Journal of **N**onlinear **A**nalysis and **O**ptimization
Vol. 15, Issue. 1, No.6 : 2024
ISSN : **1906-9685**

ADVANCED SEARCH AND SUMMARIZATION OF EDUCATIONAL DOCUMENTS USING MACHINE LEARNING

Shreyas Gaddam B.Tech. Scholar (CSE) Shri Shankaracharya Institute of Professional Management and Technology Raipur, India g.shreyas@ssipmt.com
Lakhan Baheti B.Tech. Scholar (CSE) Shri Shankaracharya Institute of Professional Management and Technology Raipur, India lakhan.baheti@ssipmt.com
Lavanya Gokhale B.Tech. Scholar (CSE) Shri Shankaracharya Institute of Professional Management and Technology Raipur, India lavanya.gokhale@ssipmt.com
Yogesh Kumar Rathore* Assistant Professor (CSE) Shri Shankaracharya Institute of Professional Management and Technology Raipur, India y.rathore@ssipmt.com∗

Abstract—
This paper presents experiments and study for improving text summarization by fine-tuning existing language models. We aim to improve a language model's ability to extract key information from diverse texts. Using advanced models and a detailed comparison, we aim to identify the most effective fine- tuned model. To address the importance of summarization in enhancing comprehension, we curate a specialized dataset for science and literature domains. Our findings provide valuable insights in terms of efficiency and performance for students, teachers, researchers and practitioners looking to optimize ad- vanced search and summarization technologies across different fields.
Index Terms—Natural Language Processing, Summarization, Semantic Search, Transformers

I. INTRODUCTION

Summarization serves as a crucial tool, playing a vital role in helping people understand large amounts of information. This process involves extracting the most important elements from lengthy articles, making it easier for individuals to grasp the subject matter. By removing unnecessary details, summarization not only improves user retention but also simplifies the communication of complex concepts, ultimately leading to better-quality information. Additionally, summariza- tion techniques significantly reduce the time required for in- depth study.

Semantic search represents a departure from conventional keyword-based retrieval systems by incorporating a deeper understanding of the meaning behind user queries and docu- ment content. Unlike traditional methods which rely solely on syntactic matching, semantic search leverages natural language processing and machine learning techniques to comprehend context, relationships, and the inherent semantics of both queries and documents. This approach provides great potential of significantly improving the metrics and relevance of search results, thereby enhancing the overall user experience.

This paper also aims to explore the transformative impact of semantic search on document retrieval, delving into its un- derlying mechanisms, advantages, and potential applications. By examining the fusion of linguistic semantics and advanced algorithms, we aim to shed light on how semantic search can better cater to the nuanced information needs of users, offering a more intelligent and context-aware retrieval experience.

In summary, this research aims to contribute to both the academic discussion on summarization techniques and their practical application, especially in the fields of science and literature. Our multifaceted approach, which includes dataset curation, model fine-tuning, and comprehensive

evaluation, is geared towards improving the sophistication and utility of summarization and search processes for better information comprehension.

This paper follows a structured approach, beginning with an extensive dataset curation from diverse sources. In the second phase, the paper delves into the model training process, lever- aging the state of the art Transformers [1] architecture. The third key component of the research involves the incorporation of the RAG (Retrieval-Augmented Generation) [2] framework, which enhances the tool's search and summarization capabil- ities.

II. RELATED WORK

The field of automatic text summarization has witnessed significant advancements over the past few decades [3], with researchers exploring various approaches to distill relevant information from large volumes of text. In this section, we review existing literature and categorize related work based on key themes and methodologies in the domain of machine learning (ML) for text summarization.

A. Extractive Summarization Approaches

- Traditional Methods: Early efforts in extractive summa- rization predominantly focused on heuristic-based meth- ods, utilizing features such as sentence importance scores and term frequency-inverse document frequency (TF- IDF). Classic algorithms like TextRank and LexRank emerged as notable contenders, emphasizing graph-based ranking algorithms for sentence extraction.
- Supervised Learning Techniques: Building upon tradi- tional methods, researchers began to explore supervised learning techniques for extractive summarization. Support Vector Machines (SVM), Decision Trees, and Random

Forests have been applied to classify sentence impor- tance, often relying on handcrafted features.

B. Abstractive Summarization Models

- Neural Sequence-to-Sequence Models: The advent of neural networks revolutionized abstractive summariza- tion, with sequence-to-sequence (Seq2Seq) models gain- ing prominence. These models employ recurrent or trans- former architectures to generate coherent and contextu- ally rich summaries. Notable variants include the atten- tion mechanism, which enhances the model's ability to focus on relevant parts of the input text.
- Pre-trained Language Models: Recent developments have seen a shift towards leveraging pre-trained language models, such as BERT (Bidirectional Encoder Repre- sentations from Transformers) [4] and GPT (Generative Pre-trained Transformer) [5], for abstractive summariza- tion tasks. These models, initially designed for contex- tual word embeddings and language understanding, have shown promising results in capturing intricate relation- ships within the text, enabling the generation of more coherent and context-aware summaries.

III. METHODOLOGY

In our experiment, we curated a summarization dataset from various other datasets which are aimed towards scientific and educational literature. We fine-tune multiple models on this curated dataset and compare the performance across models. We carefully selected the models which are suitable for summarization task. To perform our experiments, we've utilized Hugging Face Transformers library to train our models [6], [7].

A. Dataset Preparation

We compiled a moderately sized summarization dataset with around 15,000 training samples and 1500 validation samples. We only used small subsets of each dataset since the individual datasets are quite large in size. Table I indicates the datasets which we've used. The reasons for choosing these datasets are as follows:

- PubMed Scientific Papers [8]: This is a collection of publications and research articles from the biomedical literature. The articles in this dataset are dense in scien- tific terminologies which may seldom be present in other datasets.
- ArXiv Scientific Papers [8]: ArXiv is a free platform for distribution of scholarly articles. A subset of such articles from various fields will be useful for our purpose.

• BookSum [9]: it is a collection of datasets for long-form narrative summarization. This dataset has covered source documents from the literature domain, such as plays, stories and novels, and includes highly abstractive, human written summaries on multiple levels of granularity of increasing difficulty.

• ScisummNet [10], [11]: This dataset is a large annotated corpus of scientific papers with detailed summarizations.

• WikiHow [12]: It is an online platform which features how-to articles on a variety of topics. This dataset is a collection of many of such posts. Each post is of various lengths and they contain an overview which essentially is the summary of the post.

TABLE I
DATASET INFORMATION

Dataset	Total Samples
PubMed Scientific Papers	4000
ArXiv Scientific Papers	4000
BookSum	4000
ScisummNet	1009
WikiHow	4000

B. Model Selection

The models which we have chosen are based on the current state-of-the-art Transformer [1] architecture. We have specifically chosen models which include both encoder and decoder layers. The source text will be input to the encoder and the summary will be the input to the decoder layer. It was also important that we considered models with long context length so the text doesn't get truncated and even if it does, it's not significant. We also paid attention to the type of positional embeddings used in our models as relative positional embeddings are useful in long context situations.

Table II mentions the various sizes of models we used and Table III denotes the sequence lengths for inputs and outputs.

The models which we have chosen are as follows:

• Flan-T5: The T5 (Text-to-Text Transfer Transformer) model from Google [13] is a powerful and versatile sequence-to-sequence transformer architecture that can be fine-tuned for various natural language processing tasks, such as translation, summarization, question answering, and classification. It treats every task as a text generation problem, using the same model and objective for all tasks. Flan-T5 [14] is a variant of T5 specifically trained on a diverse range of instruction-tuning datasets to perform a wide variety of tasks specified by natural language instructions. We've trained our model on both the base model (Flan-T5 Base) and a smaller version (Flan-T5 Small).

• BART: Bidirectional and Auto-Regressive Transformers (BART) model [15] is a denoising autoencoder for pre- training sequence-to-sequence models. It combines the benefits of bidirectional and autoregressive transformers by corrupting text with an arbitrary noising function and then learns to reconstruct the original text. This method allows BART to handle a broad set of NLP problems like text generation, machine translation, and summarization while being robust to noise and achieving strong performance. We've used the base variant of this model.

TABLE II
MODEL PARAMETERS

Model	Number of Parameters
Google Flan-T5 Small	77M
Google Flan-T5 Base	248M
Facebook BART Base	139M

TABLE III
INPUT AND OUTPUT MAXIMUM SEQUENCE LENGTHS

Model	Input Max Length	Output Max Length
Flan-T5 Base	2048	256
Flan-T5 Small	2048	256
BART Base	1024	256

C. Metrics

The metric we've used to measure the performance of our models is ROUGE (Recall-Oriented Understudy for Gisting Evaluation) [16]. It is a set of metrics used for the automatic evaluation of machine-generated text against reference sum- maries. The following metrics are employed to evaluate our results:

- Rouge-1: Measures the overlap of unigram (single-word) tokens between the generated text and the reference summary.
- Rouge-2: Focuses on bigram (two-word) token overlap to assess the similarity between the generated and reference text.
- Rouge-L: Evaluates the longest common subsequence (LCS) of words between the generated and reference summaries.
- Rouge-Lsum: Computes the LCS of words while consid- ering stemming and stopwords, providing a more lenient evaluation.
- GenLen: Computes the mean of the lengths of summaries generated by the model.

D. Retrieval Augmented Generation (RAG)

Retrieval-augmented generation (RAG) [2] is a technique used in natural language processing and machine learning to generate more accurate, contextually relevant responses by incorporating information from an external knowledge source during the generation process. This approach combines two main components: a retriever model that identifies relevant documents or passages from a large corpus of text, and a generator model that uses this contextual information along with its own internal parameters to produce a response.

The retriever model plays a crucial role in RAG as it identifies pertinent sources of information based on the input prompt provided to the system. The retriever can be built using various techniques such as sparse vector representations, dense embeddings, or neural networks. Once the most relevant documents are selected, they are passed on to the generator model which then produces a response while taking into account both the original prompt and the additional context from the retrieved documents.

The use of an external knowledge source allows the RAG model to overcome some limitations faced by traditional generative models that rely solely on their internal parameters for generating outputs. Traditional models may struggle when confronted with complex questions requiring specific facts or domain expertise beyond what's been learned during training. By integrating retrieved information, RAG models have a better chance at providing accurate answers even if they haven't encountered similar examples during training.

Moreover, RAG models also offer improved generalization capabilities compared to standalone generative models since they can leverage up-to-date information present in the re- trieved data. As new documents get added to the corpus, the performance of RAG models improves without needing any updates to the underlying architecture or re-training the model itself.

E. Summary Generation Methodology

To generate our summaries, we have used beam search decoding. Beam search is a heuristic search algorithm used for sequence prediction, such as in machine translation or speech recognition. It maintains a set of the most promising candidate sequences at each time step and incrementally extends them by one element until a termination condition is met. The algorithm keeps track of the highest scoring sequence seen so far to find an optimal solution according to a given objective function.

The generation parameters which we have used across the models are as follows:

- max length: the maximum number of tokens set at 200.
- min length: the minimum number of tokens set at 50.

• number of beams: number of candidate hypotheses maintained during beam search decoding. A higher value generally leads to better quality outputs, but also increases computation time.

• no repetition of n-gram size: the size of n-grams that should not be repeated within the generated sequence. This helps prevent repetition and improve diversity in the generated text. We set the value to 3.

• length penalty: a value which controls the trade-off between brevity and fluency in the generated sequence. Increasing this value encourages the model to produce shorter sequences, while decreasing it encourages longer sequences.

F. Semantic Search

When a long-form document is given as an input, we apply RAG based on the user query. To apply RAG, we need to first chunk the document into groups and calculate vector embeddings on individual groups. The steps we've adopted to generate vector embeddings are as follows:

• Sentence Splitting and Grouping: The document is split into individual complete sentences. We then iterate over these sentences using stride length of 25 i.e. each group has 25 sentences. The length of individual sentence can vary based on the content.

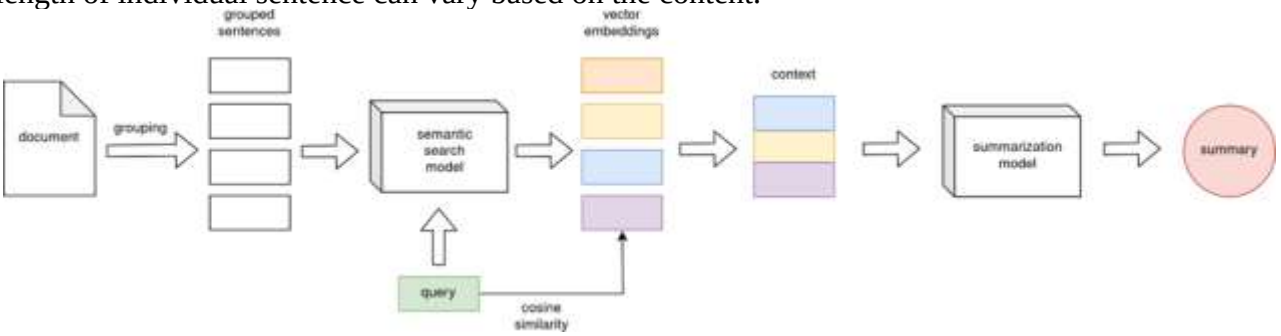

Fig. 1. workflow of search and summarization system TABLE IV

RESULTS

Model Name	Rouge-1	Rouge-2	Rouge-L	Rouge-Lsum	Mean generated summary length
Flan-T5 Base	0.3689	0.1398	0.2383	0.2386	112.20
Flan-T5 Small	0.2337	0.0708	0.1591	0.1593	65.75
BART Base	0.3788	0.1385	0.24	0.3138	135.22

• Generating Vector Embeddings: Once the groups have been created, each group is treated as an individual paragraph and then embeddings are calculated using the embedding model msmarco-distilbert-base-v4 from the Sentence Transformers library [17].

• Ranking and forming Context: Based on the user query, the query embeddings are compared with the group embeddings using Cosine-Similarity. Based on our ex- periments, we found that concatenating the top 3 groups to form the context/input to our summarization model worked the best.

IV. RESULTS

From the results that we have obtained in Table IV, it is important to keep in mind that our models will be deployed and the inference times should be minimal. Based on the inference, Flan-T5 variants evaluation was much faster as compared to BART. For hosted inference, Flan T5 Small being a relatively smaller model as compared to its Base variant performs faster but the length of summarization is much smaller. The ROUGE scores are based on limited amount of time spent in training. Usually the number of samples in summarization datasets are well over 100k. We chose very small subsets of various datasets and with RAG, we have been able to achieve decent performance over all the models in terms of summarization quality.

V. CONCLUSION

In conclusion, this research endeavor addresses the critical role of summarization in information comprehension and ex- plores the transformative potential of semantic search in doc- ument retrieval. The presented methodology encompasses the curation of a specialized summarization dataset tailored

to the intricate domains of science and literature. Through the fine- tuning of various models on this dataset and a comprehensive comparative analysis, we illuminate the nuanced differences in performance and efficiency. The success of the Advanced Search and Summarization Tool marks the beginning of a promising journey towards enhancing information retrieval and synthesis. As we look forward, several avenues for future research and development emerge, each with the potential to further augment the tool's capabilities and contribute to the broader field of information management. In the future, this can be further improved using techniques such as "Enhanced Multimodal Integration" to accommodate diverse types of content, including images, videos, and audio, the tool can be extended to support multimodal information retrieval and summarization. This involves developing algorithms that can effectively process and synthesize information from various modalities, providing users with a comprehensive understand- ing of the content. Real-time collaboration and sharing features among users is a vital aspect of modern knowledge man- agement systems. Future iterations of the tool could include features for collaborative document editing, annotation, and seamless sharing of summarized content, fostering efficient teamwork and knowledge dissemination. Additional features such as user personalization to tailor the tool to individual user preferences and requirements can significantly enhance its usability.

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," 2023.

[2] P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Ku¨ttler, M. Lewis, W. tau Yih, T. Rockta¨schel, S. Riedel, and D. Kiela, "Retrieval-augmented generation for knowledge-intensive nlp tasks," 2021.

[3] A. P. Widyassari, S. Rustad, G. F. Shidik, E. Noersasongko, A. Syukur, A. Affandy, and D. R. I. M. Setiadi, "Review of automatic text summarization techniques methods," Journal of King Saud University - Computer and Information Sciences, vol. 34, no. 4, pp. 1029–1046, 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1319157820303712

[4] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," 2019.

[5] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert- Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. M. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, and D. Amodei, "Language models are few-shot learners," 2020.

[6] T. Wolf, L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi, P. Cistac, T. Rault, R. Louf, M. Funtowicz, J. Davison, S. Shleifer, P. von Platen, C. Ma, Y. Jernite, J. Plu, C. Xu, T. L. Scao, S. Gugger, M. Drame, Q. Lhoest, and A. M. Rush, "Transformers: State-of-the-art natural language processing," in Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations. Online: Association for Computational Linguistics, Oct. 2020, pp. 38–45. [Online]. Available: https://www.aclweb.org/anthology/2020.emnlp-demos.6

[7] Q. Lhoest, A. Villanova del Moral, Y. Jernite, A. Thakur, P. von Platen, S. Patil, J. Chaumond, M. Drame, J. Plu, L. Tunstall, J. Davison, M. Sˇasˇko, G. Chhablani, B. Malik, S. Brandeis, T. Le Scao, V. Sanh, C. Xu, N. Patry, A. McMillan-Major, P. Schmid, S. Gugger, C. Delangue, T. Matussie`re, L. Debut, S. Bekman, P. Cistac, T. Goehringer, V. Mustar, F. Lagunas, A. Rush, and T. Wolf, "Datasets: A community library for natural language processing," in Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing: System Demonstrations. Online and Punta Cana, Dominican Republic: Association for Computational

Linguistics, Nov. 2021, pp. 175–184. [Online]. Available: https://aclanthology.org/2021.emnlp-demo.21
[8] A. Cohan, F. Dernoncourt, D. S. Kim, T. Bui, S. Kim, W. Chang, and N. Goharian, "A discourse-aware attention model for abstractive summarization of long documents," Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers), 2018. [Online]. Available: http://dx.doi.org/10.18653/v1/n18-2097
[9] W. Krys´cin´ski, N. Rajani, D. Agarwal, C. Xiong, and D. Radev, "Book- sum: A collection of datasets for long-form narrative summarization," 2021.
[10] M. Yasunaga, J. Kasai, R. Zhang, A. Fabbri, I. Li, D. Friedman, and
D. Radev, "ScisummNet: A large annotated corpus and content-impact models for scientific paper summarization with citation networks," in Proceedings of AAAI 2019, 2019.
[11] M. Yasunaga, R. Zhang, K. Meelu, A. Pareek, K. Srinivasan, and
D. R. Radev, "Graph-based neural multi-document summarization," in
Proceedings of CoNLL 2017, 2017.
[12] M. Koupaee and W. Y. Wang, "Wikihow: A large scale text summariza- tion dataset," 2018.
[13] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena,
Y. Zhou, W. Li, and P. J. Liu, "Exploring the limits of transfer learning with a unified text-to-text transformer," Journal of Machine Learning Research, vol. 21, no. 140, pp. 1–67, 2020. [Online]. Available: http://jmlr.org/papers/v21/20-074.html
[14] H. W. Chung, L. Hou, S. Longpre, B. Zoph, Y. Tay, W. Fedus, E. Li,
X. Wang, M. Dehghani, S. Brahma, A. Webson, S. S. Gu, Z. Dai,
M. Suzgun, X. Chen, A. Chowdhery, S. Narang, G. Mishra, A. Yu,
V. Zhao, Y. Huang, A. Dai, H. Yu, S. Petrov, E. H. Chi, J. Dean,
J. Devlin, A. Roberts, D. Zhou, Q. V. Le, and J. Wei, "Scaling instruction-finetuned language models," 2022. [Online]. Available: https://arxiv.org/abs/2210.11416
[15] 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," CoRR, vol. abs/1910.13461, 2019. [Online]. Available: http://arxiv.org/abs/1910.13461
[16] C.-Y. Lin, "ROUGE: A package for automatic evaluation of summaries," in Text Summarization Branches Out. Barcelona, Spain: Association for Computational Linguistics, Jul. 2004, pp. 74–81. [Online]. Available: https://www.aclweb.org/anthology/W04-1013
[17] N. Reimers and I. Gurevych, "Sentence-bert: Sentence embeddings using siamese bert-networks," in Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics, 11 2019. [Online]. Available: https://arxiv.org/abs/1908.10084