Proceedings of the Sixth International Conference on Trends in Electronics and Informatics (ICOEI 2022)
IEEE Xplore Part Number: CFP22J32-ART; ISBN: 978-1-6654-8328-5

Meeting Summarizer using Natural Language Processing

[1]Geethika Vadlamudi, [2]Naveena Vemuru, [3]Surendhra Vangapalli, [4]Ravi Kishan Surapaneni, [5]Sailaja Nimmagadda

[1,2,3,4]*Department of Computer Science & Engineering*

[1,2,3,4]*VR Siddhartha Engineering College*

[1,2,3,4]Vijayawada, India

[1]geethika302000@gmail.com, [2]naveenavemuru@gmail.com, [3]surendhra2000@gmail.com, [4]suraki@vrsiddhartha.ac.in, [5]nsailaja@vrsiddhartha.ac.in

2022 6th International Conference on Trends in Electronics and Informatics (ICOEI) | 978-1-6654-8328-5/22/$31.00 ©2022 IEEE | DOI: 10.1109/ICOEI53556.2022.9777155

***Abstract*— There is a demand for meeting summary, when entering the virtual world. Meeting transcripts, such as Microsoft Teams transcripts and Google Meet Transcripts, are already available. It's tough to read long transcripts, thus summary transcript production makes it easier to acquire a concise document. A summary can assist in reaching a meeting's key conclusion. The main purpose of this project is to construct a summary from the transcript. This is based on the methodical integration of different natural language processing (NLP) approaches, such as meeting summarization, that were previously developed individually and tested offline using standard datasets. This project takes into account Microsoft Teams transcripts. The Term Frequency Inverse Document Frequency (TF-IDF) method and the PageRank algorithm were employed in this research as NLP tools.**

Keywords*— *Natural Language Processing, Summarization, Transcript, Term Frequency- Inverse Document Frequency, PageRank

I. INTRODUCTION

Meetings are the most common method of employee involvement and contact. A meeting is a time for exchanging ideas and engaging in in-depth discussions. Meeting summaries are crucial because they convey the significant information of conversations in a concise manner. Meeting transcripts are easily accessible through online meeting platforms such as Microsoft Teams, Google Meet, and others. In general, reading and comprehending the entire document takes time. As a result, summaries are critical because readers are only interested in the relevant context of debates. As a result, a meeting summarizer is suggested. The meeting summarizer aids in the summarization of transcripts that are accessible.

Document summarization differs from key phrase extraction and topic modeling in that it focuses on the content of the document. In this scenario, the end output is still a document, but it's just a few sentences long, depending on how long the summary is to be. This is akin to a research paper's abstract or executive summary. The basic goal of automated document summary is to complete it without the use of human input, with the exception of executing computer code. By examining the content and context of documents, mathematical and statistical models aid in the construction and automation of the work of document summarization.

There approaches of document summarization are extractive and abstractive summarization techniques. Extractive summarization employ mathematical and statistical principles like as SVD to extract a critical subset of content from the original document, with this subset containing the most important information and serving as the document's focal point. This material can take the form of words, phrases, or even complete sentences. The end result of this method is a brief executive summary based on a few words taken from the original document. This technique does not create any new content, hence the name extraction-based.

Abstraction summarization techniques are more advanced and complex. They employ language semantics to produce representations and natural language generation (NLG) approaches, in which the computer generates text and summaries on its own using knowledge bases and semantic representations, just like a human would. These strategies thanks can be easily implemented to deep learning, but they require a lot of data and computation.

The paper focus to summarize the transcript. Here, Microsoft Teams transcript is considered and pre-processed accordingly. It includes various techniques like TextRank, and TF-IDF.

A. Problem Statement

Meetings are an essential aspect of every institution's activities, whether they are held online or in person. Meeting transcription and summarization standards, on the other hand, is always a nuisance because it necessitates arduous human labour. Summary always gives us a brief description about the meeting instead of going through long transcripts. Also in every organization persons who plays roles like managers have to report the summary of their meeting with their team mates. This becomes difficult when one have to

summarize every meeting manually. As a result, automatic meeting transcript summarization systems are required. This helps to reduce the human effort as well as time.

II. LITERATURE SURVEY

Smart Meeting is a tool utilized in [1] that allows users to automatically record, transcribe, summarize, and organize material in an in-person meeting. Transcription by ASR, transcript enrichment, and meeting summarization are the three processes. ASR transcription is to convert the speech recordings made by each participant's device into text by considering hybrid ASR models. Transcript enrichment is used to do the segmentation and grouping based on the speakers' voiceprints for voice separation and speaker identification. WSNeuSummary uses an unsupervised summarising model. In [2], a comparison of the NLTK, word embedding, T5 models and findings is presented, resulting in a more concise overview. For extractive summarization, the NLTK model employs the sentence ranking algorithm, the Word Embedding model employs pre-trained Glove Embeddings. The T5 model uses transformer architecture to achieve abstractive summarization. In [3], a system is created that combines topic modeling and phrase selection in call transcripts with punctuation restoration to provide better readable summaries. It has ten steps which are transcript channel separation, partial punctuation restoration, document preparation, topic modeling, dominant topic identification, significant term selection, Summary generation, punctuation restoration, summary tabulation, summary efficacy determination. The authors of [4] presented a comparison of the three techniques, TF-IDF, TextRank, and Latent Dirichlet Allocation (LDA). Clearly, when employed in an unsupervised manner and according to the properties of each approach used in the implementation, TextRank outperforms TF-IDF and LDA. In [5], a technique called Maximal Marginal Relevance (MMR) is used for identifying the important sentences in the input. an NLP system was developed employing powerful AI algorithms to extract meta-data from speech transcripts in [6]. Conversational casual dialogues are identified, essential concepts are identified, phase extraction, noise removal, phase normalization, phase ranking, and multi label document labeling are all part of the technique. The research object of the keyword extraction approach in [7] is English news text. TF-IDF and TextRank algorithm are combined to extract keywords. The word graph model, word frequency and inverse document frequency are constructed. The algorithm's performance is measured by the recall rate, precision rate, and macro average value. In terms of performance parameters and extraction impact, the results suggest that combining TF-IDF and the TextRank algorithm surpasses the standard technique significantly. [8] introduced BERT, which stands for Bidirectional Encoder Representations from Transformers. This mainly contains two steps: pre-training and fine tuning. A neural extractive summarization model is introduced in [9] on which the latent model is dependent. A sentence compression model is then described. Latent model uses this and finally the latent model is presented. [10] built a modified graph-based approach to extracting summary of a text content in this study. When calculating sentence similarity, instead of using the standard TF-IDF, is f-weighted cosine similarity is employed, which yields interesting results when tested with news items. After careful survey of many journals and research papers, combination of TF-IDF and TextRank is found to give effective extractive summary.

III. PROPOSED SYSTEM

A. Proposed Architecture

"Fig. 1" shows the proposed architecture of Meeting Summarizer using NLP. The system takes transcript document as an input and performs operations to produce summary document as an output. Microsoft Teams transcripts are considered for this project. The time stamps are removed from the transcript and every sentence is matched to every person. Then the text is split into sentences which are used further. In pre-processing text is standardized and stop words are removed. The document term feature matrix is built using Term Frequency – Inverse Document Frequency (TF-IDF). Term Frequency of each word will be calculated. Term Frequency refers to the number of times a word appears in a document. The inverse document frequency refers to how common or rare the word is in a document. This can be calculated by dividing the total number of documents with the number of documents which contains that word. By multiplying TF and IDF, TF-IDF score is obtained. The higher the score for a word, higher the importance of word in that document. A document similarity matrix is built by multiplying document term feature matrix and its inverse. A document similarity graph is generated. These sentences are used as the vertices and the similarities between each pair of documents as the weight or score coefficient and are fed to the PageRank algorithm. The score for each sentence is calculated. The top sentences are given as output.

"Fig. 2" shows the flow chart of the proposed model. The flowchart starts with the input transcript document. Extraction of sentences and tokenization is performed in order to use the sentences further. Then TF – IDF is used to produce document similarity matrix. This is fed to TextRank Algorithm which gives ranking for each sentence. There after the top ranked sentences are used to produce the output summary document.

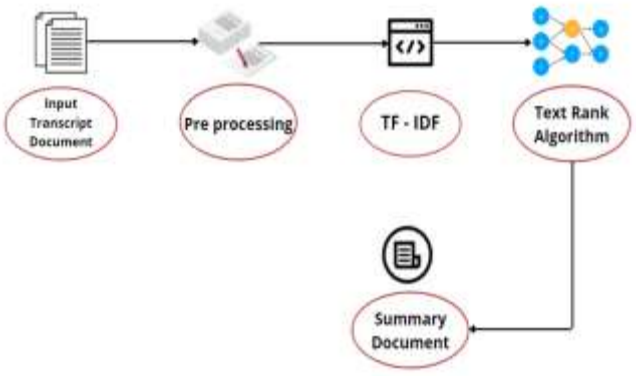

Fig. 1. Architecture of the proposed system.

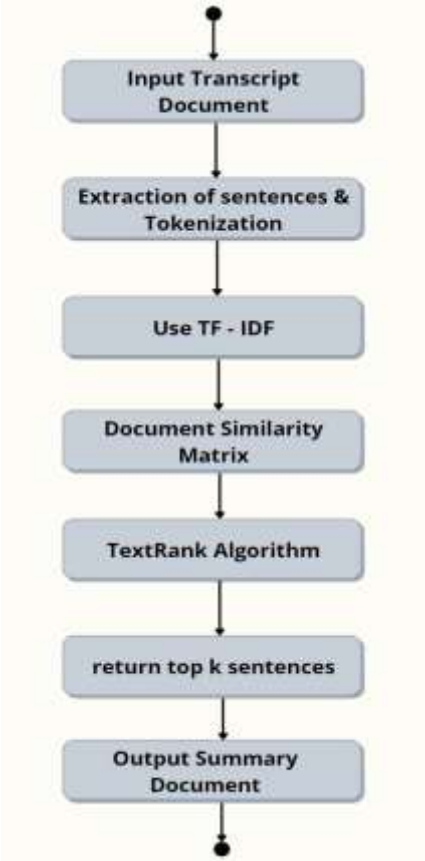

Fig. 2 Flow Chart of the proposed model

The transcript is tokenized using method of nltk. The stop words are removed and stemming is performed for the words in the document. These are vectorized using numpy. So the normalized sentences are obtained. Tfidfvectorizer is imported from sklearn module. A document term frequency is formed with the help of this. From dt matrix inverse document frequency is obtained by doing its transpose. The total number of sentences is considered for the output based on the number of sentences in the input after pre-processing. If the number of sentences is greater than 30 then the maximum number of sentences in the output are 20% of the total number of sentences in the input transcript. Otherwise 30% is considered. A similarity matrix is obtained by multiplying dt matrix and inverse of it. The PageRank algorithm is feeded with similarity graph. This assigns rankings for each sentence based on their importance. Based on the rankings, the top n sentences will be given as output.

B. TextRank Algorithm

TextRank is a graph-based ranking algorithm that has been successfully used in citation analysis, similar to Google's PageRank algorithm. It can also be used for text processing, such as finding the most relevant sentences in a text and finding keywords. Keyword extraction, automatic text summarization, and phrase ranking are all common uses for text rank. To locate the most relevant sentences in text, a graph is created, with the vertices representing each phrase in the document and the edges linking sentences based on content overlap, i.e. computing the number of words shared by two sentences. The sentences are sent into the Pagerank algorithm, which finds the most important sentences, based on this network of sentences. Only the most important sentences can be extracted when creating a summary of the text. The TextRank algorithm creates a word network to locate relevant keywords. This network is built by examining which words are connected to one another. If two words appear frequently next to each other in the text, a link is created between them. If these two words appear more frequently next to each other in the text, the link is given more weight. The PageRank algorithm is applied to the generated network to determine the significance of each word. The top third of all of these terms is maintained and deemed significant. Following that, a keywords table is created by putting the relevant terms together if they exist in the text after one another. “Fig. 3” shows the TextRank architecture.

C. Term Frequency – Inverse Document Frequency (TF-IDF)

The TF-IDF technique was created for document search and retrieval. The TF-IDF is a statistical measure that assesses the relevance of a word to a document in a set of documents. It works by raising the number of times a word appears in a document proportionally. The raw count of times a word appears in a document is used to compute the term frequency of that word. How prevalent or unusual a word is throughout the complete document set determines the inverse document frequency of the word across a group of documents. The TF*IDF weight of a phrase is equal to the product of its TF and IDF ratings. As a result, the larger the weight, the more uncommon the term in a document.

D. Algorithm

Step 1: Remove time stamps from the document, split into sentences and match every sentence with the person
Step 2: Pre-process the sentences obtained
(text-standardization, stemming, stop words removal)
Step 4: Build term feature matrix using TD-IDF
Step 5: Build document similarity matrix by multiplying term feature matrix and its inverse
Step 6: Obtain similarity graph from similarity matrix
Step 7: Feed document similarity graph to TextRank
Step 8: Obtain ranks for each sentence and sort them
Step 9: Consider top sentences as an output summary

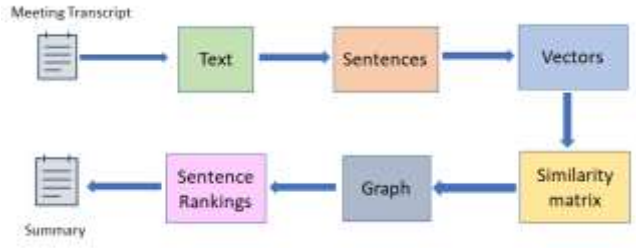

Fig. 3. TextRank Algorithm for summarization.

IV. IMPLEMENTATION

This section presents the output and results of the proposed system. A web application was developed using python flask app where user will be able to upload transcript document and when he/she hits the summarize button the summary will be displayed. The user will also be able to download the summary document. Two formats of input are considered which are .txt, .doc. The output document will be in .txt format.

A. Manual Evaluation

The suggested method's summaries have been manually tested and confirmed for substance and readability by a variety of users in the teaching and software industries. The goal was to see if the summaries were thought to be valuable in general for the purposes they were utilized for in the individual use cases. The evaluation was subjective and the method was informal. User (customer) feedback was depended on and complete control over their evaluation process and satisfaction levels was provided to them.
Answers for queries that are both generic and particular to the use cases are primarily required during user assessments. Some instances of the two types are given below.
Generic:

- Did the summaries assist users understand the transcripts content better than the original transcripts?
- Did the summaries cover any additional issues (themes) in addition to the main topic? If so, how many are there?
- How did our summaries compare to manual summaries, if any were available?

The method and results are improved by incorporating feedback from each user.
Other measures like recall and precision are calculated. The recall value obtained is around 0.42. This is compared with other models. The performance is quite good but other models like BERT, sequence to sequence models, transformer models can perform much better when implemented effectively.

Summary:

Mahesh:Well, I've been thinking of switching to an industry that has at least few decades of growth left.

Mahesh:I realize that, and I've been leaning toward digital marketing because in that industry I can carry over some of my skills from the current job.

Rohit: So are you thinking of making the transition in near future?

Mahesh: I'm 80-90 percent sure I'll go with digital marketing as the industry to reskill in, but in the next 2-3 weeks I'll take more opinions on other options, after all I wouldn't want to change the industry again.

Mahesh: And once I finalize the industry, I'll explore different options to reskill while keeping my current job.

Transcript:

00:01:04.220 --> 00:01:08.220

Rohit

Hi Mahesh, you look bit down. What's the matter?

00:01:09.260 --> 00:01:10.220

Mahesh

Nothing much.

00:01:10.220 --> 00:01:13.220

Rohit

Looks like something isn't right.

00:01:02.600 --> 00:01:09.770

Mahesh

Ya. It's at the job front. You know that the telecom industry is going through a rough patch because of falling prices and shrinking margins. These factors along with consolidation in the industry is threatening the stability of our jobs. And even if the job remains, career growth isn't exciting.

00:01:10.220 --> 00:01:13.220

Rohit

I know. I've been reading about some of these issues about your industry in the newspapers. So have you thought of any plan?

00:01:13.880 --> 00:01:24.260

Mahesh

I've been thinking about it for a while, but haven't concretized anything so far.

00:01:24.900 --> 00:01:39.300

Rohit

What have you been thinking, if you can share?

00:01:40.490 --> 00:01:41.550

Mahesh

Well, I've been thinking of switching to an industry that has at least few decades of growth left.

00:01:42.240 --> 00:01:50.600

Rohit

That's the right approach, but you need to reskill yourself for the industry you're targeting.

00:01:51.300 --> 00:02:02.450

Mahesh

I realize that, and I've been leaning toward digital marketing because in that industry I can carry over some of my skills from the current job. Another reason for this inclination is that digital marketing requires far less hardcore technical skills, which will make it relatively easier for me to acquire new skills.

00:02:04.430 --> 00:02:04.890

Rohit

Your choice makes sense. So are you thinking of making the transition in near future?

00:02:07.280 --> 00:02:07.520

Mahesh

Not immediately. I need to keep the job, as I've EMIs to pay. I'm 80-90 percent sure I'll go with digital marketing as the industry to reskill in, but in the next 2-3 weeks I'll take more opinions on other options, after all I wouldn't want to change the industry again. And once I finalize the industry, I'll explore different options to reskill while keeping my current job.

00:02:13.230 --> 00:02:14.020

Rohit

Sounds like a plan. If you need I can put you in touch with few friends who can help you finalize your future industry.

00:02:15.890 --> 00:02:16.380

Mahesh

That will be awesome. Thanks so much.

00:02:18.230 --> 00:02:19.770

Rohit

You're welcome.

V. CONCLUSION AND FUTURE WORK

The Meeting Summarizer collects all of the material from meetings into a brief and succinct summary. There are two types of summarization: abstractive and extractive. The TextRank, term frequency-inverse document frequency, and PageRank algorithms were used in this project to suggest an extractive summarization. As a result of the transcript preprocessing, better text is generated, which is then utilised to rank the sentences and provide a summary. As a result, the meeting summarizer relieves the management of the stress of physically writing the summary and reduces manual labour. The effects of the suggested system on abstractive summarization should be examined in future research. Multiple person dialogues will be considered in addition to abstractive summarization. Future study should include a comparison of our model to other research and calculating the correctness of the proposed system.

REFERENCES

[1] Yash Agrawal, Atul Thakre, Tejas Tapas, Ayush Kedia,YashTelkhade and Vasundhara Rathod, Comparative analysis of NLP models for Google Meet Transcript summarization, EasyChair Preprint no. 5404, 2021.

[2] Yuanfeng Song, Di Jiang ,Xuefang Zhao ,Xiaoling Huang, QianXu, Raymond Chi-Wing Wong, Qiang Yang, SmartMeeting: Automatic Meeting transcription and summarization for in-person conversations. ACM International Conference on Multimedia 2021, Pages 2777-2779.

[3] Pratik K. Biswas, AleksandrIakubovich, Extractive summarization of call transcripts, arXiv Preprint arXiv: 2103.10599, 19 March, 2021.

[4] Rani, Ujjwal and Karambir Bidhan, Comparative assessment of Extractive summarization: textrank, tf-idf and lda, Journal of Scientific Research 65.1 (2021): 304-311.

[5] Nenkova A., McKeown K. (2012) ,A survey of text summarization Techniques. In: Aggarwal C., Zhai C. (eds) Mining Text Data. Springer, Boston, MA.

[6] AravindChandramouli, SiddharthShukla, Neeti Nair, ShivenPurohit, ShubhamPandey and MuraliMohana Krishna Dandu, Unsupervised paradigm for information extraction from transcripts using BERT, arXiv Preprint arXiv:2110.00949,13 september 2021.

[7] Yao, L., Pengzhou, Z., & Chi, Z. Research on News keyword extraction Technology based on tf-idf and textrank, IEEE/ACIS 18th International conference on computer and information science (ICIS), June 2019.

[8] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre□training of deep bidirectional Transformers for language understanding, arXiv Preprint arXiv:1810.04805.

[9] Xingxing Zhang, MirellaLapata, Furu Wei, and Ming Zhou. Neural latent Extractive document summarization, conference on empirical methods in Natural Language Processing,2018.

[10] Mallick C., Das A.K., Dutta M., Das A.K., Sarkar A. (2019) Graph-based text summarization using modified textrank. In: Nayak J., Abraham A., Krishna B., Chandra Sekhar G., Das A. (eds) soft computing in Data Analytics. Advances in intelligent systems and Computing, vol 758. Springer, Singapore.

[11] Yue Dong, Andrei Romascanu, Jackie C. K. Cheung, HipoRank: Incorporating hierarchical and positional Information into Graph-based Unsupervised long document Extractive summarization, arXiv, 2020,volume:abs/2005.00513.

[12] Derek Miller,Leveraging BERT for Extractive text summarization on lectures, arXiv preprint, arXiv:1906.04165,7 july 2019.

[13] Wen Xiao and Giuseppe Carenini , Extractive summarization of long documents by combining global and local context, arXiv preprint, arXiv:1909.08089, 17 September 2019.

[14] Li Manling, Zhang Lingyu, Radke, Richar J, Ji Hend,Keep meeting summaries on topic: Abstractive multi-modal meeting summarization. 57th conference of the association for computational linguistics, 2021.

[15] Shashi Narayan, Shay B. Cohen, Mirella Lapata, Ranking sentences for Extractive summarization with reinforcement learning. arXiv preprint, arXiv:1802.08636,23 february 2018.