A Study on Several Text Summarization Approaches

Abstract. Text summarization refers to the process of condensing lengthy texts into concise and coherent summaries, focusing on capturing the document's main points. Automatic text summarization represents a critical challenge within the domains of machine learning and natural language processing (NLP). Given the vast volume of digital data available today, there is a growing demand for machine-learning algorithms capable of automatically condensing extensive texts into accurate and comprehensible summaries that effectively convey the intended message. Machine learning models are typically trained to comprehend documents, extracting essential information to produce the desired summarized output. Moreover, the application of text summarization offers several advantages, including the reduction of reading time, accelerated information retrieval, and the ability to store more information efficiently. In the realm of NLP, two primary methods for text summarization exist: extractive and abstractive. The extractive approach involves identifying key phrases within the source document and assembling them to form a summary without altering the text's content. On the other hand, the abstractive technique entails paraphrasing and condensing sections of the source document. In deep learning applications, abstractive summarization can overcome issues related to grammar inconsistencies often encountered in the extractive method. In our analysis, we have examined various existing methods for text summarization, including unsupervised, supervised, semantic, and structure-based approaches. We have critically assessed their potential and limitations. Our findings highlight specific challenges, such as the anaphora and cataphora problems, interpretability issues, and readability concerns for long texts. To address these challenges, we propose solutions aimed at improving the quality of the dataset by addressing outliers through the integration of corrected values obtained from human-generated inputs. Text summarization stands as an intriguing and evolving field within machine learning, gaining increasing attention. As research in this domain progresses, we anticipate the emergence of innovative breakthroughs that will contribute to the seamless and accurate summarization of lengthy textual documents.

Keywords: Summarization · Automatic summary · Abstractive summarization · Extractive summarization · Deep learning · Unsupervised summarization · Anaphora · Cataphora · Semantic summarization · Structure-based summarization

1 Introduction

Summarization is the process of separating the key bits from a larger piece of material while retaining the core ideas and concepts. It is required in a variety of settings. In the world of journalism, news pieces are frequently abbreviated to highlight the most important aspects of an event, allowing readers to keep informed despite their hectic schedules. It allows researchers to quickly comprehend the important findings and techniques of relevant studies, allowing for a more efficient examination of the current literature. Long business reports are synthesized into simple summaries in the corporate environment for executives who need a quick overview before making critical choices. Summarization algorithms are used by social media platforms to give users condensed updates that capture the substance of discussions.

Oftentimes the algorithms used by summarization models are not sufficient to find the optimal result the user is looking for. The main reason might be the usage of a single template for summarization that often fails to capture the detail and distinctness of the document. The user may have to input particular details or summarize the paper in segments to get the relevant useful information he is looking for. Other problems include a lack of model accuracy. It may be necessary to spend a lot of time comprehending complex topics before even attempting to condense them because creating a summary demands a comprehensive mastery of the original material.

All these factors combined motivated us to ease the research work of our fellow researchers and make an all-inclusive comprehensive document on summarization.

The overall structure of the paper is mentioned below. The history of tachygraphy is written in Section 2. Thereafter, we have shown different types of summarization methods in Section 3. Challenges, important features, and a brief discussion have been described in Sections 4, 5, and 6. Finally, Section 7 contains the future scopes and closing remarks.

2 Background Study

The subsequent section delves into various approaches and techniques employed in text summarization, culminating in a comprehensive overview of the document. We conducted an exhaustive examination of the current landscape of summarization scopes and methodologies. This was undertaken with the aim of gaining insight into the complexities associated with existing methods and identifying the factors contributing to the suboptimal performance of certain approaches.

Upon conducting a survey analysis of the existing methods of summarization across various research papers it was found that a lot of summarization models are still in use. The oldest two methods of summarization are Extractive and Abstractive summarization. Extractive summarization in its essence uses a ranking algorithm that produces the most occurring words from the content exactly as they appear [5]. How they work is that they calculate the frequency of each word in the document and rank them in descending order, taking only the first 'n' into consideration, for the sake of simplicity let us assume 'n' is ten here. Then they rank the sentences based on how many such important words the sentence contains. If a sentence has all 10 of these words, which generally is not the case, but let us assume an ideal situation, then that sentence will be of the highest priority. Let us assume the summarized document will have 'm' such sentences, here 'm' is selected based on how much we want to compress the document. Then the sentences are shown as they appear. Abstractive summarization initially follows the same principle but the place where it differs is that instead of exactly displaying the sentences it tries to formulate new sentences by incorporating the meaning it understood [4].

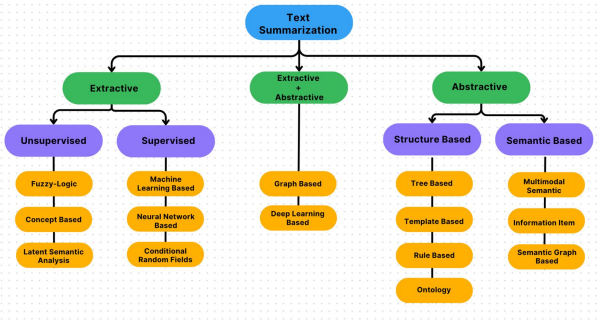

Fig. 1: Conventional methods

Now let us dive into some of the technicalities of each of these methods. In 1969, an automatic text summarization system was developed that went beyond the conventional keyword-based approach, which relied on frequency-dependent weights [7]. This pioneering system incorporated three additional methods to calculate the weights of sentences:

1. Cueing Method - This is predicated on the supposition that the existence or when specific cue words are absent in the dictionary determines the importance of a phrase.
2. Tile Method - This method calculates sentence weight based on all content words found in the text's title and subheadings.
3. The Location Method is founded on the premise that sentences appearing at the beginning of both the text and a paragraph are more likely to carry significance.

The Trainable Document Summarizer of 1995 was designed to perform sentence extraction tasks using a variety of weighting heuristics [7]. These included the Paragraph Feature,Fixed-Phrase Feature,Sentence Length Cut-Off Feature, Uppercase Word Feature and Thematic Word Feature,.

3 Summarization Methods

3.1 Extractive Summarization:

1. **Term Frequency-Inverse Document Frequency (TF-IDF) method:** Using the conventional weighted term-frequency and inverse sentence-frequency approach, where sentence-frequency counts the number of sentences containing a specific term, a sentence-level bag-of-words model is constructed. Then, only the phrases with the highest ratings are chosen to be included in the summary. These sentence vectors are assessed based on their similarity to the query, effectively applying the Information Retrieval paradigm directly to the summarization process [6].
   Summarization is typically tailored to specific queries, but it can be adapted to a more generic form. Frequently occurring non-stop words within the document(s) can serve as query words to generate a generic summary. These phrases yield general summaries as they encapsulate the essence of the text. In the context of sentence term frequency, it often remains at 0 or 1, as a single content word doesn't usually appear frequently within one sentence. The transition from query-based summary generation to generic summarization occurs when users formulate query words similarly to how they do in information retrieval.
2. **Cluster Method:** Documents are typically structured to cover various themes in an organized sequence. Consequently, it's logical to expect that summaries would encompass these diverse "topics" present in the papers. Some summarization methods take into account this factor by utilizing clustering techniques. Term frequency-inverse document frequency (TF-IDF) scores are employed to represent words within documents [17]. Within the context of these clusters, the term frequency represents the average number of occurrences per document. IDF values are calculated using the entire corpus. The summarization process takes clustered documents as input, with each cluster representing a distinct topic. To illustrate each topic, the words with the highest term frequency-inverse document frequency (TF-IDF) scores within that cluster are selected.
   Sentence selection is based on several criteria. Firstly, sentences are chosen based on their similarity to the central idea of cluster Ci. Secondly, the position of the sentence within the document denoted as Li, is considered. Lastly, a sentence's similarity to the first sentence in the document referred to as Fi, is taken into account. A sentence's overall score, denoted as Si, is calculated as the weighted average of these three variables:
   Si = (W1*Ci) + (W2*Fi) + (W3*LiSi) = (W1*Ci) + (W2*Fi) + (W3*Li)
3. **Graph Based Approach:** In the context of the graph-theoretic approach, sentences within documents are transformed into nodes within an undirected network, following standard preprocessing techniques. Each sentence is represented as a unique node. When two sentences share specific common terms or their similarity (e.g., cosine similarity) surpasses a predefined threshold, they are connected by an edge in the network. Selected sentences from the relevant sub-graph are used exclusively for query-specific summaries. The sub-graphs can be selected for general summaries, in contrast.
   Another important outcome of graph-theoretic analysis is the identification of key sentences within a document. In this context, significant sentences within the partition are nodes with a high degree of connectivity (i.e., a large number of edges connecting to that node). These highly connected nodes are accorded greater priority for inclusion in the summary [1].
4. **Latent Semantic Analysis (LSA) Method:** The singular value decomposition (SVD) is an exceptionally efficient mathematical technique for identifying the principal orthogonal dimensions within multidimensional data. Even if the documents do not include the same exact phrases, SVD is particularly good at grouping them together when they are semantically similar. When words appear in the same singular vectors, they are connected together because they frequently occur in comparable situations. As a result, topic words and content sentences can be extracted from documents using SVD, contributing to a deeper understanding of the data's structure and semantics.

Coming at our second approach we have a slightly advanced approach of Abstractive Summarization. Abstractive summarization is a text summarization technique where a machine generates a concise summary of a longer document by interpreting and rephrasing the content in a human-like manner. Abstractive summarising uses natural language creation to provide summaries that may include innovative words and phrases, as opposed to the extractive summary, which chooses sentences verbatim from the original text. This method is more adaptable and

creative but also more difficult to execute effectively because it seeks to capture the core meaning and context of the source text. Let us take a look at the most popular methods of abstractive summarization.

3.2 Abstractive Summarization:

1. **Seq2Seq (Sequence-to-Sequence) Models:** The model takes a variable-length input sequence, such as a sentence in one language, and uses an encoder to convert it into a fixed-length context vector or hidden representation. The most important details from the input sequence are encoded in this context vector [3]. It provides a summary of the input and includes the pertinent background information needed to produce the output sequence. The context vector and a unique "start token" are used to initialize the decoder. This prepares the groundwork for producing the output sequence. The decoder creates the output sequence step by step, predicting each element (such as a word or symbol) while taking the context vector and the elements that have already been produced into account. To concentrate on particular segments of the input sequence, attention mechanisms may be used. The decoder creates elements repeatedly, each prediction being dependent on the preceding one. It keeps on until either a unique "end token" is produced or a predefined maximum sequence length is reached [13].
2. **Pointer-Generator Networks:** Like other Seq2Seq models, Pointer-Generator Networks consist of an encoder and a decoder. The encoder processes the input document, while the decoder generates the summary. A crucial feature of Pointer-Generator Networks is the ability to copy words directly from the source document. This capability is achieved through the implementation of a copy mechanism, which grants the model the decision-making power to choose between generating a word from its own vocabulary or copying it directly from the source document. The model maintains its vocabulary of words that it can generate, but it also employs an attention mechanism to identify which words from the source document should be copied. This mechanism enables the model to effectively handle out-of-vocabulary words. For every word in the vocabulary or source document, the model computes both a generation probability and a copy probability, allowing for flexible and accurate word selection during the summarization process. It combines these probabilities to decide what to output in the summary. To prevent excessive copying of the same words, Pointer-Generator Networks often use a coverage vector that keeps track of which source words have been attended to. This helps in maintaining diversity in the generated summary [11].
3. **BERTSUM:** BERTSUM first tokenizes the input document into word pieces and encodes them into contextualized embeddings using a pre-trained BERT model. It computes the salience scores for each sentence in the document. This is done by applying a feedforward neural network to the BERT embeddings of each sentence, capturing their importance. The model employs an intra-sentence and inter-sentence scoring mechanism to refine the sentence importance scores. Intra-sentence scoring assesses the importance of each word within a sentence, while inter-sentence scoring measures the importance of sentences relative to each other.BERTSUM selects sentences with the highest importance scores, effectively identifying the most salient sentences in the document. Selected sentences are compressed by removing less relevant words and retaining the most informative ones. The compressed sentences are concatenated to form the summary. BERTSUM does not generate abstract summaries; it extracts and combines important sentences.

3.3 Deep Learning Summarization:

1. BART (Bidirectional and Auto-Regressive Transformers): BART is a transformer-based model that has been purposefully designed for sequence-to-sequence tasks, with a particular focus on summarization. BART has demonstrated its capability to achieve state-of-the-art performance in abstractive summarization tasks.
2. GPT (Generative Pre-trained Transformer): GPT models, including GPT-2 and GPT-3, can be fine-tuned for abstractive summarization. They generate summaries by predicting words or phrases that capture the essential information from the source text.
3. T5 (Text-To-Text Transfer Transformer): T5, another transformer-based model, adopts a unique approach by treating summarization as a text-to-text problem. In this framework, the input text is transformed into a summary text. T5 offers the flexibility of being fine-tuned for a wide range of summarization tasks, making it a versatile and powerful tool for various summarization applications.

Algorithms			Limitations
Extractive	**Unsupervised**	Fuzzy logic	To improve the quality of summarization, it is necessary to remove redundancies in the post-processing phase.
		Concept-based	The summary should use similarity measures to reduce redundancy, which can affect quality.
		Latent-Semantic	LSA-generated summaries take a long time.
	Supervised	Machine Learning	To make good summaries, it has to be trained and improved on a large set of data.
		Neural Network	Both the training phase and the application phase are quite slow with neural networks. Training data also requires human interruption.
		Conditional Random Fields	Linguistic features are not taken into account in the use of CRF. It also needs an external domain specific corpus.
Abstractive	**Structure-based**	Tree-based	It fails to recognise the relationship between sentences as a result of ignoring the context and important phrases in the text. Another issue is that it consistently emphasises syntax rather than meaning.
		Template based	As the templates are pre-defined using this technique, the summaries lack variation.
		Rule-based	The process of creating the regulations is time-consuming. The requirement to manually write the rules presents another difficulty.
		Ontology-based	It takes a long time to prepare an appropriate ontology, and it cannot be applied to other domains.
	Semantic-based	Multi-model semantic	The framework must be automatically analysed because humans now manually evaluate it.
		Information item	Generating grammatical and meaningful sentences from the material is difficult. Due to faulty parses, summaries also have very poor linguistic quality.
		Semantic graph	Limited to single document abstractive summarization.

Table-1: An overview of different summarization methods

4. Pegasus: It is a transformer-based model explicitly designed for abstractive summarization. It incorporates pre-training and fine-tuning stages and has shown strong performance in various summarization benchmarks.

Finally, we summarize the results of the papers for the reader's convenience in Table 1.

4 Challenges

The primary objective of any ATS (Automatic Text Summarization) system should be to generate summaries that closely resemble human-generated summaries. Nonetheless, achieving this goal poses significant challenges for existing ATS systems. These challenges include:

1. **Evaluation:** Assessing the quality of automatic text summaries is a complex task. Different datasets and metrics can yield varying results and might favor specific summarization techniques. While common datasets and metrics can produce satisfactory results, they come with their own issues. Metrics like precision and recall can be misleading and might not effectively evaluate sentences with semantic or syntactic errors. This can lead to high scores for unimportant sentences while overlooking grammatically incorrect yet meaningful ones.
2. **Important Sentence Selection:** Identifying the most crucial sentences in a text is subjective. Standardizing the selection process according to benchmarks can affect the resulting summary. Incorporating user-specific data can help address this challenge in professional summarization. Despite efforts to use vector representations and similarity matrices, there's no foolproof method for determining the most important sentences.
3. **Anaphora Problem:** Replacement of subjects with synonyms and pronouns is a common challenge in text summarization. Addressing this problem entails the identification of which pronoun corresponds to a specific word, a task that can pose considerable complexity for machine-based systems [10].

4. **Predefined Template:** While natural language processing has made remarkable progress in ATS, these methods often rely on predefined templates for summarization tasks. They cannot generate entirely new sentences independently, necessitating the use of specific templates.
5. **Long Sentences and Jargon:** Current text summarization models excel at summarizing shorter sentences but may struggle with longer sentences and specialized jargon. Addressing this limitation requires the development of architectures capable of effectively summarizing longer sentences and handling domain-specific terminology.
6. **Interpretability:** Abstractive models are designed to generate concise representations of source content, but they can encounter difficulties when it comes to capturing the nuances of human language and the expression of emotions in written text. Achieving interpretability with abstractive models presents a challenging task due to the inherent complexity of human language and its emotional dimensions.
7. **Cataphora Problem:** Ambiguity in words, stemming from multiple meanings or different contexts, can impact sentence summarization accuracy. This issue, known as the Cataphora problem, can be mitigated using disambiguation algorithms to match acronyms with their intended topics.

Additional challenges include ensuring that summary sentences are meaningful and impactful to users and creating robust representations that handle difficulties encountered by the system. Ongoing research in text summarization focuses on achieving higher levels of abstraction, offering numerous opportunities for researchers and linguists to explore solutions [2].

5 Overview of Important Features

Text representation models have gained popularity in enhancing how input documents are understood. In the field of natural language processing (NLP), these models transform words into numerical forms, allowing computers to detect patterns within language [8]. They establish connections between selected phrases and the context within documents. Among the most prominent text representation techniques are Bag of Words, Term Frequency-Inverse Document Frequency,N-gram and Word Embedding.

1. N-gram: As it does not need extensive linguistic preprocessing, it is ideal for multilingual situations. It groups words or characters into N components to create a model. The resulting vector representation is reasonably sized and manageable. However, N-grams have limitations; they become more effective with larger N, which increases processing demands and requires substantial RAM. Furthermore, N-grams yield a sparse language representation relying on term co-occurrence probabilities. Words not in the training corpus receive a zero probability.
2. Bag of words (BoW): BoW can lead to vector sparsity, potentially impacting model performance. Additionally, BoW disregards word order within sentences, which can be crucial for text comprehension. To address these limitations, more advanced models like N-grams and Word Embedding have been developed.
3. Term Frequency-Inverse Document Frequency(TF-IDF): IDF gauges word importance by considering its frequency across the corpus, while TF measures its occurrence in a specific document. By combining both metrics, TF-IDF determines a term's importance within a document relative to its significance in the entire corpus. However, TF-IDF has its constraints, such as slow performance with large vocabularies and an assumption that term counts are independent indicators of similarity.
4. Word Embedding: Word embedding algorithms map words or phrases into fixed-dimensional vectors within a continuous space. Similar words or phrases are represented by closely located vectors [9] and [12]. Few famous algorithms are:

   - FastText
   - Global Vectors for Word Representation (GloVe)
   - Word2Vec

6 Discussion

In this paper, we have highlighted how this technology addresses the challenge of condensing extensive textual information into concise and coherent summaries. This has far-reaching applications in various sectors, including journalism, research, business, and social media.

We have explored the two primary approaches to text summarization: extractive and abstractive. Extractive methods focus on selecting key sentences or phrases directly from source text and arranging them into forming a

summary. While abstractive methods involve interpreting and rephrasing the content in a more human-like manner to generate summaries. The flexibility and adaptability of abstractive summarization are emphasized.

The paper talks about various techniques and methodologies used in text summarization. It mentions methods such as TF-IDF, clustering, graph-based approaches, and deep learning models. These techniques cater to different requirements and complexities of summarization tasks, offering a diverse range of tools for summarization practitioners.

Lastly, we have identified several challenges associated with text summarization. These include difficulties in evaluating the quality of automatic summaries, the subjective nature of sentence selection, and linguistic challenges like anaphora and cataphora problems. It also discusses issues related to handling long sentences and specialized jargon. The future scope of our video-based text summarization approach is ripe with potential for innovation and expansion. By continually pushing the boundaries of what is possible, we can empower users with more advanced, customizable, context-aware video summarization solutions.

The paper provides a comprehensive overview of text summarization, highlighting its utility, various methodologies, challenges, and promising future developments. It emphasizes the role of this technology in addressing the growing need to efficiently distill information from vast textual data sources.

7 Conclusion and Future Scope

Looking ahead to future scope, our innovative approach to video-based text summarization opens up exciting possibilities for continued research and development. While we have made significant strides in extracting audio, converting it into text, and summarizing video content, there are several promising avenues for further exploration and refinement:

1. **Enhanced Multimodal Analysis:** Expanding our capabilities to include not just audio but also visual elements within videos will provide a more comprehensive understanding of the content. Incorporating image recognition and object detection techniques can lead to richer and more contextually relevant summarization.
2. **Real-Time Summarization:** Developing real-time video summarization algorithms will be crucial for applications such as live broadcasts, social media streaming, and surveillance. This will require optimizing processing speed without compromising summarization quality.
3. **Customizable Summaries:** Enabling users to customize the level of detail and specific aspects they want in their summaries can make the technology more user-centric. Tailoring summaries to individual preferences and needs will be a valuable feature.
4. **Multilingual Summarization:** Expanding our system to support multiple languages will broaden its utility and accessibility on a global scale, allowing users to summarize videos in languages other than English.
5. **Deep Learning Advancements:** As deep learning techniques continue to evolve, exploring the integration of the latest models and architectures can lead to significant improvements in summarization accuracy and coherence.
6. **Application Diversification:** Further exploring the diverse applications of video summarization, such as education, content recommendation, and market research, will open up new opportunities for commercial and societal impact.
7. **Ethical Consideration:** As video summarization technology becomes more widespread, addressing ethical concerns surrounding privacy, bias, and misinformation detection will be crucial to ensure responsible deployment.
   It can be further used to analyze the mental and sentimental health of the user which will further help us analyse the psychological health of the user.

References

1. Aker, A., Cohn, T., Gaizauskas, R.: Multi-document summarization using a* search and discriminative learning. In: Proceedings of the 2010 conference on empirical methods in natural language processing. pp. 482–491 (2010)
2. AL-Banna, A.A., AL-Mashhadany, A.K.: Natural language processing for automatic text summarization [datasets]-survey. Wasit Journal of Computer and Mathematics Science **1**(4), 156–170 (2022)
3. Dey, M., Mondal, A., Das, D.: Ntcir-12 mobileclick: Sense-based ranking and summarization of english queries. In: Ntcir (2016)
4. Edmundson, H.P.: New methods in automatic extracting. Journal of the ACM (JACM) **16**(2), 264–285 (1969)
5. Ferreira, R., de Souza Cabral, L., Lins, R.D., e Silva, G.P., Freitas, F., Cavalcanti, G.D., Lima, R., Simske, S.J., Favaro, L.: Assessing sentence scoring techniques for extractive text summarization. Expert systems with applications **40**(14), 5755–5764 (2013)

6. García-Hernández, R.A., Ledeneva, Y.: Word sequence models for single text summarization. In: 2009 Second International Conferences on Advances in Computer-Human Interactions. pp. 44–48. IEEE (2009)
7. Kupiec, J., Pedersen, J., Chen, F.: A trainable document summarizer. In: Proceedings of the 18th annual international ACM SIGIR conference on Research and development in information retrieval. pp. 68–73 (1995)
8. Mondal, A., Cambria, E., Dey, M.: An annotation system of a medical corpus using sentiment-based models for summarization applications. In: Computational Intelligence Applications for Text and Sentiment Data Analysis, pp. 163–178. Elsevier (2023)
9. Mondal, A., Dey, M., Mahata, S.K., Sarkar, D.: An automatic summarization system to understand the impact of covid-19 on education. In: Applications of Machine intelligence in Engineering, pp. 379–386. CRC Press (2022)
10. Mridha, M.F., Lima, A.A., Nur, K., Das, S.C., Hasan, M., Kabir, M.M.: A survey of automatic text summarization: Progress, process and challenges. IEEE Access **9**, 156043–156070 (2021)
11. Syed, A.A., Gaol, F.L., Matsuo, T.: A survey of the state-of-the-art models in neural abstractive text summarization. IEEE Access **9**, 13248–13265 (2021)
12. Tas, O., Kiyani, F.: A survey automatic text summarization. PressAcademia Procedia **5**(1), 205–213 (2007)
13. Zhang, Y., Zincir-Heywood, N., Milios, E.: Narrative text classification for automatic key phrase extraction in web document corpora. In: Proceedings of the 7th annual ACM international workshop on Web information and data management. pp. 51–58 (2005)