Nanotechnology Perceptions
ISSN 1660-6795
www.nano-ntp.com

Text Summarization: Exploring Classical, Machine Learning, and Deep Learning Models

Dr. Aniket K. Shahade[1], Dr. Priyanka V. Deshmukh[1], Disha Sushant Wankhede[2], Amruta Vikas Patil[3], Dr. Nitin N. Sakhare[2], Pritam H. Gohatre[4]

[1]*Symbiosis Institute of Technology, Pune Campus, Symbiosis International (Deemed University), Pune, India*
[2]*Vishwakarma Institute of Information Technology, India*
[3]*Research Scholar, SJJTU, India*
[4]*Research Scholar, VNIT, India*
E-mail: aniket.shahade11@gmail.com

The exponential growth of textual data across digital platforms necessitates efficient summarization techniques to extract relevant information and present it concisely. Text summarization, an essential task in Natural Language Processing (NLP), aims to generate coherent and meaningful summaries while preserving core information from the source text. This paper provides a comprehensive review of state-of-the-art text summarization models, categorizing them into extractive and abstractive approaches. Extractive models compile significant sentences directly from the original text, while abstractive models generate summaries by interpreting and rephrasing content.
The paper reviews some of the developments of such models, starting with historical roots such as statistical models and moving up to the modern intelligent machine learning and deep learning models. Main techniques, such as TF-IDF, LSA, and Transformer-based models, are described along with the explanation of their main concepts. It looks at the difficulties that come with each of them; for example, cohesiveness, pertinency, and working with language subtleties. The paper also reviews several big data and key metrics used in the evaluation of the summarization models, thereby pointing out trends and developments in text summarization, in relation to the hybrid models and pre-trained language models. It provides the required knowledge of the text summarization techniques, progress and issues to the researchers and practitioners to steer the future work.

Keywords: Text Summarization, Natural Language Processing, Extractive Summarization, Abstractive Summarization, Machine Learning, Deep Learning.

1. Introduction

1.1 Background

A massive increase in the creation and distribution of content in the last few years means that the current world is drowning in textual information published on news websites, social media platforms, academic databases, and business archives. Such a spurt in information calls for effective approach of cross-extracting and summarizing essential content so that the end-users are able to obtain the relevant information without much hassle of going through huge volumes of information. Text summarization, an important subtask of Natural Language Processing (NLP), solves this problem by providing summaries of the text documents keeping only the necessary information and discarding the rest (Nenkova & McKeown, 2012).

Text summarization can be broadly categorized into two main approaches: In the process of text summarization, the two most common methods include extracting and abstracting. The extractive type of summarization entails identifying useful individual sentences or phrases from the original text and joining them to create a summary (Gupta & Lehal, 2010). This approach works based on the feature selection that is based on certain factors like term frequency or sentence importance. On the other hand, abstractive summarization creates new summaries in the form of complete sentences which means that it involves the generation of new text that contains the main information of the source text; it is usually more complex than extractive summarization as it demands more comprehension of language and content (See et al., 2017). This method copies the abstracting skills of a human being in that the generated summary may contain paraphrased or synthesized information.

1.2 Historical Evolution of Text Summarization

The generation of text summarization techniques has gone through many changes in the last couple of decades. The initial strategies focused mainly on heuristic and statistical strategies. For example, methods like Term Frequency-Inverse Document Frequency (TF-IDF) and Latent Semantic Analysis (LSA) were used to locate and extract the most important sentences based on the statistical measures (Mihalcea & Tarau, 2004). While these methods were quite beneficial in some instances, the problem of cohesion and focus frequently reared its head in the summarization output.

Before machine learning, new and advanced techniques for text summarization came up. Naïve Bayes and Support Vector Machines (SVM) were also used to enhance the performance of extractive summarization since supervised learning was applied in this case (Nallapati et al., 2016). However, these models entailed feature engineering and large annotated data sets to work and were not very scalable.

1.3. Rise of Deep Learning in Summarization

Deep learning marked a tremendous development in the process of text summarization. RNNs, LSTMs and CNNs started to emerge as new and more effective methods for working with sequences and context (Cheng & Lapata, 2016). These models made it possible to produce summaries that are more sensitive to the context of the original text, at a considerable cost in terms of computational complexity.

In recent years, the Transformer framework, including BERT and GPT, has become the new generation of text summarization. These models use attention mechanism to capture long-range dependencies and context that are useful in extractive and abstractive summarization with high accuracy and natural language generation (Liu & Lapata, 2019). Transformers have raised the bar in this regard by showcasing their prowess in most of the summarization tasks and datasets.

2. Types of Text Summarization

Text summarization can be broadly categorized into two main types: The two main types of summarization are; The extractive summarization and the abstractive summarization. The two are in some way similar in that both are used to re-write long texts into compact forms but they are in many ways different in terms of the way they work and the results they produce.

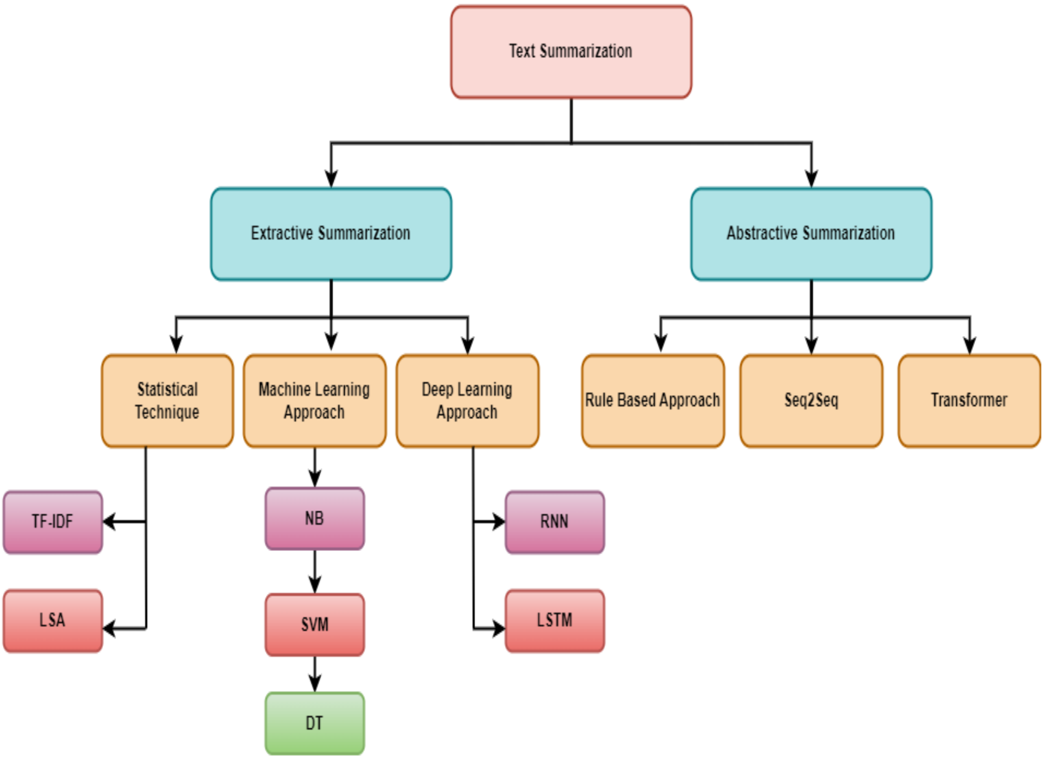

Figure 1. Types of Summarization Techniques

2.1 Extractive Summarization

In extractive summarization, what is summarized are individual sentences, phrases or segments of the source text that are deemed significant. It assigns a score to the textual units according to their importance and relation to the main content, and then selects these units to construct a summary.

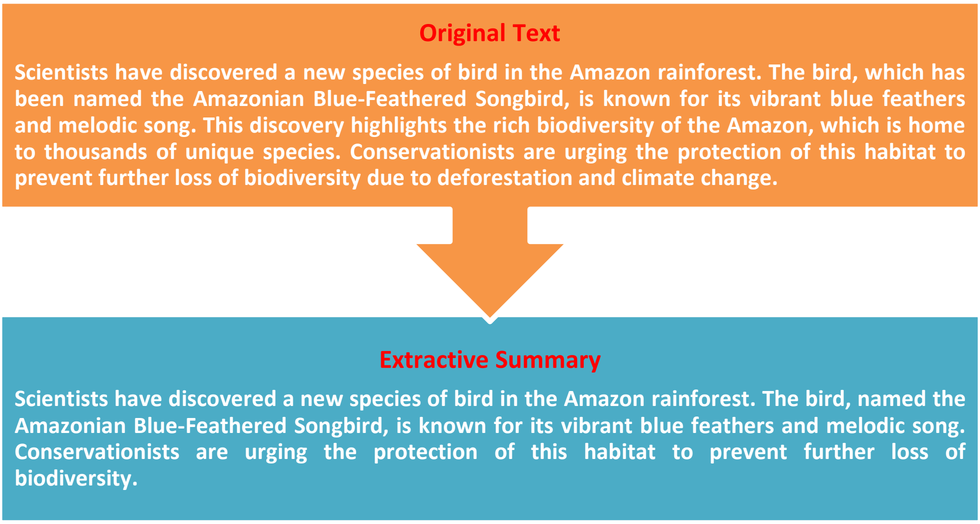

Figure 2. Extractive Summarization

2.1.1 Methodology

The methodology behind extractive summarization can be divided into traditional statistical techniques, machine learning-based approaches, and modern deep learning models:

- Statistical Techniques: The first approaches relied on basic statistical analysis to find out the important sentences. For example, Term Frequency-Inverse Document Frequency (TF-IDF) measures the significance of a word in a given document with regard to a collection of documents; thus, it can assist in the identification of the most important sentences that contain words with high TF-IDF score (Salton & McGill, 1986). Latent Semantic Analysis (LSA) decreases the dimension of the term-document matrix and performs singular value decomposition to obtain the relationship between terms and documents (Gong & Liu, 2001).

- Machine Learning Approaches: Since the introduction of machine learning, extractive summarization methods advanced to include supervised learning techniques. Naive Bayes classifiers, Support Vector Machines (SVM), and decision trees have also been employed for the classification of the sentences depending on the features such as position of the sentence, length of the sentence and term frequencies (Kupiec, Murray, Renals & Carletta, 2005). Such models are learnt from annotated data in which the sentences are tagged based on their relevance.

- Deep Learning Models: New developments in deep learning have enhanced extractive summarization in the following ways. Recurrent Neural Networks (RNNs) and their derivatives like Long Short-Term Memory (LSTM) networks, capture the sequence of sentences and estimate the probability of each sentence being in the summary (Nallapati et al., 2017). Convolutional Neural Networks (CNNs) process the text as a signal and then convolve the signal with filters to extract local features; thus, it is effective in identifying important features within the sentences (Cao et al., 2015). Recently, transformer-based models such as BERT (Bidirectional Encoder Representations from Transformers) have achieved state of the art results using attention to capture bidirectional context and generate good quality sentence embeddings (Liu & Lapata, 2019).

2.1.2 Applications and Challenges

Extractive summarization is applied in numerous fields because of its efficiency and ease of implementation. For instance, it is applied in news summarization to produce brief news summaries which can help the readers to easily understand the most important aspects of the article (Nallapati et al., 2016). While used in the legal and medical professions, summarization tools assist the professionals in a way that they extract essential information from long documents, thus enabling them to make decisions in the shortest time possible (Hennig et al. , 2015). Extractive summarization is also employed by businesses to review customers' feedback and comments, extracting major comments that represent overall tone (Hu & Liu, 2004).

However, extractive summarization has some problems. The idea of keeping the extracted summaries coherent and fluent is challenging as it is literally impossible to just string together the high scoring sentences. In turn, extractive methods are not capable of identifying the underlying contextual meaning or the additional information that is often implicit in the text, thus providing summaries that are not very detailed.

2.2 Abstractive Summarization

Abstractive summarization entails the creation of entirely new sentences that will represent the summary of the entire text. This approach involves the translation of the content, which sometimes may involve paraphrasing, and therefore one must have a good understanding of language and semantics. Abstractive methods are designed to generate more coherent and fluent summaries as compared to the summary generated by an actual human.

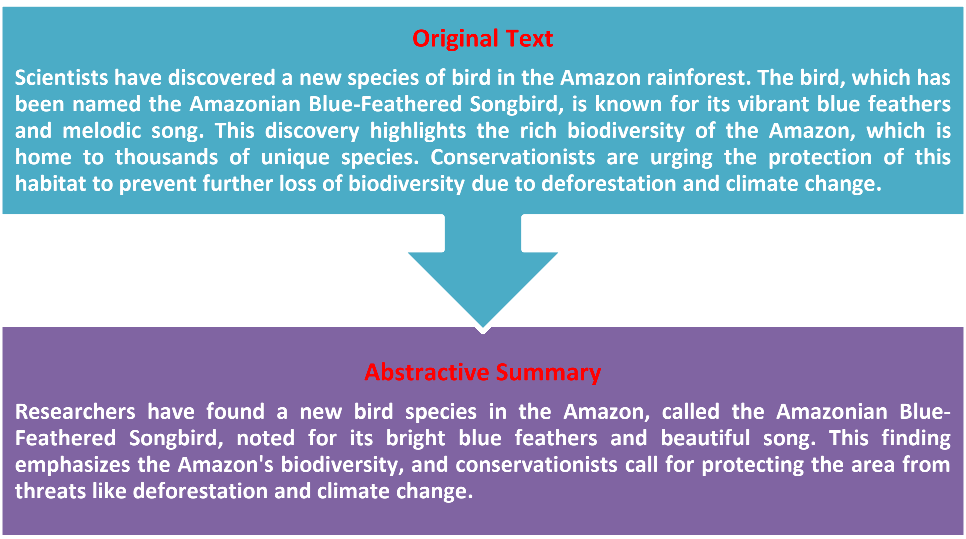

Figure 3. Abstractive Summarization

2.2.1 Methodology

The methodology behind abstractive summarization has evolved from rule-based systems to advanced deep learning models:

- Rule-based Systems: The first approaches used in the production of summaries were based on the rules and templates. These systems employed the linguistic knowledge and syntactic rules for paraphrasing and summarizing the text (Jing & McKeown, 2000). Rule based systems although were successful in certain domains but they had a major drawback of not being able to adapt to different types of text.

- Sequence-to-Sequence Models: The Seq2Seq models paved way for the next level of abstractive summarization. Seq2Seq models used often involve RNNs or LSTMs and are capable of generating new sentences as the output. Attention mechanisms were later incorporated to enhance the ability to pay attention to important sections of the input text while generating (Bahdanau, Cho, & Bengio, 2015).

- Transformer-based Models: The Transformer architectures like the Transformer model by Vaswani et al. (2017) brought a drastic change in the abstractive summarization. Self-attention mechanisms are used in transformers to capture long-range dependencies and contextual information. Several models such as BERT and GPT have been fine tuned for summarization tasks and have been reported to perform better as compared to several other models (Lewis et al., 2020; Zhang et al., 2019).

These models create summaries that are of high quality because they comprehend the content and then rewrite it in fluent natural language.

2.2.2 Applications and Challenges

Thus, abstractive summarization has a lot of uses in different fields. In journalism, it is used to develop brief summaries of news and headlines, which gives the readers a summary of the highlights (Nallapati et al., 2016). In academia, the summarization aids the researchers in preparing abstracts for long papers, thus helping in the process of literature reviews and sharing of knowledge (Cohan et al., 2018). Abstractive methods are also used in customer service to produce answers based on a large amount of query information to improve the efficiency of support systems (Radford et al., 2019).

However, abstractive summarization has some challenges as explained below. Creating summaries that are grammatically correct and semantically meaningful is challenging, and that is why language understanding is important. Another major concern is to keep the text coherent and unaltered in meaning while rephrasing the content. Furthermore, the assessment of abstractive summaries is challenging as the commonly used metrics such as ROUGE may not effectively measure the quality and informativeness of the generated text.

Extractive and abstractive summarization methods have their benefits and drawbacks, and the choice depends on the specifics of the task. On the other hand, extractive methods are easier to implement and are less complex as compared to the other methods since they may not be as comprehensive and well-structured. While abstractive methods are more human-like in terms of the summaries they create, they are less accurate and fluent and need complex models. It is important to know these types and their methodologies for the further progress of the field and the creation of suitable summarization instruments in NLP.

3. Classical Approaches

Some of the current sophisticated methods of summarizing texts can be attributed to the classical approaches to the process. These methods are mainly statistical methods and graph-based methods which provide different approaches to extract important information from a text.

3.1 Statistical Methods

Mathematical and statistical approaches are used in statistical methods to determine the significance of words, sentences, and phrases in the document. Two of the most common statistical models that are applied in extractive summarization are the TF-IDF and LSA.

3.1.1 TF-IDF

Term Frequency-Inverse Document Frequency (TF-IDF) is a quantitative method used to determine the relevance of a word in a document with regards to a group of documents (Salton & McGill, 1986). The importance rises with the number of occurrences of the word in the document but descends with the frequency of the word in the whole collection. This assists in finding out words that are quite relevant in the context of the document while not frequently used in other documents.

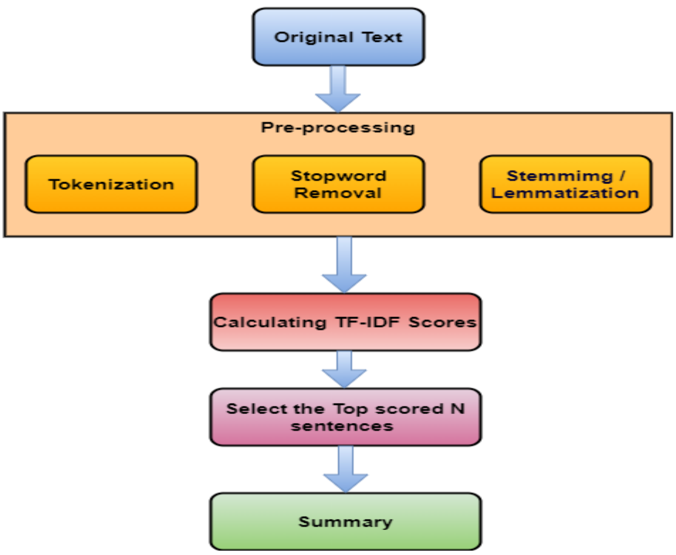

Figure 4. TF-IDF Summarization process

- Term Frequency (TF): This calculates the rate of occurrence of a term in a document. Thus, the higher the TF value, the more often the given term appears in the document.

$$\mathrm{TF}(t,d) = \frac{\text{Number of times term t appears in document d}}{\text{Total number of terms in document d}}$$

- Inverse Document Frequency (IDF): This evaluates the significance of the term in the whole document. This lowers the importance attached to the terms that are commonplace in many documents and raises the importance of the terms that are scarce.

$$\mathrm{IDF}(t,D) = \log\left(\frac{\text{Total number of documents}}{\text{Total number of documents containing term t}}\right)$$

- TF-IDF Calculation: TF IDF score of a term is calculated as the product of Term Frequency and Inverse Document Frequency.

$$\mathrm{TF} - \mathrm{IDF}(t,d,D) = \mathrm{TF}(t,d) * \mathrm{IDF}(t,D)$$

Due to its simplicity as well as efficiency in identifying significant words and phrases in the document, TF-IDF has been popularly employed in information retrieval and text mining.

3.1.2 Latent Semantic Analysis (LSA)

Latent Semantic Analysis (LSA) is a technique that employs singular value decomposition (SVD) of the term- document matrix to obtain a reduced rank matrix that best preserves the relationships between terms and documents (Gong & Liu, 2001). In LSA, the relationships between terms and concepts are examined to distinguish between the actual associations.

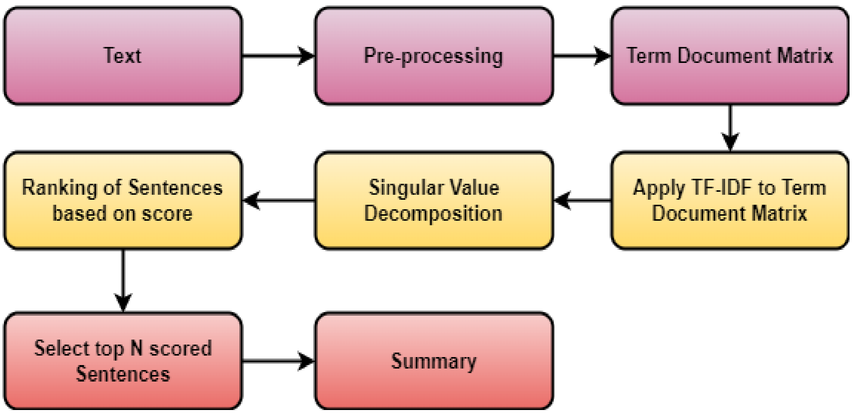

Figure 5. Summarization using LSA

- Term-Document Matrix: This is a matrix in which rows correspond to the terms and columns correspond to documents. Every cell holds the count of the term in the document.

- Singular Value Decomposition (SVD): This breaks the term-document matrix into three matrices known as U, $\sum$ and V^T.

The matrix $\sum$ contains singular values which are indicative of the importance of the various dimensions.

$$A = U\sum V^T$$

Because of the ability to truncate the ∑ matrix and certain columns of U and VT, LSA eliminates noise and focuses on the strength of the relationships, which allows to obtain semantically important sentences (Landauer et al., 1998).

Text summarization is one of the most areas in which LSA has been successfully applied, especially in the extraction of materials from large document collections (Gong & Liu, 2001)

LSA helps to cut down noise and stress on the most important co-occurrences, which allows to extract semantically meaningful sentences (Landauer et al., 1998).

LSA has been used in many text summarization applications especially in extraction of relevant information from large documents (Gong & Liu, 2001).

3.2 Graph-Based Methods

Graph-based methods utilize the theory of graphs in which a text is constructed as a graph to locate the significant sentences. Nodes are elements of a sentence, and edges refer to the connection between the sentences depending on their contents. With reference to graph-based methods, two of them are TextRank and LexRank.

3.2.1 TextRank

TextRank is another rank-based algorithm developed from PageRank that is unsupervised and based on a graph (Mihalcea & Tarau, 2004). It assigns a rank to a sentence in a text based on how much relevant it is and how it is related to other sentences.

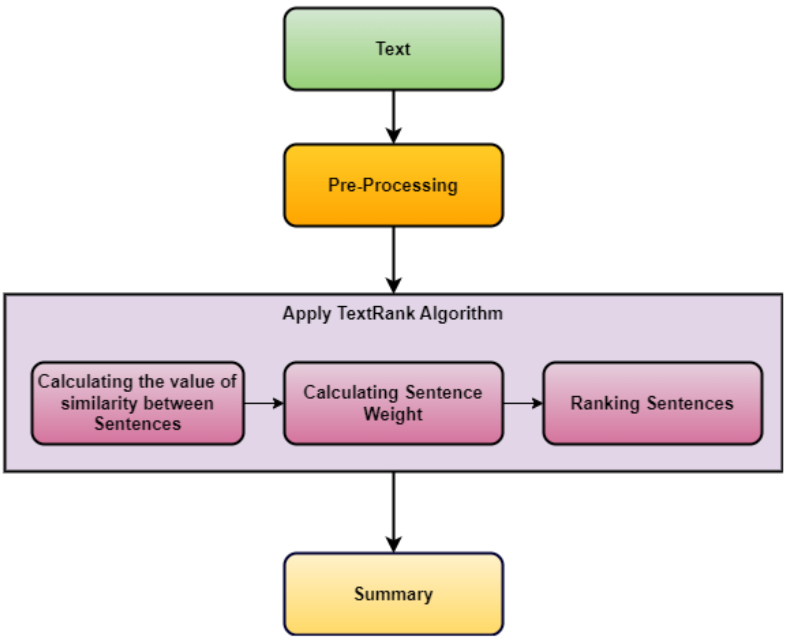

Figure 6. Summarization suing TextRank

- Graph Construction: In the case of sentences, the nodes are the sentences and the edges between the nodes are created based on the similarity of the sentences commonly computed using cosine similarity of the TF-IDF vectors.

- Ranking Algorithm: The algorithm of the TextRank used for calculating the importance score of the particular sentence, which depends on the importance of the connected sentences. The degree values obtained at the final step of the process are used to rank the sentences and the higher value signifies the higher importance of the information.

$$S(Vi) = (1 - d) + d \times \sum_{V_j \mathcal{E} adj(Vi)} \frac{S(Vj)}{L(Vj)}$$

Where S(Vi) is the score of sentence i, d is a damping factor (usually set to 0.85), adj(Vi) are the adjacent sentences to i, and L(Vj) is the number of edges from sentence j.

TextRank is useful for the selection of the most important sentences for summarizing and it was successfully applied because of its simplicity and stability (Mihalcea & Tarau, 2004).

3.2.2 LexRank

Another graph based method is LexRank that ranks the sentences according to their importance in the text using the eigenvector centrality (Erkan & Radev, 2004). Thus, LexRank differs from TextRank by taking into account the connectivity of the sentence graph and the relative importance of the sentences.

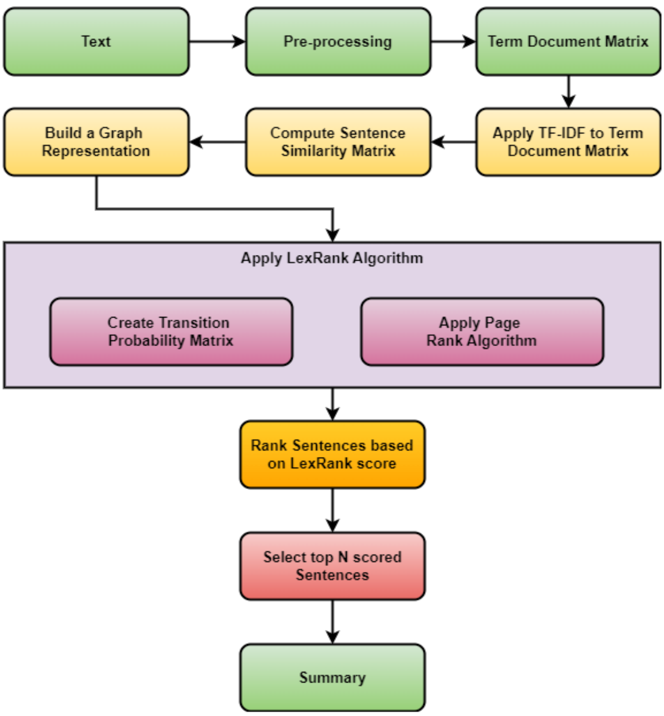

Figure 7. Summarization using LexRank

- Graph Construction: Like TextRank, LexRank also creates a graph with the nodes as sentences and edges between the sentences based on the content similarity.

- Centrality Measure: LexRank determines the centrality of a node by Eigenvector centrality formula which not only considers first neighbors of the sentence but also the second neighbors or the importance of the nodes to which the sentence is connected.

$$\mathrm{C(Vi)} = \frac{1}{\mathrm{N}} + \sum_{\mathrm{V_i \varepsilon adj(V_i)}} \frac{\mathrm{C(Vj)}}{\mathrm{d(Vj)}}$$

where C(Vi) is the centrality score of sentences i, N is the total number of sentences, and d(Vj) is the degree of sentence j.

It has been applied mostly in multi-document summarization case and it has been proved that LexRank can generate clear and meaningful summaries (Erkan & Radev, 2004).

Statistical and graph-based methods are the traditional methods that have given basic approaches to determining important information in texts. While the type of TF-IDF and LSA methods are based on word and sentence importance by frequency and semantic relationship between words, the method based on the graph like TextRank and LexRank takes into account the structure and connectivity of the sentences. Acquaintance with these classical methods is crucial for the formation of more progressive and efficient strategies of summarization.

4. Machine Learning Approaches

Supervised and unsupervised learning methods are used in the application of ML in text summarization in order to produce summaries independently. These approaches involve the use of labeled and unlabeled data for learning and information extraction as well as information condensation from the text.

4.1 Supervised Learning Models

Training based on supervised learning in text summarization meant for developing a function that maps the input text with the output summary using the training data set. Two of the most used supervised learning methods are Support Vector Machines (SVM) and Random Forests.

4.1.1 Support Vector Machines (SVM)

Support Vector Machines abbreviated as SVM is a supervised learning algorithm used for classification and for regression, and text summarization. SVMs have been designed to find the hyperplane that could effectively classify the classes of data points as described by Cortes and Vapnik (1995).

- Text Summarization: In text summarization, SVMs are used to learn from features extracted from texts including tf-idf scores or word embeddings to predict the importance relevance of the sentences in summarization (Murray et al., 2005).

- Feature Extraction: In text categorization, SVMs depend on proper feature extraction methods to put the textual data into the appropriate format for classification. The features could be n-grams, syntactic features or semantic embeddings as described by Joachims (1998).

- Advantages: SVMs also perform well when dealing with large number of features and can easily be trained on the text summarization data sets since they capture the input-output mapping of the problem domain well.

4.1.2 Random Forests

Random Forests is another type of learning algorithm that creates many decision trees at the time of learning and delivers the mode of the classes or mean estimate of the trees that is created during the learning process (Breiman, 2001).

- Text Summarization: In text summarization, Random Forests can be applied to sorting the sentences by their score or relevance for extractive summaries. In decision tree for each sentence in the text, a score is produced and all the scores are summed up to give the final summary (Murray et al., 2005).

- Ensemble Learning: Random Forests reduce the problem of overfitting and enhance the model's stability through averaging the results of a number of decision trees. Every decision tree is built separately from a different portion of the data, thus increasing the variety and decreasing the variance.

- Advantages: They have good performance even when the training data contains noise and are applicable for large numbers of features. Due to their ability to identify complicated dependencies between input variables and output variables, they can be used to summarize various kinds of texts.

4.2 Unsupervised Learning Models

The unsupervised learning models in text summarization work without using any training data and in fact, they try to learn inherent structures in the text. A widely used group of methods can be referred to as clustering-based techniques.

4.2.1 Clustering-based Methods

In clustering techniques similar sentences or documents are clustered together depending on the content of the documents. These methods work decomposing the text into related groups of sentences and selecting the representative ones or centroids as summary candidates (Blei et al., 2003).

- Text Summarization: In text summarization, the clustering methods like K-means clustering or hierarchical clustering are used to group the sentences based on similarity measure like TF-IDF cosine similarity or semantic embeddings.

- Centroid Extraction: After clustering, centroid or a best representative sentence from cluster is chosen as the candidate summary. These sentences are the main ideas or subject matters of the text, which gives a summary of the original content of the text. (Blei et al., 2003).

- Advantages: The clustering-based methods are cheap in terms of time and computational resources, and are also non-parametric. Summarization of large amount of text can be done quickly and the method is relatively insensitive to the length and organization of documents.

5. Deep Learning Approaches

Recent studies have made great progress in text summarization with the help of deep learning techniques based on neural network models that can capture complex relationships and dependencies in the text. This section explores three key deep learning paradigms used in text summarization: Recurrent Neural Networks (RNNs), Encoder-Decoder Architectures also known as Sequence-to-Sequence (Seq2Seq) Models, and Attention Mechanisms.

5.1 Recurrent Neural Networks (RNNs)

Recurrent Neural Networks (RNNs) are a type of neural network mostly used in dealing with sequential data making it ideal for use in text summarization.

5.1.1 LSTM and GRU-based Models

LSTM and GRU are two types of RNNs that have been improved to minimize the vanishing gradient problem in long sequences.

- LSTM: LSTM networks have memory cells that enable them to keep information from one sequence to the other hence making it easier for them to summarize texts in as much as they retain relevant information as they go through the texts (Hochreiter & Schmidhuber, 1997).

- GRU: While less complex than LSTMs, GRUs have proved to be almost as effective in sequence modelling tasks. They employ update and reset gates to regulate the flow of information in the network as stated by Cho et al., 2014.

- Text Summarization: In extractive and abstractive summarization, LSTM & GRU based models were exposed where they can learn to generate summaries from sequential input data and predict the important information (Nallapati et al., 2016).

5.2 Sequence-to-Sequence (Seq2Seq) Models

Seq2Seq models are the encoder-decoder type of models and can be used where input sequence is transformed into the output sequence and hence, the application of Seq2Seq can be seen in text summarization.

5.2.1 Encoder-Decoder Architecture

The Encoder-Decoder architecture comprises two main components: an encoder which maps the input sequence into a fixed size vector, and a decoder that maps this vector into an output sequence (Sutskever et al., 2014).

- Text Summarization: Seq2Seq models have been used in abstractive summarization where the encoder takes the input text and the decoder that generates the summary learns the probability of each word in the output sequence (Nallapati et al., 2016).

- Advancements: New modifications such as the Transformer architecture have also advanced Seq2Seq models by integrating attention mechanisms to better address long distance dependences and extract contextual information (Vaswani et al., 2017).

5.3 Attention Mechanisms

Some of the natural language processing tasks that have been transformed by attention mechanisms include text summarization in that it allows models to pay attention to parts of sequence that are important.

5.3.1 Transformer Models

Transformer models, proposed by Vaswani et al. (2017), operate solely with self-attention mechanisms to capture the relations between the input and output sequences and get rid of recurrence or convolution, which improves the model's parallelism and performance.

- BERT and GPT: Transformer models include BERT and GPT which have been used to solve extractive and abstractive summarization tasks (Devlin et al., 2019; Radford et al., 2019).

- Advantages: Long-range dependencies and context are crucial to modelling, and transformer models are excellent in these aspects, which is why they are perfect for summarizing texts of different domains and lengths.

6. Conclusion

The shift from the classical to deep learning techniques in text summarization is a result of the developments in NLP and AI fields. Although previous methods, including TF-IDF, LSA, TextRank, and LexRank, merely put the foundation of text summarization, machine learning and deep learning have taken text summarization to new heights: synthesizing more precise and contextually relevant summaries.

Supervised learning models such as SVM and the Random Forests brought in the concept of using labeled data to improve performance, albeit with the con of needing large amounts of labeled data. Clustering based methods belong to the unsupervised learning models that offered scalable as well as versatile solutions for all types of problems without requiring labeled data but the main difficulties included the problems related with the optimization of the similarity measures and the clustering algorithms.

Deep learning is considered to be the next big step in development. The use of RNN-based models (LSTM and GRU) and Seq2Seq models proved the model's capacity to deal with long term dependency and produce contextual and coherent abstractive summaries. The introduction of attention mechanisms in the Transformer models, BERT and GPT helped in improving the overall ability of models to focus on context and come up with good quality summaries.

However, there are always some issues to be solved even if the progress is rather great. Deep learning models are computationally expensive and need a large training data set which is a challenge to some applications. Also, the making of abstractive summaries remain a difficult task to accomplish, especially when it comes to the factual aspects of the document and possible concise language in it.

There are several directions for the further investigation: enhancing the efficiency of deep learning models, increasing the factual accuracy of the abstractive summaries, and studying the applicability of the hybrid approaches based on the classical, machine learning, and deep learning methods. Thus, the solutions for these problems can help extend the capacity and versatility of text summarization systems for multiple domains.

References

1. Cheng, J., & Lapata, M. (2016). Neural Summarization by Extracting Sentences and Words. In Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics.
2. Cohan, A., Dernoncourt, F., Kim, D. S., Bui, T., Kim, S. N., & Chang, W. (2018). A Discourse-Aware Attention Model for Abstractive Summarization of Long Documents. In Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies.
3. Dang, H. T. (2005). Overview of DUC 2005. In Proceedings of the Document Understanding Conference.
4. Gupta, V., & Lehal, G. S. (2010). A Survey of Text Summarization Extractive Techniques. Journal of Emerging Technologies in Web Intelligence, 2(3), 258-268.
5. Hennig, P., Hossbach, M., & Reitz, F. (2015). Analyzing Documents by Extracting Informative Sentences. In International Conference on Business Information Systems.
6. Liu, Y., & Lapata, M. (2019). Text Summarization with Pretrained Encoders. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing.
7. Mihalcea, R., & Tarau, P. (2004). TextRank: Bringing Order into Texts. In Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing.
8. Nallapati, R., Zhou, B., Gulcehre, C., & Xiang, B. (2016). Abstractive Text Summarization using Sequence-to-Sequence RNNs and Beyond. In Proceedings of the 20th SIGNLL Conference on Computational Natural Language Learning.
9. Nenkova, A., & McKeown, K. (2012). A Survey of Text Summarization Techniques. In Mining Text Data (pp. 43-76). Springer.
10. See, A., Liu, P. J., & Manning, C. D. (2017). Get to the Point: Summarization with Pointer-Generator Networks. In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics.
11. Bahdanau, D., Cho, K., & Bengio, Y. (2015). Neural Machine Translation by Jointly Learning to Align and Translate. In Proceedings of the 3rd International Conference on Learning Representations (ICLR).
12. Cao, Z., Wei, F., Li, S., Li, W., & Zhou, M. (2015). Learning Summary Prior Representation for Extractive Summarization. In Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics.
13. Gong, Y., & Liu, X. (2001). Generic Text Summarization Using Relevance Measure and Latent Semantic Analysis. In Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval.
14. Hu, M., & Liu, B. (2004). Mining and Summarizing Customer Reviews. In Proceedings of the 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining.
15. Jing, H., & McKeown, K. (2000). Cut and Paste Based Text Summarization. In Proceedings of the 1st North American Chapter of the Association for Computational Linguistics

Conference.
16. Kupiec, J., Pedersen, J., & Chen, F. (1995). A Trainable Document Summarizer. In Proceedings of the 18th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval.
17. Lewis, M., Liu, Y., Goyal, N., Ghazvininejad, M., Mohamed, A., Levy, O., ... & Zettlemoyer, L. (2020). BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics.
18. Murray, G., Renals, S., & Carletta, J. (2005). Extractive Summarization of Meeting Recordings. In Proceedings of Interspeech.
19. Nallapati, R., Zhai, F., & Zhou, B. (2017). SummaRuNNer: A Recurrent Neural Network Based Sequence Model for Extractive Summarization of Documents. In Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence.
20. Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., & Sutskever, I. (2019). Language Models are Unsupervised Multitask Learners.
21. Salton, G., & McGill, M. J. (1986). Introduction to Modern Information Retrieval. McGraw-Hill.
22. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). Attention is All You Need. In Proceedings of the 31st International Conference on Neural Information Processing Systems.
23. Zhang, X., Zhao, J., & LeCun, Y. (2019). Text Summarization with Pretrained Encoders. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing.
24. Erkan, G., & Radev, D. R. (2004). LexRank: Graph-based Lexical Centrality as Salience in Text Summarization. Journal of Artificial Intelligence Research, 22, 457-479.
25. Landauer, T. K., Foltz, P. W., & Laham, D. (1998). An Introduction to Latent Semantic Analysis. Discourse Processes, 25(2-3), 259-284.
26. Lin, C. Y. (2004). ROUGE: A Package for Automatic Evaluation of Summaries. In Proceedings of the Workshop on Text Summarization Branches Out (WAS 2004).
27. Salton, G., & Buckley, C. (1988). Term-weighting Approaches in Automatic Text Retrieval. Information Processing & Management, 24(5), 513-535.
28. Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent Dirichlet Allocation. Journal of Machine Learning Research, 3, 993-1022.
29. Breiman, L. (2001). Random Forests. Machine Learning, 45(1), 5-32.
30. Cortes, C., & Vapnik, V. (1995). Support-vector networks. Machine Learning, 20(3), 273-297.
31. Joachims, T. (1998). Text categorization with support vector machines: Learning with many relevant features. In Proceedings of the European Conference on Machine Learning.
32. Hochreiter, S., & Schmidhuber, J. (1997). Long Short-Term Memory. Neural Computation, 9(8), 1735-1780.
33. Cho, K., van Merrienboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., & Bengio, Y. (2014). Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation. In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP).
34. Shahade, A.K.; Walse, K.H.; Thakare, V.M. Deep learning approach-based hybrid fine-tuned Smith algorithm with Adam optimiser for multilingual opinion mining. Int. J. Comput. Appl. Technol. 2023, 73, 50–65.
35. Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL).
36. Wankhede, D. S., & Shelke, C. J. (2024). An experimental study using deep neural networks to predict the recurrence risk of brain tumor glioblastoma multiforme. Journal of Electrical

Systems, 20(1s), 498-507. https://doi.org/10.52783/jes.793

37. Wankhede, D. S., Shelke, C. J., Shrivastava, V. K., Achary, R., & Mohanty, S. N. (2024). Brain tumor detection and classification using adjusted InceptionV3, AlexNet, VGG16, VGG19 with ResNet50-152 CNN model. EAI Endorsed Transactions on Pervasive Health and Technology, 10.
38. Shahade, A., Walse, K., Thakare, V. M., & Atique, M. (2023). Multi-lingual opinion mining for social media discourses: An approach using deep learning based hybrid fine-tuned Smith algorithm with Adam optimizer. International Journal of Information Management Data Insights, 3, Article 100182.
39. Bhattacharjee, K., et al. (2020). Survey and gap analysis of word sense disambiguation approaches on unstructured texts. In 2020 International Conference on Electronics and Sustainable Communication Systems (ICESC) (pp. 323-327). IEEE. https://doi.org/10.1109/ICESC48915.2020.9155947
40. Chavhan, M. P. G., Patil, R. V., & Mahalle, P. N. (n.d.). Context mining with machine learning approach: Understanding, sensing, categorizing, and analyzing context parameters.