Review of Computer Engineering Research
2022 Vol. 9, No. 2, pp. 83-95
ISSN(e): 2410-9142
ISSN(p): 2412-4281
DOI: 10.18488/76.v9i2.3038

A GENERAL APPROACH FOR MEETING SUMMARIZATION: FROM SPEECH TO EXTRACTIVE SUMMARIZATION

Neslihan Akar[1+]
Metin Turan[2]

[1,2]*Graduate School of Natural and Applied Sciences, Istanbul Commerce University, Istanbul, Turkey.*
[1]*Email: nesliarslan2506@gmail.com Tel: +9005344077782*
[2]*Email: mturan@ticaret.edu.tr Tel: +9005327623554*

(+ Corresponding author)

Article History
Received: 4 April 2022
Revised: 20 May 2022
Accepted: 8 June 2022
Published: 24 June 2022

Keywords
Automatic extractive Summarization
Meeting summarization
Speech recognition
Speech summarization
Speech to text
TF-IDF.

ABSTRACT

Developing technologies and techniques have increased the amount of information and enabled easier access to information resources. However, due to the ever-growing amount of information sources, it has become difficult to access the information needed in a limited time. Consequently, the need for summary information has become important. This research is focused on the extraction of inferential written summaries of communications that occur in oral environments such as meetings, lectures and conferences. However, since this type of problem requires conversion from audio to text, it also includes issues such as the human factor, sound recording environments, and language-specific problems. This study aimed to take the audio recordings of the meetings, especially the IT sector, to process and summarize. Spontaneous conversations were converted into audio recordings and the obtained texts were summarized using extractive summarization techniques. The motivation of the study is to catch the important points that may escape the attention of the individuals at the meeting and to summarize the main agenda items for the personnel who could not attend the meeting. The experimentally generated dataset (converted from audio recordings to text) was summarized by three different human summarizers and compared with the summaries obtained from the developed inferential summative model. The results obtained are remarkable and it is seen that approximately 71% success was achieved.

Contribution/Originality: This research proposes a model to summarize a meeting using Natural Language Processing text summarization techniques (Extractive Summarization) supported by special dictionary usage. It can also be extended to other meetings as well as experimented on IT sector.

1. INTRODUCTION

Information is the resolution of uncertainty [1]. Humans want to remove or decrease uncertainty because uncertainty leads to misconception. Humanity has been struggling for information since its first existence. The transfer of information from generation to generation through oral or written tradition is a necessary phenomenon for the continuation of vitality, such as preserving the experiences of human beings and maintaining their lives.

Today, increasing communication resources have enabled faster and more effective dissemination of information. However, the increase in the speed of information dissemination, thanks to the increasing technological tools, has created a disadvantage after a point. Increasing resources cause information pollution and exposure to unnecessary information. To reach sufficient information when needed, it is necessary to understand what to look for and where.

Always scanning huge archives to capture the essence of information is a waste of time and energy. Summary information has always kept its importance in social life, education life and professional business life. With the increase in natural language processing techniques and tools, the importance of studies in this field has been increasing. Even summarizing written texts, oral sources and videos is possible with current technologies [2].

According to the nature of the professional business life, we have to come together with large or small groups at meetings. Even when working in small groups, we cannot prevent information from being aggregated and becoming a huge pile. Sometimes we are exposed to much more information than we need and we cannot even manage with small issues. We need to know what's at the core of the job and divide our work into small pieces. This situation is experienced commonly, especially in companies where meetings are held continuously and work is produced according to the decisions taken as a result of those meetings. Extended meeting hours and consecutive meetings actually consist of irregular sets of information, many of which are repetitive. A real meeting may contain complex and overlapped voices, interrupted or abandoned utterances [3]. The ability to automatically summarize meetings and to extract important (key) information can greatly increase the efficiency in many areas of business [4]. The motivation of the study is to take advantage of the developing technical possibilities and developing natural language processing algorithms in this field.

There are two approaches to summarizing a meeting. In the first approach, two-stage work was carried out. In the first stage, the sounds were collected and translated into writing. In the second stage, the obtained texts were summarized. Switching from speech to text caused many problems. This was because spontaneous speech is formless and is different from written text. Spontaneous speech often includes unnecessary information such as fluency, fillings, repetitions, corrections, and word fragments [5]. Any researcher using real meeting datum will largely have trouble transitioning from speech to text, and there will be losses that will cause loss of meaning because there are too many people in the meeting environment and there are many factors such as interference, talking at the same time, malfunctions caused by voice recorders and noise. According to the report published in the research with the ICSI corpus, 19 utterances other than 9 were observed to be missing due to interruption by another speaker, and high speaker overlap was observed in many regions [3].

Many text-based and speech-based features have been proposed in speech summarization systems to summarize different speech datum. Hori and Furui studied automatic speech summarization based on word significance and linguistic likelihood. Their aim was to summarize Japanese broadcast news. They focused on a method of finding important words using the number of characters relative to the target compression ratio [6]. For speech summarization, it is possible to benefit not only from the lexical features of words, but also from their acoustic features. This is because unlike written language, speech recordings contain intonation. It offers us the opportunity to extract the most important sections according to this intonation [7]. For example, Zhang et al. conducted a study comparing two types of speech, Mandarin Broadcast News and Cantonese Parliament Speech, using acoustic/prosodic, linguistic and structural features for speech summarization [8]. Having tried many approaches to improve query-based meeting summarization performance, Huebner et al. have expanded their approach to locate-then-summarize approach of QMSum [9]. They also investigated the effectiveness of pre-training the model with a large news summary dataset using HMNet [10].Finally, they validated their approach by comparing the performance of their model with BART, a state-of-the-art language model that is efficient for summarizing. They observed improved performance by using clustering methods to extract key information [4].

In the second approach, speech summary can be obtained from the direct speech. The techniques applied in speech-to-speech automatic summarization are different. Original spoken sentences are analyzed separately as words and filler units [5]. It is aimed to extract important information by processing the audio recordings taken instantaneously and to make them more meaningful by reducing the repetitive sentences and concepts to one in extended meetings. In a study, Baykal et al. benefited from the repetitive features of sounds. Repetitive structures like the chorus in musical audio can be observed in meetings. Accurate detection of repetitions can improve performance

in automatic speech summarization. In the presented method, it transforms the 1-D time-domain speech signal into a 2-D image representation, a (dis)similarity matrix and detects possible repetitions in the matrix using appropriate computer vision techniques. According to the results obtained, speech signals catch the key-concept (or topic) and computational analysis performed well. Also, the method does not convert the speech signal into words, phrases or sentences. Therefore, it can be generalized as a speech-to-speech summarization method, in which summarizing results are presented by speech rather than text. Also, it does not need any prior knowledge of language and grammar [11]. In addition, it is possible to benefit not only from the spoken, but also from the video features. For example, Li et al. developed an abstractive meeting summarizer from both video and audios of meeting records. They proposed a multi-modal hierarchical attention mechanism across three levels: topic segment, utterance and word. They took advantage of speaker interaction and participant feedback to discover salient utterances [12].

In addition to all these academic studies, commercially developed automated tools are available. Thanks to the developed automatic tools, meeting datum can be recorded and stored. They help to increase, the productivity of both meeting attendees and non-participants. Moreover, they allow evaluation of all meeting content by an independent third eye [13, 14]. One of them is the meeting recognition and learning assistant, called Calo, which was developed for this purpose. The audio stream from each meeting participant is transcribed to text using two separate recognition systems. The first recognition is achieved when creating live copies with a real-time recognizer. The other recognition is applied when the meeting ends, another transcript is generated, it is a second offline recognition system. The results are obtained by using the prosodic and contextual information of the obtained texts [13]. The other example is MeetingVis, which summarizes not only what is spoken, but also the materials (presentation slides) used, has added as another dimension to this issue. MeetingVis introduced a visual narrative approach to summarization [14].

The purpose of the meeting summary is to cover the important points and to create the main agenda items for personnel that cannot attend the meeting. Also reviewing the meeting notes before the next meeting for successive meetings can provide quick recall for better planning and preparation. In general studies, repetitions and emphases have been given importance and extractive summary algorithms have been developed. On the other hand focusing only on repetitive information is not the right approach. As Shannon mentioned in the theory of information, less seen information may be more important [1]. Many experiments were conducted by changing the weights of the words that occur less frequently to try to improve the results. Since this study aims to summarize the meetings in the IT sector, the "A Computer Science Academic Vocabulary List" dictionary [15] that was obtained by Roesler as a result of their thesis study was used. Experimentally generated meeting texts dataset (converted from audio recordings to text) were summarized by three different human summarizers at the rates of 40% and 20%. The experiments showed that program extracted summaries have a sentence similarity success rate of approximately 71% compared to human summaries.

The rest of this paper is organized as follows: Section 2 includes the details of the methodology, in two parts, the former is speech to text conversion and the latter is text to summarization. Section 3 describes the dataset features and preparation steps. Section 4 sets out the results and evaluations. Finally, section 5 has discussions (drawbacks) and future works of the research.

2. METHODOLOGY

Figure 1 clearly shows the work steps through the proposed solution. All meeting audios files were first applied to speech to text operation. Later, the texts obtained were summarized as 40% and 20% of the whole text. Each meeting text was also summarized in the same proportions by 3 different human summarizers. Finally, human and machine extractive summaries were compared with selection of similar sentences in the original text files. The ratio of similar sentences over total sentences in the extractive summary gives the success of the summarization in terms of one human summarizer. The average of ratios with all summarizers determines the success of our extractive summarizer.

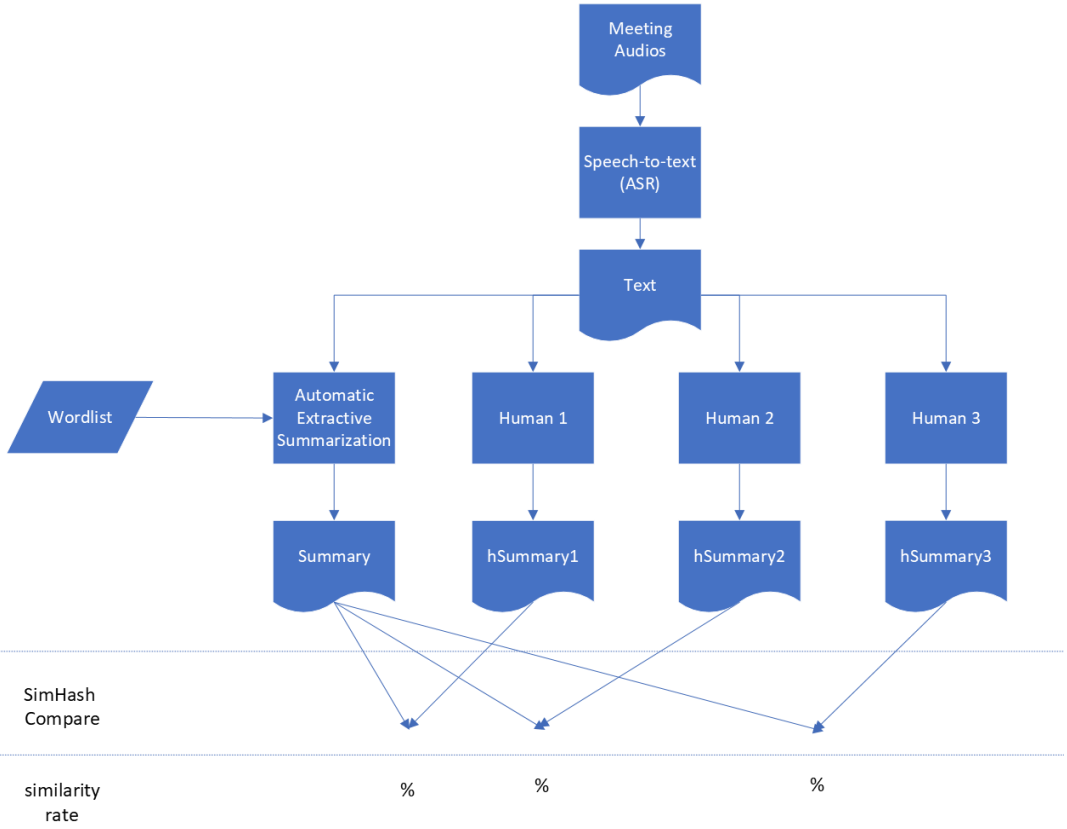


Figure 1 Summarization process map.

A. Speech to Text

It is a methodology that provides the textual output of speech given as input to a computer program. Depending on the number of speakers and the style of speech, there are various forms of speech. They include:

- Broadcast news.
- Lecture.
- Public speaking.
- Interview.
- Telephone conversation.
- Meeting.

This research focuses on the conversations in the meeting in which there will be the voices of two or more people in the audio recording. The frequency and intonation of each of these sounds will be different. Also, spontaneous speech includes unnecessary information such as gaps, repetitions, corrections, and word fragments.

There are many tools available for speech to text translation. In addition to commercial tools, for example Python offers us a free library that's called Speech Recognition. We could obtain limited text by using the features of this library. Processing an audio recording of mutual conversations caused data loss. Moreover low-frequency sounds were recognized as noise when translated into text. We used Google's cloud Application Programming Interface (API) as the results from the Python library were not sufficient. Google's cloud API seems the most advanced infrastructure, because it provides lots of flexibility. It offers the ability to accurately convert speech to text using the API powered by Google's AI technologies. The infrastructure using the latest technology applies the most advanced deep learning neural network algorithm. It enables testing, creation and management of custom resources. It can be deployed either in the cloud with the API or on-premises anywhere. The API activated with Python coding provided the conversion of all the texts we wanted to put into writing.

B. Text to Summarization

Speech summarization is the process of obtaining information that will usefully present key points to the end users in the shortest possible way. For speech summarization, speech recognition technique is used and natural language processing algorithms are applied to summarize the texts obtained from the recognition system [16]. There are two main summarization techniques: extractive and abstractive [17]. The extractive summarization technique is based on the selection of best original parts of text from within summarization size. The sentences are directly extracted into the summary. Whereas in abstractive summarization, the summarization is regenerated using all information in the text [18]. In the proposed model, TF-IDF algorithm is used to select the best original parts of the text (extractive summarization).

TF-IDF is the product of two statistics, term frequency and inverse document frequency. Term Frequency (TF) – Inverse Document Frequency (IDF) is a technique for measuring words in a set of documents. Its aim is define the importance of a keyword or phrase within a document [19].

The text summarization process steps applied are showed in Figure 2. The details of summarization using TF-IDF algorithm are explained below stepwise (a) through (f).

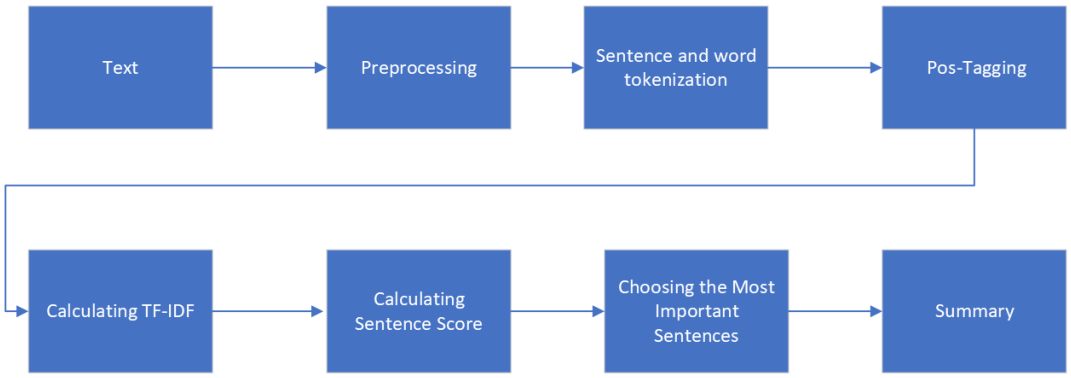


Figure 2. Summarization process steps.

a) Preprocessing

Preprocessing is the step for text clearance and optimization. It is the first step of NLP projects. In this step, distribution of data is analyzed to decide which techniques are needed, and how deep cleaning should be done. The preprocess stages depend on the text content and problem definition. The preprocessing steps used in this study were as follows:

- Remove punctuation characters.
- Lowercase conversion.
- Remove numerals.
- Spelling corrections.
- Singularization.
- Converting all words into base form.
- Removal of stop-words [20].

b) Tokenization

Tokenization is dividing large blocks of text into smaller units, generally called tokens. For example, sentences in a paragraph and words (sometimes called terms) in sentences are extracted by tokenization. Weights are assigned to tokens. These weights can depend on many parameters such as frequency, uniqueness, etc. In this study, the sentence tokenization approach was used. At the end of this step a matrix containing all the identified tokens with their weights was created. This matrix was used to calculate sentence importance.

c) Pos-Tagging

POS Tagging (Parts of Speech Tagging) is the process of mark-up words in text form for a particular part of a speech according to its definition and context. Each word is tagged according to its type grammatically [21]. NN - noun (singular), NNS – noun (plural), NNP – proper noun (singular), VB – verb (base form), VBD – verb (past tense), VBG – verb (gerund / present participle), VBN – verb (past participle), VBP – verb (sing. present, non-3d) and VBZ – verb (3rd person sing. present). Tags were only evaluated in the model as due to the nature of the problem, nouns and verbs were considered to be the valuable. The other words were not included.

d) Calculating TF-IDF

The symbols used in the formulae below are that lowercase d represents a document in the D document set, where t is selected word (noun or verb) in document d. The words listed in decreased frequency order and only a percentage of top frequent words (20% and 50% word ratios were the best) were selected for further evaluation. This is a technique to eliminate garbage words when you need to produce list of words for your model.

- All tagged words' frequency values were calculated using Equation 1. " t' " indicates one of the words included in similar document.

$$tf(t,d) = \frac{f_{t,d}}{\sum_{t' \in d} f_{t',d}} \quad (1)$$

- Also the Inverse Document Frequency (IDF) was calculated for each word in the all text using Equation 2.

$$idf(t,D) = \log \frac{N}{|\{d \in D : t \in d\}|} \quad (2)$$

- The TF-IDF score of each word was calculated by multiplying the word's TF score by its IDF score using Equation 3.

$$tfidf(t,d,D) = tf(t,d) \cdot idf(t,D) \quad (3)$$

e) Scoring the Sentences

The score of each sentence was calculated using Equation 4, where the TF-IDF values of the words belonging to this sentence is the summed up.

$$Sentence\ Score = \left(\sum tfidf(t,d,D)\right) \quad (4)$$

The context of the problem is important when evaluating the words effect in the score evaluation. The special dictionary includes the specific words in the context is the one of the techniques used in such problems to better evaluate scores or, in other words, normalizing. The terms that occurs in the context dictionary [15] were used with a learning alpha coefficient as an additional weight for sentence score. As a result, the ranking of sentences in importance worked better than the first classical approach. The final form of the formula for scoring sentences is given in Equation 5.

$$Sentence\ Score = \left(\sum tfidf(t,d,D)\right) + \alpha * \sum_{for\ all\ t \in d} \begin{Bmatrix} 1\ if\ t \exists\ dictionary \\ 0\ if\ t \nexists\ dictionary \end{Bmatrix} \quad (5)$$

All the processing steps are explained by real data. For example, "I am testing the current project." sentence was processed in meeting 8.

1. First of all preprocessing was applied step by step.
2. Next tokenization was applied. Tokens were obtained as a list of words "test", "current" and "project".
3. Pos Tagging was also applied.

Table 1 Presents the Sample sentence preprocessing steps.

Table 1. Sample sentence preprocessing steps.

Preprocessing Step	Output
Remove punctuation characters:	I am testing the current project
Lowercase conversion:	I am testing the current project
Remove numerals:	No
Spelling corrections:	No
Singularization:	No
Converting all words into base form:	I am test the current project
Stop-words removal:	test current project

Table 2 Presents pos tagging in sample sentence.

Table 2. Pos tagging in sample sentence.

Word	Pos Tagging
"test"	VB – verb (base form)
"current"	not tagged, because it is not a verb or a noun
"project"	NN - noun (singular)

4. Evaluation requires counts, so that $tfidf$ values were calculated for remaining words.

The $tfidf$ value of "test" is 0.065

The $tfidf$ value of "project" is 0.12

5 Finally, sentence scoring was obtained using the formula given in (5).

The sentence score without dictionary= (0.065 + 0.12) + 0= 0.185

The sentence score with dictionary usage= 0 + {(10.2 x 0.065) + (10.2 x 0.12) = 1.887

Table 3. Results of sentence score with/out dictionary.

Approaches	Test	Project	Sentence Score
Standard Tf-Idf approach	0.065	0.12	0.185
Added Dictionary	10.2 x 0.065	10.2 x 0.12	1.887

Table 3 Calculates the results of sentence score with and without dictionary.

The alpha coefficient was chosen as 10.2, it presented the best optimized model among others using brute force algorithm for maximizing the expectation. It did not work for smaller alpha values. The multiplier alpha 10 can be explained by shifting the sentence score to the left at least one digit as dictionary words effect. It was observed that the most optimal candidate was 10.2. Although the effect of the proposed model is shared in the Results & Evaluation section, it is a pleasure to share the increase in similarity from 0.48 to 0.75 with the human summarizers for the dictionary model for 20% extractive summary.

A 3.5 million-word corpus of academic computer science textbooks and journal articles were used by Roesler to produce the dictionary. In this dissertation, help was taken from the Roesler's dictionary called "A Computer Science Academic Vocabulary List" [15]. This dictionary has words such as; system, data, algorithm, model, case, program, information, code, function, process, application, software, etc. commonly used in computer engineering, development and design.

f) Generating the Summary

Finally, the sentences were put in order using the calculated sentence scores in increasing order. The extractive summary is obtained putting sentences into the summary following the order until the summary size is reached.

3. DATA

One of the most difficult aspects of working on this research was the difficulty of accessing the real dataset. For this purpose, 10 meeting texts on real issues were prepared. Since these meetings were experimental, they had a length of approximately 30-90 sentences. The number of sentences and audio recording length for each meeting are given below in the Table 4.

Table 4. Information of prepared meetings.

#Meeting	#Sentences	#Min
Meeting 1	58	3' 33"
Meeting 2	57	3' 19"
Meeting 3	43	2' 55"
Meeting 4	53	2' 58"
Meeting 5	68	4' 50"
Meeting 6	50	3' 04"
Meeting 7	81	4' 16"
Meeting 8	53	3' 07"
Meeting 9	59	4' 06"
Meeting 10	37	3' 03"

Speech to text process contains problems caused by speaker's accent. The language support of automatic speech recognition tools is inadequate for non-native speakers. The output of these tools does not produce satisfactory text outputs. Sometimes it uses homonymous word, a very similar word or the closest word to the original. Moreover, wrong conversion occurs due to misunderstanding the speaker's accent. The voice of English native speaker via on-line tool called Freetts was used to produce video recordings to produce consistent data.

A special style for naming the summary files was organized. The file name starting with "hSum" is used for human summarizers and the file name starting with "sum" is used for the extractive summarizer which is followed by the meeting number. The next notation "L" means summary size which is followed by its percentage. The next symbol "W" means word ratio which is followed by its percentage.

For example, meeting 10 was summarized using four different parameters by the extractive summarizer given in Table 5. All of them were produced as 40% summaries. The first only uses TF-IDF algorithm. The second is the dictionary form that doesn't use a "word ratio". The third one used the dictionary form with 50% "word ratio". The last one used the dictionary form with 20% "word ratio".

Table 5. Sample data.

#37 total sentences	hSum310-L40%
Sum10-L40%-tf-idf	0.53
Sum10-L40%	0.64
Sum10-L40%-W50%	0.65
Sum10-L40%-W20%	0.73

The summaries produced by humans were compared in terms of similarity. Figure 3 gives the comparison of 40% summaries, whereas Figure 4 presents the comparison of 20% summaries. Similarity charts show that the data is not contradictory and distribution is consistent. This average of similarities was accepted as a comparison benchmark for the success of model summaries.

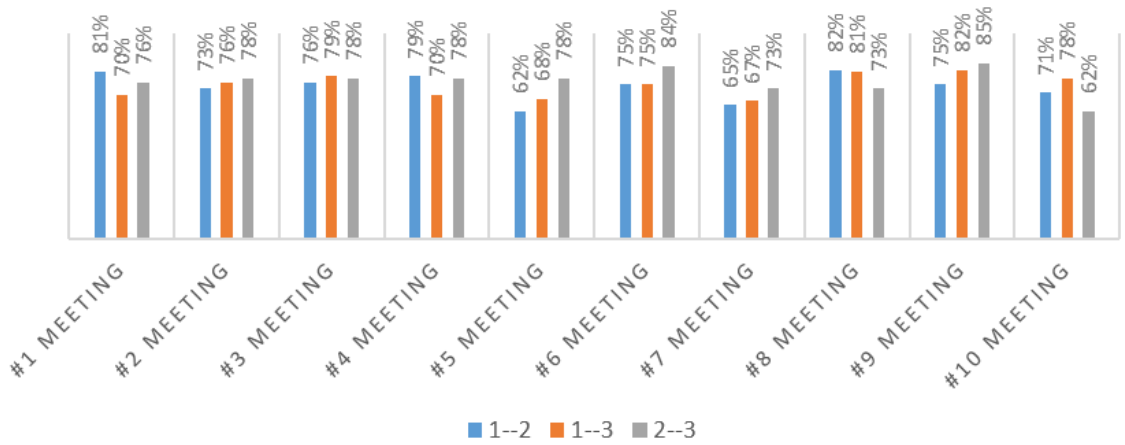


Figure 3. Similarity rates of human summaries of 40% length.

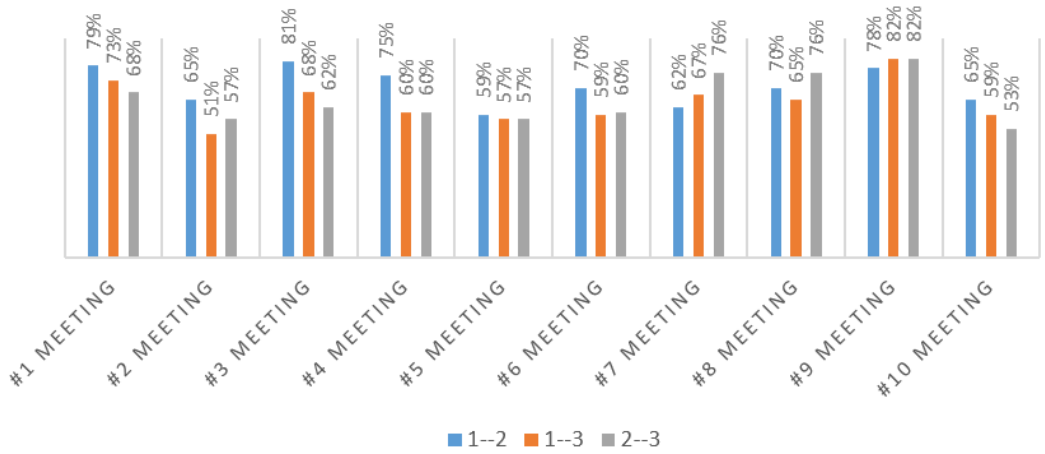


Figure 4. Similarity rates of human summaries of 20% length.

As a result, similarity rates of human summaries ranged between 60% and 86%, 75% on average for 40% summaries. Additionally, similarity rates of human summaries ranged between 50% and 83%, 67% on average for 20% summaries. The blue line represents the similarity ratio of the 1st and 2nd human summaries; the orange line represents the similarity ratio of the 1st and 3rd human summaries; the gray line represents the similarity ratio of the 2nd and 3rd human summaries.

4. RESULTS & EVALUATIONS

When evaluated the results obtained across the whole experimental studies, it was observed that the success rate was at the expected level in the text summaries of 40% and 20% in length. It is clearly seen in Figure 6 and Figure 8 that we achieve the most optimal results when 50% of the word list obtained from the TF-IDF approach is used.

Approximately 71% success was achieved. The success rate increased when the average of all comparison results were calculated. The optimum results of the study are given in Figure 5 and Figure 7.

Figure 5 shows the similarity between human and extractive summaries for the 40% length of summaries. The blue line point represents only using TF-IDF algorithm, the orange line represents using a dictionary form that doesn't use a "word ratio", the gray line represents adding a dictionary with 50% "word ratio" and the yellow line represents using a dictionary form with 20% "word ratio".

	mean1	mean2	mean3	mean4	mean5	mean6	mean7	mean8	mean9	mean10
sumX-L40%-tf-idf	0.74	0.64	0.71	0.69	0.58	0.70	0.66	0.63	0.72	0.57
sumX-L40%	0.74	0.69	0.68	0.64	0.57	0.69	0.57	0.74	0.71	0.63
sumX-L40%-W50%	0.69	0.69	0.62	0.69	0.67	0.69	0.63	0.78	0.75	0.68
sumX-L40%-W20%	0.74	0.70	0.59	0.70	0.67	0.70	0.70	0.72	0.75	0.75

Figure 5. Similarity chart human & extractive summaries of 40% length.

To show the increase more clearly, when the arithmetic mean of all averages of the results is plotted, the graph was like this in Figure 6.

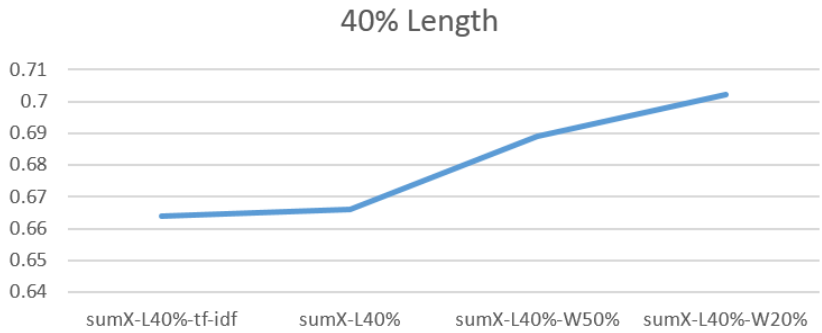
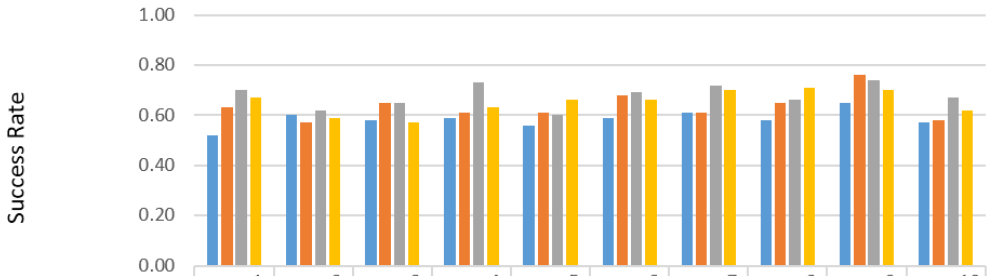


Figure 6. Mean of all averages of the similarity ratios 40% length.

The graph for the length of 20% was as follows in Figure 7.



	mean1	mean2	mean3	mean4	mean5	mean6	mean7	mean8	mean9	mean10
sumX-L20%-tf-idf	0.52	0.60	0.58	0.59	0.56	0.59	0.61	0.58	0.65	0.57
sumX-L20%	0.63	0.57	0.65	0.61	0.61	0.68	0.61	0.65	0.76	0.58
sumX-L20%-W50%	0.70	0.62	0.65	0.73	0.60	0.69	0.72	0.66	0.74	0.67
sumX-L20%-W20%	0.67	0.59	0.57	0.63	0.66	0.66	0.70	0.71	0.70	0.62

Figure 7. Similarity chart human & extractive summaries of 20% length.

The arithmetic mean of all averages of the results that has 20% length of text was as in Figure 8.

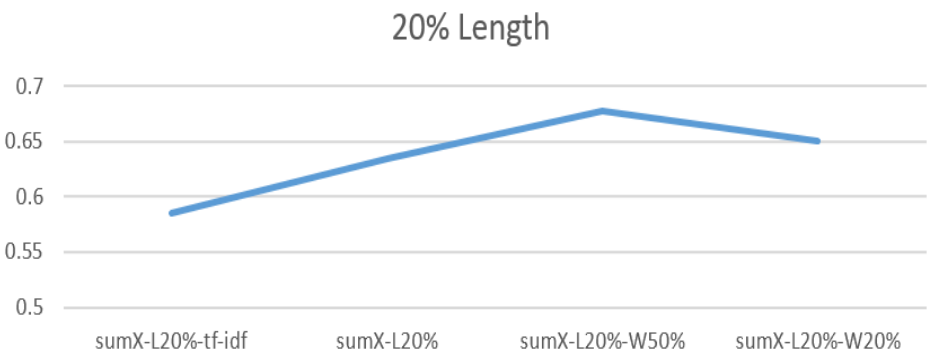


Figure 8. Mean of all averages of the similarity ratios 20% length.

When the Figure 6 and Figure 8 checked carefully, the increase in the success rate including the dictionary is notable. In Figure 5 and Figure 7 blue lines represent success rate without dictionary, where orange lines present success rate with dictionary. It is observed that if the selective words were determined better (the best ratio seems to be 50%), sentences these are more suitable for the summary could be selected.

5. CONCLUSION & FUTURE WORK

As the evaluation criteria, extractive summaries produced in the length of 40% and 20% of the texts by three different human summarizers were used. Similarity rates are given in Figure 3 and Figure 4. After extractive summarize and human summarize comparison were interpreted by an average of all similarity rates. The increase is in success rate clearly seen trending upward in Figure 6 and Figure 8.

The graphs are in the ascending direction as using a dictionary causes more appropriate sentence selection. Additionally, by using word ratios of 20% and 50% generated summaries that were similar to human summaries. While calculating the importance score of the sentence, the frequency values of the words are summed. Therefore, long sentences are more advantageous in the standard approach. It's not the length of the sentence that matters, but the effectiveness of the dictionary approach. More appropriate sentences were extracted using keywords. Approximately 71% similarity was achieved on average. If this is compared with the average similarity of summaries produced by summarizers, it can be seen that the proposed model produced almost the same similarity (71%) to a human summarizer. This result encourages us to do further work to achieve a result that sis better than a human being. Furthermore, future work may also include all the colloquial rules in the model that help improve speech summarization performance. This model can be used in many situations such as summarizing long lectures, conferences and meetings as it automatically converts speech to text and summarizes it. It can help both employees and students take notes in situations such as training, seminars and symposiums. Multi-language support can be developed (especially non-native speakers), and it can be used together with other languages other than English. It would be useful to provide not only English but also other language support. Also the information of who belongs to the summary sentence can be kept by making a personalized label using speech detection algorithms. This study was done in English, but its development in Turkish and its support with artificial intelligence algorithms will be a very good contribution to our language and will be a work to inspire new generations of researchers.

Funding: This study received no specific financial support.
Competing Interests: The authors declare that they have no competing interests.
Authors' Contributions: Both authors contributed equally to the conception and design of the study.

REFERENCES

[1] C. E. Shannon, *An algebra for theoretical genetics, Ph.D. dissertation, dept. of mathematics.* Cambridge, MA: Massachusetts Institute of Technology, 1940.

[2] M. Elfeki, L. Wang, and A. Borji, "Multi-stream dynamic video summarization," in *2022 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, 2022, pp. 185-195.

[3] E. Shriberg, R. Dhillon, S. Bhagat, J. Ang, and H. Carvey, "The ICSI meeting recorder dialog act (MRDA) corpus," in *Proceedings of the SIGDIAL 2004 Workshop - 5th Annual Meeting of the Special Interest Group on Discourse and Dialogue (ACL)*, 2004, pp. 97–100.

[4] A. Huebner, W. Ji, and X. Xiao, "Meeting summarization with pre-training and clustering Methods. Retrieved from: http://arxiv.org/abs/2111.08210," 2021.

[5] S. Furui, T. Kikuchi, Y. Shinnaka, and C. Hori, "Speech-to-text and speech-to-speech summarization of spontaneous speech," *IEEE Transactions on Speech and Audio Processing*, vol. 12, pp. 401-408, 2004.Available at: https://doi.org/10.1109/tsa.2004.828699.

[6] C. Hori and S. Furui, "Automatic speech summarization based on word significance and linguistic likelihood," in *2000 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings (Cat. No. 00CH37100)*, 2000, pp. 1579-1582.

[7] J. Zhang and H. Yuan, "Speech summarization without lexical features for mandarin presentation speech," in *Proceedings - 2013 International Conference on Asian Language Processing, IALP 2013*, 2013, pp. 147–150.

[8] J. Zhang. and H. Yuan, "A comparative study on extractive speech summarization of broadcast news and parliamentary meeting speech," in *Proceedings of the International Conference on Asian Language Processing 2014, IALP 2014 (Institute of Electrical and Electronics Engineers Inc., 2014)*, 2014, pp. 111–114.

[9] A. Awadallah, H. Celikyilmaz, A. Jha, X. R. Qiu, Y. Liu, M. Mutuma, D. Radev, and D. e. a. Yin, "Qmsum. Retrieved from: https://github.com/Yale-LILY/QMSum/tree/main/data/ALL," 2021.

[10] C. Zhu, R. Xu, M. Zeng, and X. Huang, "A hierarchical network for abstractive meeting summarization with cross-domain pretraining," in *Findings of the Association for Computational Linguistics Findings of ACL: EMNLP 2020 194–203 (Association for Computational Linguistics (ACL)*, 2020.

[11] M. Sert, B. Baykal, and A. Yazici, "A Combining structural analysis and computer vision techniques for automatic speech summarization," in *Proceedings - 10th IEEE International Symposium on Multimedia, ISM 2008*, 2008, pp. 515–520.

[12] M. Li, L. Zhang, H. Ji, and R. J. Radke, "Keep meeting summaries on topic: Abstractive multi-modal meeting summarizationi," in *ACL 2019 - 57th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference (Association for Computational Linguistics (ACL)*, 2020, pp. 2190–2196.

[13] G. e. a. Tur, "The CALO meeting speech recognition and understanding system," in *2008 IEEE Spoken Language Technology Workshop*, 2008, pp. 69-72.

[14] Y. Shi, C. Bryan, S. Bhamidipati, Y. Zhao, Y. Zhang, and K.-L. Ma, "Meetingvis: Visual narratives to assist in recalling meeting context and content," *IEEE Transactions on Visualization and Computer Graphics*, vol. 24, pp. 1918-1929, 2018.Available at: https://doi.org/10.1109/tvcg.2018.2816203.

[15] D. Roesler, "A Computer Science Academic Vocabulary List," Dissertations and Theses, Dep. Applied Linguistics, Portland State Univ., Paper 5540, 2020.

[16] A. NithyaKalyani and S. Jothilakshmi, "Intelligent speech signal processing, editor(s): Nilanjan Dey," ed: Academic Press, 2019, pp. 113-138.

[17] R. Ferreira, L. de Souza Cabral, R. D. Lins, G. P. e Silva, F. Freitas, G. D. Cavalcanti, R. Lima, S. J. Simske, and L. Favaro, "Assessing sentence scoring techniques for extractive text summarization," *Expert Systems with Applications*, vol. 40, pp. 5755-5764, 2013.Available at: https://doi.org/10.1016/j.eswa.2013.04.023.

[18] K. U. Manjari, S. Rousha, D. Sumanth, and D. J. Sirisha, "Extractive text summarization from web pages using selenium and TF-IDF algorithm," in *2020 4th International Conference on Trends in Electronics and Informatics (ICOEI)(48184)*, 2020, pp. 648-652.

[19] A. Rajaraman and J. D. Ullman, "Data mining (PDF). Mining of massive datasets," ed: Cambridge University Press, 2011, pp. 1–17.

[20] J. P. Verma and A. Patel, "Evaluation of unsupervised learning based extractive text summarization technique for large scale review and feedback data," *Indian Journal of Science and Technology*, vol. 10, pp. 1-6, 2017.Available at: https://doi.org/10.17485/ijst/2017/v10i17/106493.

[21] T. Gungor, "Part-of-speech tagging. In ¨ Handbook of Natural Language Processing," 2nd ed: Chapman & Hall/CRC, 2010, pp. 205–235.