A METICULOUS ANALYSIS OF EXTRACTIVE TEXT SUMMARIZATION APPROACHES AND EVALUATION METRICS FOR GENERATED SUMMARIES

Pravesh S. Patel[1]*, Dr. Paresh M. Solanki[2]

Abstract –
Text summarization's objective is to condense the original text into a concise document utilizing multiple semantics. It may also assist the reader in determining whether the original material is worth reading in its entirety. Having an abundance of information available on the internet for any given topic, creating a summary of the important details would be beneficial to many people. Text summarization approaches that implemented in a variety of ways. Text summarizing techniques can be categorized in different ways, such as the type of input (e.g. single or multi-document), output (e.g. extractive or abstractive), and purpose (e.g. generic, domain-specific, or query-based). This paper reviews extractive approaches for text summarization, focusing on the output results and sentence scoring techniques. It also provides an overview of the techniques and methods used by researchers for comparison and development. Finally, it reviews summary evaluation methods and provides suggestions for potential opportunities and challenges in text summarization research.

Keywords – Text Summarization, Natural Language Processing, Extractive Text Summarization, Sentence Scoring, Text Summary assessment

[1]*Faculty of Engineering and Technology Ganpat University, India
E-mail:- pravesh.patel@ganpatuniversity.ac.in
[2]Faculty of Engineering and Technology, Ganpat University, India
E-mail:- paresh.solanki@ganpatuniversity.ac.in

***Corresponding Author**: - Pravesh S. Patel
*Faculty of Engineering and Technology Ganpat University, India
E-mail:- pravesh.patel@ganpatuniversity.ac.in

DOI: - 10.31838/ecb/2023.12.si5.0117

I. INTRODUCTION

Nowadays, the quantity of textual documents generated and made available in electronic form has gradually risen over time, thanks to recent advancements in web-based applications[1]. The task of analysing extensive amounts of data to discover useful insights can be a huge challenge, and thus requires the utilization of automated processes to help manage the current data set. To communicate the key information in the original text, a good text summary technique should grasp the entire text, reorganise information, and create cohesive, useful, and impressively short summaries[2]. By filtering out unnecessary or repetitive material and choosing the most important sections of the text, these algorithms are able to create useful yet short summaries[3]. The difficulty posed by the increasing expansion of the internet and papers, as well as the limits of existing text summarising algorithms, are all covered in this review. Research in text summarisation has grown significantly in the fields of information retrieval and natural language processing, offering a variety of applications in data mining, web-based information retrieval, generating abstracts of technical papers, and generating highlights of news stories. When a text input is given, the summarized output is produced through pre-processing steps such as sentence segmentation, tokenization, stop word removal and word stemming. Stop words are common words like 'a', 'an', 'the', etc. that are eliminated, while word stemming removes suffixes and prefixes. After the pre-processing, each sentence is represented by a vector of feature attributes. In Fig.1 shows general model for text summarization[4].

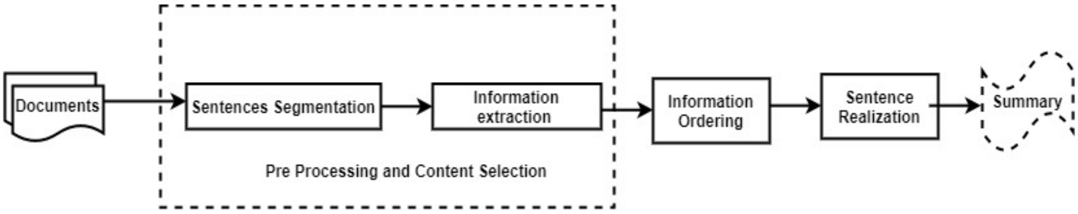

Fig.1 General model for text summarization

The text summarization process content three sections:

- Content Section: Select sentences based on sentence weigh to extract from the document
- Information Ordering: Select an order for sentences to appear in the generated summary.
- Sentence Realization: clean up the sentence.

II. TEXT SUMMARIZATION

Fig 2 demonstrates the various taxonomies of text summarization which are dependent on the frequency of the input sources, the goal of the summary, the type of summary produced, the language of the input sources, and the category. There are main two different groups of Text Summarization based on summary contents: Inductive and Informative. A summary of information generally only contains the main concept of the document(s), usually making up 5 to 10 percent of the text. On the contrary, an informative summarization system provides a more detailed overview of the text, usually 20 to 30 percent of the original text. Much of the research published about generated summaries is separated into two types: extractive and abstractive text summarization. Abstractive text summarization is a complicated task that requires a careful examination of the text in multiple stages, such as semantic analysis, lexical relationships, and named entity recognition. This process necessitates a thorough understanding of the meaning and connections of words, as well as the ability to interpret implications and generate sentences. As such, creating abstracts that serve as summaries has become challenging. Extractive text summarization relies on taking sentences from the source text and using sentence score to decide which ones to include. No condensing of the text takes place. Recently, researchers have been exploring a combined method of extractive and abstractive summarization. Depending on the scope of the source material, summarization can be classified as either single document or multi-document. When only one document is provided as an input for summarization, this can be referred to as single document text summarization. Conversely, when multiple documents are submitted to create a summary, this is referred to as multi-document summarization.

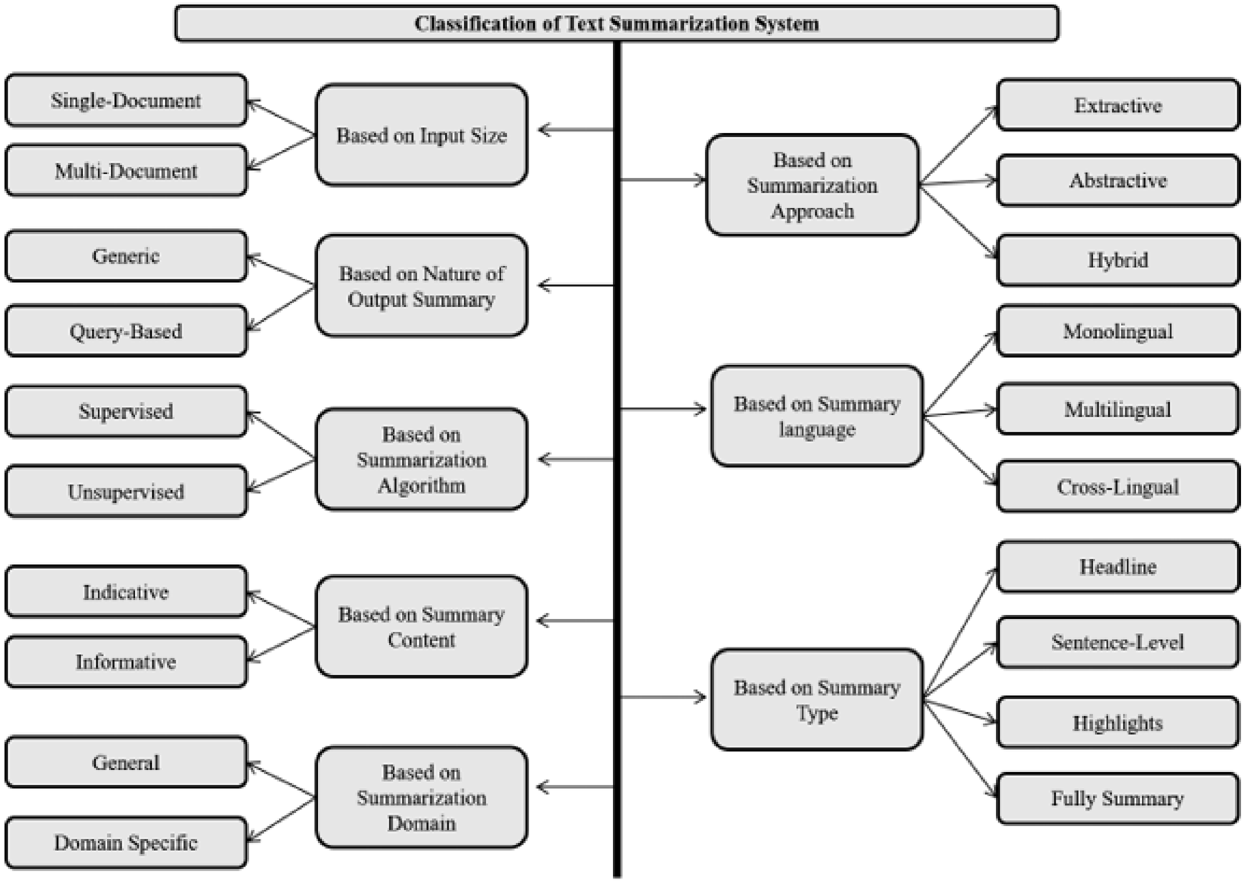

Fig 2 Taxonomies of Text Summarization

When it comes to summarization, domain specific techniques use information about a certain subject to generate a summary, whereas domain independent summarization relies on general features to identify the key points in text. In recent times, most researchers have shifted to domain specific summarization. Supervised text summarization algorithms need to be trained, which necessitates annotated training data. This data needs to be manually annotated, which is both laborious and costly. In contrast to supervised algorithms, unsupervised algorithms do not necessitate a training phase or training data. Text summarization can be classified into monolingual, multi-lingual, and cross-lingual summaries. A monolingual summary occurs when the source and target documents are written in the same language. A multi-lingual summary is generated in multiple languages (such as English, Arabic, and French) when the source text is composed in multiple languages. Finally, a cross-lingual summary is generated in a separate language (e.g. French or Arabic) when the source text is written in one language (e.g. English). Summarization techniques are implemented depending on the type of summary desired, such as Headlines, Sentence-Levels, Highlights, or Full Summaries. The length of the generated summaries is determined by the objective of the ATS system. For example, Headline generation typically produces a shorter result than a sentence.[5]. A sentence-level summarization generates a single sentence from the input text which is usually an abstractive sentence[5]. A condensed overview resulting in a report that is generally in the form of bullet points is created by a highlights summarization, which has a very brief and succinct style[6]. The highlights of the main points offers the reader a concise summary of the most important data in the source material[6]. In conclusion, the amount of words in the summary or the degree of compression is typically used as a guide for generating a comprehensive summary.

III. Extractive Text Summarization

Extractive summarizers analyse a text and extract phrases that best represent the message buried within it. The principle behind most extractive summarising approaches is to locate keywords and extract sentences with more keywords than the rest[7]. The most common method of keyword extraction is to extract relevant terms with a higher frequency than others. Fig.3 shows the diametric representation of extractive text summarization approach. In pre-processing phase, on single or multiple documents, the text content divided into paragraphs then sentences and then applied for word analysis, where its consists word stemming, stop word removal, tokenization, lemmatization and also compute the contribution of words[8][9].

Extractive text summary generated based on score of particular sentence, there are several features existing for sentence scoring[10][11][12] presented in Table 1. After sentence scoring, the rank of sentence calculated for entire text document(s) and based on compression ratio, the sentences are selected for final summary.

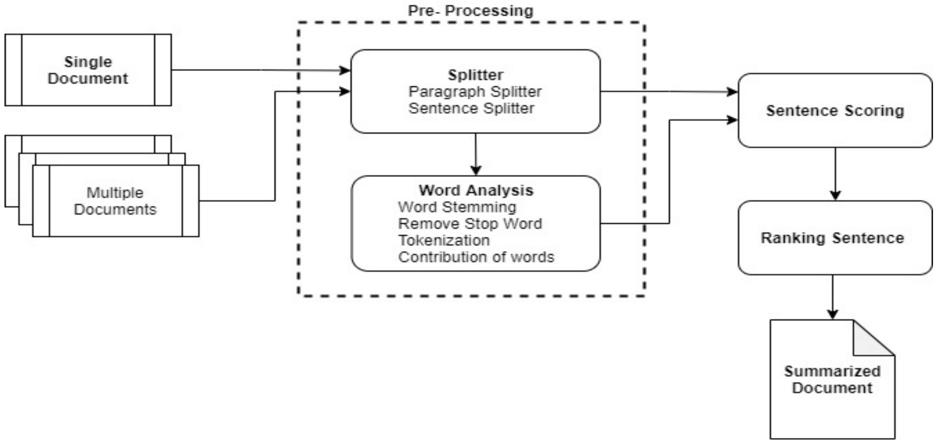

Fig 3 Extractive Text Summarization

Sentence Scoring

Sentence scoring is a method used to analyse the features of a sentence in order to determine its importance in the context of a larger text. In the research field of text summarization, researchers have addressed the question of how a system can assess the relevance of sentences in a given text. Table 1 outlines the various sentence scoring methods currently being utilized to assess the importance of particular statements.

Table 1 Sentence Scoring Methods

Sentence Scoring Features	Description
Title Word[13]:	Sentences containing terms that are the same as the title are also used to determine the document's subject matter. These sentences are more likely to be included in a summary.
Word frequency	The method of word frequency scoring is aptly named as it implies that words which appear more often in the document will receive a higher score. This means that sentences containing the most common words in the text have a better chance of being chosen as part of the final summary. It follows that words with a high frequency are likely to be related to the main theme of the document.
Content word [14](Keyword)	Content words, often known as keywords, are nouns. The word frequency - inverse document frequency - can be used to determine them (TF-IDF). Sentences with keywords have a better probability of being included in the summary.
Sentence Length:	In a summary, extremely long and very short sentences are not taken into account.
Sentence: position[4]	The first and/or last sentence of a text document's first and/or last paragraph are usually more essential and have a better probability of being included in the summary.
Upper-case word:	The summary includes sentences that use acronyms or proper names.
Proper Noun:	Sentences that contain proper nouns are more likely to be included in a summary than those without, which falls under the upper case Method. This could include a person's name, a place name, or the name of a concept, among other things.

Cue-Phrase:	Summaries are most likely to contain sentences that contain any cue phrase.
Biased Word	If a word in a sentence comes from a selective list of terms, the phrase is significant. The weighted word list is pre-determined and may include domain-specific terms.
Font based	Words printed in upper case, bold, italics, or underlined typefaces are regarded more essential in sentences.
Pronouns	Pronouns like "they, it, he, and she" can't be included in the summary unless they're extended into nouns that fit.
Presence of non-essential information	If a phrase begins with one of the words "because," "furthermore," or "additionally," it can be assumed that the phrase contains unnecessary information, and should be assigned a value of "true" or "1." However, if the phrase does not include any of these words, it should be labelled as "false" or "0."
Sentence-to-Sentence Cohesion[15]	The document's similarity between "s1" and each other sentence s is computed for each sentence. The raw value of this characteristic may be determined for a given sentence by adding all of the similarity values together. For each sentence, the process is repeated.
Sentence-to-Centroid Cohesion[14]	To calculate the centroid of a document, take the arithmetic average of all the coordinate values associated with the document. Then, calculate the similarity between the centroid and each phrase in the document. The raw value of this feature can then be retrieved for each sentence, and the process is repeated.
Discourse analysis[16]	One of the good features for text summarising is discourse level information in a text. In order to write a clear, confident summary.

IV. Extractive Text Summarization Approaches

Extractive text summarization is a machine learning classification issue in which the input text is broken down into sentences and each sentence is assessed as either suitable for summary or not. It may be trained in either supervised or unsupervised mode, depending on whether both source text and extractive summary are present in the input data. Manually built features such as TF-IDF, convolutional neural network, recurrent neural network, and recurrent with attention mechanism can be used to map words to vectors. Current state-of-the-art approaches for phrase encoding employ transformer-based models with a stack of self-attention layers, which outperform earlier methods substantially. Unsupervised extractive summarization is possible. Sentences are also mapped into vector space in this scenario, and a clustering technique is used to choose sentences that are closest to the cluster centre. Each cluster is meant to have sentences related to a specific topic. Finally, in the summary, a single sentence from each cluster is presented. The importance of sentences is established by examining their statistical and linguistic features[17][18]. Extractive summaries are produced by taking key pieces of text (sentences or passages) from the original text, using a statistical analysis of either individual or blended surface-level characteristics such as word/phrase frequency, position, or cue

words to determine the chosen segments[19]. The Table 2. presents an overview of several extractive text summarization techniques.

Statistical-Based Methods:
These techniques pull out relevant sentences and words from the source material based on a statistical analysis of a collection of characteristics. [20][21].

The sentence which is deemed to be the most important can be determined by factors such as its positioning, its frequency of use, and other like criteria.

Concept-Based Methods:
Methods are used to identify concepts contained in a text from outside sources of information[22] such as Word Net, How Net, Wikipedia, and so on

Topic-Based Methods:
This method is most popular strategies for illustrating the topic of a document are Term Frequency, Term Frequency-Inverse Document Frequency (TF-IDF), lexical chains, and topic word approaches. These involve creating a simple chart to represent the topic of the document, as well as assigning weights to the words that are

associated with the topic. All of these strategies are based on recognizing what the main subject of the document is [23].

Sentence Centrality or Clustering-Based Methods:

This multi-document summarizer technique involves finding the most pertinent and meaningful sentences in a group of documents that touch on the main topic[24]. The significance of the sentences is evaluated by looking at the centrality of the words they contain[25].

Graph-Based Methods:

These methods[26][22] involve constructing graphs based on sentences in order to represent a single document or a group of related documents. Using a graph to visualize the sentences from an input document, each node representing a sentence, and an edge connecting each pair of sentences with a weight equal to the cosine similarity of the two sentences, a ranking algorithm can be used to determine the importance of each sentence.

Semantic-Based Methods:

Latent Semantic Analysis (LSA) is a popularly applied semantic-driven extractive automation technique. It is an unguided process that expresses the sense of written words through the noted occurrences of words together. [27][28].

Machine-Learning-Based Methods:

This process transforms the issue of summarization into a supervised categorization issue concerning individual sentences. [29]. The system is taught to differentiate between summary and non-summary sentences in a test document, based on examples from a training set of documents that have been labelled with their human-generated summaries.

Deep-Learning-Based Methods:

In [30],Kobayashi et al. present a summarization system that utilizes document-level similarity, which is based on embedding. Words are given a representation of their meaning in the form of an embedding, and documents are seen as a collection of sentences, and sentences as a collection of words. This is formalized as a sub-modular function, which is the negative summation of the distances between the nearest neighbours of the embedding distributions, which is the set of word embedding that are in the document. The conclusion that was drawn was that document-level similarity can discover more intricate meanings than sentence-level similarity.[31].

Optimization-Based Methods:

In this method the summarization process is converted into an optimization problem. As an example, a general extractive multi-document ATS system can be seen as a multi-dimensional challenge[32][9].

Fuzzy-Logic-Based Methods:

The application of fuzzy logic in ATS is advantageous because it mimics the way humans think. This type of logic allows for the representation of sentence characteristics that cannot be accurately expressed using only the binary values of 0 and 1.[33]. The amalgamation of multiple techniques is advantageous for summarization because it leverages the merits and counteracts the drawbacks of each approach. Moreover, incorporating multiple characteristics likely leads to more precise weighting of the input sentences[25][34].

Table 2 Extractive text summarization Approaches

Approach	Documents	Description	Observations
TF-IDF[35]	Single	The number of sentences in the document that contain that phrase is used to determine sentence frequency in this method. The sentences with the highest scores are included in the summary.	Non-stop terms that occur most frequently in the content can be used as query words to produce a general summary.
Cluster Based[4]	Single /Multiple	First, sentences are grouped, after which sample sentences are constructed and retrieved depending on each cluster.	In the job of automated key word extraction, narrative text categorization is used.
Naïve Bayes [36]	Single	Using a naive-bayes classifier, the classification function classifies each sentence as worth extracting or not. Only the best sentences were removed after each sentence was given a score.	The best abstract was given by the position of the sentence, cue characteristics, sentence length, and sentence feature.
Decision Tree Based[37]	Single/ Multiple	This method takes into account the importance of a characteristic known as sentence position.	The effectiveness of the position approach was demonstrated by a high level of matching.

Hidden markov model[38]	Single	A hidden markov model(HMM) was used to extract a phrase from a document.	It's utilised to account for sentence dependencies on the local level.
Log Linear Model[12]	Single	Using log-linear models, it was possible to demonstrate that the summarizing system produced superior summaries.	The F-score was employed to evaluate the summaries generated. Criteria such as word combinations, phrase length, sentence location, and discourse features like in the introduction or conclusion were taken into account.
Neural Network based [39][9]	Single /Multiple	taught to examine the relevant aspects of sentences that can be picked in the article summary	the neural network is tweaked to integrate and generalize the key features evident in summary sentences. Finally, the updated neural network is applied as a filter to generate news article summaries.
Graph Theoretic [40] [13]	Single	To create a summary, the graph-based method is utilized to extract the most important sentences from the original content. This method is designed to take use of both the local and global characteristics of sentences.	The local property can be seen as a collection of significant words in each phrase, while the global property is the connection of all sentences in the text. These two features are then joined together to generate a single indicator of sentence relevance.
Latent semantic Analysis Based[41] [42]	Single /Multiple	Singular Value Decomposition (SVD) is a method which reveals the principal orthogonal dimensions of multidimensional data. It is also known as Latent Semantic Indexing (LSI) due to its ability to group documents that share similar meanings even when they do not contain the same words. This technique is also used in applications such as text summarization systems, Principal Component Analysis (PCA), etc.	SVD-derived techniques recognize sentence vectors with mutually perpendicular attributes, selecting a relevant sentence from each of these dimensions ensures that it accurately reflects the document and that it is not repetitive. It is noteworthy that this attribute only applies to data that naturally has major components.
Concept Obtained [43]	Single	Instead of using a word, it employs an idea as a feature. This method creates a preliminary summary using a theoretical vector space model, and then calculates the degree of semantic comparison of sentences to reduce repetition.	Calculate the significance of each sentence and reduce the duplication of summaries to get the final summary.
Fuzzy logic based[10] [34]	Single/Multiple	This approach to summarization employs a fuzzy system to analyse various characteristics of a text, such as sentence length, similarity to a key word, and so on. The generated rules are then stored in the system's knowledge base. Afterward, based on the sentence attributes and rules in the knowledge base, each sentence in the output is assigned a value between zero and one. Finally, the summary is generated by taking into consideration the value assigned to each sentence as a measure of its importance.	The de-fuzzier takes the output from the inference of the linguistic variables and, using the membership function, transforms them to clear numerical values that display the final score of the sentence.

Genetic Algorithm and regression model based [44]	Single	This method makes use of a trainable summarizer, which generates summaries by taking into account a variety of factors. Each sentence feature's impact on the summarizing job was examined. Then, to create an appropriate mix of feature weights, all of the characteristics were combined to train the Genetic Algorithm (GA) and Mathematical Regression (MR) models.	In order to generate a summarizer for each model, parameters of the features were applied to instruct the Feed Forward Neural Network (FFNN), Probabilistic Neural Network (PNN), and Gaussian Mixture Model (GMM).
Query – biased [45][46]	Single	The system incorporates the document structure, namely the sectional hierarchy, into the output summary. A query-biased approach is used to choose both the structural information and the content to be shown in the summary. Both in the summarizing process and in the output summaries, the system leverages structural and linguistic information from the documents.	For summarization, the system employs natural language processing techniques such as recognizing phrases as better content carriers than single words.
Lexical chain based[36] [47]	Single	The text summarization is done using linguistic analysis.	where semantically relevant sequences in the document were discovered, and numerous lexical chains that formed a representation of the original document were extracted
Ranking based Clustering [48]	Single	Instead of being considered as a sentence feature, a word is handled as a text object (which is self-contained).	The top rated sentence from the highest ranked theme cluster to the lowest ranking theme cluster is used to create summaries, followed by the second highest ranked sentences from theme clusters in decreasing order of their ranks, and so on.
Based on Word embedding [4][49][14]	Multiple	The framework was created by expanding a single document summarizing approach based on kp centrality. It involves two different strategies. I. To generate a final summary, a single layer method combines summaries from each input document. II. The waterfall method integrates summaries in a cascading pattern, based on the chronological order of documents.	Evaluation is performed using rouge-1 and user study
Graphsum [50]	Multiple	During the summarization process, a graph-based summarizer analyzes a set of documents and uses association rules to identify the connections between different words. Graph sum utilizes a procedure that can discriminate between positive and negative correlations between terms.	A Page Rank is used to rank the network nodes, which indicate combinations of two or more words. The sentence selection procedure is then carried out using the resulting node rating.

V. SUMMARY EVALUATION METHODS

Automatic summary generation is difficult because we don't know which parts of the information should contribute to the summary. The varying perspective of summary makes evaluating automatically generated summary, even from a trained human, more difficult. Someone may consider a particular point to be important, while others may consider it to be unimportant[13]. The purpose of the summary can aid in the evaluation of an automatically generated summary. As described in the survey paper, summary evaluation can be broadly classified as follows:

Extrinsic text summary evaluation is a method of measuring the quality of a summary of a text document. It is based on comparing the summary produced to a set of reference summaries created by humans[21]. The quality of the generated summary is measured by the degree to which it matches the reference summaries in terms of content and style. This type of evaluation is used in Natural Language Processing (NLP) and Information Retrieval (IR) applications to evaluate the performance of automatic summarization algorithms. It is also used to identify potential improvements that can be made to the algorithms[51].

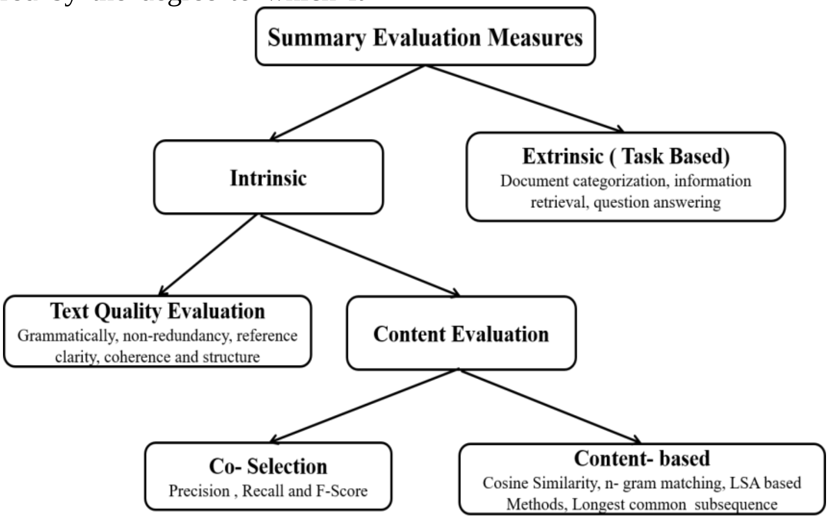

Fig. 4 Summary Evaluation Techniques[21]

Intrinsic text summary evaluation is the process of evaluating a summary of a text or document by comparing it to the original text. It is typically done by measuring the semantic similarity between the summary and the original text, or by using natural language processing (NLP) techniques to assess the accuracy of the summary in terms of its content and structure[21]. The most commonly used metrics for intrinsic text summary evaluation are ROUGE (Recall-Oriented Understudy for Gisting Evaluation) and BLEU (Bilingual Evaluation Understudy). These metrics measure how much the summary contains the content of the original text, and how well it conveys the meaning of the original text.

ROUGE is a set of metrics used to evaluate the quality of text summarization. It is often used to evaluate the performance of automatic summarization systems[52]. The ROUGE metrics compare the candidate summaries to the reference summaries and calculate the number of overlapping words and phrases. The metrics include ROUGE-N, ROUGE-L, and ROUGE-W, which measure the number of overlapping n-grams, the longest common sequence, and the weighted average of the overlapping score of n-grams, respectively.

BLEU is a method for evaluating the quality of a text summarization system, developed by researchers at IBM. It measures the similarity between a text summary and a reference text by comparing the n-grams (word sequences of length n) of the summary and the reference text[53]. The higher the BLEU score, the more similar the summary is to the reference text. BLEU is used in many areas of Natural Language Processing (NLP) and is a popular metric for text summarization because it is relatively easy to compute and is language-independent.

Precision and recall are two important metrics used to evaluate the performance of a text summarization system. Precision measures the proportion of summaries that are accurate, while recall measures the proportion of the relevant summaries that are retrieved. In the case of text summarization, the precision metric is usually calculated by dividing the number of correct summaries by the total number of summaries generated. The recall metric, on the other hand, is typically calculated by dividing the number of correct summaries by the total number of ground-truth summaries.

$$Rouge\ recall = \frac{no.of\ matching\ words\ in\ generated\ summary\ by\ system}{no.of\ words\ in\ golden\ summary}$$

$$Rouge\ precision = \frac{no.of\ matching\ words\ in\ generated\ summary\ by\ system}{no.of\ words\ in\ generated\ summary\ by\ system}$$

$$Rouge\ F1 - score = 2\frac{precision * recall}{precision + recall}$$

The F-score is a metric that combines the precision and recall metrics into a single score. It is calculated by taking the harmonic mean of the two scores, which is equal to the product of the two scores divided by their sum. Therefore, an F-score of 1 indicates perfect precision and recall, while an F-score of 0 indicates perfect inaccuracy.

VI. Future Score

Assessing the quality of summaries, be it manually or automatically, is no easy feat. The biggest issue in this evaluation is developing a standard against which the results of the systems being compared can be judged. It is also hard to ascertain what an accurate summary should look like, as a system may create a superior summary that differs from any human-made summary that is used to measure performance. Both extractive and abstractive approaches to text summarization have their own challenges. Extractive summarization compromises the readability of the text generated. The future of this research domain heavily depends on being able to identify effective methods of automatically evaluating the systems.

VII. CONCLUSION

Content summarization is an increasingly important part of NLP, as the amount of data available on the internet is huge. Exact data allows for faster, more efficient searches. Therefore, summarization is a necessity for businesses, government organizations, students, and teachers. This paper discusses both the extractive and abstractive approaches, as well as their performance. Content summarization is important for both businesses and research. Abstractive summarization is more complex, but produces better summaries than extractive summarization. Text summarization has many practical applications and its potential should be explored. This could be highly beneficial for everyday work.

References

1. I. Kadhim, "Survey on supervised machine learning techniques for automatic text classification," Artif. Intell. Rev., vol. 52, no. 1, pp. 273–292, 2019, doi: 10.1007/s10462-018-09677-1.
2. S. Biswas, R. Rautray, R. Dash, and R. Dash, "Text Summarization: A Review," in 2018 2nd International Conference on Data Science and Business Analytics (ICDSBA), 2018, pp. 231–235, doi: 10.1109/ICDSBA.2018.00048.
3. P. Widyassari et al., "Review of automatic text summarization techniques & methods," J. King Saud Univ. - Comput. Inf. Sci., May 2020, doi: 10.1016/J.JKSUCI.2020.05.006.
4. S. Lamsiyah, A. El Mahdaouy, B. Espinasse, and S. El Alaoui Ouatik, "An unsupervised method for extractive multi-document summarization based on centroid approach and sentence embeddings," Expert Syst. Appl., vol. 167, p. 114152, Apr. 2021, doi: 10.1016/ J.ESWA.2020.114152.
5. F. Dernoncourt, M. Ghassemi, and W. Chang, "A Repository of Corpora for Summarization," May 2018, [Online]. Available: https:// aclanthology. org/L18-1509.
6. K. Woodsend and M. Lapata, "Automatic Generation of Story Highlights," in Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics, Jul. 2010, pp. 565–574, [Online]. Available: https://aclanthology.org/P10-1058.
7. J. R. Thomas, S. K. Bharti, and K. S. Babu, "Automatic keyword extraction for text summarization in e-newspapers," ACM Int. Conf. Proceeding Ser., vol. 25-26-Augu, 2016, doi: 10.1145/2980258.2980442.
8. J. N. Madhuri and R. Ganesh Kumar, "Extractive Text Summarization Using Sentence Ranking," in 2019 International Conference on Data Science and Communication (IconDSC), 2019, pp. 1–3, doi: 10.1109/IconDSC.2019.8817040.
9. M. Tomer and M. Kumar, "Multi-document extractive text summarization based on firefly algorithm," J. King Saud Univ. - Comput. Inf. Sci., 2021, doi: https://doi.org/ 10.1016/ j.jksuci.2021.04.004.
10. F. Kyoomarsi, H. Khosravi, E. Eslami, P. K. Dehkordy, and A. Tajoddin, "Optimizing Text Summarization Based on Fuzzy Logic," in Seventh IEEE/ACIS International Conference

on Computer and Information Science (icis 2008), 2008, pp. 347–352, doi: 10.1109/ ICIS. 2008.46.

11. F. Chen, K. Han, and G. Chen, “An approach to sentence-selection-based text summari-zation,” in 2002 IEEE Region 10 Conference on Computers, Communications, Control and Power Engineering. TENCOM ’02. Proceedings., 2002, vol. 1, pp. 489–493 vol.1, doi: 10.1109/TENCON. 2002. 1181320.
12. M. Osborne, “Using maximum entropy for sentence extraction,” Proc. ACL-02 Work. Autom. Summ. 4 (ACL), pp. 1–8, 2002, doi: 10.3115/1118162.1118163.
13. Z. Xu et al., “Improving readability through extractive summarization for learners with reading difficulties,” Procedia Comput. Sci., vol. 21, no. 3, p. 102536, 2020, doi: https:// doi.org/10.1016/j.procs.2019.09.393.
14. S. Lamsiyah, A. El Mahdaouy, B. Espinasse, and S. El Alaoui Ouatik, “An unsupervised method for extractive multi-document summarization based on centroid approach and sentence embeddings,” Expert Syst. Appl., vol. 167, p. 114152, 2021, doi: https://doi.org/ 10. 1016/j.eswa.2020.114152.
15. T. M. Anh, N. T. My, and N. T. T. Trang, “Enhanced genetic algorithm for single document extractive summarization,” ACM Int. Conf. Proceeding Ser., pp. 370–376, 2019, doi: 10.1145/3368926.3369729.
16. R. Ferreira et al., “A four dimension graph model for automatic text summarization,” Proc. - 2013 IEEE/WIC/ACM Int. Conf. Web Intell. WI 2013, vol. 1, pp. 389–396, 2013, doi: 10.1109/WI-IAT.2013.55.
17. Harbola, “A Survey on Extractive Text Summarization Techniques,” 2018.
18. S. Saziyabegum and P. S., “Literature Review on Extractive Text Summarization Approaches,” Int. J. Comput. Appl., vol. 156, no. 12, pp. 28–36, 2016, doi: 10.5120/ ijca 2016912574.
19. W. Guan, I. Smetannikov, and M. Tianxing, “Survey on Automatic Text Summarization and Transformer Models Applicability,” ACM Int. Conf. Proceeding Ser., pp. 176–184, 2020, doi: 10.1145/ 343780 2. 3437832.
20. V. Gupta and G. S. Lehal, “A survey of text summarization extractive techniques,” J. Emerg. Technol. web Intell., vol. 2, no. 3, pp. 258–268, 2010.
21. M. Gambhir and V. Gupta, “Recent automatic text summarization techniques: a survey,” Artif. Intell. Rev., vol. 47, no. 1, pp. 1–66, 2017, doi: 10.1007/s10462-016-9475-9.
22. N. Moratanch and S. Chitrakala, “A survey on extractive text summarization,” in 2017 International Conference on Computer, Communication and Signal Processing (ICCCSP), 2017, pp. 1–6, doi: 10.1109/ ICC CSP.2017.7944061.
23. Nenkova and K. McKeown, “A Survey of Text Summarization Techniques,” in Mining Text Data, C. C. Aggarwal and C. Zhai, Eds. Boston, MA: Springer US, 2012, pp. 43–76.
24. P. Mehta and P. Majumder, “Effective aggregation of various summarization techniques,” Inf. Process. \& Manag., vol. 54, no. 2, pp. 145–158, 2018.
25. N. Nazari and M. A. Mahdavi, “A survey on automatic text summarization,” J. AI Data Min., vol. 7, no. 1, pp. 121–135, 2019.
26. W. Wali, B. Gargouri, and A. Ben Hamadou, “Enhancing the sentence similarity measure by semantic and syntactico-semantic knowledge,” Vietnam J. Comput. Sci., vol. 4, no. 1, pp. 51–60, 2017, doi: 10.1007/s40595-016-0080-2.
27. K. Al-Sabahi, Z. Zhang, J. Long, and K. Alwesabi, “An enhanced latent semantic analysis approach for arabic document summarization,” Arab. J. Sci. Eng., vol. 43, no. 12, pp. 8079–8094, 2018.
28. M. Mohamed and M. Oussalah, “SRL-ESA-TextSum: A text summarization approach based on semantic role labeling and explicit semantic analysis,” Inf. Process. \& Manag., vol. 56, no. 4, pp. 1356–1372, 2019.
29. N. Moratanch and S. Chitrakala, “A survey on abstractive text summarization,” in 2016 International Conference on Circuit, Power and Computing Technologies (ICCPCT), 2016, pp. 1–7, doi: 10.1109/ICCPCT. 2016. 7530193.
30. H. Kobayashi, M. Noguchi, and T. Yatsuka, “Summarization based on embedding distributions,” in Proceedings of the 2015 conference on empirical methods in natural language processing, 2015, pp. 1984–1989.
31. L. Chen and M. Le Nguyen, “Sentence selective neural extractive summarization with reinforcement learning,” in 2019 11th International Conference on Knowledge and Systems Engineering (KSE), 2019, pp. 1–5.
32. J. M. Sanchez-Gomez, M. A. Vega-Rodr\'\iguez, and C. J. Pérez, “Experimental analysis of multiple criteria for extractive multi-document text summarization,” Expert Syst. Appl., vol. 140, p. 112904, 2020.

A. Kumar and A. Sharma, “Systematic literature review of fuzzy logic based text summarization,” Iran. J. fuzzy Syst., vol. 16, no. 5, pp. 45–59, 2019.

33. Patel, S. Shah, and H. Chhinkaniwala, "Fuzzy logic based multi document summarization with improved sentence scoring and redundancy removal technique," Expert Syst. Appl., vol. 134, pp. 167–177, Nov. 2019, doi: 10.1016/J.ESWA.2019.05.045.
34. M. A. Ullah, S. M. Marium, S. A. Begum, and N. S. Dipa, "An algorithm and method for sentiment analysis using the text and emoticon," ICT Express, vol. 6, no. 4, pp. 357–360, Dec. 2020, doi: 10.1016/ j.icte. 2020.07.003.
35. N. Ramanujam and M. Kaliappan, "An Automatic Multidocument Text Summarization Approach Based on Naïve Bayesian Classifier Using Timestamp Strategy," Sci. World J., vol. 2016, p. 1784827, 2016, doi: 10.1155/2016/1784827.
36. C.-Y. Lin, "Training a Selection Function for Extraction," in Proceedings of the Eighth International Conference on Information and Knowledge Management, 1999, pp. 55–62, doi: 10.1145/319950.319957.
37. J. M. Conroy and D. P. O'leary, "Text Summarization via Hidden Markov Models," in Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, 2001, pp. 406–407, doi: 10.1145/ 383952.384042.
38. T. Shi, Y. Keneshloo, N. Ramakrishnan, and C. K. Reddy, "Neural Abstractive Text Summarization with Sequence-to-Sequence Models," ACM/IMS Trans. Data Sci., vol. 2, no. 1, pp. 1–37, 2021, doi: 10.1145/3419106.
39. A.-V. Luong, N.-T. Tran, V.-G. Ung, and M.-Q. Nghiem, "Word Graph-Based Multi-sentence Compression: Re-ranking Candidates Using Frequent Words," in 2015 Seventh International Conference on Knowledge and Systems Engineering (KSE), 2015, pp. 55–60, doi: 10.1109/KSE.2015.65.
40. M. I. Yeh JY, Ke HR, Yang WP, "ext summarization using a trainable summarizer and latent semantic analysis," Inf. Process. Manag. 41(1), pp. 75–95, 2005.
41. R. Rani and D. K. Lobiyal, "An extractive text summarization approach using tagged-LDA based topic modeling," Multimed. Tools Appl., vol. 80, no. 3, pp. 3275–3305, 2021, doi: 10.1007/s11042-020-09549-3.
42. Zamanifar, B. Minaei-Bidgoli, and M. Sharifi, "A New Hybrid Farsi Text Summarization Technique Based on Term Co-Occurrence and Conceptual Property of the Text," in 2008 Ninth ACIS International Conference on Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing, 2008, pp. 635–639, doi: 10.1109/ SNPD.2008.57.
43. M. A. Fattah and F. Ren, "Automatic Text Summarization," World Acad. Sci. Eng. Technol. Int. J. Comput. Electr. Autom. Control Inf. Eng., vol. 2, pp. 90–93, 2008.
44. C. Pembe, "Automated Query-biased and Structure-preserving Text Summarization on Web Documents," 2007.
45. M. Afsharizadeh, H. Ebrahimpour-Komleh, and A. Bagheri, "Query-oriented text summarization using sentence extraction technique," in 2018 4th International Conference on Web Research (ICWR), 2018, pp. 128–132, doi: 10.1109/ICWR. 2018. 8387 248.
46. Chowanda, R. Sutoyo, Meiliana, and S. Tanachutiwat, "Exploring Text-based Emotions Recognition Machine Learning Techniques on Social Media Conversation," in Procedia Computer Science, Jan. 2021, vol. 179, pp. 821–828, doi: 10.1016/j.procs. 2021. 01.099.
47. L. Yang, X. Cai, Y. Zhang, and P. Shi, "Enhancing sentence-level clustering with ranking-based clustering framework for theme-based summarization," Inf. Sci. (Ny)., vol. 260, pp. 37–50, 2014, doi: https://doi.org/ 10.1016/j.ins.2013.11.026.
48. L. Marujo et al., "Exploring Events and Distributed Representations of Text in Multi-Document Summarization," Know.-Based Syst., vol. 94, no. C, pp. 33–42, Feb. 2016, doi: 10.1016/j.knosys.2015.11.005.
49. Baralis, L. Cagliero, N. A. Mahoto, and A. Fiori, "GraphSum: Discovering correlations among multiple terms for graph-based summarization," Inf. Sci., vol. 249, pp. 96–109, 2013.
50. M. Mojrian and S. A. Mirroshandel, "A novel extractive multi-document text summarization system using quantum-inspired genetic algorithm: MTSQISGA," Expert Syst. Appl., vol. 171, p. 114555, 2021, doi: https:// doi. org/10.1016/j.eswa.2020.114555.
51. C.-Y. Lin, "ROUGE: A Package for Automatic Evaluation of Summaries," 2004.
52. K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "Bleu: a method for automatic evaluation of machine translation," in Proceedings of the 40th annual meeting of the Association for Computational Linguistics, 2002, pp. 311–318.