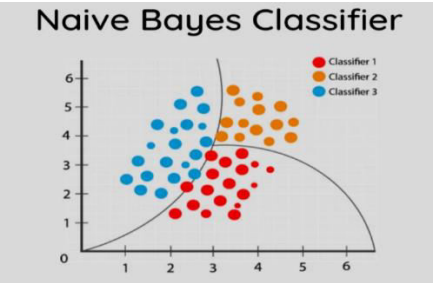


Fig 2: Naïve Bayes

Naïve Bayes - Key Terminologies

- **Bayes' Theorem –** Formula used for probability estimation: P(A|B)=P(B|A)P(A)P(B)P(A|B) = \frac{P(B|A) P(A)}{P(B)}
- **Prior Probability (P(A)P(A)) –** Initial probability of a class before considering evidence.
- **Likelihood (P(B|A)P(B|A)) –** Probability of given features occurring in a specific class.
- **Posterior Probability (P(A|B)P(A|B)) –** Updated probability of a class after considering evidence.
- **Class Conditional Independence –** Assumption that features are independent given the class label.
- **Feature Vector –** A set of attributes used to classify an instance.
- **Types of Naïve Bayes:**
- **Multinomial NB –** Used for word frequency in text classification.
- **Bernoulli NB –** Works with binary feature data (word presence/absence).
- **Gaussian NB –** Assumes features follow a normal distribution (continuous data).

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 14, Issue 4, April 2025

|DOI: 10.15680/IJIRSET.2025.1404389|

How does a Naïve Bayes work?
Naïve Bayes works by applying Bayes' theorem to classify data based on probabilities. First, during training, it calculates the prior probability of each class and the likelihood of each feature appearing within that class. When a new instance is given, the algorithm computes the posterior probability for each class by multiplying the prior probability with the likelihood of the observed features, assuming feature independence (a key assumption of Naïve Bayes). The class with the highest probability is assigned to the new data. This approach is widely used in text classification, such as spam filtering and sentiment analysis, due to its efficiency and effectiveness in handling large datasets despite its independence assumption.

5.1.2 Decision Tree Regression in Python

We will now go through a step-wise Python implementation of the Decision Tree Regression algorithm that we just discussed.

Importing necessary libraries: The first step will always consist of importing the libraries that are needed to develop the ML model. The Numpy, Matplotlib and the Pandas libraries are imported.

Importing the data set: For this problem, we will be loading a CSV dataset through a HTTP request (you can also download the dataset from here). We will be loading the data set using the read_csv() function from the pandas module and store it as a pandas DataFrame object.

Separating the features and the target variable: After loading the dataset, the independent variable and the dependent variable need to be separated.Our concern is to model the relationships between the (Crop,Season, Average_income, etc..) and the target variable (Production) in the dataset.

Splitting the data into a train set and a test set: We use the train_test_split() module of scikit-learn for splitting the data into a train set and a test set. We will be using 20% of the available data as the testing set and the remaining data as the training set.

Fitting the model to the training dataset: After splitting the data, let us initialize a Decision Tree Regressor model and fit it to the training data. This is done with the help of DecisionTreeRegressor() module of scikit-learn.
Calculating the loss after training: Let us now calculate the loss between the actual target values in the testing set and the values predicted by the model with the use of a cost function called the Root Mean Square Error (RMSE). The RMSE of a model determines the absolute fit of the model to the data.

5.1.3 SVM (Support Vector Machine)

Support Vector Machine (SVM) is a supervised machine learning algorithm used primarily for classification tasks. It works by finding the optimal hyperplane that best separates different classes in a dataset while maximizing the margin between them. The data points closest to the hyperplane, known as support vectors, play a crucial role in defining the boundary. SVM can handle both linear and non-linear data using kernel tricks, such as the RBF kernel, to transform data into higher dimensions where it becomes separable. It is widely used in text classification and sentiment analysis, such as analysing YouTube comments, due to its high accuracy in handling high-dimensional data. However, SVM can be computationally expensive for large datasets and requires careful tuning of hyperparameters for optimal performance.

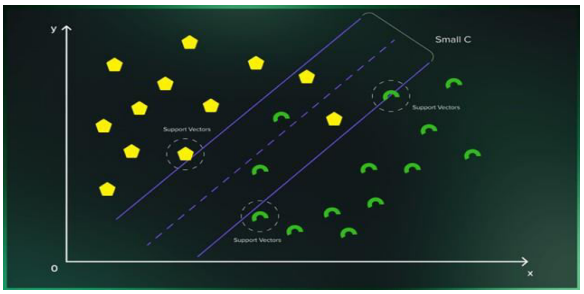


Fig 3: Support Vector Machine

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 14, Issue 4, April 2025

|DOI: 10.15680/IJIRSET.2025.1404389|

SVM aims to find the best hyperplane that separates data points belonging to different classes. The goal is to maximize the margin, which is the distance between the closest data points (support vectors) and the hyperplane.
Hyperplane – A decision boundary that separates different classes.
Support Vectors – The data points closest to the hyperplane, which influence its position.
Margin – The distance between the hyperplane and the nearest data points from each class. A **larger margin** improves the model's generalization.

Types of SVM
Linear SVM – Used when data is linearly separable (can be separated by a straight line or plane).
Non-Linear SVM – Used when data is not linearly separable; it uses a kernel trick to map data into higher dimensions where it becomes separable.

Common SVM Kernels
Linear Kernel – Used when data is already separable in its original space.
Polynomial Kernel – Maps input features into a higher-degree polynomial space.
Radial Basis Function (RBF) Kernel – Maps data into an infinite-dimensional space, commonly used for complex patterns.

Advantages of SVM
- Effective in high-dimensional spaces (e.g., text classification).
- Works well even with small datasets.
- Robust to overfitting when properly tuned.

Disadvantages of SVM
- Computationally expensive for large datasets.
- Choosing the right kernel and hyperparameters can be challenging.

The following figures present the sequence of screenshots of the results.

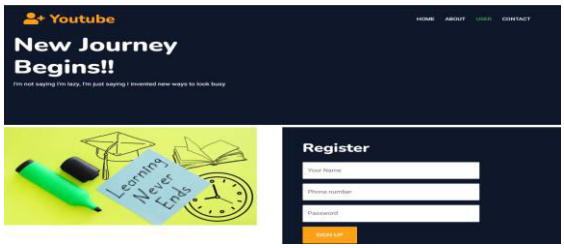


Fig 2a: Registration Page

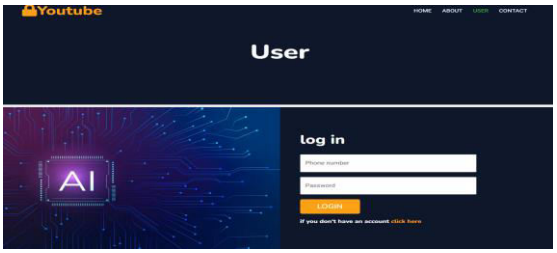


Fig 2b: Login page

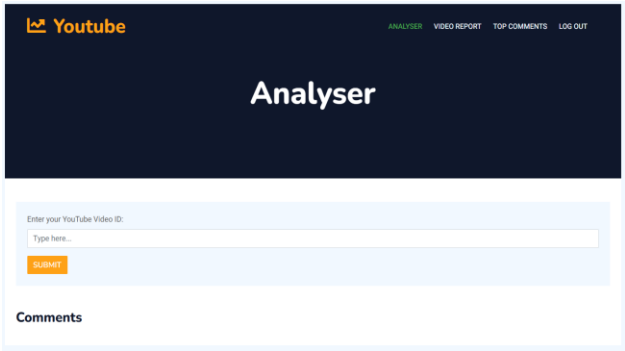


Fig 2c: login with user details extract you tube video ID using you tube API

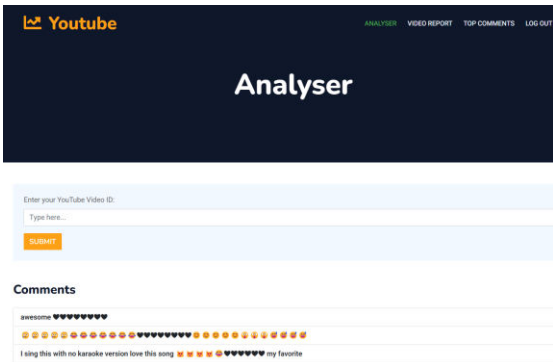


Fig 2d: report of comments extracted from you tube API

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 14, Issue 4, April 2025

|DOI: 10.15680/IJIRSET.2025.1404389|

Fig 2e: analysis of comments

Fig 2f: Top 10 comments POSITIVE

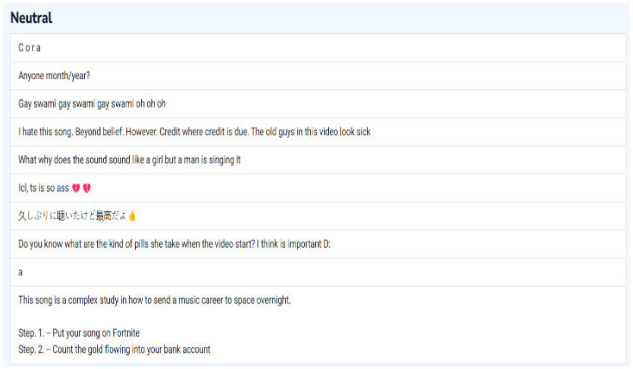


Fig 2g: Top 10 comments NEUTRAL

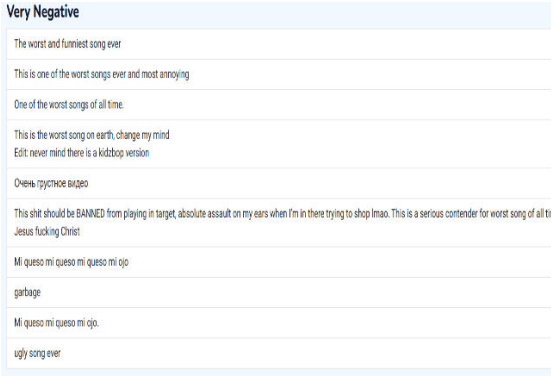


Fig 2h: Top 10 comments NEGATIVE

VI. CONCLUSIONS AND FUTURE WORK

6.1 CONCLUSIONS

The study of public sentiment analysis through YouTube comments using machine learning provides valuable insights into audience opinions, emotions, and behavioural patterns. As social media platforms like YouTube continue to grow, analysing user-generated content becomes crucial for businesses, content creators, and researchers to better understand audience engagement and improve digital strategies.

By leveraging advanced machine learning algorithms, such as Support Vector Machine (SVM) and Naïve Bayes (NB), along with Natural Language Processing (NLP) techniques, we can effectively classify sentiments into categories like positive, negative, and neutral**.** These models process large volumes of comments to identify key trends, user satisfaction levels, and potential concerns, helping content creators refine their content strategies, brands improve their customer interactions, and policymakers detect online toxicity.

Despite the effectiveness of machine learning in sentiment analysis, challenges remain. Contextual ambiguity**,** sarcasm detection, and multilingual analysis present significant hurdles, as machine learning models often struggle to fully grasp the nuanced nature of human language. Additionally, noisy and unstructured data, such as slang, emojis, and abbreviations, can impact classification accuracy. Addressing these challenges requires continuous improvements in deep learning models**,** transformer-based architectures (such as BERT or GPT), and enhanced feature engineering techniques.

Overall, this study highlights the growing importance of machine learning-driven sentiment analysis in understanding user emotions and shaping digital interactions. With further advancements in AI and NLP, sentiment analysis can become even more accurate, context-aware, and scalable, ultimately contributing to more meaningful and impactful engagement in the digital landscape.

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 14, Issue 4, April 2025

|DOI: 10.15680/IJIRSET.2025.1404389|

6.2 FUTURE WORK

While this study demonstrates the effectiveness of machine learning in analyzing public sentiments through YouTube comments, several areas require further exploration and improvement. One key challenge is enhancing sentiment detection, particularly in handling sarcasm, irony, and contextual ambiguity. Future research can leverage advanced transformer-based models**,** such as BERT, RoBERTa, and GPT, to improve contextual understanding and classification accuracy. Additionally, YouTube hosts a diverse global audience, making multilingual and cross-language sentiment analysis a crucial area of focus. Implementing models like mBERT or XLM-R can enhance the accuracy of sentiment classification across various languages and dialects.

Another promising direction is aspect-based sentiment analysis (ABSA)**,** which moves beyond general sentiment classification to detect sentiments related to specific aspects, such as video quality, content relevance, or engagement level. Similarly, emotion detection and fine-grained sentiment analysis can be explored to classify emotions like joy, anger, excitement, or disappointment, providing deeper insights into audience reactions. Addressing noisy and unstructured data remains a challenge, as YouTube comments often include slang, emojis, abbreviations, and misspellings. Future research should focus on improving text preprocessing techniques and utilizing advanced word embeddings like Word2Vec, FastText, or contextual embeddings to handle informal language more effectively.

Developing real-time sentiment analysis systems integrated with live data streaming and dashboard visualization can further enhance the usability of sentiment analysis for content creators and businesses. Moreover, the detection of hate speech, toxicity, and fake comments should be incorporated into future models to create safer online environments. Reinforcement learning and human-in-the-loop approaches could help improve model robustness against harmful content. Another critical area is explainability and bias reduction in sentiment models, as many machine learning algorithms function as black boxes. Utilizing interpretability techniques such as SHAP or LIME can improve transparency and fairness in sentiment classification.

By addressing these challenges and leveraging cutting-edge AI advancements, future research can significantly enhance the accuracy, scalability, and applicability of sentiment analysis. This will make sentiment analysis a more powerful tool for businesses, researchers, and policymakers in the rapidly evolving digital landscape, allowing for more informed decision-making and content strategy development.

REFERENCES

1. Pang, B., & Lee, L. (2008). Opinion Mining and Sentiment Analysis. Foundations and Trends in Information Retrieval, 2(1-2), 1-135. This paper provides a comprehensive review of sentiment analysis techniques, including supervised and unsupervised learning methods used in text classification.
2. Sebastiani, F. (2002). Machine Learning in Automated Text Categorization. ACM Computing Surveys, 34(1), 1-47. This study explores various machine learning approaches for text classification, with a focus on their applications in sentiment analysis.
3. Liu, B. (2012). Sentiment Analysis and Opinion Mining. Synthesis Lectures on Human Language Technologies, 5(1), 1-167. This book discusses fundamental techniques and challenges in sentiment analysis, covering lexicon-based and machine learning-based approaches.
4. Cortes, C., & Vapnik, V. (1995). Support-Vector Networks. Machine Learning, 20, 273-297. This foundational work introduces the Support Vector Machine (SVM) algorithm, which is widely used in sentiment classification tasks.
5. Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. Proceedings of NAACL-HLT 2019, 4171-4186. This paper presents the BERT model, a deep learning architecture that significantly improves sentiment analysis by understanding word context.
6. Mohammad, S. M., & Turney, P. D. (2013). Crowdsourcing a Word-Emotion Association Lexicon. Computational Intelligence, 29(3), 436-465. The authors introduce a large-scale lexicon that helps in mapping words to emotions, aiding in fine-grained sentiment analysis.
7. YouTube API Documentation. (n.d.). Retrieved from YouTube Developer Portal. This official documentation provides guidelines on accessing YouTube comments using APIs, essential for data collection in sentiment analysis research.

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 14, Issue 4, April 2025

|DOI: 10.15680/IJIRSET.2025.1404389|

8. Feldman, R. (2013). Techniques and Applications for Sentiment Analysis. Communications of the ACM, 56(4), 82-89. This paper discusses practical applications of sentiment analysis in industries like marketing, social media
9. monitoring, and political analysis.
10. Zhang, L., Wang, S., & Liu, B. (2018). Deep Learning for Sentiment Analysis: A Survey. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, 8(4), e1253. The authors review deep learning techniques for sentiment classification, comparing different neural network architectures.
11. Sharma, S., & Dey, S. (2012). A Machine Learning Approach to Sentiment Analysis in Online Reviews. Proceedings of the 2012 International Conference on Advances in Social Networks Analysis and Mining (ASONAM), 202-209. This study evaluates different machine learning models, including SVM and Naïve Bayes, for sentiment classification in online user review.

INTERNATIONAL STANDARD SERIAL NUMBER INDIA

INNO SPACE SJIF Scientific Journal Impact Factor

doi cross ref

निस्केयर NISCAIR

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY

9940 572 462 6381 907 438 ijirset@gmail.com

www.ijirset.com

Scan to save the contact details