Communications on Applied Nonlinear Analysis
ISSN: 1074-133X
Vol 32 No. 9s (2025)

A Comprehensive Study of Text Summarization with Advent of Large Language Models

[1]Dr. Geetanjali Vinayak Kale, HOD Computer Engineering, [2]Prof. Pranali Rajendra Navghare, Computer Engineering, [3]Atharva Sadanand Litake, Computer Engineering,[4]Apurva Ajit Kulkarni, Computer Engineering, [5]Kishanlal Chhelaram Choudhary, Computer Engineering, [6]Aditya Ashru Darade, Computer Engineering

[1.] Pune Institute of Computer Technology, gvkale@pict.edu

[2.] Pune Institute of Computer Technology, prnavghare@pict.edu

[3.] Pune Institute of Computer Technology, C2K21106776@ms.pict.edu

[4.] Pune Institute of Computer Technology, C2K2221@ms.pict.edu

[5.] Pune Institute of Computer Technology, C2K21106773@ms.pict.edu

[6.] Pune Institute of Computer Technology, C2K21106899@ms.pict.edu

Article History:

Received: 12-01-2025

Revised: 15-02-2025

Accepted: 01-03-2025

Abstract:
Introduction: Communication is at the heart of the human race. With the growth of social media and other communication platforms, the globe is now connected at a single click. People communicate and tend to share information through these platforms. A massive amount of data is being generated and being analysed every second. To tackle the problem of analysing Big Data and withdraw insights from it, is a difficult task. Text summarization is the process of concise representation of textual data so as to extract the most important information out of it. Text summarization plays a major role in analysing big data and taking decisions based upon the insights drawn. With the advent of Large Language Models, the techniques used for summarization have been enhanced to a large extent. The following paper surveys old techniques for text summarization and studies the new methodologies using Large Language Models. The paper aims to deliver the most up-to-date survey of text summarization and enhancement in it using Large Language Models.

Objectives: The objective of this paper is to provide a comprehensive study on text summarization, focusing on its evolution and the advancements brought by Large Language Models (LLMs). It aims to analyse the development of summarization techniques, including both extractive and abstractive methods, while exploring the mathematical algorithms and machine learning models that underpin these approaches. Additionally, the paper discusses the impact of natural language processing (NLP) advancements, particularly LLMs, in enhancing the accuracy and efficiency of summarization. A comparative analysis of traditional and modern approaches is presented, evaluating their effectiveness using various datasets. Furthermore, the study highlights key research contributions in the field and identifies current challenges, paving the way for future innovations in text summarization.

Methods: Text summarization research has evolved significantly, exploring extractive and abstractive methods using machine learning and LLMs. Mark Dredze et al. used LSA and LDA for email summarization, while Mohamed Abdel Fattah et al. trained ML models for sentence extraction. Jan Ulrich et al. applied regression-based learning for email thread summaries. Pete Burnap et al. focused on summarizing real-world events from social media.

Derek Miller et al. used BERT and K-Means for lecture summarization, and Rahim Khan et al. leveraged K-Means and TF-IDF for news summarization. Mingxi Zhang et al. optimized TextRank for keyword extraction. Jingqing Zhang et al. introduced PEGASUS for abstractive summarization with advanced pre-training techniques. Zhang et al. also explored long-dialogue summarization using retrieval-based and hierarchical encoding methods.

Conclusions: The advent of Large Language Models (LLMs) has significantly advanced text summarization by addressing the shortcomings of earlier extractive and abstractive techniques. Traditional extractive methods often produced fragmented summaries, while early abstractive approaches struggled with coherence and redundancy. LLMs, powered by transformers and self-attention mechanisms, have enabled more fluent, contextually aware, and human-like summaries. These models can effectively capture long-range dependencies, rephrase content, and generate concise yet meaningful summaries. As a result, LLMs have expanded the scope of text summarization, making it more applicable and reliable across various domains, including news, research, and automated content generation.

Keywords: Large Language Models, Text Summarization, Summarization Survey, Big Data

1. Introduction

World is growing at a faster speed and so does networking amongst individuals. Social media has now become one of the essential necessities of life. With the advent of Social Media applications like Whatsapp, Twitter, Facebook and many more, people around the globe grew more interconnected. Regardless of borders and distance, now people are a single click away from each other. Communication is at the heart of all these Social Media platforms.

Communication amongst individuals can take place via messages, tweets, emails and other functionalities. To get connected and stay up-to-date with all the real-world events, every individual uses these communication media. These platforms not only help in establishing contact with near and dear ones but provide various updates from various sectors such as science & technology, medical, agriculture and countless others. They act as a rich source of information from all walks of life. But, with the increasing number of users on these platforms, a huge amount of data is generated. According to a survey, 500 million tweets are made every day on twitter. On an average, a person receives 121 emails every day. Looking at the growing number of users and data generated, it is humanly impossible to analyse and understand each and every message and gain information from it.

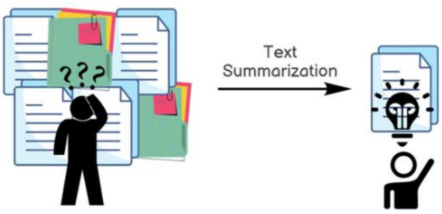

Fig 1. Role of Text Summarization

As a solution to the above problem, the role of Text Summarization and techniques was taken into consideration. Text summarization is the process of distilling the most important information from a source to produce an abridged version for a particular user and task. Text summarization is a concise representation of a huge amount of data without hampering meaning and context of the data. It creates a short and coherent version of data, through which understanding and analysing the data becomes easier. It enhances the process of understanding and information extraction. In today's fast-paced world, where everyone is constantly racing against time, text summarization serves as a valuable tool. Growing research in the field of Natural Language Processing (NLP) and its application in Text Summarization have opened wide doors for enhancement in Text summarization techniques. New methodologies based upon Large Language models have been developed and used in recent times. The following paper aims to represent a comprehensive study on text summarization, its development and advancement with the advent of Large Language Models. It also aims to briefly discuss all the old and new methodologies used for text summarization.

2. Related Survey

Many research papers have been published in the domain of text summarization since the start of the 21st century. Text Summarization research papers focus on both abstractive and extractive text summarization methodologies and all the statistical and machine learning models involved. Various mathematical algorithms and Large Language models have been used for Text Summarization. This section depicts the overview of various research papers published since the beginning of the 21st century in the field of Text Summarization. Mark Dredze and co-authors [5] state that with increasing networking and communications within individuals, people have started using mail service to a large extent. Number of emails generated has been increasing substantially. It is humanly impossible to read and understand each and every email received from a variety of sources. To address this concern, the author along with co-authors proposes an unsupervised learning approach to generate summary of keywords for emails using latent topic models. The goal of this technique is to provide the user the gist of email in a very concise and precise way. To achieve this, the paper proposes use of Latent Semantic Analysis (LSA) and Latent Dirichlet Allocation (LDA) for generating summary keywords. The four methodologies namely LDA-doc, LSA-doc, LDA-word, and LSA-word are employed and compared against the TF-IDF as baseline. Evaluation of these methodologies is done by using Enron email corpus open-source dataset which contains around 25,000 emails from 150 users.

Mohamed Abdel Fattah et al. [6] studied automatic text summarization using a trainable summarizer based on sentence-level features like position, keyword presence, length, and centrality. Their goal was to develop a system that extracted important sentences for summaries by training ML models to weigh these features. To achieve this, the authors described four techniques: Feature Extraction, Trainable Summarizer, Genetic Algorithm (GA), Mathematical Regression (MR), and Testing and Summarization. Feature Extraction identified key features, and the Trainable Summarizer implemented two overarching models. GA used natural selection to optimize feature weights, while MR correlated sentence features with manually constructed summaries. In Testing and Summarization, sentences were ranked and selected based on scores, considering a compression ratio. Training data consisted of 50 manually summarized religious articles, while the testing data included 100 English religious articles from the Internet archive.

Jan Ulrich et al. states the need to summarize email conversations due to the growing volume of email each individual receives. This paper [17] emphasizes the use of a regression-based machine learning approach for email thread summarization. The research also focuses on a novel feature - speech acts and their usefulness in email summarization. It also compares various regression-based classifiers and proves how they perform better than binary based classifiers. To validate the performance, it uses 10-fold cross validation for evaluation and employs weighted recall as the evaluation metric. The author uses BC3 corpus dataset and Enron email corpus open-source dataset [1] to compare performance of various regression-based classifiers.

Pete Burnap et al. [1] addresses the increase in social networking amongst individuals via social media platforms such as Twitter, Threads and many more. To stay aware and up-to-date with all the real-world events, each individual seeks information via these platforms. This paper proposes summarization of real-world event tweets and provides gist of all the recent tweets to users. To achieve this, the author states three techniques - Temporal TF-IDF, Retweet Voting and Temporal Centroid Representation for automatic summarization of generated tweets. Using Temporal TF-IDF, the words and their importance are evaluated based upon timelines whereas in the Retweet Voting approach, consider a tweet as a representative message taking the number of retweets and their timeline into consideration. The Temporal Centroid method evaluates similarity and centrality of various tweets across different clusters. Around 75,000 tweets dataset was used for evaluating various techniques and validated based upon a variety of evaluation metrics.

Derek Miller and co-authors [8] created a lecture summary service that used BERT for extractive summarization in hopes of identifying key sentences from lecture transcripts using machine learning techniques such as BERT for text embeddings and K-Means clustering for sentence selection. It addressed some of the inconveniences related to existing systems by providing dynamic summaries for lectures. Text preprocessing involved tokenizing the transcript and discarding useless information. BERT provided improved sentence embeddings based on the pre-last layer, while K-Means clustering selected sentences closest to the centroid for the summary. It had a RESTful API based on Python that allowed users to submit transcripts and specify the preferred summary length. The authors did not name a specific dataset but mentioned that the service was tested on lecture transcripts, possibly from Udacity. Rahim Khan and co-authors [11] developed clustering methods for extractive text summarization, particularly K-means clustering and TF-IDF. They aimed at reducing the workload of large texts and extracting key sentences for use in automatic summarization. Basic methods were included, preprocessing including tokenization and stemming. The Elbow method and Silhouette method were used to determine the best numbers of clusters (K values), grouping with respect to the importance of sentences based on the scores from TF-IDF, and their performance has been evaluated in terms of precision metrics against other summarization approaches. experimented on news articles demonstrated improved summary accuracy and readability.

Mingxi Zhang and co-authors [22] studied the performance of TextRank for keyword extraction, experimenting with parameters like co-occurrence window size, iteration count, decay factor, and text length. Their goal was to optimize TextRank for tasks such as text classification, information retrieval, and summarization. Text preprocessing involved removing stop words using the XPO6 list and applying Porter Stemmer for word stemming. The TextRank model treated keywords as nodes and cooccurrences as edges, using PageRank to rank their importance. Experiments focused on evaluating window size, iterations, and performance using Precision, Recall, and Accuracy. The authors used the Hulth2003 dataset with manually assigned keywords and the Krapivin 2009 dataset of 2304 computer science papers.

Jingqing Zhang along with co-authors [21] showcase their work in the domain of abstractive summarization and tailoring pre-training objectives. The recent work in the domain of extractive summarization showed great success but the sector of abstractive summarization has not been explored up to its full potential. This paper from Google AI emphasizes on pre-training large Transformer-based-encoder-decoder models on massive text corpora with advanced self-supervised objectives. It introduces a State-of-the-Art algorithm for abstractive summarization with novel pre-training objectives. It employs the use of both Gap Sentence Generation (GSG) and Masked Language Model (MLM) as pre-training objectives. PEGASUS model removes/masks important sentences from an input document and generates all together a singular output sequence from all the remaining sentences. The author tests the model across 12 different datasets related to various domains like media, science, medical, governmental and many more. The results of the model were validated using human evaluation.

Zhang and co-authors [24] conducted a study on long-dialogue summarization, addressing the limitations of current transformer-based models in handling lengthy dialogues. The authors experimented with Long Former models, focusing on a retrieve-then-summarize pipeline and hierarchical dialogue encoding models like HMNet, to improve summarization performance in tasks such as meeting and media interview summarization. Preprocessing involved tokenization and truncation of dialogues when input limits were exceeded. In the summarization task, utterances were treated as nodes, with edges representing connections between two utterances, and importance was ranked using various retrieval methods, such as TF-IDF and BM25. Performance was measured using ROUGE. The datasets used included QMSum, MediaSum, and SummScreen, and the experiments highlighted the impact of retrieval and encoding methods on summarization quality.

Table 1. Overview of methodologies used for Text Summarization

No.	Title	Author	Approach	Dataset	Year
1	Generating Summary Keywords for Emails Using Topics	Mark Dredze, Hanna M. Wallach, Danny Puller, Fernando Pereira	Latent Dirichlet Allocation (LDA)	Enron Email Corpus , containing 25,000 emails from about 150 users.	2008
2	Automatic Text Summarization	Mohamed Abdel Fattah, Fuji Ren	Genetic Algorithms and Mathematical Regression	Training Data: 50 English articles Testing Data: 100 English religious articles available on the Internet.	2008

3	Regression-Based Summarization of Email Conversations	Jan Ulrich, Giuseppe Carenini, Gabriel Murray, Raymond Ng	Regression based Classifiers.	BC3 Corpus - British Columbia Conversation Corpus and Enron Email Corpus	2009
4	Automatic Real-World Event Summarization via Twitter	Nasser Alsaedi, Pete Burnap, Omer Rana	Temporal TF-IDF and Temporal Centroid Representation	A dataset of 75,000 tweets from 50 trending topics and a dataset of 2.7 million tweets.	2016
5	Leveraging BERT for Extractive Text Summarization on Lectures	Derek Miller	BERT for extractive summarization and K-Means clustering for selecting key sentences.	No specific dataset, used transcription from lectures.	2019
6	Extractive Based Text Summarization Using K-Means and TF-IDF	Rahim Khan, Yurong Qian, Sajid Naeem	K-means clustering with TF-IDF in extractive text summarization	No specific dataset used, experimented on news articles.	2019
7	An Empirical Study of TextRank for Keyword Extraction	Mingxi Zhang, Xuemin Li, Shuibo Yue, Liuqian Yang	TextRank Algorithm	Hulth2003 corpus and Krapivin2009 dataset of 2304 full papers.	2020
8	PEGASUS: Pre-training with Extracted Gap-sentences for Abstractive Summarization	Zhang, J., Zhao, Y., Saleh, M., Liu, P. J.	Transformer based encoder-decoder model - PEGASUS	Tested across 12 different datasets CNN/DailyMail, XSum, PubMed, ArXiv, BigPatent, GovReport	2020

9	An Exploratory Study on Long Dialogue Summarization: What Works and What's Next	Yusen Zhang, Ansong Ni, Tao Yu, Rui Zhang, Chenguang Zhu, Budhaditya Deb, Asli Celikyilmaz, Ahmed Hassan Awadallah, Dragomir Radev	TF-IDF and Best Matching 25 (BM25)	The datasets used included QMSum, MediaSum, and SummScreen.	2021

3. Definition of text summarization

Text summarization is the process of representing a huge amount of textual data in the form of a concise way retaining information and other main ideas. The ultimate objective of using text summarization is to gain useful information and understand data without giving a thorough read. Text summarization is used to shorten lengthy texts without losing its original meaning. Text summarization mainly can be divided into two types - Extractive Text Summarization and Abstractive Text Summarization.

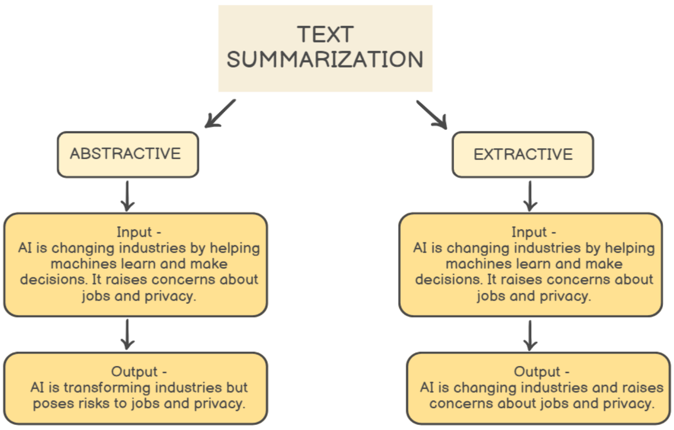

Fig 2. Types of Text summarization with example

Extractive Text Summarization approach directly selects words, phrases and sentences from the text without rephrasing it and forms a summary out of it. The selected phrases are not at all altered or changed in any way but rearranged in a way that makes summary coherent. The process of selection of statements lies at the core of Extractive Text Summarization. It relies on various frequency based and supervised machine learning algorithms for selection of appropriate statements.

Abstractive Text Summarization approach rewrites the original text in shortened form with rephrasing of original sentences or phrases extracted. This methodology does not rely upon rearranging the phrases extracted from the text. It uses NLP techniques to understand the meaning of the text and rewrites it in a concise way without losing the original meaning of the text. Abstractive Text

Summarization is used to devise more human-like summaries and works in a similar approach to how humans will summarize the text.

4. Text Summarization Techniques

Before the rise of LLM, the Text Summarization approach relied more upon statistical approaches and traditional rule-based NLP techniques. These approaches were based upon both abstractive and extractive text summarization which used simpler rule-based algorithms. The succeeding section evaluates all the methodologies of Text summarization and describes newer Text Summarization methodologies based upon LLMs.

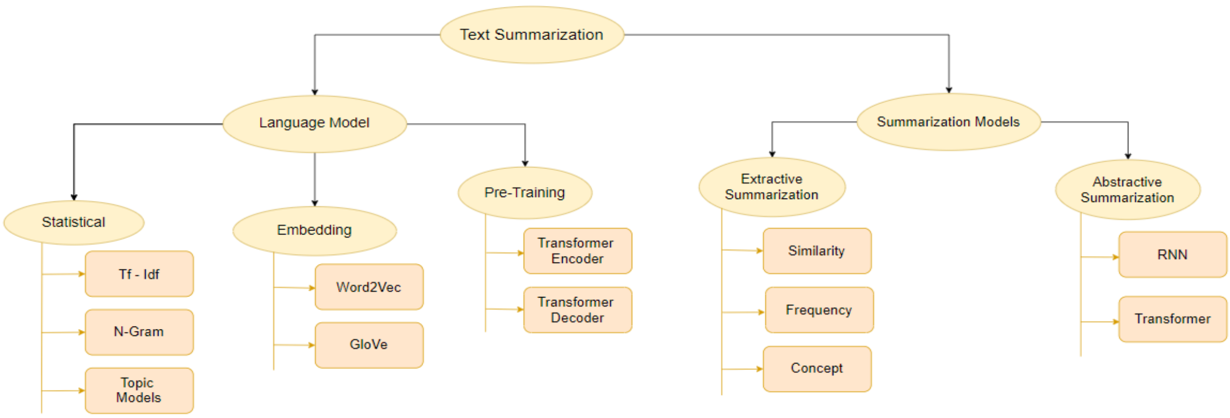

Fig 3. Older Text Summarization methodologies

4.1. Term-Frequency (TF) and Inverse Document Frequency (IDF)

TF is a measure of how many times a word occurs in a document and is majorly used in summarization techniques and in retrieving information. TF is calculated by dividing how many times a word occurs in a document by the total number of words in the document. The more words appear in the document, it increases its relevance to the document. TF is often used with Inverse Document Frequency (IDF), which measures how common a term occurs across a collection of documents. The TF-IDF score is calculated as TF*IDF.[19] used TF-ISF (Term Frequency- Inverse Sentence Frequency) approach to extract sentences.[1] used Temporal TF-IDF approach to summarize Twitter posts.TF-IDF, works to identify the main content words in documents by weighing them with relative frequency in the document and rarity across a group of documents. Sentences or phrases bearing important words that happen to score high on TF-IDF will serve a great purpose in identifying key sentences that will feature among the summaries while common words that lose relevance are generally ignored.TF-IDF is simple and fast, being able to identify important terms based on their rarity, but ignores word order and context, thus not suited for capturing nuances like synonyms.

$$Term\ Frequency = \frac{number\ of\ t\ in\ d}{total\ number\ of\ terms\ in\ d} \quad (1)$$

$$IDF = log\frac{total\ number\ of\ documents}{number\ of\ documents\ with\ t} \quad (2)$$

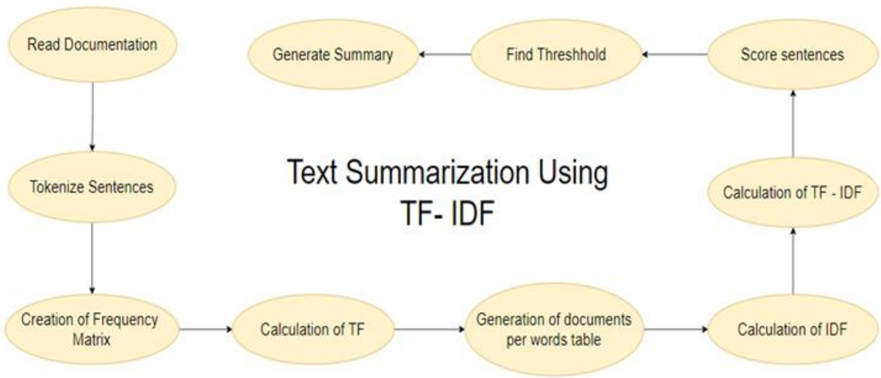

Fig 4. Diagrammatic flow of Text Summarization using TF-IDF

4.2. N-gram Model

In text summarization, n-grams are contiguous sequences of n words that allow one to represent patterns and context in a text. By analyzing unigrams (single words), bigrams (two-word combinations), trigrams (three-word combinations), and higher-order n-grams, the summarization process can effectively assess word relationships, improve sentence coherence, and isolate meaningful phrases or sentences. The approach allows preservation of context with a reduction in redundancy within the summary.[16] uses N-gram for representation of sentences.[13][14] uses bi-gram text fragments of length 2 to preserve semantic relationships. N-grams help solve this problem by maintaining word sequences and context, but those who would use higher-order n-grams would incur a hefty computational cost along with data sparsity.

4.3. Topic Model

Topic modelling makes the identification and grouping of dominant topics of a document possible. In identifying the most dominant topics to present information, summarization can be appropriately directed toward such content, so that the gist of the summarization reflects key themes and ideas in the text instead of bits of information in the text. This produces more coherent and informative summaries, and it may operate for longer or multi-topic documents.[20] describes applying LSA (Latent Semantic Analysis) to build a word-sentence matrix, from which relationships between words and sentences were extracted by means of decomposition. The summary was generated by first filtering sentences through a classifier. Topic models such as LDA or NMF perform well at identifying key themes across large-or-multi-topic documents, providing a coherent summary for readers though they are complex yet difficult to implement and might struggle with short text or fine-grained detail.

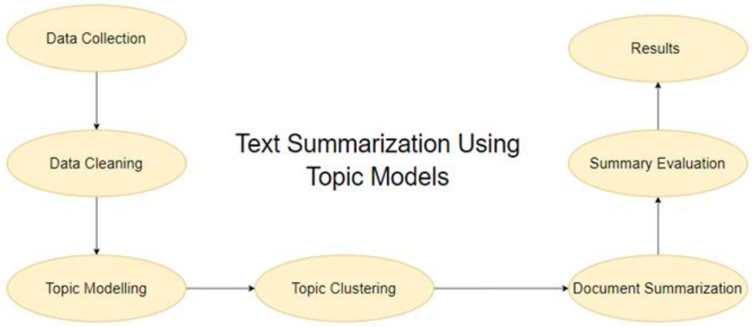

Fig 5. Topic Model Flow Diagram

4.4. Word Embedding Model

Word Embedding Models, for example Word2Vec, GloVe, or BERT, convert words into dense vectors in a continuous vector space, which takes semantic relationships and meaning into account. Thus, the system can recognize that two words might have meant in common, even if they have different word sequences or word order. Such word embeddings are used in text summarization to generate summaries that are even more accurate and contextually rich. These summaries contain not only those high-quality phrases, but semantically rich sentences as well which, albeit of different words, express similar things. Therefore, we can create a better and more coherent summary from a larger dataset. [9] used Word2Vec model and [15] used GloVe model for summarization.

4.5. Transformer Encoder-Based Model

BERT and T5, two transformer models, exhibit phenomenal results in text summarization due to their capacity to model context and relationships among words across a document. Based on how they influence the words, self-attention conveniently provides weights towards words depending on their contextualized meaning. This offers them a far-reaching understanding of how they formulate their meaning and structure in a sentence. In text summarization, by establishing key sentences/phrases that most explicitly cover the main idea being stressed by the text, the transformer gets to generate summaries that are coherent, context-contingent, and succinct. From another perspective, they handle long-range dependencies pipeline, thus usable for summarizing huge amounts of data. [4] used the BERT model for capturing deep conceptual semantic information. The transformer encoder-based models are capable of gathering context and even handle long-range dependencies successfully, thus generating coherent summaries, but they generally need high computational resources and still lack fluency in text generation.

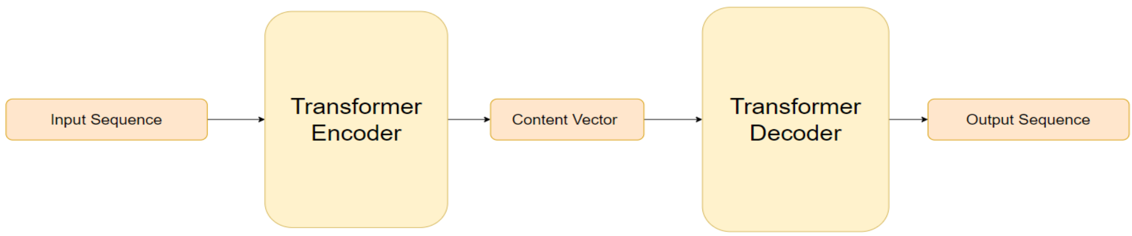

Fig 6. Transformer Encoder - Decoder

4.6. Transformer Decoder-Based Model

Text generation models based on a transformer decoder, such as GPT (Generative Pre-trained Transformer) [2], served principally for text summarization. They are fed with the context of either an encoder or previous outputs, generating coherent summaries through a word-completing process. The generated summaries could be concise, coherent, and contextually relevant. Powerful text generation is owing to the decoder's ability to consider the entire context for the summarization process. Thereafter, it tries to focus on the important details while ignoring less relevant information so as to provide high-level summaries that portray the gist of any original text. The transformer decoder-based models excel in generating fluent and contextually relevant summaries but at the risk of occasionally overlooking essential parts of the text.

4.7. Similarity-based Summarization

In similarity-based extractive summarization, the approach relies on measuring the similarity between sentences in the document. Techniques such as cosine similarity or sentence embeddings are used to compare the relationship between different parts of the text. By clustering similar sentences and identifying the most representative ones, the model creates a summary that captures the core ideas of the document. Popular methods include the TextRank algorithm, which builds a graph of sentence similarities and ranks sentences based on their importance in the overall structure.

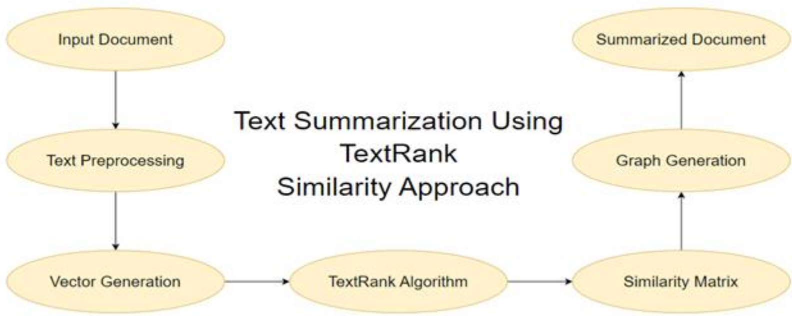

Fig 7. TextRank Algorithm

4.8. Frequency-based Summarization

Frequency-based summarization works on the assumption that the most frequent terms in a document are often the most important. This method calculates the frequency of words or phrases, ranks sentences based on how many high-frequency words they contain, and selects the top sentences for inclusion in the summary. Common techniques like TF-IDF (Term Frequency-Inverse Document Frequency) are frequently used to identify significant terms. Though simple, this method can be highly effective for text summarization when dealing with documents where repetition highlights key themes or topics.

4.9. Concept-based Summarization

Concept-based summarization shifts the focus from individual word frequency to the identification of overarching ideas or concepts in a text. This approach seeks to understand the key themes or concepts that represent the entire document and selects sentences that best express those ideas. Techniques like

Latent Semantic Analysis (LSA) or Latent Dirichlet Allocation (LDA) can be used to uncover hidden topics or concepts within the text. By summarizing the document at the conceptual level, this method offers a more insightful reduction of the content, especially useful in long or complex texts.

4.10. Recurrent Neural Network (RNN)-based Summarization

Abstractive summarization models initially employed Recurrent Neural Networks (RNNs), particularly through sequence-to-sequence (Seq2Seq) architectures. In these models, the text is encoded into a fixed-length vector, which is then passed through a decoder to generate the summary. RNN-based models, particularly Long Short-Term Memory (LSTM) and Gated Recurrent Units (GRU), allow for the generation of summaries based on the understanding of the sequence of words in a document. The introduction of attention mechanisms greatly improved the performance of these models by allowing the model to focus on different parts of the input text while generating the output, thereby improving the relevance and coherence of the summary.

4.11. Transformer-based Summarization

Transformer-based models have dramatically advanced the field of abstractive summarization. Unlike RNNs, transformers use a mechanism called self-attention, which enables the model to focus on different parts of the input text simultaneously. This allows the model to handle longer documents more effectively. Models like BERT, GPT, T5, and BART have demonstrated state-of-the-art performance in abstractive summarization tasks. For example, BART (Bidirectional and Auto-Regressive Transformers) combines the bidirectional encoding of BERT with an auto-regressive decoding mechanism, making it highly effective for summarization. T5 (Text-to-Text Transfer Transformer) frames summarization as a text-to-text problem, where input text is transformed into a summarized output. Additionally, Pegasus, designed specifically for summarization tasks, uses a masked language model pre-training method that enables it to generate high-quality summari5.es by predicting entire masked sentences in a document.

5. Revolution in text summarization before and after all

The advent of Large Language Models (LLMs) has revolutionized text summarization by significantly improving upon traditional approaches. Prior to LLMs, summarization was largely dominated by extractive methods, where the system would select and stitch together sentences directly from the original text based on factors like term frequency, similarity metrics, or conceptual relevance. While this approach ensured grammatical correctness, the resulting summaries often lacked coherence, as they did not account for the flow or contextual relationships between sentences. Moreover, extractive models could not condense information effectively, leading to verbose and sometimes redundant outputs. Early abstractive methods, primarily based on Recurrent Neural Networks (RNNs) or Long Short-Term Memory (LSTM) models, aimed to generate summaries by rephrasing content but faced significant limitations. These models struggled to maintain fluency, produced repetitive content, and were unable to capture long-range dependencies within the text due to their sequential nature and limited memory capacity. As a result, the summaries generated were often incoherent or incomplete. The introduction of LLMs such as BERT, GPT, T5, and BART has drastically enhanced both extractive and abstractive summarization. These models leverage transformer architecture, which includes a self-attention mechanism that allows the model to weigh different parts of the text simultaneously. This enables LLMs to capture contextual relationships across the entire document, improving the relevance and coherence of the generated summaries. Abstractive models, in particular, have benefitted from LLMs as they are now capable of synthesizing and generating new sentences that better encapsulate the meaning of the original text. LLMs are pre trained on vast amounts of text data, making them adept at understanding a wide range of language patterns and structures, which results in more natural and fluent summaries. This has allowed for more concise, accurate, and contextually aware summarizations, particularly in complex domains like research papers, news articles, and technical documents. With the ability to understand deeper semantics and generate human-like summaries, LLMs represent a significant leap forward in the field of text summarization, setting a new standard for both accuracy and fluency.

6. Applications of text summarization

The end goal of using text summarization is to enhance the process of withdrawing insights from data that require a thorough reading. Text summarization finds its application in the field of email thread summarization, research paper analysis, media monitoring systems and many other sectors in which understanding from data is utmost important.

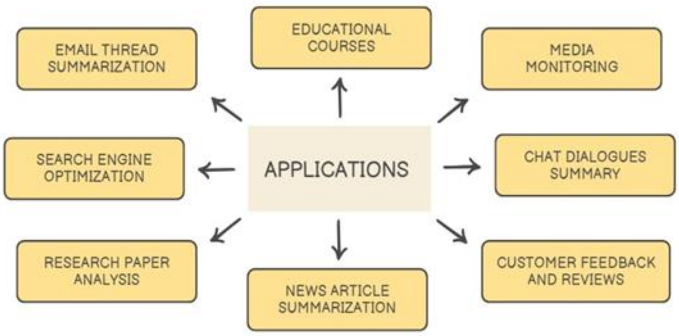

Fig 8. Applications of text summarization

1. Email thread summarization: Text summarization finds a wide-open domain in the field of electronic messaging systems. To understand the gist of the email is the main objective while reading the email. Text summarization

plays a vital role in summarizing all email threads and providing the user with the gist of the email.

2. Media Monitoring: For any individual and organization working in a space, it is important to analyse recent news and trends and make decisions upon it. With the growing market of e-newspapers and other electronic media, a huge amount of data is generated which increases cognitive load upon such individuals. Text Summarization condenses such a large volume of data and represents a concise summary for individuals and organizations to work upon. Additionally, summarization aids to identify recent trends amongst news and articles which proves to be beneficial to analyse them in a short span of time.

3. Chat dialogue Summary: On an average, a person of age group between 18 years to 24 years old receives 138 texts per day. To analyse the texts and know the importance of each text by giving a read to it, is a tiresome task. Extractive Text Summarization helps to extract important chats and

contents from the dialogue whereas Abstractive Text Summarization helps to understand the gist of the dialogue and their priority with respect to time.

4. Research Paper Analysis: A huge amount of research papers in a variety of domains get published every year at various journals and conferences. To study a topic and propose a new methodology on it required extensive study of all these documents. Text Summarization proves to be a boon for new researchers to get acquainted with their domain and start research at a faster pace.

7. Conclusion

The emergence of Large Language Models (LLMs) has dramatically changed the landscape of text summarization. Before LLMs, summarization techniques primarily relied on extractive methods, where sentences were selected directly from the text based on factors like frequency, similarity, or importance. While effective in identifying key points, these methods often produced summaries that were disjointed and lacked coherence. Early attempts at abstractive summarization, using RNNs or LSTMs, aimed to generate new sentences but struggled with fluency, repetition, and incomplete understanding of the text. The introduction of LLMs such as BERT, GPT, and T5 marked a significant improvement. These models, powered by transformers and self-attention mechanisms, can capture long-range dependencies within the text and generate summaries that are both contextually aware and fluent. They have the ability to rephrase content, offering more concise and coherent summaries that are closer to human-like understanding. This shift has not only improved the quality of summarization but has also broadened its applicability across various domains, including news, research, and content generation.

References

[1] Alsaedi, N., Burnap, P., & Rana, O. (2016). Automatic summarization of real-world events using twitter. In Proceedings of the International AAAI Conference on Web and Social Media (Vol. 10, No. 1, pp. 511-514).

[2] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal,A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin,S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, and D. Amodei, “Language models are few-shot learners,”in Advances in Neural Inf. Process. Systems, H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, Eds., vol. 33. Curran Associates, Inc., 2020, pp. 1877–1901.

[3] Hernández-Castañeda, Á., García-Hernández, R. A., Ledeneva, Y., & Millán-Hernández, C. E. (2020). Extractive automatic text summarization based on lexical-semantic keywords. IEEE Access, 8, 49896-49907.

[4] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “BERT: Pre-training of deep bidirectional transformers for language understanding,” in Proc. 2019 Conf. North American Chapter Assoc. Comput. Linguistics: Human Lang. Technol., Volume 1 (Long and Short Papers), J. Burstein, C. Doran, and T. Solorio, Eds. Minneapolis, Minnesota: Assoc. Comput. Linguistics, Jun. 2019, pp. 4171–4186.

[5] Dredze, M., Wallach, H. M., Puller, D., & Pereira, F. (2008, January). Generating summary keywords for emails using topics. In Proceedings of the 13th international conference on Intelligent user interfaces (pp. 199-206).

[6] Fattah, M. A., & Ren, F. (2008). Automatic text summarization. World Academy of Science, Engineering and Technology, 37(2), 192.

[7] Feng, X., Feng, X., & Qin, B. (2021). A survey on dialogue summarization: Recent advances and new frontiers. arXiv preprint arXiv:2107.03175.

[8] Miller, D. (2019). Leveraging BERT for extractive text summarization on lectures. arXiv preprint arXiv:1906.04165.

[9] T. Mikolov, K. Chen, G. S. Corrado, and J. Dean, “Efficient estimation of word representations in vector space,” in Int. Conf. Learning Representations, 2013.

[10] Jin, H., Zhang, Y., Meng, D., Wang, J., & Tan, J. (2024). A comprehensive survey on process-oriented automatic text summarization with exploration of llm-based methods. arXiv preprint arXiv:2403.02901.

[11] Khan, R., Qian, Y., & Naeem, S. (2019). Extractive based text summarization using k-means and tf-idf. International Journal of Information Engineering and Electronic Business, 12(3), 33.

[12] Langston, O., & Ashford, B. (2024). Automated summarization of multiple document abstracts and contents using large language models. Authorea Preprints.

[13] W. Luo, F. Liu, Z. Liu, and D. Litman, “Automatic summarization of student course feedback,” in Proc. 2016 Conf. North American Chapter Assoc. Comput. Linguistics: Human Lang. Technol., K. Knight, A.Nenkova, and O. Rambow, Eds. San Diego, California: Assoc.Comput. Linguistics, Jun. 2016, pp. 80–85.

[14] X. Qian and Y. Liu, “Fast joint compression and summarization via graph cuts,” in EMNLP, D. Yarowsky, T. Baldwin, A. Korhonen, K. Livescu, and S. Bethard, Eds. Seattle, Washington, USA: Assoc. Comput. Linguistics, Oct. 2013, pp. 1492–1502.

[15] J. Pennington, R. Socher, and C. Manning, “GloVe: Global vectors for word representation,” in EMNLP, A. Moschitti, B. Pang, and W. Daelemans, Eds. Doha, Qatar: Assoc. Comput. Linguistics, Oct. 2014, pp. 1532–1543.

[16] C. Rioux, S. A. Hasan, and Y. Chali, “Fear the REAPER: A system for automatic multi-document summarization with reinforcement learning,” in EMNLP, A. Moschitti, B. Pang, and W. Daelemans, Eds. Doha, Qatar: Assoc. Comput. Linguistics, Oct. 2014, pp. 681–690.

[17] Ulrich, J., Carenini, G., Murray, G., & Ng, R. (2009, March). Regression-based summarization of email conversations. In Proceedings of the international AAAI conference on web and social media (Vol. 3, No. 1, pp. 334-337).

[18] van Schaik, T. A., & Pugh, B. (2024, July). A Field Guide to Automatic Evaluation of LLM-Generated Summaries. In Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval (pp. 2832-2836).

[19] K. Wang, T. Liu, Z. Sui, and B. Chang, "Affinity-preserving random walk for multi-document summarization," in EMNLP, M. Palmer, R. Hwa, and S. Riedel, Eds. Copenhagen, Denmark: Assoc. Comput. Linguistics, Sep. 2017, pp. 210–220.

[20] J.-Y. Yeh, H.-R. Ke, W.-P. Yang, and I.-H. Meng, "Text summarization using a trainable summarizer and latent semantic analysis," Inf. Process. & Manage., vol. 41, no. 1, pp. 75–95, 2005, an Asian Digital Libraries Perspective.

[21] Zhang, J., Zhao, Y., Saleh, M., & Liu, P. J. (1912). Pegasus: pre-training with extracted gap-sentences for abstractive summarization (2019). arXiv preprint ArXiv:1912.08777.

[22] Zhang, M., Li, X., Yue, S., & Yang, L. (2020). An empirical study of TextRank for keyword extraction. IEEE access, 8, 178849-178858.

[23] Zhang, H., Yu, P. S., & Zhang, J. (2024). A Systematic Survey of Text Summarization: From Statistical Methods to Large Language Models. arXiv preprint arXiv:2406.11289.

[24] Zhang, Y., Ni, A., Yu, T., Zhang, R., Zhu, C., Deb, B., ... & Radev, D. (2021). An exploratory study on long dialogue summarization: What works and what's next. arXiv preprint arXiv:2109.04609.