Published by :
http://www.ijert.org

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 13 Issue 10, October 2024

Unveiling Insights: Advanced Techniques for Youtube Comment Analysis

Harsh Bhimate
Dept. of AI
G H Raisoni College of Engineering
Nagpur 440016, India

Jayesh Nehare
Dept. of AI
G H Raisoni College of Engineering
Nagpur 440016, India

Laxmikant Yenchalwar
Dept. of AI
G H Raisoni College of Engineering
Nagpur 440016, India

Yash Bhatnagar
Dept. of AI
G H Raisoni College of Engineering
Nagpur 440016, India

Durgaprasad Roy
Dept. of AI
G H Raisoni College of Engineering
Nagpur 440016, India

***Abstract* — The massively growing volume of user-generated content on social media platforms like YouTube has highlighted the need for effective tools to analyze them, especially categorization of comments. This paper presents the development of an Enhanced YouTube Comments Classifier that categorizes comments into six categories: appreciation, normal, suggestion, question, trolling, and other languages. We built a dataset of nearly 10500 comments in English and Hinglish, using the YouTube Data API and supplemented it with a Kaggle dataset for better generalization. Text features were extracted using the TF-IDF vectorizer, and the Random Forest Classifier was employed for classification, achieving an accuracy of 91.71%. The model is capable of categorizing both English and Hinglish comments while identifying non-English and non-Hinglish entries as "other languages." The classifier is deployed using Flask, enabling real-time predictions and enhancing content moderation and audience engagement on YouTube.**

***Keywords*— YouTube, Kaggle, categorize, TF-IDF, Flask.**

I. INTRODUCTION

YouTube is an application that serves as a platform for video sharing. Here users are allowed to view, upload, share, and comment on videos. It is owned by Google, launched in 2005. There is variety of user-generated and corporate media content, short video and TV clips, music and documentary films, movie teasers and trailers, live streams, and more content is posted on this platform. It is universally one of the most visited websites. These are some of the features of this platform:

- Content Variety: YouTube provides various range of content, making it a versatile platform.
- User Interaction: Users can interact with videos through likes, dislikes, and comments. Feature of subscription helps the users to stay updated with new content from their favorite creators.
- Monetization Policy: Through channel membership, ads Super Chat and premium revenue, there is an opportunity for content creators to earn income.
- Community Guidelines and Policies: YouTube has a set of community guidelines and policies to maintain a safe and respectful environment.
- Analytics and Insights: Content creators have access to YouTube Analytics through YouTube studio, which provides detailed insights into video performance, engagement metrics, audience demographics and more.
- Search and Recommendation Algorithms**:** YouTube uses advanced algorithms to recommend videos to users based on their viewing history, likes, preferences, and trending content. This personalized experience helps users discover relevant and engaging content.

Over the past few years, YouTube has become major hub for education, entertainment and information sharing. For content creators, it is the most popular and preferred platform. Massive numbers of Youtubers or the content creators has been posting lot of videos. These videos are reaching to large number of audience and more than millions of people are watching it. Due to which the Youtubers have to constantly keep upgrading, managing and maintaining the quality of their content. So, there is a feature of comment in YouTube that allows viewers to comment or give feedback in comment section.

In order to improve audience engagement, feedback given in comments serves information on aspects of the content that might need improvement. Many Youtubers face the challenge of not having enough time to read through all the comments on their videos. However, understanding audience feedback is crucial for maintaining public interest in their content. Our work offers a solution to this dilemma, by making it easier for Youtubers to read comments effectively. Our approach is to first extract all the comments from a video and classify them into multiple categories based on: Suggestion, Trolling, Question, Appreciation, normal and other languages (excluding Hinglish and English). These categories can help Youtubers focus only on those comments which is of their interest. Also a new section is created for further analysis of comments which include graphs to show the occurrence and importance of

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 13 Issue 10, October 2024

particular type of category and summarize furthermore all the comments of each category.

Many studies, including the renowned analysis of sentiment in Twitter data [1], YouTube polarity trend analysis [2], and user comment sentiment analysis on YouTube [4], have been conducted in the field of sentiment analysis. Categorizing comments into different types is difficult due to factors such as informal language, grammatical errors, punctuation mistakes, unformatted texts, and trivial comments. There are occasions when a single comment contains multiple sentences in various languages and categories. This problem presents a distinct difficulty in analyzing sentiment using different types of sentences and emotions. One way to tackle the issue is to categorize comments using NLP through lexicon [3], for example, spotting interrogative comments by looking for words like what, how, and why. In the same way, optimistic statements can be recognized by terms such as excellent, top, and superb. Nonetheless, this method does not tackle the distinct obstacles posed by casual writings and various languages. Furthermore, if a comment contains several categories, this technique will not work effectively. Using neural networks [5] can categorize these comments more effectively, serving as a potential solution, but they are challenging to adjust, complicated, and lack easy explanation.

II. LITERATURE REVIEW

[4] Singh et al. (2021) implemented sentiment analysis on YouTube comments using an existing annotated corpus. She applied data preprocessing techniques, including normalization, to thoroughly clean the data. Various machine learning algorithms such as KNN, Random Forest, SVM, LR, and Decision Trees were implemented. The performance of the models was evaluated using F-score and accuracy metrics.

[6] Pokharel and Bhatta (2021) collected YouTube comments using the YouTube Data API, gathering 10,000 comments from tutorial videos. After preprocessing the data to remove noise, the comments were manually labeled into six categories: Positive, Negative, Interrogative, Imperative, Corrective, and Miscellaneous. Feature extraction was performed, and classification models were trained. Hyperparameter tuning was implemented to achieve optimal results. Models like Linear SVC, Logistic Regression, Multinomial Naive Bayes, Random Forest, and Decision Trees were tested. The best results were achieved with Linear SVC, which achieved a maximum F1 score of 0.86 and a cross-validation score of 0.84.

[7] Kavitha et al. (2020) extracted YouTube comments, video descriptions, and author names using the YouTube API. They focused on videos from three domains: NLP, Sports, and Movies. Data Preprocessing involved tokenization and stop word removal. Features were extracted using Bag of Words and Association Word List methods. The user comments were classified into four categories: relevant, irrelevant, positive, and negative. Both feature extraction methods were applied across the three domains, and their performance was evaluated.

[8] Khoo et al. (n.d.) worked with a dataset of 160 email dialogues from Hewlett-Packard's help desk, containing 1,486 sentences classified into 14 categories, such as apology, salutation, request, question, and suggestion. Various experiments were conducted using Bag of Words, followed by NLP techniques like stop word removal, tokenization, and lemmatization. Classification algorithms including Naive Bayes, Decision Trees, and SVM were applied, with SVM delivering the most consistent performance. Feature selection methods such as chi-squared, information gain, and binomial separation were also explored.

[9] Yasmina et al. (2016) used YouTube API v3 to gather comments reflecting informal, emotion-laden communication. Comments were retrieved using targeted keywords, and stemming techniques were applied during preprocessing to standardize word forms. An unsupervised method was used to classify emotions by calculating the relatedness between words and emotions using normalized PMI (Point wise Mutual Information). The method effectively distinguished between neutral and emotional words, adapting to the context of negation and evolving language. The approach, which classifies emotions at the word level, showed a high average precision of 92.75%, improving upon previous methods by considering the frequency of word occurrences in relation to emotion categories.

III. METHODOLOGY

The project's goal is to categorize YouTube video comments into six categories, namely, appreciation, normal, suggestions, questions, trolling and other languages. The Enhanced YouTube Comments Classifier has been implemented using Python language. The various python packages used for analysis are Pandas, scikit-learn, NumPy, NLTK (Natural Language Tool Kit) and pickle. For system implementation, we utilized the scikit-Learn machine learning library, which is built in Python. Well-known machine learning library scikit-Learn has an intuitive user interface and is deeply linked with the Python programming language. Our methodology is shown in Fig. 1.

.

Published by :
http://www.ijert.org

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 13 Issue 10, October 2024

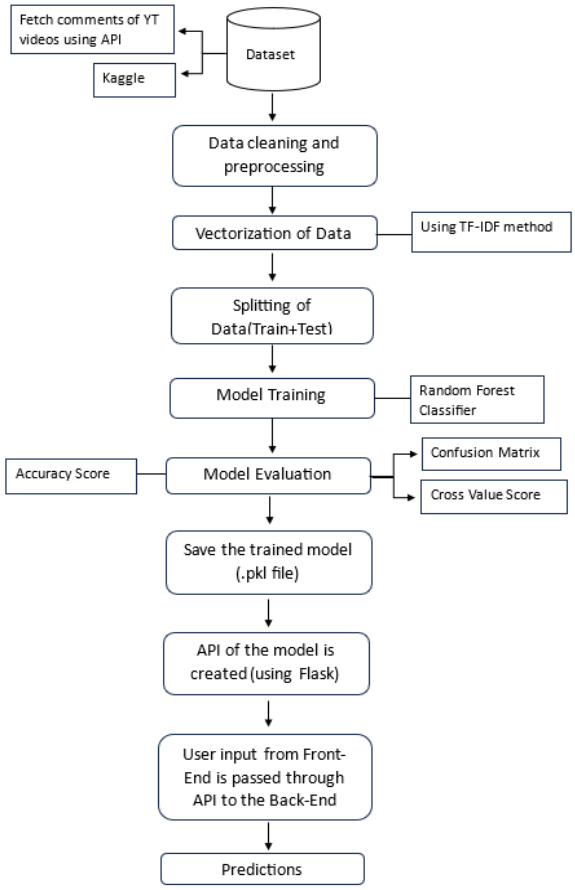

Fig 1: Proposed methodology

A. Data Collections

The dataset which we used for our system is created manually. Our dataset consists of 10504 comments of English and Hinglish language with their corresponding sentiments. We have retrieved the comments of different videos using YouTube's Data API, which provides methods to retrieve video comments, details, and other information. We have set up the Google Cloud Project and the YouTube API. To retrieve comments from a YouTube video, we made a request to the API. Additionally, we augmented our dataset with the Kaggle YouTube comments dataset for better generalization of our model. Fig 2 shows the description of dataset

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10504 entries, 0 to 10503
Data columns (total 2 columns):
 #   Column     Non-Null Count  Dtype
---  ------     --------------  -----
 0   comment    10496 non-null  object
 1   sentiment  10381 non-null  object
dtypes: object(2)
memory usage: 164.2+ KB
```

Fig 2: Dataset description

B. Data Pre-processing and cleaning

Performing preliminary data processing is crucial, especially when it comes to predicting through labeled dataset. This stage deals with problems that may compromise data efficiency, such as missing values and impurities in the data.

The main goal of data pre-processing is to improve post-mining quality and efficiency. While using machine learning algorithms on a dataset, this phase is crucial to ensure accurate results and good predictions. Data cleaning is the process of taking raw data and refining it using several methods, such as eliminating duplicates and unnecessary entries.

C. Converting texts into features

Document frequency, hashing vectorizer, TF-IDF vectorizer and Word2Vec are some of the well-known methods for vectorizing a corpus of text. The process of vectorizing data involves transforming non-numeric data, like text or images, into a numerical format that can be used by machine learning algorithms. Most ML algorithms, especially those based on linear algebra require input features to be represented as numerical vectors. Text data needs to be converted into numerical representations for ML models to process.

For this paper, we have chosen to use the TF-IDF vectorizer. TF-IDF has the ability to include terms that are not frequently found in the document. It extends the Bag of Words (Bow) approach by keeping in mind the frequency of words across documents, giving more weight to rare but important words. It allocates a weight based on the reciprocal document frequency (TF) (IDF) and phrase frequency (TF) to each word in a document. Higher weight values indicate that a word is more important.

$$\text{TF(t,d)} = \frac{f(t,d)}{\sum_{t' \in d} f(t',d)}$$

$$\text{IDF(t)} = \log \frac{N}{DF(t)}$$

$$\text{TF-IDF(t,d)} = \text{TF(t,d)} * \text{IDF(t)}$$

Where, $f(t,d)$ = Term frequency of term t in document d

N = Total number of document in the corpus

DF(t) = Document frequency

D. Splitting of Data

Splitting of data ensures that the model generalizes well to unseen data. Usually, there are several subsets of the dataset, such as training and testing data. For our research, we have maintained 80:20 as the ratio. 20% of the dataset is used for testing, and the remaining 80% is used for training.

E. Model Training

Following the preparation of the data, machine learning technique is applied. In order to categorize the comments, the Random Forest Classifier is employed for the prediction of emotions of comments.

For decision tree-based tasks such as regression and classification, Random Forest, also referred to as Random Decision Forest, a supervised machine learning approach is used.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 13 Issue 10, October 2024

Managing large and complex datasets, controlling high-dimensional feature spaces, and providing information about the importance of specific attributes are all made possible by random forests. The ability of the Random Forest Classifier algorithm is to minimize over fitting while maintaining a high level of accuracy of predictions.
The Random Forest Classifier creates a series of decision trees from a randomly selected subset of the training set. A selection of decision trees (DT) chosen at random from the training set serves as the initial step. Each decision tree's votes are totaled to get the final prediction.

F. Connect Front-End to Back-End

The trained model is saved as .pkl file. Flask framework has been used to deploy the ML model. Using Flask, an API endpoint is created. When a user sends comment through the front end, this comment is sent as a POST request to the API, this runs the model on the data and returns predictions as the corresponding emotion.

IV. RESULTS

A. Count of Words in a Comment

Fig 3 shows a histogram of word count in the comments illustrates the distribution of comment lengths across the dataset. Most comments have between 5 to 20 words, with a notable number having less than 10 words. This gives an idea of the brevity and structure of the comments being classified. Understanding the distribution of word counts helps us comprehend how well the model may perform across both shorter and longer comments.

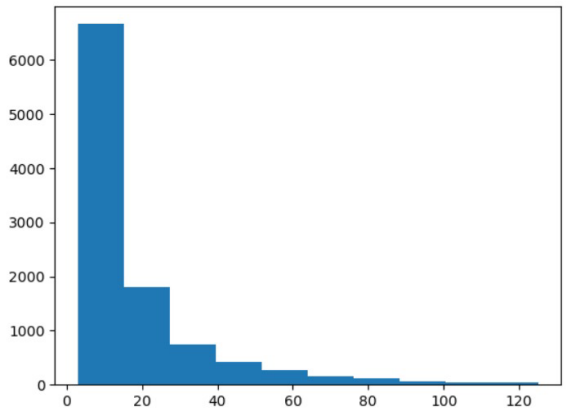

Fig 3 : Histogram of word count in comment

B. Category Counts in Dataset

The dataset contains comments categorized into six types of sentiments. Fig 4 shows a higher number of question type and neutral sentiments like normal, with a significant presence of appreciation and questions. Trolling type comments has occurred the least time.

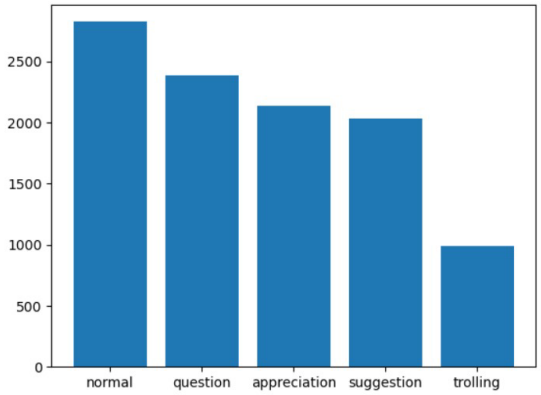

Fig 4: Category counts in dataset

C. Predicted Categories and Model Performance

The pie chart of the predicted categories after applying the Random Forest Classifier shows the following distribution:

- Appreciation: 24.4%
- Normal: 22%
- Trolling: 9.76%
- Suggestion: 12.2%
- Question: 17.1%
- Other Languages: 14.6%

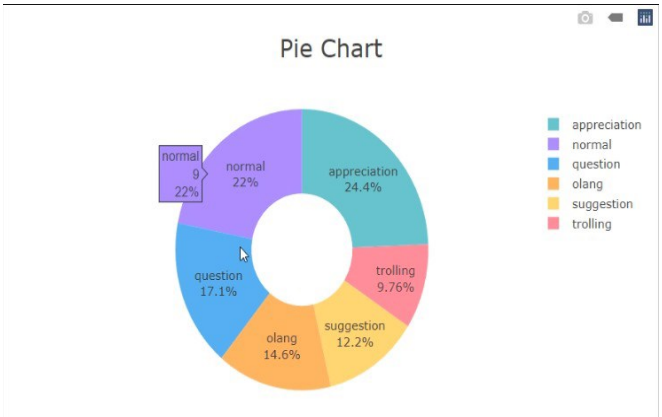

Fig 5: pie chart of predicted categories

We noticed that the Random Forest Classifier algorithm yields 91.71% of Accuracy that is helpful for medical experts to correctly identify the emotions of the comments. Our model categorizes the comments based on their emotions into 6 categories namely, ‘appreciation’, ‘normal’, ‘trolling’, ‘suggestion’, ‘question’ and ‘other languages’. The model works exceptionally well on the English as well as Hinglish language comments and categorizes the comment as ‘other language’ when any other language apart from English and Hinglish is detected.

D. Front-End Integration

A user-friendly front-end was developed using Flask framework, which allows the trained model to categorize comments in real-time on a YouTube clone interface. Users can input comments, and the system will display the predicted sentiment in one of the six categories, enhancing user experience and providing an intuitive way to view categorized comments directly.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 13 Issue 10, October 2024

E. Summary of important questions and suggestions

Our system encompasses an additional feature that presents a summary of the most significant questions and suggestions from all the comments. This "important of important" summary extracts the key insights, allowing content creators to quickly identify crucial feedback and recurring queries. They do not need to go through all the comments or even a particular classification box of suggestion or question. By providing a concise view of the most relevant interactions, the feature ensures that no essential comment is missed, even in large volumes of feedback. This helps creators make informed decisions and respond promptly, improving overall engagement with their audience. Fig 6 shows the feature of important summary at our clone website.

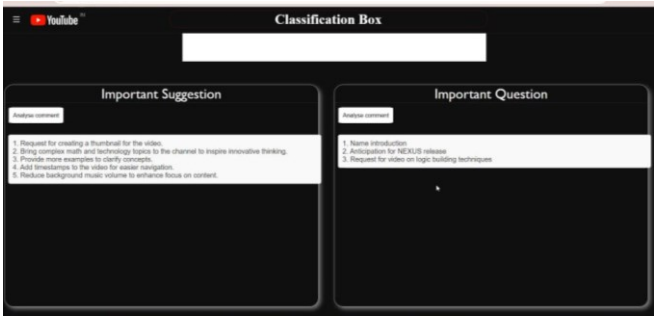

Fig 6: Summary box of questions and suggestions

V. CONCLUSION

In this paper, we developed a comment categorization model with enhanced analysis for YouTube videos, utilizing the Random Forest Classifier machine learning model. The model is capable of classifying comments into six distinct categories, attaining an accuracy rate of 91.71%. We were able to capture the most relevant features from the text data, by using the TF-TDF vectorizer and hence enhancing the model's performance. Our approach works efficiently on both English and Hinglish language comments, and it can distinguish comments in other languages as well. The model's robust performance showcases its potential in aiding content creators and platform managers to better understand audience engagement and sentiments. This can ultimately help improve user interaction and content management.

VI. REFERENCES

[1] Wang, Yili & Guo, Jiaxuan & Yuan, Chengsheng & Li, Baozhu. (2022). Sentiment Analysis of Twitter Data. Applied Sciences. 12. 11775. 10.3390/app122211775.

[2] Amar Krishna, Joseph Zambreno, and Sandeep Krishnan. 2013. Polarity Trend Analysis of Public Sentiment on YouTube. In Proceedings of the 19th International Conference on Management of Data (Ahmedabad, India) (COMAD '13). Computer Society of India, Mumbai, Maharashtra, IND, 125–128.

[3] Khin Zezawar Aung and Nyein Nyein Myo. 2017. Sentiment analysis of students' comment using lexicon based approach. In 2017 IEEE/ACIS 16th International Conference on Computer and Information Science (ICIS). 149–154. https://doi.org/10.1109/ICIS.2017.7959985

[4] Singh, R., Indian Institute of Technology Delhi, Ayushka Tiwari, & SRM Institute of Science and Technology, Ghaziabad. (2021). YOUTUBE COMMENTS SENTIMENT ANALYSIS. International Journal of Scientific Research in Engineering and Management (IJSREM), 05–05(05), 1–2. https://www.researchgate.net/publication/351351202

[5] Chunting Zhou, Chonglin Sun, Zhiyuan Liu, and Francis Lau. 2015. A C-LSTM neural network for text classification. arXiv preprint arXiv:1511.08630 (2015).

[6] Pokharel, R., & Bhatta, D. (2021, October 31). Classifying YouTube Comments Based on Sentiment and Type of Sentence. arXiv.org. https://arxiv.org/abs/2111.01908

[7] Kavitha, K., Shetty, A., Abreo, B., D'Souza, A., & Kodana, A. (2020). Analysis and Classification of User Comments on YouTube Videos. Procedia Computer Science, 177, 593–598. https://doi.org/10.1016/j.procs.2020.10.084

[8] Khoo, A., Marom, Y., Albrecht, D., & Faculty of Information Technology, Monash University. (n.d.). Experiments with Sentence Classification. In Proceedings of the 2006 Australasian Language Technology Workshop (ALTW2006) (pp. 18–25). https://aclanthology.org/U06-1005.pdf

[9] Yasmina, D., Hajar, M., & Hassan, A. M. (2016). Using YouTube Comments for Text-based Emotion Recognition. Procedia Computer Science, 83, 292–299. https://doi.org/10.1016/j.procs.2016.04.128

[10] A. Agrawal and A. An, "Unsupervised Emotion Detection from Text Using Semantic and Syntactic Relations," 2012 IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology, Macau, China, 2012, pp. 346-353, doi: 10.1109/WI-IAT.2012.170. keywords: {affective computing;emotion detection},