New Jersey Institute of Technology

# Reform text articles to condensed summaries by applying transformer model capabilities with traditional NLP methods

Jhilmit Asri (ja844)

Submitted to: Dr. Arashdeep Kaur

# 1 Performance Evaluation Parameters

## 1.1 ROUGE (Recall-Oriented Understudy for Gisting Evaluation)

[16]ROUGE is a set of metrics for evaluating the quality of summaries in comparison to the reference summaries provided in the dataset. ROUGE scores are calculated based on the overlap of n-grams, word sequences, and word pairs between generated and reference summaries. A total of 3 types of ROUGE scores are calculated in different implementations. They are defined as:

- ROUGE-1 (Unigram Overlap): Measures informativeness by counting overlap over single words.

$$\text{ROUGE-1 Precision} = \frac{NumberOfOverlappingUnigrams}{TotalUnigramsGeneratedInSummary} \tag{1}$$

- ROUGE-2 (Bigram Overlap): Measures fluency by counting overlap over two-word sequences (bigrams).

$$\text{ROUGE-2 Precision} = \frac{NumberOfOverlappingBigrams}{TotalBigramsGeneratedInSummary} \tag{2}$$

- ROUGE-L (Longest Common Subsequence): Evaluates frequency and grammatical structure by identifying the longest common subsequence (LCS) between generated and reference sub-summaries.

$$\text{ROUGE-L Precision} = \frac{LCSLength}{TotalWordsInReferenceSummary} \tag{3}$$

## 1.2 BLEU - Bilingual Evaluation Understudy Score

[17]BLEU is primarily used for machine translation but it is also applied to sequence-to-sequence tasks such as text summarization. It measures n-gram precision between generated and reference summaries, incorporating a penalty to discourage excessive short summaries. It is given by:

$$BLUE = BP \cdot \exp\left(\sum_{n=1}^{N} w_n \log p_n\right) \tag{4}$$

where:

- $BP$ is the brevity penalty, used to penalize short summaries.

- $p_n$ is the precision for each n-gram length.
- $w_n$ is the weight assigned to each n-gram (commonly $\frac{1}{N}$ for equal weighting).

BLUE scores range from 0-1, 1 being the perfect match.

## 1.3 BERT Score

[18]BERTScore is a metric that computes the similarity between the embeddings of tokens in generated and reference summaries using pre-trained BERT embeddings. This metric captures similarity beyond n-gram overlap. It is given by:

$$BERTScore = \text{Cosine Similarity}(Emb_{gen}, Emb_{ref}) \tag{5}$$

## 1.4 METEOR Score - Metric for evaluation of Translation with Explicit Ordering

[19]METEOR Score is a metric often used for translation and summarization. It calculates similarity by incorporating stemming, synonymy matching, and paraphrase matching in addition to exact matching. It is given by:

$$METEOR = F_{mean} \cdot (1 - \gamma \cdot \text{Penalty}) \tag{6}$$

- $F_{mean}$: The harmonic mean of precision and recall.
- Penalty: Accounts for fragmentation in matching.

## 1.5 Precision

Precision evaluates the fraction of correctly classified instances among those classified as positives. It is defined as:

$$\text{Precision} = \frac{TP}{TP + FP} \tag{7}$$

where **TP** represents True Positives, and **FP** represents False Positives.

## 1.6 F1 Score

The F1 Score is the harmonic mean of Precision and Recall, providing a balance between them. It is given by:

$$F1 = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \times 100 \tag{8}$$

- $TP$: True positives
- $FP$: False positives

### 1.7 Recall

Accuracy is the harmonic mean of Precision and Recall, providing a balance between them.

$$\text{Precision} = \frac{TP}{TP + FN} \tag{9}$$

- $TP$: True positives
- $FN$: False negatives

### 1.8 Accuracy

$$\text{Precision} = \frac{TP + TN}{TP + TN + FP + FN} \tag{10}$$

- $TP$: True positives
- $FN$: False negatives
- $TN$: True negatives
- $FN$: False negatives

## 2 Literature Review

Author Ashish Vaswani in 2017 introduced [1]"Attention Is All You Need" the Transformer Model, an architecture based entirely on attention mechanisms, eliminating the need for recurrence or convolutions in the sequence tests. The key innovation in this study is self-attention, which allows the model to focus on different parts of the input sequence without considering the distance between them. The transformer model consists of an encoder-decoder structure, where both the encoder and decoder use self-attention layers to analyze dependencies in the input-output sequences. The paper demonstrates the effectiveness of the transformer on machine translation tasks and provides extraordinary results in translation quality as well as training efficiency. The transformer model has now become a foundation for various NLP tasks, including text summarization, due to its scalable nature and parallelization capabilities.

Authors Gidiotis and Tsoumakas in 2020 proposed a different approach to summarize large text documents. Their idea - [3]”A divide and conquer approach to the summarization of long documents” segments a long document into several sections, generating source and target pairs for each segment to train the summarization model. The model can produce partial summaries for each section which can then be combined to create a final summary. This method reduces computational complexity and enables parallelization. The authors demonstrated that this approach achieves competitive results on datasets like []arXiv and []PubMed, using both sequence-to-sequence (RNNs) and Transformer-based models.

Mike Lewis, Yinhan Liu, and Naman Goyal along with other authors presented [5]”BART” in 2019. BART is a denoising autoencoder model designed to pre-train sequence-to-sequence tasks such as summarization, translation, QnA, etc. BART combines the power of bidirectional encoders and autoregressive decoders. It uses an interesting technique of pertaining the model by corrupting/masking the input text and learning to reconstruct the original text. This strategy allows BART to perform well across a wide range of tasks, including summarization. The model is evaluated on various summarization datasets and it achieves state-of-the-art performance, particularly on abstractive summarization tasks. BART’s performance gains are the result of its noising strategy during retraining, which forces the model to connect long-range dependencies and formulate sentence structure. The paper also reports significant Improvements in summarization tasks, achieving high ROUGE scores compared to previous models.

Abigail See, Peter J. Liu, and Christopher D. Manning proposed a hybrid model in the research paper [2]”Get to the point” in 2017 that combines abstractive and extractive summarization techniques using pointer generator networks. Their model addresses two problems in abstractive summarization, the inaccurate reproduction of factual details and the tendency for models to repeat phrases. This proposed network allows the model to selectively copy words from the source test while retaining the ability to generate new words, thus balancing extraction and abstraction. Moreover, the authors introduced a coverage mechanism to track parts of the source text that have already been summarized, reducing repetition in general summaries. The model is applied to the CNN/Daily-mail dataset and achieves noticeable improvements over previous abstractive models in terms of ROUGE scores. This hybrid approach provides a better, more accurate, and coherent way to generate summaries, overcoming some limitations of purely abstractive models.

Authors Liu and Lapata 2019 explored the application of pre-trained language models, specifically BERT to summarize texts. They propose a novel framework in [4]"Text Summarization with Pretrained Encoders" for both extractive and abstractive summarization using pre-trained encoders. Their model uses BERT as a document-level encoder and provides the capability of capturing semantic relationships between sentences, For the extractive summarization part, they stack inter-sentence transformer layers on top of the BERT encoder to capture document-level features. For abstractive summarization, they deployed an encoder-decoder architecture, where the BERT encoder is combined with a transformer encoder. The authors also introduced a two-stage fine-tuning process that focuses on the extractive task before fine-tuning for the abstractive task. Experimental results across several datasets show that the model achieves sizable performance, highlighting the effectiveness of pre-trained encoders for summarization tasks.

The paper "T-BERTSum: Topic-Aware Text Summarization Based on BERT" [6] enlists the hurdles in automatic text summarization mainly with long-term text dependencies and latent topic mapping. T-BERTSum conveys the contextual description through the BERT's pre-trained language model, and NTM enhances the clear and focused topic summaries. The paper focused on two aspects which are extractive summarization, recognizing key sentences, and abstractive summarization, refining extracted sentences into a clear and concise topic summary. To evaluate long-term dependencies in the text transformers and to handle a sequence of the text during extraction, an LSTM layer is incorporated whereas to reduce redundancy during abstractive summarization the gated network is used. CNN Daily Mail and XSum datasets are analyzed and tested through this model, where they represent the performance improvement on previous summarization models through the rouge metrics. To calculate the overlap of unigrams, bigrams, and the longest common subsequence between the generated and reference summaries the rouge metrics with ROUGE-1, ROUGE-2, and ROUGE-L scores were used. They also incorporated the manual prediction for further evaluation of the model through the salience, coherence, and redundancy. They fine-tuned the BERT model to improve efficiency and generate concise textual documents with a two-stage approach. The pre-trained transformers, LSTM layers, and topic modeling represent how will result in analyzing text.

The paper "Autoregressive Decoder with Extracted Gap Sessions for Sequential/Session-Based Recommendation" [7]incorporates a PEGASUS-based transformer model. To enhance interactions with extracted gap sessions and session recommendations the gap-session transformer" (GST), combines sequential recommendation systems (SRS) with session-based recommendation systems (SBRS). They used GST and adapted Pegasus a transformer model as an abstractive summarization technique for the experiment. They

signify that the GST enables the capture of complex relationships and time-sensitive transitions between items through the encoding. The autoregressive decoder from Pegasus improves performance and masks the limitations in detecting high-order relationships between items to produce session recommendations. They represented the corpus theme for the model which helps for stronger embedding and thematic learning within sessions.
MovieLens1M and Yoochoose datasets were analyzed through the GST with metrics of Hit Ratio@5 and @10 for sequential recommendation and Precision@5 and @10 for session-based tasks. The GST illustrates the model performance through the 21 baselines achieving the highest accuracy. They showed that the incorporation of PEGASUS's abstractive summarization technique with GST can be able to generate contextually extracted sessions improving the accuracy. They demonstrated that Pegasus along with the GST slack off limits and enhances the performance in SRS and SBRS contexts with the gap session generator to improve session encoding and relational summarization.

The article "Automatic text summarization models using transformer-based architectures", [8] focuses on the models of BART (Bidirectional and Auto-Regressive Transformer) and T5 (Text-to-Text Transfer Transformer) as a primary model and they also worked on Seq2Seq + CGU, ALONE and Reinforced-ConvS2S. They worked on the CNN daily mail dataset with these two models converting large text into precise information which consists of over 286,000 training pairs, 13,368 validation pairs, and 11,487 test pairs They address the challenges of filtering large text into concise summaries through fine-tuning the model. The NLP techniques and transformer models improve the text summarization efficiency. They trained the BART by manipulating the original text and fine-tuning the loss and the T5 was trained using both supervised and unsupervised tasks for generating text. They evaluated the performance through rouge scores. BART gained a ROUGE-1 score of 37.555 percent, which was higher than T5's score of 36.45 percent and other models trained. BART's architecture balances the. The model outperformed earlier models by margins of 16.79 percent over Takase and Kobayashi (2020) models, 3.44 percent over Lin et al. (2018) models and 20.54 percent over Wang et al.

The article "Stacked Denoising Variational Auto Encoder Model for Extractive Web Text Summarization," [9] uses the Stacked Denoising Variational Autoencoders (SD-VAE) model for improving the efficiency of extractive web text summarization text. They incorporated the stacked architecture which denoise the autoencoders to generate input text. The model was trained on a variety of datasets, and the noise was added to the training process to enhance the model's capacity to recognize and extract key phrases from the documents. This method helped to increase the coherence and relevance of the generated summaries. They used the Twitter sentiment-text dataset, containing 27,482 tweets with positive, negative, and neutral sentiments. To handle large data they used tokenization, stop word removal, lemmatization, and CBoW for preprocessing. The SD-VAE model with hierarchical attention extracts the keywords and improves the accuracy of feature extraction in summarization. The Multilayered Competitive Probable Modular Perception Model and Graph-based Quadruplicate Lexicon Summarization methodologies are also further evaluated. The model achieved a maximum accuracy of 98.3 percent, F1-Score of 98.4 percent, precision of 98 percent, and recall of 98.8 percent. and ROUGE scores. They conclude that the model efficiently evaluates the web-specific summarization.

The article " Investigating the Pre-Training Bias in Low-Resource Abstractive Summarization" [10] evaluates the efficiency of pre-training transformer-based summarization models mainly in low-resource settings. The article focuses on different datasets like CNN/Daily Mail, ArXiv, WikiHow, BigPatent, and Booksum. These datasets summarize difficulties because of their structural variations and differing degrees of detail. Each model is benchmarked using 10–1000 instances on few-shot learning tasks as part of the experimental setting. Model effectiveness is measured using metrics like ROUGE, ALTI scores, and Extractive Oracle comparisons. The metrics will analyze extractive and attention allocation patterns to analyze whether the model can work in the low-resource setting. The paper specifies that the BART will perform better than the other models in low resource conditions. Pegasus-SP illustrates the improved adaptability and increased attention alignment in datasets along with BART.

Ramesh Nallapati, Bowen Zhou, and Bing Xiang along with other authors in 2016 tried to explore [11] "Abstractive Text Summarization using Sequence-to-Sequence RNNs and Beyond", to produce paraphrased summaries of central content points in an original document. The main goals of abstractive summarisation differ slightly from the goals of machine translation so specific models are proposed, such as mechanism for hierarchical content structures and the leveraging of attention mechanisms at multiple levels. Features added by the authors increase the quality of the summaries, which include a feature-rich encoder, a generator-pointer switching model that accounts for addressing out-of-vocabulary words, and a hierarchical attention model where sentences and words within long documents are captured.
The proposed models on the Gigaword, DUC, and CNN/Daily Mail corpora were evaluated using ROUGE metrics (ROUGE-1, ROUGE-2, and ROUGE-L) and reported substantial gains over previous benchmarks. For example, 'feats-lvt2k-2sent-ptr' gave ROUGE-1, ROUGE-2, and ROUGE-L scores of 36.40, 17.77, and 33.71 on Gigaword, outperforming previous state-of-the-art approaches including ABS+. On DUC 2004, the model was also competitive: scoring 28.61 (ROUGE-1), 9.42 (ROUGE-2), and 25.24

(ROUGE-L) – providing a stronger sense of content relevance while retaining sentence coherency.
The authors also provide a new dataset with multi-sentence summaries extracted from CNN and Daily Mail, providing a standard benchmark for future research on multi-sentence summarizing tasks. Results reveal that the hierarchical attention model and a temporal attention model designed to minimize repetitive content showed effectiveness, which suggests that the new methods are promising for generating summaries that sound more natural and fluent on larger and more complex datasets. The study establishes the models as a good starting point for future research on deep learning approaches to summarization.

Kryscinski, Bryan McCann, Caiming Xiong, and Richard Socher(2019) proposed [12]"Evaluating the Factual Consistency of Abstractive Text Summarization" which can identify factual inconsistencies in abstractive text summarization by designing a weakly-supervised, BERT-based model, FactCC. The model is trained on weakly-supervised data that is generated by rule-based transformations of source sentences and allows it to detect inconsistencies in summarization outputs. The model highlights these inconsistencies in the source document, which helps identify factual errors that are missed by current evaluation metrics (e.g., ROUGE) between portions of the input and summary. This reminds us how the current emphasis on developing domain-specific consistency checks in summarization is an appropriate direction to consider as up to 30% of abstractive summaries have errors.
In experiments with the CNN/Daily Mail test set, FactCC demonstrated accuracy that far surpassed conventional natural language inference (NLI) and fact-checking with an accuracy of 74.15% and an F1 score of 0.5106, compared to baseline scores of around 52% accuracy and 0.0882 F1 from NLI models. Additionally, human evaluation finds that the span-selection feature helps humans assess factual consistency and speeds up task completion and agreement between annotators. The findings suggest that FactCC not only improves factual accuracy in generated summaries but also supports human fact-checking efforts through meaningful explanations.

In the paper [13]"An effective sentence-extraction technique using contextual information and statistical approaches for text summarization," Ko and Seo (2008) proposed a method that combines contextual information and statistical techniques to improve sentence extraction for both single and multi-document summarization tasks. Their approach addresses the feature sparseness issue by creating "Bi-gram pseudo-sentences" which connect consecutive sentences to preserve contextual information during extraction. The authors combine statistical techniques such as title matching and aggregation similarity, To assess the importance of sentences based on location, word frequency, and similarity to search terms. By focusing on bi-gram pseudo sentences in an initial extrac-

tion stage, followed by extracting individual sentences from these pairs, this approach aims to improve summaries coherence and informativeness.
The results show that the proposed method outperforms conventional methods such as DOCUSUM and Microsoft Word's summarization, especially in cases and absence of title information. For single-document summarization, their method achieved F1 scores of 53.4 for 10% summaries and 55.3 for 30% summaries when titles were present, surpassing the performance of alternative statistical methods. For multi-document summarization, the hybrid method reached an F1 score of 51.6, outperforming the popular MMR (Maximal Marginal Relevance) technique, which scored around 48.2-48.3. This indicates that it can adapt to both document type and summarization needs.

In [14]" Text Summarization using Latent Semantic Analysis," Ozsoy, Alpaslan, and Cicekli (2011) explore Latent Semantic Analysis (LSA) as an approach to automatic text summarization. LSA is an unsupervised, algebraic statistical method that Uses singular value decomposition (SVD) to identify patterns of words and phrases that co-occur within text. The authors introduce two new algorithms within the LSA framework, focusing on single and multiple document summarization. They applied and evaluated these methods on Turkish and English text datasets. It aims to establish language independence and highlight the potential of LSA for cross-linguistic summarization tasks.
This study uses ROUGE scores to compare the performance of the proposed LSA-based method with existing summarization techniques. This reveals that the "cross" method consistently gives the best results on a wide range of datasets. In particular, the cross method on the Turkish dataset with long documents achieved a higher ROUGE-L score compared to existing algorithms such as the Gong-Liu method, and similar results were found on the English datasets, including DUC2002 and DUC2004 benchmarks. However the cross-method is less effective for small documents, indicating that LSA's power is more evident in the Summary where the context covers several sentences.
The evaluation concludes that although LSA-based summarization methods may not outperform knowledge-based approaches based on machine learning in some situations, It is still highly effective for language-independent summarization using only internal document data. And demonstrate robustness and adaptability across languages.

In [15]"PEGASUS: Pre-training with gap-splitting sentences for abstract generalization," Zhang and other authors (2020) present PEGASUS, a pre-training model specifically designed for abstractive text summarization tasks. The model covers entire sentences using a new method called "Gap sentences" and predicts them using the remaining context, simulating a summarization objective. This approach differs from conventional models covering random words or phrases in that it is more amenable to summarization tasks. PEGASUS uses a Transformer-based encoder-decoder architecture and pre-trains it on large datasets, such as C4 (Common Crawl) and HugeNews corpora,

optimizing for a broad range of domains, from news to legislative documents.
The model's performance is evaluated on 12 datasets, including news articles, scientific papers, and patents. PEGASUS extracts results from each dataset. The ROUGE score measures this. For instance, on the XSum dataset, PEGASUS reaches ROUGE-1, ROUGE-2, and ROUGE-L scores of 47.21, 24.56, and 39.25, respectively. The model also demonstrates strong zero-shot and low-resource capabilities, outperforming previous models with as few as 1,000 training samples on select tasks. Additionally, the PEGASUS model trained on the HugeNews dataset outperformed the model trained on C4 in most datasets. This underscores the importance of pre-training data.
Human evaluations also confirmed the effectiveness of PEGASUS, showing that the summaries match human-written ones in readability and informativeness, particularly for datasets like XSum and CNN/DailyMail. The article concludes that PEGASUS provides an efficient and effective method for producing high-quality summaries in a wide variety of domains.

# 3 Comparison Table

| Sno | Ref. | Year | Methodology | Rouge-1 | Rouge-2 | Rouge-L | BLEU | BERTScore |
|---|---|---|---|---|---|---|---|---|
| 1 | [13] | 2008 | Single/Multiple Document extractive Summarization | NR | NR | NR | NR | NR |
| 2 | [14] | 2011 | LSA based Text Summarization | NR | 0.19260 | 0.37229 | NR | NR |
| 3 | [11] | 2016 | Abstractive(Sequence-to-sequence encoder-decoder with RNN) | 35.46 | 13.30 | 32.65 | NR | NR |
| 4 | [1] | 2017 | Self Attention, Transformer Model | NR | NR | NR | 28.4 (E-G)<br>41.8 (E-F) | NR |
| 5 | [2] | 2017 | Pointer generator network, Seq2Seq Model | 43.71 | 26.40 | 39.38 | NR | NR |
| 6 | [12] | 2019 | FactCC Model, Sentence-Level Consistency Classification | NR | NR | NR | NR | NR |
| 7 | [15] | 2020 | Gap-Sentences Generation (GSG), Principal Sentence Selection, Pre-training, Fine-tuning | 47.21 (Xsum) | 24.56 (Xsum) | 39.25 (Xsum) | NR | NR |
| 8 | [5] | 2020 | Denoising Autoencoder, Seq2Seq Model | 48.91 (CNN), 45.41 (Xsum) | 30.84 (CNN), 19.18 (Xsum) | NR | NR | 89.3 |
| 9 | [4] | 2020 | Two Staged FineTuning, Encoder-Decoder Architecture | 49.02 | 31.02 | NR | NR | 90.2 |
| 10 | [3] | 2020 | Segment Wise Processing, Transformer Architecture | 46.69 (arxiv), 45.15 (pubnet) | 17.44 (arxiv), 19.75 (pubnet) | 33.77 (arxiv), 38.11 (pubnet) | NR | NR |
| 11 | [6] | 2021 | T-BERTSum | 39.90 (xsum), 43.58 (Cnn/daily mail) | 17.48 (xsum), 20.45 (CNN/daily mail) | 32.18 (xsum), 39.80 (CNN/daily mail) | NR | NR |
| 12 | [7] | 2023 | PEGASUS | NR | NR | NR | NR | NR |
| 13 | [8] | 2024 | BART,T5 | 37.55, 36.45 | NR | NR | NR | NR |
| 14 | [9] | 2024 | Stacked Denoising Variational Autoencoder, CBoW Text Vectorization Model | 0.521 (proposed work) | 0.256 (proposed work), | 0.491 (proposed work) | NR | NR |
| 15 | [10] | 2024 | BART, PEGASUS | 29.43±6.83 (BART) | 8.24±4.79 (BART) | 17.10±4.04 (BART) | NR | NR |

Table 1: Evaluation Metric Table

| Sno | Ref. | Precision | Recall | F1Score | Accuracy |
|---|---|---|---|---|---|
| 1 | [13] | NR | 14.75% | 47.9 with 10% summary 50.4 with 30% | NR |
| 2 | [14] | NR | NR | NR | NR |
| 3 | [11] | NR | NR | NR | NR |
| 4 | [1] | NR | NR | NR | NR |
| 5 | [2] | NR | NR | NR | NR |
| 6 | [12] | NR | NR | 0.5106 | 74.15 |
| 7 | [15] | NR | NR | NR | NR |
| 8 | [5] | NR | NR | NR | NR |
| 9 | [4] | NR | NR | NR | NR |
| 10 | [3] | NR | NR | NR | NR |
| 11 | [6] | NR | NR | NR | NR |
| 12 | [7] | 0.5999 (yoochoose) p@5, 0.2999 (yoochoose) p@10 | NR | NR | NR |
| 13 | [8] | 87.55(BART) | 89.07 | 88.31 | NR |
| 14 | [9] | 98 | 98.8 | 98.4 | 98.30 |
| 15 | [10] | 27.91 (BART) | 35.29 | 28.14 | NR |

Table 2: Evaluation Metric Table - Continued

# 4 References

1. A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, ”Attention is All You Need” (2017)
   Available: `https://arxiv.org/abs/1706.03762`

2. A. See, P. J. Liu, and C. D. Manning, ”Get to the Point: Summarization with Pointer Generator Networks” - (2017)
   Available:`https://ui.adsabs.harvard.edu/abs/2017arXiv170404368S/abstract`

3. A. Gidiotis and G. Tsoumakas, ”A Divide and Conquer Approach to the Summarization of Long Documents” - (2020)
   Available: `https://ieeexplore.ieee.org/document/9257174`

4. Y. Liu and M. Lapata, ”Text Summarization with Pretrained Encoders” - (2019)
   Available: `https://ui.adsabs.harvard.edu/abs/2019arXiv190808345L/abstract`

5. M. Lewis, Y. Liu, N. Goyal, M. Ghazvininejad, A. Mohamed, O. Levy, V. Stoyanov, and L. Zettlemoyer, ”BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension” - (2020)
   Available: `https://arxiv.org/abs/1910.13461`

6. T. Ma, Q. Pan, H. Rong, Y. Qian, Y. Tian, and N. Al-Nabhan, ”T-BERTSum: Topic-Aware Text Summarization Based on BERT,” - (2021)
   Available: `doi:10.1109/TCSS.2021.3088506`

7. J. Chung, J. H. Lee, and B. Jang, ”Autoregressive Decoder With Extracted Gap Sessions for Sequential/Session-Based Recommendation” - (2023)
   Available: `doi:10.1109/ACCESS.2023.3297204`

8. Rao, R., Sharma, S., and Malik, N. Automatic text summarization using transformer-based language models. - (2024)
   Available: `https://doi.org/10.1007/s13198-024-02280-4`

9. Yadav, M., Katarya, R. Stacked Denoising Variational Auto Encoder Model for Extractive Web Text Summarization. - (2024)
   Available: `https://doi.org/10.1007/s40998-024-00751-9`

10. D. Chernyshev and B. Dobrov, ”Investigating the Pre-Training Bias in Low-Resource Abstractive Summarization” - (2020)
    Available: `doi:10.1109/ACCESS.2024.3379139`

11. Ramesh Nallapati, Bowen Zhou, Cicero dos Santos, Bing Xiang, Caglar Gulcehre, ”Abstractive Text Summarization using Sequence-to-Sequence RNN’s and Beyond”, 2016
Available: https://arxiv.org/abs/1602.06023

12. Wojciech Kryscinski, Bryan McCann, Caiming Xiong, Richard Socher, ”Evaluating the Factual Consistency of Abstractive Text Summarization” - (2019)
Available: https://arxiv.org/abs/1910.12840

13. Youngjoong Ko and Jungyun Seo, ”An effective sentence-extraction technique using contextual information and statistical approaches for text summarization” - (2008)
Available: https://www.sciencedirect.com/science/article/pii/S0167865508000676

14. Makblue Gulcin Ozsoy, Freda Nur Alpaslan, IIyas Cicekli, ”Text summarization using Latent Semantic Analysis” - (2011)
Available: https://www.researchgate.net/publication/220195824_Text_summarization_using_Latent_Semantic_Analysis

15. Jingqing Zhang, Yao Zhao, Mohammad Saleh, Peter J. Liu,” PEGASUS: Pre-training with Extracted Gap-sentences for Abstractive Summarization” - (2020)
Available: https://arxiv.org/abs/1912.08777

16. Chin-Yew Lin ”ROUGE: A Package for Automatic Evaluation of Summaries” - (2004)
Available: https://aclanthology.org/W04-1013/

17. Kishore Papineni, Salim Roukos, Todd Ward, Wei-Jing Zhu, “BLEU: a method for automatic evaluation of machine translation“ - (2002)
Available: https://dl.acm.org/doi/10.3115/1073083.1073135

18. Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, Yoav Artzi, “BERTScore: Evaluating Text Generation with BERT“ - (2019)
Available: https://arxiv.org/abs/1904.09675

19. Abhaya Agarwal, Alon Lavie, “METEOR: An Automatic Metric for MT Evaluation with Improved Correlation with Human Judgments“ - (2007)
Available: https://dl.acm.org/doi/10.5555/1626355.1626389