An Automated Framework for Summarizing YouTube Videos Using NLP

Siri *Dharmapuri*[*1],Sashank *Desu*[1], Karthik *Alladi*[1], *Harika* Gummadi[1], Harshit *Gupta*[2] , *S. Noor Mohammad* Shareef[3]

[1]Gokaraju Rangaraju Institute of Engineering and Technology, Department of Computer Science, Hyderabad, Telangana, India
[2]Uttaranchal School of Computing Sciences, Uttaranchal University, Dehradun, 248007, India
[3]KG Reddy College of Engineering & Technology, Hyderabad

Abstract. In recent times, YouTube has increasingly become the preferred platform to consume educational content. In order to learn complex and intricate concepts, a student must sit through many of hours of YouTube videos where an average video length is about 20 minutes. To see if the content of a given YouTube video is relevant to what the user is looking for, YouTube Video Summarizer was conceptualized. YouTube Video Summarizer is a Chrome Extension tool which can be used to quickly generate the summary of a YouTube video using the English-language transcript of the video Automation. This allows for a seamless generation of a synopsis without spending hours watching the content to determine its relevancy.

1. Introduction

YouTube is an online video-sharing platform founded by Chad Hurley in 2005. It is the second most visited website, right behind Google search. Since its inception, it has been frequently used as an educational platform with a goal of teaching millions of students across the globe.

YouTube Video Frame Summarizer [1]–[4] generates the summary of a YouTube video by extracting chunks of information and piecing it together through an NLP model using the automatically generated English-language transcript provided with the video.

1.1 Rationale

The intention behind developing this project was to make YouTube videos easier to understand at a glimpse by providing a gist of the content and then leaving it to the user to decide whether the content is relevant and informative. This makes the tedious process of scouring through a long video for information easier.

* Corresponding author: siri1686@grietcollege.com

1.2 Goal

The goal of the project is to output a concise, grammatically correct summary of a given YouTube video in the form of text which indicates the relevancy of the content without the need to watch the video. Relevancy is deduced by the end user based on what kind of content they are looking for.

2. Literature Review

Web is the most frequently used networking aid which satisfies the requirements of all types of users; it provides a solution for any type of problem definition. While developing a web portal the appearance of web portal makes a development more critical. The good appearance of a web can easily attract more number of visitors which is a success of web portal. For designing and developing such well structured and with the good appearance of web we have to choose a proper technology. The technological needs of satisfying a good web portal can be fulfilled by "python" and "flask"[1].

Youtube has now became one of the biggest platforms for entertainment , study , cooking , and many more stuff like that. Although the user base varies from young to old, YouTube is most popular among young people who prefer the variety of content, interactive features and instant gratification of YouTube video content to regular television. Many use it for entertainment purposes, to learn how to do something (teaching), to keep up with the latest music videos from their favourite artists, and more. YouTube is available in almost all countries and in more than 50 languages. Like Google, all you need to create and use a YouTube account is a Google account. But the only problem is at times it happens that we do not have that much amount of free time to watch a particular video . So , that is where this paper comes into the picture which allows us to provide a short summary of the youtube video which not only saves our time but also gives us crisp content which we can refer to while writing a summary or a synopsis without wasting large amount of time [2].

3. Methodology

3.1 Python

Python was initially developed by Guido van Rossum, and first implemented in December 1989. Van Rossum was the lead developer for the project until 12th July 2018. Python is the primary programming language for this project. The local webserver and the core summarizing logic were developed in Python using Flask [5]–[7] and HuggingFace Transformers [8] library.

a. Machine Learning

Machine Learning [9] plays a key role. It is used in tasks like perceiving human appearances or self-driving vehicles. With the growing proportions of data, there is substantial research and findings to acknowledge that Machine Learning is now a fundamental aspect for technological progression.

b. Natural Language Processing

It is used for activities like mail spam detection, texting etc. It helps computers converse with humans. It helps in speech recognition and text analytics. NLP is a technology which can understand the human language. NLP has two phases - Data Preprocessing and Algorithm Generation. The former involves the process of cleaning the raw data and transforming it into formats that the machine can interpret.

NLP involves two different techniques. There are syntax and semantic analysis. Syntax includes- Stemming, Parsing, Sentence Breaking, Morphological Segmentation, Word Segmentation. Semantics includes- Named Entity Recognition, Natural Language Generation, Word Sense Disambiguation.

The main functions of NLP are:

- Text classification
- Text extraction
- Machine translation

c. Natural Language Processing Tasks

Some of the NLP tasks are:

- Speech recognition
- Language translation
- Content summarization
- Text suggestions
- Content filtering

d. Natural Language Processing Tools

i. HuggingFace Transformers

HuggingFace Transformers [8], [10] is a collection of APIs and tools to download state-of-the-art pretrained NLP models. This project utilized the T5 [11], [12] model, developed by Google. The T5 is a text-to-text transformer that has a variety of features that can be applied to a text like translation, summarization etc. It is an encoder-decoder type model. There are 5 sizes of T5 – t5-small, t5-base, t5-large, t5-3b, t5-11b. The t5-base model was used in this project for the purpose of transcript summarization.

ii. Preprocessing

The practice for arranging actual data to be used for a Machine learning algorithm is referred to as data preprocessing. Empirical data can contain disturbance, null data, and is in an unsuitable form, making it impossible to utilize in machine learning models directly. Preprocessing the data is an essential task for cleaning the data and making it viable for a machine learning model to process useful data which improves the model's efficacy. The steps involved in preprocessing are:

- Removing punctuations like: . , ! $( ) * % @
- Removing Stop words
- Tokenization

iii. Tokenization

- Converts a sentence into collection of words, called tokens
- Breaks the text into smaller portions
- It discovers the meaning of the text by inspecting the words and their sequences

4. Results

To measure the accuracy of the generated summaries [13]–[20] 12 YouTube videos of various genres were randomly picked. The outputs were then measured against summaries generated by humans, which were used as reference summaries.

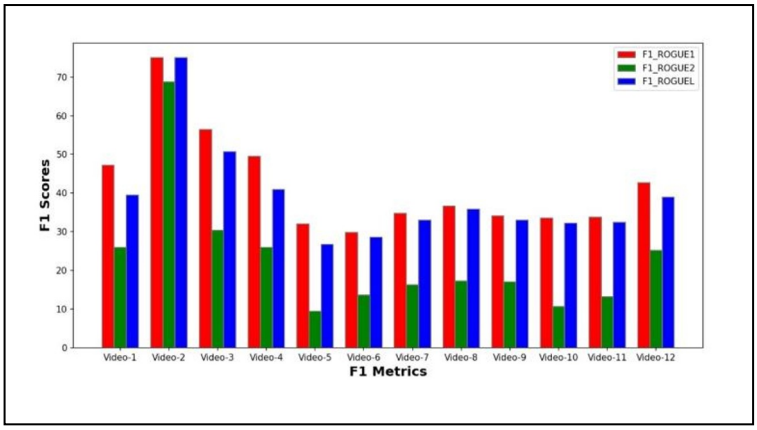

Fig. 1. F1-Scores for 12 YouTube videos

A technique called ROGUE [21]–[24] was utilized to evaluate the performance of the current model. The metrics tested were – ROGUE-N (1 and 2) & ROGUE-L [25]. We compute the F1-Scores for each of these metrics through precision and recall values. The F1-Scores give us a valid measure of our model's performance since it depends on the model gathering essential words – recall, while avoiding irrelevant words – precision.

Table 1 consists of the list of videos whose transcripts were extracted and summarized. Column that represents Transcript Length indicates the total number of words spoken in the YouTube video. Column that represents Summary Length indicates the total number of words in the summary generated. These columns represent the efficiency of the summarization model for a given transcript length.

Figure 1 illustrates the F1 Scores of individual videos selected for summarization. Selected videos are indicated by their respective numbers in Table 1. The 3 metrics being tested – ROGUE-1, ROGUE-2, ROGUE-L – have all been given F1-Scores for each video.

ROGUE is considered a good metric for estimating summarization and translation capabilities of the model. It is effective in assessing the accuracy of the summarized text, but if we have multiple sentence sequences that use synonyms to convey the same meaning, the summary is generally given a low score. ROGUE only looks for exact matches and does not take semantics into account, hence the resulting F1-Scores in Figure 1. Despite this flaw, our model outperforms other transcript summarizing models by achieving relatively high F1-Scores for baseline summarization [26].

Table 1. List of videos selected for summarization

Video Number	Video Name	Duration (minutes)	Transcript Length (number of words)	Summary Length (number of words)
1	Iphone14	9	1814	314

Video Number	Video Name	Duration (minutes)	Transcript Length (number of words)	Summary Length (number of words)
2	Dynamic Programming	15	2334	470
3	Stacks	18	2742	425
4	BitcoinExplained	11	1575	324
5	RussiaUkraineWar	9	1448	228
6	VeritasiumHorses	16	2791	352
7	VeritasiumImaginary	23	3715	460
8	TEDxParadox	5	700	113
9	DBCooperDisappearance	20	3013	389
10	TomScottAirtag	21	3524	372
11	JHFrance	14	2443	298
12	RRRReview	7	1423	223

5. Conclusion

Automatic Framework YouTube Video Summarizer provides users with a coherent, syntactically, and grammatically accurate summary of any YouTube video Frames with a default English-language transcript. It helps users understand the crux of the video by summarizing the words spoken throughout its duration, at a simple glance.It is able to achieve the task of summarization using the tools provided by the HuggingFace library, and with the help of the Google T5 NLP model. After fine-tuning the model for video summarization, the results achieved were greater than the average F1-Scores of alternate models and numerous other implementations of the same model.

6. Future Scope

There is scope for further tuning and evaluating the performance of the model and implementing the following features –

- Automatic Subtitle Generation using Speech-To-Text NLP Models
- Multilingual Support For Summary Generation
- Deploying The Project On A Cloud Database
- Displaying Additional Content Like Video Transcript, Transcript Length

References

1. P. Mehta, Creating Google Chrome Extensions. Berkeley, CA: Apress, 2016, ch. 1, doi: 10.1007/978-1-4842-1775-7_1.K. Murai and V. Klyuev, (2015).
2. M. Frisbie, Berkeley, CA: Apress, (2023). https://doi.org/10.1007/978-1-4842-8725-5_6.
3. "YouTube-Transcript-API," PyPI. [Online]. Available: https://pypi.org/project/youtube-transcript-api/.
4. F. Aslam, H. Mohammed, and P. Lokhande, "in IJARCS, vol**. 6 ,** (2015)
5. A. Mani, "Building Restful Apis with Flask in Python," Atma's Blog, Oct. 20, 2019. [Online]. Available: https://atmamani.github.io/blog/building-restful-apis-with-flask-in-python/.
6. J. Chan, R. Chung, and J. Huang, Packt Publishing Ltd, (2019).
7. T. Wolf *et al.*, *arXiv:1910.03771 [cs]*, Jul. 14, (2020).
8. T. M. Mitchell, *Machine Learning, vol. **1**.,* New York: McGraw-Hill, (2007).
9. L. Tunstall, L. Von Werra and T. Wolf,*O'Reilly Media, Inc.*, (2022).
10. T5, Hugging Face Transformers documentation, [Online]. Available: https://huggingface.co/docs/transformers/model_doc/t5.
11. "Exploring Transfer Learning with T5: The Text-to-Text Transfer Transformer," *Google AI Blog*, 20-Feb-2020. [Online]. Available: https://ai.googleblog.com/2020/02/exploring-transfer-learning-with-t5.html.
12. U. Khandelwal *et al.*, *arXiv:1905.08836 [cs]*, May 21, (2019).
13. U. Kapoor," *DSpace @ Delhi Technological University: SUMMARISATION OF YOUTUBE VIDEO USING TRANSFORMERS AND API*, 01-May-2022. [Online]. Available: http://dspace.dtu.ac.in:8080/jspui/handle/repository/19635.
14. P. Theron, J. Dentan, and L. Gautier,May (2022), doi: 10.13140/RG.2.2.20076.85125.
15. C. Raffel *et al.*, *arXiv preprint arXiv:1910.10683*, (2020).
16. H. K. Dhalla, "A Performance Analysis of Native JSON Parsers in Java, Python, MS.NET Core, JavaScript, and PHP," *2020 16th International Conference on Network and Service Management (CNSM)*, Izmir, Turkey, (2020), doi: 10.23919/CNSM50824.2020.9269101.
17. A. Nafies, *Medium, The Startup*, 25-Oct-2020. [Online]. Available: https://medium.com/swlh/parsing-rest-api-payload-and-query-parameters-with-flask-better-than-marshmallow-aa79c889e3ca.
18. Y. Agrawal *et al.*, *EasyChair Preprint (5404)*, (2021).
19. P. Dwivedi, *Towards Data Science*, Sep. 9, (2020). [Online]. Available: https://towardsdatascience.com/fine-tuning-a-t5-transformer-for-any-summarization-task-82334c64c81.
20. J. Briggs," *Towards Data Science*, 02-Sep-2021. [Online]. Available: https://towardsdatascience.com/the-ultimate-performance-metric-in-nlp-111df6c64460.
21. T. Berg-Kirkpatrick, D. Burkett, and D. Klein, "*An empirical investigation of statistical significance in NLP,"* in Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning. Jeju Island, Korea: Association for Computational Linguistics, Jul. (2012).
22. R. Wang and J. Li, "*Bayes Test of Precision, Recall, and F1 Measure for Comparison of Two Natural Language Processing Models*," in Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, (2019).

23. Sokolova, Marina & Japkowicz, Nathalie & Szpakowicz, Stan, *"Beyond Accuracy, FScore and ROC: A Family of Discriminant Measures for Performance Evaluation,"*Australasian Joint Conference on Artificial Intelligence.Vol. **4304. 1015-1021. 10.1007/11941439_114.**
24. G. Kavita, "An Intro to Rouge, *FreeCodeCamp.org*, 24-Oct-2017. [Online]. Available: https://www.freecodecamp.org/news/what-is-rouge-and-how-it-works-for-evaluation-of-summaries-e059fb8ac840/.
25. P. Shruti *et al.,arXiv:1906.07901 [cs.CL]*, Jun 19, (2019). Australasian Joint Conference on Artificial Intelligence
26. *Conference Paper Deep Learning Framework for Liver CT Image Segmentation and Risk Prediction,*Gundavarapu, M.R., Saginala, R., Varma, M.A., ...Bodduluri, A.S., (2023**), 645** LNNS, pp. **189–201**
27. Dusa, D., Gundavarapu, M.R.8th *International Conference on Advanced Computing and Communication Systems*, ICACCS 2022, (2022), pp. **1023–1028**
28. Tejaswini Priyanka, Y. Reshma Reddy, D. Vajja, G. Ramesh and S. Gomathy (2023). *A Novel Emotion based Music Recommendation System using CNN. 2023 7th International Conference on Intelligent Computing and Control Systems* (ICICCS), Madurai, India, 592-596, doi: 10.1109/ICICCS56967.2023.10142330.
29. Ramesh, G., Reddy, K.S.S., Ramu, G., Reddy, Y.C.A.P., Somasekar, J. (2023). *An Empirical Study on Discovering Software Bugs Using Machine Learning Techniques.*
30. In: Buyya, R., Hernandez, S.M., Kovvur, R.M.R., Sarma, T.H. (eds) *Computational Intelligence and Data Analytics. Lecture Notes on Data Engineering and Communications Technologies,* vol **142**. Springer, Singapore. https://doi.org/10.1007/978-981-19-3391-2_14.
31. Chandrika Lingala, and Karanam Madhavi, "*A Hybrid Framework for Heart Disease Prediction Using Machine Learning Algorithms ",* E3S Web of Conferences 309, 01043 (2021), ICMED 2021. https://doi.org/10.1051/) e3sconf/202130901043 SCOPUS
32. Chandra Sekhar Reddy P , Sakthidharan G, Kanimozhi Suguna S, Mannar Mannan J, Varaprasada Rao P, IJEAT. **8,** (2019)