Journal of Artificial Intelligence and Capsule Networks (ISSN: 2582-2012)
www.irojournals.com/aicn/

YouTube Comment Sentiment Classification System

Jana Sorupaa J[1], Anto Nivedha J[2], Arsha R[3], Muthulakshmi K[4]

[1,2,3]Student, Department of Artificial Intelligence and Data Science, Kamaraj College of Engineering and Technology, Virudhunagar, Tamil Nadu, India

[4]Assistant Professor, Department of Computer Science and Engineering, Kamaraj College of Engineering and Technology, Virudhunagar, Tamil Nadu, India

Email: [1]janasorupaa@gmail.com, [2]nivedhaanto@gmail.com, [3]arsharajarathinam02@gmail.com, [4]muthulakshmicse@kamarajengg.edu.in

Abstract

With more than 2 billion viewers per month, YouTube is the most widely used video-sharing website worldwide. On this website, users can watch, upload, and share videos covering a wide range of subjects. YouTube comments include facts, opinions, and responses to videos in addition to starting discussions. The number of YouTube comments makes it difficult to manually analyze them all. The study of reading, understanding, and creating text in human languages encompasses a broad range of methods and techniques under the umbrella of natural language processing or NLP. The primary goal of the research is to find and analyze YouTube comments, which, when used with natural language processing algorithms, might be beneficial for the channels' continued development. One of the NLP methods used for this research was tokenization, which is used to break down text into individual words or tokens. Stemming and lemmatization are used to reduce the root words and normalize the variation. Categorization is performed by identifying the named entities, such as people, organizations, and locations. The machine translation was used to convert the comments from one language to another.

Keywords: NLP (Natural Language Processing), Stemming and Lemmatization, Tokenization, YouTube Comments, Named Entities, Machine Learning.

DOI: https://doi.org/10.36548/jaicn.2024.1.007
Received: 12.02.2024, received in revised form: 01.03.2024, accepted: 14.03.2024, published: 02.04.2024

1. Introduction

A crucial role in comprehending public sentiment and enhancing online conversations is played by YouTube comment analyzers. Valuable insights from YouTube comments, including sentiment, topics, and entities, are extracted by these analyzers by utilizing natural language processing (NLP)[1-5]. The contributions to this endeavour are made by tools such as YouTube polarity trend analysis [11], user comment sentiment analysis on YouTube [12], and similar tools. The significance of YouTube comment analyzers stems from several factors. Firstly, they are used to aid in understanding how individuals respond to YouTube videos and identifying prevailing trends in public sentiment. This information has proved invaluable to content creators, businesses, and researchers, enabling them to enhance their content, develop innovative products and services, and study social phenomena [7,9]. Secondly, YouTube comment analyzers act as a means by which harmful content, such as spam, hate speech, and misinformation, can be identified and eliminated. A safer and more positive online environment for all users is fostered by this [10]. Lastly, the improvement of online discussions is facilitated by these analyzers by identifying engaged and informative commenters. This knowledge can be utilized to devise effective tools and strategies that promote meaningful interactions [14]. In our research, we first analyze the comments to determine their sentiment (positive, negative, or neutral) and extract important topics and keywords to identify trends and patterns. Additionally, we aim to identify spam and abusive comments and categorize comments based on topic, category, or language. Furthermore, the analysis includes identifying the most influential commenters based on the extracted keywords. Lastly, the sentiment of comments will be compared [16-21].

The rest of the manuscript is organized with related works in Section 2, the proposed work in Section 3, the experimental results in Section 4, and the conclusion and future work in Section 5.

2. Related Work

The annotators in [15] manually labeled the comments as positive, negative, or neutral. Subsequently, the comments were categorized into positive-related, negative-related, neutral-related, or neutral-unrelated categories. Following preprocessing, the comments were segmented into 4,132 positive, 1,074 negative, and 780 neutral comments. Finally, they applied three classifiers—SVM-RBF, Bernoulli NB, and KNN—to classify the comments.

For these classifications, the proposed method employs NLP techniques. Numerous studies have investigated feature extraction from text. For instance, [12] examined viewers' commenting behaviour on coding tutorial videos, while [18] employed lexicon-based methods for sentiment analysis. Their approach involved creating a dictionary of words manually ranked on a scale from -5 to +5 to determine sentiment polarity.

In [8], the author employed Naïve Bayes classification for polarity (sentiment) analysis on the IMDB dataset. Their method incorporates the category of each comment during the feature selection process, rendering it unable to compute the polarity of a new comment without prior knowledge of its category.

In this polarity analysis, SentimentIntensityAnalyzer from the NLTK library was used to analyze the sentiment of the comment. [13] Introduced an advanced concept by showing the relationship among the words in their logical meaning as well as sentimental sense. The word vectors were obtained from an unsupervised neutral language model Word2vec [6, 8, 17].

Bags of Word Approach - The task of fully understanding text is hardly easy since it involves a variety of complex concepts that are difficult to implement in machines. The bag of words approach follows a simple methodology i.e. to count the number of times each word appears in the given text and associate sentiment weight to each word depending on the overall sentiment value of the text. It emphasizes the idea of having one feature for each word which proves effective for sentiment analysis. It is used as a baseline in text analytics projects and natural language processing. Pre-processing stages can dramatically improve the performance of the bag of words approach.

3. Proposed Work

The Table.1 and 2 shows the hardware and the software requirements of the proposed work.

Table.1 Hardware Requirements

Hardware Requirements	
Processor	Intel i5
Operating System	Windows 11
RAM	4GB
Hard Disk	512 GB

Table 2. Software Requirements

Software Requirements
Python Packages - Googleapiclient. discovery, Re, Sentiment Intensity Analyzer, matplotlib, numpy, pandas, seaborn, translator, NLTK
Python 3.10
Google Colab

3.1 System Requirements

3.1.1 Google Colab

Collaboratory, or "Colab" for short, is a product from Google Research. Colab allows anybody to write and execute arbitrary Python code through the browser and is especially well suited to machine learning, data analysis, and education. More technically, Colab is a hosted Jupyter notebook service that requires no setup to use, while providing access free of charge to computing resources including GPUs.

3.1.2 Googleapiclient. Discovery

Googleapiclient. discovery package is used in the YouTube Comment Analyzer project to interact with the YouTube Data API. This package allows developers to easily make requests to Google APIs, including the YouTube Data API, and also it is used for retrieving comments from YouTube videos.

3.1.3 Re

The ‘re’ module can be used for filtering comments in the context of the YouTube Comment Analyzer project.

3.1.4 Sentiment Intensity Analyzer

This package is part of the NLTK sentiment module, which is used for sentiment analysis in natural language processing (NLP). Specifically, it's a component of the Natural Language Toolkit (NLTK), a library in Python widely used for text analysis and NLP tasks.

3.1.5 Matplotlib

This package is used for plotting the raw data into pictorial data.

3.1.6 Seaborn

It is used to create a scatter plot to visualize the relationship between comment length and sentiment score

3.1.7 Translator

It is used to handle multilingual text data or provide language translation functionality to your users.

3.1.8 NLTK

Natural Language Toolkit (NLTK) is a popular open-source Python library used for working with human language data. It provides tools, resources, and programs for processing and analyzing textual data in natural language.

3.2 System Design

The system design is illustrated in figure 3.1

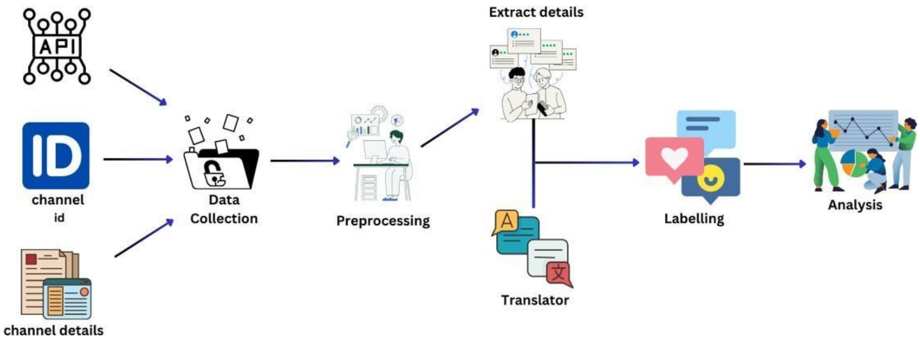


Figure 3.1. System Design

3.2.1 API Key

We created an API key to access various channel IDs. By creating an API key, we create an account in 'Google Cloud Platform' and create a new project. After the new project is created, the API key is enabled in YouTube Data API v3. It provides access to YouTube Data such as videos, playlists, etc.

3.2.2 Channel ID

API provides access to various data on YouTube, including information about channels, videos, playlists, etc. The API provides pagination parameters like 'pageToken' to navigate through pages of results. For each page of results, extract the channel IDs from the response data. The channel ID uniquely identifies each YouTube channel.

3.2.3 Channel Details

We use the channels. list' method to request information about a specific channel and retrieve it, such as snippets, statistics, etc. The response from the API requests to extract the desired channel details, such as title, description, subscriber count, view count, video count, playlist_id, likes, dislikes, comments, views, date of publication, etc.

3.2.4 Data Collection

The dataset could include video metadata (title, description, tags, etc.), channel information (title, description, subscriber count, etc.), and comments on specific videos obtained through the YouTube Data API. The 'channels_list' method is utilized to retrieve comments on specific videos. The data returned by the API requests is stored in your preferred format, such as CSV, in our local storage.

3.2.5 Pre-Processing

We can preprocess the comment data using Python and relevant libraries such as NLTK. We'll perform common preprocessing steps such as text cleaning, tokenization, removing stopwords, and stemming or lemmatization.

- **Tokenization** - Split the comments into individual words or tokens by using the (word_tokenize) function.
- **Removing Stop Words** - Remove common stopwords like "the," "and," "is," etc. By using set (stopwords. words('English)).
- **Stemming or Lemmatization** – Using the PorterStemmer() algorithm to reduce words to their root form using stemming or lemmatization. This step helps in normalizing the text.

Additionally, we handled the preprocessing steps such as handling emojis and dealing with non-English comments. We used this technique for fixing any bugs, adding new features, and providing technical support to customers.

- **Normalization** - It helps us to ensure that the data is consistent and can be compared for further analysis. We used text normalization to reduce the word to its base form by removing the inflectional part.

3.2.6 Translator

The transcripts are loaded to the Multilanguage model which will be useful for the user to identify and get clarity of the content in the comments.

3.2.7 Labelling

The sentiment_scores(comment, polarity) function has two parameters. Comment, which is a single comment to analyze, and polarity, which is a list to store the sentiment polarity scores. It uses SentimentIntensityAnalyzer from the NLTK library to analyze the sentiment of the comment. The sentiment score dictionary obtained from polarity_scores contains four values: positive, negative, neutral, and compound.

The script opens the file named "ytcomments.csv" for reading. It reads all lines from the file and stores them in the comments list. It checks whether the average polarity score indicates a positive, negative, or neutral response to the video. If the average polarity score is greater than 0.05, it prints "The Video has got a Positive response". If the average polarity score is less than -0.05, it prints "The Video has got a Negative response". Otherwise, it prints "The Video has got a Neutral response. The polarity score observed are depicted in Figure 3.2 and Table.3

```
Average Polarity: 0.22139110105580684
The Video has got a Positive response
The comment with most positive sentiment: اكتب إليك ..<br>ياقمري م
 with score 0.9959 and length 135
The comment with most negative sentiment: what a pile of junk
 with score -0.9524 and length 211
```

Figure 3.2. Polarity Score

Table 3. Polarity Score for the Responses

Polarity	Score	Length
Average	0.22139	101
Positive	0.9959	135
Negative	-0.9524	211

In Figure 3.2 the polarity score and length will be calculated based on the length of the comments in which it analyzes a single word in the comment and the polarity score will be generated based on the score by using the SentimentIntensityAnalyzer package in the NLTK package

3.2.8 Analysis

Finally, it counts the number of comments categorized as positive, negative, and neutral. These counts are stored in variables positive_count, negative_count, and neutral_count, respectively. It creates a list of comment_counts containing the counts of comments corresponding to each sentiment category. Finally, it displays the bar chart using plt. Show ().

3.2.9 Translation

The main purpose of the multilingual transformation phase is to help the user to understand the video content in their language. A multi-language model generally refers to a natural language processing (NLP) model that is capable of understanding and processing text in multiple languages. Such models have become increasingly important in the field of NLP due to the global nature of communication and the diversity of languages spoken across different regions. The available languages are shown to the user to select their particular language with the language code. We used the “Googletrans” model, for providing language and language code. The “Googletrans” model is the translator model and the module is available with the multiple languages that are supported by Google. The “Googletrans” model is used with packages like translators and constants which will be useful to convert the text from one to another language. The extracted transcripts are passed to the “Googletrans” module to provide the transcripts in the respective language that had been chosen by the user.

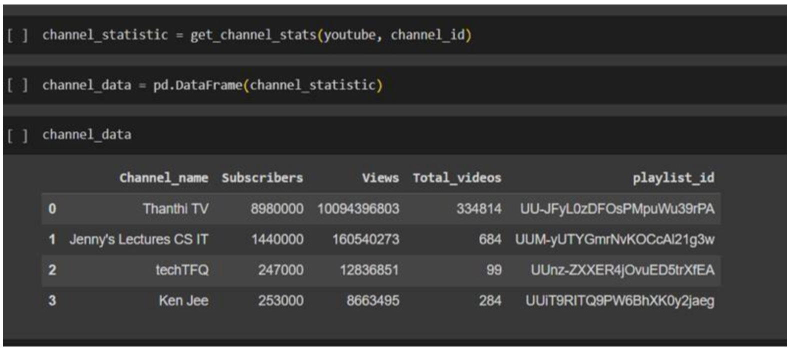
3.2.10 Benefits and Proposed Work

It improves the user experience and offers insightful information. It helps content producers better understand the attitudes, tastes, and levels of engagement of their audience, which helps them customize their content. It facilitates the discovery of trends and patterns in the comments area, which supports the optimization and development of content strategies. This analysis is a strong resource for content creators.

4. Experiment and Results

The results observed are illustrated in Figure 4.1 to 4.7

Extract Channel Details



```
[ ] channel_statistic = get_channel_stats(youtube, channel_id)
[ ] channel_data = pd.DataFrame(channel_statistic)
[ ] channel_data
```

	Channel_name	Subscribers	Views	Total_videos	playlist_id
0	Thanthi TV	8980000	10094396803	334814	UU-JFyL0zDFOsPMpuWu39rPA
1	Jenny's Lectures CS IT	1440000	160540273	684	UUM-yUTYGmrNvKOCcAI21g3w
2	techTFQ	247000	12836851	99	UUnz-ZXXER4jOvuED5trXfEA
3	Ken Jee	253000	8663495	284	UUiT9RITQ9PW6BhXK0y2jaeg

Figure 4.1. Extract Channel Details

In Figure 4.1, we have retrieved channel information using the YouTube Data API. We can utilize various endpoints provided by the API. By making authorized requests with an API key, we accessed a wealth of information about YouTube channels. Once the request is made and the response is received, we process the JSON data to extract the desired information.

Extract Video Data

```
video_data['Published_date'] = pd.to_datetime(video_data['Published_date']).dt.date
video_data['Views'] = pd.to_numeric(video_data['Views'])
video_data['Likes'] = pd.to_numeric(video_data['Likes'])
video_data['Views'] = pd.to_numeric(video_data['Views'])
video_data
```

	Title	Published_date	Views	Likes	favorite	Comments
0	The Death of the Full-Time Job (Rise of the Co...	2023-09-22	4967	338	0	70
1	How to Survive a Down Data Job Market	2023-09-05	15661	513	0	64
2	The Harsh Reality of the Data Job Market	2023-08-24	104019	2530	0	292
3	7 Industries AI will Aggressively Disrupt	2023-08-14	5597	229	0	41
4	What's in My Data Science Travel Bag? (50+ Fli...	2023-08-07	3858	122	0	47
5	7 Enticing Jobs AI Will Create	2023-07-24	4416	191	0	37
6	The ChatGPT Code Interpreter is OVERRATED	2023-07-22	3951	142	0	31
7	Exciting Announcement!	2023-06-23	6515	294	0	50
8	AI: A Customer Service Revolution?	2023-05-26	3746	117	0	35
9	Beginner Kaggle Data Science Project Walk-Thro...	2023-05-16	27264	933	0	57
10	How I Would Learn Data Science with AI (If I C...	2023-05-04	31815	1349	0	77
11	This AI question will determine our future	2023-04-06	4230	261	0	37
12	I'll Never Code the Same Again... (GPT-4 is OP)	2023-04-03	70758	828	0	79
13	Your current portfolio is actually hurting you...	2023-03-30	6684	397	0	7
14	Traditional "networking" is stupid. Do THIS in...	2023-03-27	6713	365	0	3
15	you're job searching wrong. #shorts	2023-03-23	4066	245	0	3
16	The Economics of Data Roles #shorts	2023-03-21	8000	509	0	10
17	Why your phone knows everything you say #shorts	2023-03-18	2698	121	0	7

Figure 4.2. Extract Video Data

In Figure 4.2, using the API key we have extracted the video data like Views, Likes, Comments, etc.

Visualize Video Data

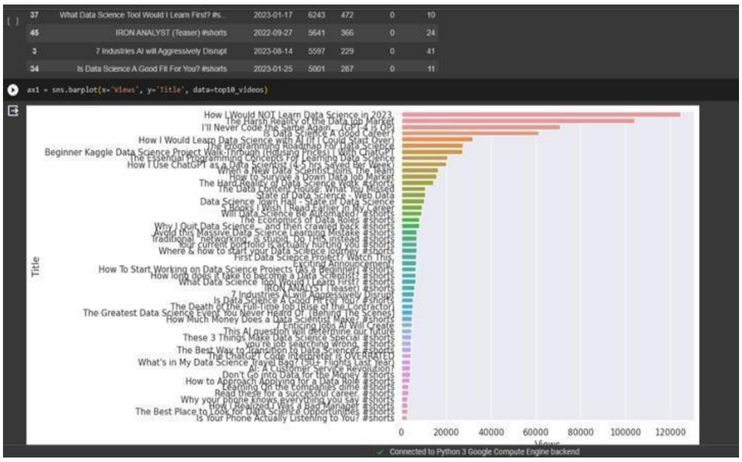
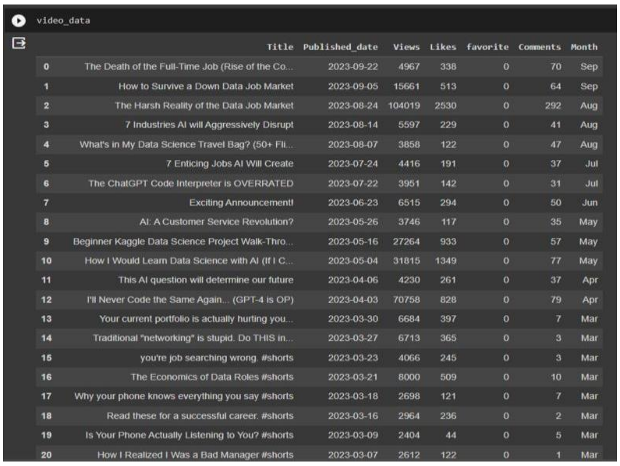


Figure 4.3. Visualize Video Data

In Figure 4.3, Providing users with valuable insights and functionalities we can visualize the channel video data.

Extract Video Data



```
video_data
```

	Title	Published_date	Views	Likes	favorite	Comments	Month
0	The Death of the Full-Time Job (Rise of the Co...	2023-09-22	4967	338	0	70	Sep
1	How to Survive a Down Data Job Market	2023-09-05	15661	513	0	64	Sep
2	The Harsh Reality of the Data Job Market	2023-08-24	104019	2530	0	292	Aug
3	7 Industries AI will Aggressively Disrupt	2023-08-14	5597	229	0	41	Aug
4	What's in My Data Science Travel Bag? (50+ Fli...	2023-08-07	3858	122	0	47	Aug
5	7 Enticing Jobs AI Will Create	2023-07-24	4416	191	0	37	Jul
6	The ChatGPT Code Interpreter is OVERRATED	2023-07-22	3951	142	0	31	Jul
7	Exciting Announcement!	2023-06-23	6515	294	0	50	Jun
8	AI: A Customer Service Revolution?	2023-05-26	3746	117	0	35	May
9	Beginner Kaggle Data Science Project Walk-Thro...	2023-05-16	27264	933	0	57	May
10	How I Would Learn Data Science with AI (If I C...	2023-05-04	31815	1349	0	77	May
11	This AI question will determine our future	2023-04-06	4230	261	0	37	Apr
12	I'll Never Code the Same Again... (GPT-4 is OP)	2023-04-03	70758	828	0	79	Apr
13	Your current portfolio is actually hurting you...	2023-03-30	6684	397	0	7	Mar
14	Traditional "networking" is stupid. Do THIS in...	2023-03-27	6713	365	0	3	Mar
15	you're job searching wrong. #shorts	2023-03-23	4066	245	0	3	Mar
16	The Economics of Data Roles #shorts	2023-03-21	8000	509	0	10	Mar
17	Why your phone knows everything you say #shorts	2023-03-18	2698	121	0	7	Mar
18	Read these for a successful career. #shorts	2023-03-16	2964	236	0	2	Mar
19	Is Your Phone Actually Listening to You? #shorts	2023-03-09	2404	44	0	5	Mar
20	How I Realized I Was a Bad Manager #shorts	2023-03-07	2612	122	0	1	Mar

Figure 4.4. Extract Video Data

In Figure 4.4, We have fetched the statistics such as likes, dislikes, and comments, which offer valuable insights into the video's engagement metrics and may be included in the response.

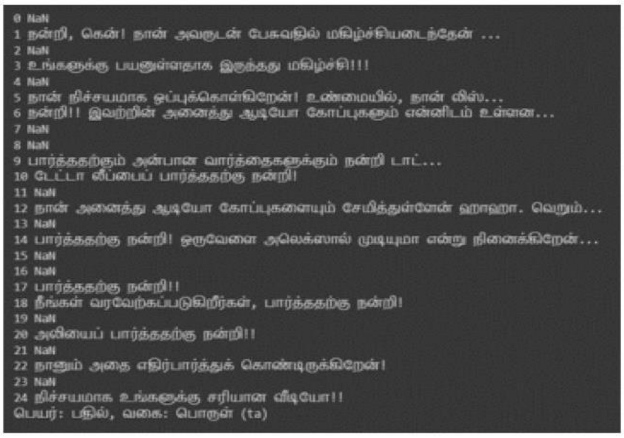
Comment Extraction

```
                  Name                                          Comment
               Ken Jee  Again, special thanks to Alex for speaking wit...
                   NaN                                              NaN
 Datascience Concepts  Ken, you never fail to amaze with such useful ...
                   NaN                                              NaN
      chinmay harkawat  Dude.. your interviews are awesome... <br>You ...
                   NaN                                              NaN
                   NaN                                              NaN
          Vinod yadav                                              Bhai
            Andrew Mo  Subscribed to @innoarchitech, his stuff is awe...
                   NaN                                              NaN
                   NaN                                              NaN
           Boris Giba  Awesome interview; it was very interesting to ...
                   NaN                                              NaN
       aditya malhotra  Thanks Ken and Alex for such a great interview...
                   NaN                                              NaN
            Why of AI  Here&#39;s the link to the AI Readiness articl...
Chai Time Data Science  Got a chance to watch the premier live and I r...
                   NaN                                              NaN
                   NaN                                              NaN
           Ali Kaiser  Thanks for such an enlightened interview Ken J...
                   NaN                                              NaN
            Why of AI  I can&#39;t wait for this to premiere! Ken&#39...
                   NaN                                              NaN
     Daniel Domínguez  Yoooo, it is my dream to work IndyCar or f1. C...
                   NaN                                              NaN
```

Figure 4.5. Comment Extraction

In Figure 4.5, YouTube comments were extracted from the dataset by using a data frame it will be easy to translate the comments.

Translation



```
0 NaN
1 நன்றி, கென்! நான் அவருடன் பேசுவதில் மகிழ்ச்சியடைந்தேன் ...
2 NaN
3 உங்களுக்கு பயனுள்ளதாக இருந்தது மகிழ்ச்சி!!!
4 NaN
5 நான் நிச்சயமாக ஒப்புக்கொள்கிறேன்! உண்மையில், நான் விஸ்...
6 நன்றி!! இவற்றின் அனைத்து ஆடியோ கோப்புகளும் என்னிடம் உள்ளன...
7 NaN
8 NaN
9 பார்த்ததற்கும் அன்பான வார்த்தைகளுக்கும் நன்றி டாட்...
10 டேட்டா லீப்பைப் பார்த்ததற்கு நன்றி!
11 NaN
12 நான் அனைத்து ஆடியோ கோப்புகளையும் சேமித்துள்ளேன் ஹாஹா. வெறும்...
13 NaN
14 பார்த்ததற்கு நன்றி! ஒருவேளை அலெக்ஸால் முடியுமா என்று நினைக்கிறேன்...
15 NaN
16 NaN
17 பார்த்ததற்கு நன்றி!!
18 நீங்கள் வரவேற்கப்படுகிறீர்கள், பார்த்ததற்கு நன்றி!
19 NaN
20 அலியைப் பார்த்ததற்கு நன்றி!!
21 NaN
22 நானும் அதை எதிர்பார்த்துக் கொண்டிருக்கிறேன்!
23 NaN
24 நிச்சயமாக உங்களுக்கு சரியான வீடியோ!!
பெயர்: பதில், வகை: பொருள் (ta)
```

Figure 4.6. Translation

In Figure 4.6, Using the "Google Trans" module we provided transcripts in the selected language. Then we used translated transcripts to show the user, allowing them to better understand the content of the video and the information being conveyed.

Comment Analysis

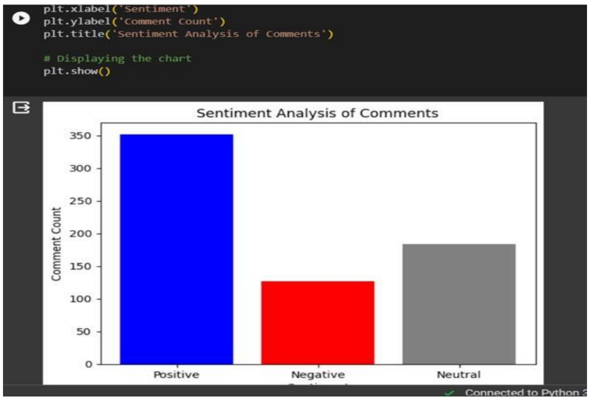


Figure 4.7. Comment Analysis

In above Figure 4.7, YouTube comments were extracted from the dataset by using a data frame and are divided into 3 phases (positive, negative neutral) which makes it easy for the content creators to observe the comments.

5. Conclusion and Future Work

In this research, the YouTube comments were classified into three categories: good, negative, and neutral, and then we represented them using a pictorial graph. This gives us the ability to gauge the mood of the audience, spot patterns, and identify key voices in online communities. The unstructured text was transformed into actionable insights using natural language processing (NLP) techniques. These insights can be used to measure the influence of influencers, analyze social phenomena, and improve the content. In this research, a good analysis of the comments in multiple languages based on the user input was achieved. The future work is to construct a model for detecting sentiment analysis on YouTube videos via user comments using deep learning.

Reference

[1] Athar, Awais, and Simone Teufel. "Detection of implicit citations for sentiment detection." In Proceedings of the workshop on detecting structure in scholarly discourse, pp. 18-26. 2012.

[2] Abdalgader, Khaled, and Aysha Al Shibli. "Experimental results on customer reviews using lexicon-based word polarity identification method." IEEE Access 8 (2020): 179955-179969.

[3] Poria, Soujanya, Erik Cambria, Alexander Gelbukh, Federica Bisio, and Amir Hussain. "Sentiment data flow analysis by means of dynamic linguistic patterns." IEEE Computational Intelligence Magazine 10, no. 4 (2015): 26-36.

[4] Xu, Guixian, Yueting Meng, Xiaoyu Qiu, Ziheng Yu, and Xu Wu. "Sentiment analysis of comment texts based on BiLSTM." Ieee Access 7 (2019): 51522-51532.

[5] Ishaq, Adnan, Sohail Asghar, and Saira Andleeb Gillani. "Aspect-based sentiment analysis using a hybridized approach based on CNN and GA." IEEE Access 8 (2020): 135499-135512.

[6] Vinodhini, G., and R. M. Chandrasekaran. "Sentiment analysis and opinion mining: a survey." International Journal 2, no. 6 (2012): 282-292.

[7] Cambria, Erik, and Bebo White. "Jumping NLP curves: A review of natural language processing research." IEEE Computational intelligence magazine 9, no. 2 (2014): 48-57.

[8] Athar, Awais. *Sentiment analysis of scientific citations*. No. UCAM-CL-TR-856. University of Cambridge, Computer Laboratory, 2014.

[9] Pokharel, Rhitabrat, and Dixit Bhatta. "Classifying youtube comments based on sentiment and type of sentence." arXiv preprint arXiv:2111.01908 (2021).

[10] Yang, Yiming, and Jan O. Pedersen. "A comparative study on feature selection in text categorization." In Icml, vol. 97, no. 412-420, p. 35. 1997.

[11] Krishna, Amar, Joseph Zambreno, and Sandeep Krishnan. "Polarity trend analysis of public sentiment on YouTube." In Proceedings of the 19th international conference on management of data, pp. 125-128. 2013.

[12] Bhuiyan, Hanif, Jinat Ara, Rajon Bardhan, and Md Rashedul Islam. "Retrieving YouTube video by sentiment analysis on user comment." In 2017 IEEE International Conference on Signal and Image Processing Applications (ICSIPA), pp. 474-478. IEEE, 2017.

[13] Alsubait, Tahani, and Danyah Alfageh. "Comparison of machine learning techniques for cyberbullying detection on youtube arabic comments." International Journal of Computer Science & Network Security 21, no. 1 (2021): 1-5.

[14] Deng, Li, and Yang Liu, eds. Deep learning in natural language processing. Springer, 2018.

[15] Mulholland, Eleanor, Paul Mc Kevitt, Tom Lunney, and Karl-Michael Schneider. "Analysing emotional sentiment in people’s YouTube channel comments." In Interactivity, Game Creation, Design, Learning, and Innovation: 5th International Conference, ArtsIT 2016, and First International Conference, DLI 2016, Esbjerg, Denmark, May 2–3, 2016, Proceedings 5, pp. 181-188. Springer International Publishing, 2017.

[16] Muhammad, Abbi Nizar, Saiful Bukhori, and Priza Pandunata. "Sentiment analysis of positive and negative of youtube comments using naïve bayes–support vector machine (nbsvm) classifier." In 2019 International Conference on Computer Science, Information Technology, and Electrical Engineering (ICOMITEE), pp. 199-205. IEEE, 2019.

[17] Savigny, Julio, and Ayu Purwarianti. "Emotion classification on youtube comments using word embedding." In 2017 international conference on advanced informatics, concepts, theory, and applications (ICAICTA), pp. 1-5. IEEE, 2017.

[18] Boudad, Naaima, Rdouan Faizi, Richard O. Haj Thami, and Raddouane Chiheb. "Sentiment Classification of Arabic Tweets: A Supervised Approach." J. Mobile Multimedia 13, no. 3&4 (2017): 233-243.

[19] Manning, Christopher D., Mihai Surdeanu, John Bauer, Jenny Rose Finkel, Steven Bethard, and David McClosky. "The Stanford CoreNLP natural language processing toolkit." In Proceedings of 52nd annual meeting of the association for computational linguistics: system demonstrations, pp. 55-60. 2014.

[20] Agrawal, Ameeta, and Aijun An. "Unsupervised emotion detection from text using semantic and syntactic relations." In 2012 IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology, vol. 1, pp. 346-353. IEEE, 2012.

[21] Jiashen Liu, Jia Liu, Xinyi Wang, and Jian Zhang. \"Sentiment analysis on social media: A survey.\" IEEE Transactions on Computational Social Systems, 7(3),2020, pp. 682-705.