Leveraging Transformers for Structured Highlight Generation in Scientific Literature

Rachana Nagaraju[*,†], Hosahalli Lakshmaiah Shashirekha[†]

Department of Computer Science, Mangalore University, Mangalore, Karnataka, India

Abstract

The rapid expansion of scientific literature has made it increasingly difficult for researchers to keep track of key contributions across disciplines. While abstracts provide useful overviews, they are often too detailed for quick comprehension. Research highlights address this gap by presenting concise, structured statements that emphasize the most important findings of a paper. Automating the generation of such highlights has the potential to accelerate knowledge discovery, improve the accessibility of research outputs, and support academic search engines and digital libraries in indexing scientific content more effectively. The *SciHigh* shared task is designed to explore the challenges of generating highlights from abstracts. In this paper, we, **Team MUCS**, present our transformer-based summarization pipeline built on the `T5-base` model, developed and submitted as part of the *SciHigh* shared task. The pipeline incorporated pre-processing, tokenization, and fine-tuning within a sequence-to-sequence (seq2seq) framework. Training is carried out using the AdamW optimizer with careful parameter selection, while generation employed constrained decoding strategies to balance informativeness and fluency. The design emphasized capturing both the lexical and semantic correspondence between abstracts and their highlights, with evaluation relying on Recall-Oriented Understudy for Gisting Evaluation (ROUGE) - a widely used metrics for summarization. Our system achieved a **ROUGE-L score of 0.2208**, placing us at **7th rank** on the official leaderboard. These results highlight the strength of fine-tuned transformer architectures for scientific highlight generation, while also revealing opportunities for further improvement through the integration of retrieval-augmented approaches, hybrid architectures, or domain-adaptive pretraining. Overall, our study demonstrates how modern neural summarization techniques can be effectively applied to generate structured highlights, ultimately contributing to more efficient scientific communication and knowledge dissemination.

Keywords

Research Highlight Generation, Scientific Summarization, Transformer Models, T5, Sequence-to-Sequence model, Automatic Text Summarization

1. Introduction

Digital repositories and online journals publish thousands of articles daily, making efficient mechanisms for information retrieval and comprehension essential [1, 2]. But the unprecedented growth of scientific literature in recent years has made it increasingly difficult for researchers to stay updated with key findings across domains. Traditionally, abstracts serve as the primary medium for summarizing research papers. While they provide valuable overviews, abstracts are often lengthy and written in prose, making them less accessible for quick scanning, particularly on mobile devices or in time-sensitive scenarios.

Research highlights are introduced by publishers as a complementary form of scientific summarization. Unlike abstracts, highlights are short, bullet-style statements that emphasize the most significant contributions of a study. They have been shown to aid rapid comprehension, improve discoverability in academic search engines, and enhance metadata quality for digital libraries [3, 4]. Automating the generation of such highlights can significantly reduce the cognitive load on researchers and streamline the scientific communication process. The difference between abstract and highlights is illustrated in Table 1. It can be observed that highlights provide concise and accessible summaries compared to longer and more detailed abstracts.

Forum for Information Retrieval Evaluation, December 17-20, 2025, Varanasi, India

[*] Corresponding author.

[†] These authors contributed equally.

rachananagaraju20@gmail.com (R. Nagaraju); hlsrekha@mangaloreuniversity.ac.in (H. L. Shashirekha)

0000-0002-9421-8566 (H. L. Shashirekha)

Table 1
Sample Abstract and Corresponding Author-Written Highlights from the SciHigh Dataset

Abstract	Highlights
We use a controlled experiment to analyze the impact of watching different types of educational traffic campaign videos on overconfidence of undergraduate university students in Brazil. The videos have the same underlying traffic educational content but differ in the form of exhibition. We find that videos with shocking content (Australian school) are more effective in reducing drivers' overconfidence, followed by those with punitive content (American school). We do not find empirical evidence that videos with technical content (European school) change overconfidence. Finally, this paper also introduces how to use machine learning techniques to mitigate the usual subjectivity in the design of the econometric specification.	• We use a controlled experiment to analyze the impact of watching different types of educational traffic campaign videos on overconfidence. • We find that videos with shocking content (Australian school) are more effective in reducing drivers' overconfidence. • We do not find empirical evidence that videos with technical content (European school) change overconfidence. • This paper also introduces how to use machine learning techniques to mitigate the usual subjectivity in the design of the econometric specification.

SciHigh[1] shared task at FIRE[2] 2025 aims to explore the challenges of developing systems capable of automatically generating research highlights directly from scientific paper abstracts. This initiative provides a benchmark setting for advancing automated scientific summarization research. In this paper, we - team **MUCS** describe the models proposed for addressing the challenges of developing systems capable of automatically generating research highlights directly from scientific paper abstracts. Our proposed approach is based on `T5-base` model. The scientific abstracts are first tokenized and encoded as the input sequence, while the corresponding highlight-style statements are used as the target sequence. The model is then fine-tuned in a seq2seq framework, where it learns to generate concise highlight-style statements from the given abstracts. Our code is available on GitHub[3] to reproduce the results and explore further. Our system achieved a **ROUGE-L score of 0.2208**, securing the **7th** position on the official leaderboard. This outcome demonstrates the effectiveness of transformer-based architectures for generating research highlights, while also highlighting opportunities for future exploration through retrieval-augmented and domain-adaptive approaches. Overall, our work illustrates the potential of modern neural models to support researchers in navigating the ever-growing volume of scholarly literature.

The subsequent sections of this paper details the related works (Section 2), methodology (Section 3), experiments, results, and implications of our approach (Section 4), declaration on generative AI (Section 6), followed by conclusion and future works (Section 5).

2. Related Works

Generation of highlights from scientific articles has received increasing attention as a subdomain of summarization, particularly motivated by the need to produce concise and domain-relevant summaries of research contributions. Various approaches have explored this task ranging from extractive models to pre-trained transformer architectures and hybrid methods. Rehman et al. [5] proposed a pointer-generator network enhanced with a coverage mechanism and SciBERT embeddings, specifically targeting highlight generation from scientific abstracts. Their model achieved ROUGE-1, ROUGE-2, and ROUGE-L scores of 41.78, 18.63, and 31.46 respectively on the CSPubSum dataset and 42.13, 18.91, and 31.58 on MixSub, outperforming traditional baselines. The system demonstrated strong handling of

[1] https://sites.google.com/jadavpuruniversity.in/scihigh2025
[2] https://fire.irsi.org.in/fire/2025/home
[3] https://github.com/rachanabn20/SciHigh-2025

inherent scientific terminology and reduced redundancy during generation. However, the reliance on pointer networks introduced an extractive bias, preventing truly abstractive and novel rephrasings. In a related study, Rehman et al. [6] explored the use of ELMo contextual embeddings to enrich the semantic representations in highlight generation models. While the method improved word-level context understanding and achieved competitive scores (ROUGE-L of 26.42 on CSPubSum), its summarization quality lagged behind transformer-based models, largely due to ELMo's limited capacity to handle long dependencies and capture sentence-level discourse structure.

Rehman et al. [7] further introduced an Named Entity Recognition (NER)-augmented summarization pipeline, where named entity information was integrated during the generation process. This technique improved content relevance and model interpretability and was shown to increase ROUGE-1 and METEOR scores by approximately 1.5–2 points over non-NER baselines. However, its effectiveness was heavily dependent on the accuracy of the underlying NER module, particularly for domain-specific entities. In a benchmarking study, Rehman et al. [8] evaluated several pre-trained summarization models and analyzed their performance on scientific text. The study reported that models such as T5 and BART struggled with domain mismatch unless fine-tuned properly, and models pre-trained on newswire often failed to generalize to scientific abstracts. Rehman et al. [9] also proposed a hybrid architecture combining extractive components with neural abstractive summarization. Though effective at maintaining key phrases from abstracts, the methodology lacked deep abstraction and was sensitive to input noise and alignment mismatches.

Xiao et al. [10] proposed PRIMERA, a pyramid-based pretraining strategy optimized for long and multi-document scientific summarization. Their model achieved impressive scores, reporting ROUGE-1/2/L as 47.2/20.8/39.6 on long document datasets, and outperformed BART and PEGASUS in terms of sentence-level coherence. PRIMERA explicitly models sentence dependencies and performs well in scenarios requiring reasoning over multiple sentences, but it comes at the cost of significant computational overhead and susceptibility to hallucination, particularly in domain-specific tasks. Lewis et al. [11] presented Retrieval-Augmented Generation (RAG), a hybrid architecture that pairs dense passage retrievers with generative models to inject factual information into generated summaries. This approach demonstrated improved factual accuracy in knowledge-intensive tasks but posed engineering challenges due to its dependency on retrieval quality and alignment with the generator. Further, performance often degraded when the retriever failed to retrieve relevant passages, especially in unseen domains.

Zhang et al. [12] introduced PEGASUS, a summarization-oriented pretraining framework where key sentences are masked and reconstructed. PEGASUS achieved state-of-the-art ROUGE-L scores (45.16 on scientific summarization benchmarks) and aligned closely with actual summarization tasks. Its pretraining objective significantly boosted performance in low-resource scenarios. However, the model's effectiveness was reliant on high-quality sentence masking heuristics, and its training demands were computationally expensive. Lewis et al. [13], in contrast, proposed BART, a denoising autoencoder that is commonly fine-tuned for summarization tasks. BART has become a popular baseline for scientific summarization due to its robustness and fluency, achieving a ROUGE-L of up to 36.93 on the arXiv dataset. However, like many generative models, it sometimes hallucinates facts and lacks grounding in the input abstract.

La Quatra et al. [14] developed THExt, a highlights extraction model trained on author-provided highlights. Their system relied on selecting sentences based on transformer-encoded importance scores. It offered efficiency and simplicity but was inherently extractive, thus incapable of performing paraphrasing or abstraction beyond sentence boundaries. Goyal et al. [15] explored few-shot and zero-shot summarization using prompt-based large language models. These models, including GPT-3 and its successors, demonstrated strong fluency and generalization ability. Without fine-tuning, they generated highlights of competitive quality. However, they require extensive computational resources and often hallucinate scientific terminology or infer conclusions not in the source text.

In summary, highlight generation techniques range from lightweight, domain-trained models [5, 6, 7] to large, general-purpose summarization frameworks [12, 13, 10]. Models such as PEGASUS and PRIMERA offer strong abstractive capacity and high ROUGE scores, but are computationally expensive and somewhat opaque. Pointer-generator and NER-augmented methods provide more interpretable and

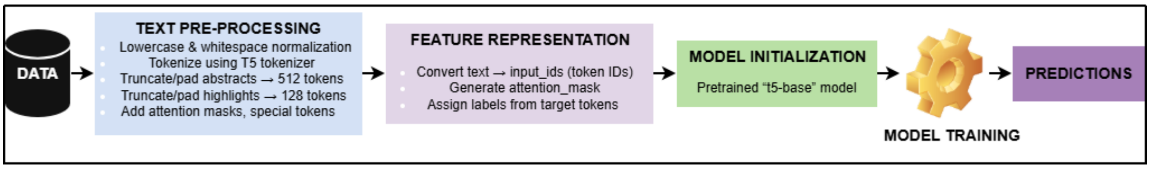
DATA
TEXT PRE-PROCESSING
Lowercase & whitespace normalization
Tokenize using T5 tokenizer
Truncate/pad abstracts → 512 tokens
Truncate/pad highlights → 128 tokens
Add attention masks, special tokens
FEATURE REPRESENTATION
Convert text → input_ids (token IDs)
Generate attention_mask
Assign labels from target tokens
MODEL INITIALIZATION
Pretrained "t5-base" model
MODEL TRAINING
PREDICTIONS

Figure 1: Proposed Framework for Highlight Generation

domain-aligned summaries while remaining lightweight, though they often trend toward extractiveness. Hybrid and retrieval-based models like RAG improve factual accuracy but struggle with complexity. Prompt-based LLMs show strong zero-shot potential and fluency but face issues with hallucination and cost. The extensive work by Rehman et al. across multiple architectures highlights the need for cost-effective and interpretable models tailored specifically to scientific summarization, balancing performance and practicality.

3. Methodology

The methodology for the SciHigh shared task is designed around a transformer-based summarization pipeline. In our approach, the abstracts are first pre-processed to ensure consistency in length and format. These pre-processed texts are then transformed into feature representations using the Text-to-Text Transfer Transformer `(T5)-base`[4] tokenizer, which encodes both the abstracts and their corresponding highlights.

Finally, a seq2seq model based on `T5-base` is fine-tuned on the training data to map abstracts into concise, highlight-style statements.

T5 model is a pre-trained seq2seq architecture introduced by Google Research. Built on the Transformer architecture, T5 is trained on a large corpus known as the C4 dataset, allowing it to learn powerful language representations. It treats every Natural Language Processing (NLP) task as a text-to-text problem, enabling a unified approach to tasks like translation, summarization, and classification. The `T5-base` variant used in our work contains 220 million parameters, balancing performance and computational efficiency.

The proposed pipeline enables the system to learn the structural differences between abstracts and highlights, ultimately producing outputs that resemble author-written highlights. The overall architecture of our proposed retrieval framework is illustrated in Figure 1.

3.1. Text Pre-processing

MixSub-SciHigh dataset - a subset of MixSub dataset [5] provided for SciHigh shared task contains research articles collected from ScienceDirect across different scientific domains, published in 2020. Each instance has an abstract along with the corresponding author-written research highlights. Since transformer models require uniform tokenized inputs, the following pre-processing steps are applied to ensure consistency:

- **Lowercasing and whitespace normalization:** All text is converted to lowercase and unnecessary whitespace characters are stripped off.

- **Input length standardization:** Abstracts are tokenized using the T5 tokenizer and truncated or padded to a maximum length of 512 tokens to fit the model's input constraints.

- **Target length standardization:** Abstracts are tokenized and truncated or padded to 512 tokens, while research highlights are standardized to 128 tokens.

[4] https://huggingface.co/docs/transformers/en/model_doc/t5

- **Special tokens:** Padding tokens and sentence delimiters (`<pad>`, `</s>`) are preserved to maintain sequence boundaries.

This pre-processing ensures that both abstracts and highlights are consistently represented as fixed-length sequences suitable for transformer input.

3.2. Feature Representation

After pre-processing, textual inputs are transformed into numerical features using the `AutoTokenizer` from HuggingFace:

- **Input IDs:** Each abstract is mapped into a sequence of integer token IDs representing words and sub-words.
- **Attention masks:** Binary masks are generated to distinguish between padded tokens and valid tokens. This prevents the model from attending to padded positions during training.
- **Labels:** Highlights are similarly tokenized and mapped to integer IDs, which serve as ground-truth labels during model training.

These feature representations capture the semantic information of abstracts while maintaining structural alignment with highlights.

3.3. Model Training

seq2seq models are widely used for text generation tasks, as they learn to transform an input sequence into a corresponding output sequence. We fine-tune a pre-trained `T5-base`[5] model using the HuggingFace `AutoModelForSeq2SeqLM` API to map abstracts (source sequence) into highlights (target sequence) using the following steps:

- **Batching:** Training and Validation sets are divided into mini-batches for efficient GPU utilization.
- **Forward pass:** Each batch of inputs (input IDs, attention masks, labels) is passed into the model. The decoder generates highlights corresponding to the abstract.
- **Loss computation:** Cross-entropy loss is computed between predicted sequences and reference highlights.
- **Optimization:** The AdamW optimizer updates model parameters, with weight decay to prevent overfitting.
- **Gradient management:** Gradients are reset after every batch using `optimizer.zero_grad()` to ensure stability.
- **Epochs:** The model is trained for two epochs, with validation performed after each epoch to track ROUGE scores.

The hyperparameters used in training the model are listed in Table 2.

4. Experiments and Results

The implementation relies on PyTorch[6] for Deep Learning, HuggingFace Transformers[7] for seq2seq modeling, and the Evaluate library[8] for computing metrics. Details of the dataset are shown in Table 3.

[5] https://huggingface.co/t5-base

[6] https://pytorch.org/

[7] https://huggingface.co/docs/transformers/index

[8] https://huggingface.co/docs/evaluate

Table 2
Hyperparameters used for model training

Hyperparameter	Value
Model name	T5-base
Maximum input length	512
Maximum target length	128
Training batch size	4
Evaluation batch size	4
Optimizer	AdamW
Learning rate	3e-4
Number of epochs	2
Loss function	Cross-entropy
Output directory	./sci-summary

The models are evaluated based on ROUGE-1, ROUGE-2, ROUGE-L, and ROUGE-Lsum scores. ROUGE is a widely used metric for summarization tasks. While **ROUGE-1** measures unigram overlap between generated and reference highlights, reflecting basic content coverage, **ROUGE-2** captures bigram overlap, providing insights into fluency and phrase-level accuracy. **ROUGE-L** evaluates the longest common subsequence (LCS), highlighting the ability of the system to preserve structural consistency and **ROUGE-Lsum** is a summary-level variant of ROUGE-L, designed to evaluate matches across multiple sentences, making it particularly suitable for highlight-style summarization. These metrics are chosen since they are widely adopted in automatic text summarization benchmarks, including the SciHigh shared task at FIRE 2025.

Table 3
Dataset statistics for MixSub-SciHigh

Split	Number of Instances
Training Set	10,000
Validation Set	1,985
Test Set	1,840

4.1. Results

ROUGE scores obtained by the prosposed models on Validation and Test sets are shown in Table 4. Our proposed model obtained a ROUGE-L score of **0.2208** on the Test set with 7th rank. The results illustrate that the model performed reasonably well on Validation set, with ROUGE-1 score 0.3333 and ROUGE-L score 0.2442. On the Test set, there is a small drop (ROUGE-1: 0.2961, ROUGE-L: 0.2208), showing that while the model generalizes, it struggles a bit with unseen abstracts from different domains.

Table 4
Performances of the proposed model on Validation and Test sets

Metric	Validation	Test
ROUGE-1	0.3333	0.2961
ROUGE-2	0.1143	0.1005
ROUGE-L	0.2442	0.2208
ROUGE-Lsum	0.2444	0.2500

Interestingly, the ROUGE-Lsum score on the Test set (0.2500) is slightly higher than on the Validation set (0.2444). This suggests that the generated highlights capture the sentence-level summary structure fairly well. Overall, the fine-tuned `t5-base` model can produce concise and structured highlights, but there is still room to improve through domain adaptation, longer training, or combining with retrieval-based methods. Figure 2 shows the performance of the all the participating teams in the shared task.

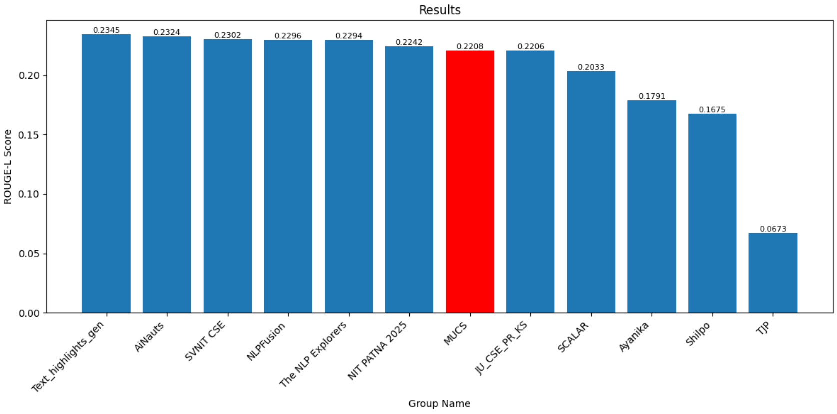

Figure 2: ROUGE-L scores of Participating Teams in the Shared Task

5. Conclusion and Future Work

In this work, we describe the model submitted to SciHigh shared task at FIRE 2025 on research highlight generation using abstracts. Our approach based on fine-tuning the `t5-base` model achieved a ROUGE-L score of **0.2208** on the official test set, placing our team- **MUCS** at the **7th rank** on the leaderboard. The results indicate that transformer-based summarization models are effective in generating concise and structured highlights, though some performance gaps remain compared to the top-ranked systems. Efforts towards improving generalization across diverse scientific domains and exploring lightweight adaptations to better capture domain-specific writing styles will be explored further. Such enhancements could further improve the quality and readability of automatically generated highlights.

6. Declaration on Generative AI

Generative AI tools are used in assisting with language refinement and formatting of certain sections of this paper. All technical content, experiments, results, and interpretations are conceived, implemented, and validated independently. The AI tool does not contribute to the design of experiments, execution of code, data analysis, or interpretation of results. Final responsibility for the accuracy and integrity of the paper remains with the contributors.

References

[1] L. Bornmann, R. Mutz, Growth Rates of Modern Science: A Bibliometric Analysis Based on the Number of Publications and Cited References, Journal of the Association for Information Science and Technology 66 (2015) 2215–2222. doi:10.1002/asi.23329.

[2] S. Fortunato, C. T. Bergstrom, et al., Science of Science, Science 359 (2018) eaao0185. doi:10.1126/science.aao0185.

[3] P. Li, W. Lam, A Survey on Abstractive Text Summarization, Proceedings of the AAAI Conference on Artificial Intelligence 34 (2020) 9380–9387. doi:10.1609/aaai.v34i05.6396.

[4] R. Nallapati, B. Zhou, C. Gulcehre, B. Xiang, et al., Abstractive Text Summarization Using Sequence-to-Sequence RNNs and Beyond, in: Proceedings of the 20th SIGNLL Conference on Computational Natural Language Learning (CoNLL), 2016, pp. 280–290. doi:10.18653/v1/K16-1028.

[5] T. Rehman, D. K. Sanyal, S. Chattopadhyay, P. K. Bhowmick, P. P. Das, Generation of highlights from research papers using pointer-generator networks and SciBERT embeddings, IEEE Access 11 (2023) 91358–91374. doi:10.1109/ACCESS.2023.3292300.

[6] T. Rehman, D. K. Sanyal, S. Chattopadhyay, Research highlight generation with ELMo contextual embeddings, Scalable Computing: Practice and Experience 24 (2023) 181–190.

[7] T. Rehman, D. K. Sanyal, P. Majumder, S. Chattopadhyay, Named entity recognition based automatic generation of research highlights, in: Proceedings of the Third Workshop on Scholarly Document Processing (SDP 2022) collocated with COLING 2022, Association for Computational Linguistics, Gyeongju, Republic of Korea, 2022, pp. 163–169.

[8] T. Rehman, S. Das, D. K. Sanyal, S. Chattopadhyay, An analysis of abstractive text summarization using pre-trained models, in: Proceedings of International Conference on Computational Intelligence, Data Science and Cloud Computing: IEM-ICDC 2021, Springer, 2022, pp. 253–264.

[9] T. Rehman, D. K. Sanyal, S. Chattopadhyay, P. K. Bhowmick, P. P. Das, Automatic generation of research highlights from scientific abstracts, in: 2nd Workshop on Extraction and Evaluation of Knowledge Entities from Scientific Documents (EEKE'21), collocated with JCDL'21, 2021.

[10] W. Xiao, G. Carenini, et al., PRIMERA: Pyramid-based Masked Sentence Pre-training for Multi-document Summarization, in: Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (ACL), 2022, pp. 7893–7909. doi:10.18653/v1/2022.acl-long.542.

[11] P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, et al., Retrieval-augmented Generation for Knowledge-Intensive NLP, in: Advances in Neural Information Processing Systems (NeurIPS), volume 33, 2020, pp. 9459–9474.

[12] J. Zhang, Y. Zhao, M. Saleh, P. J. Liu, PEGASUS: Pre-training with Extracted Gap-sentences for Abstractive Summarization, in: Proceedings of the 37th International Conference on Machine Learning (ICML), 2020, pp. 11328–11339.

[13] M. Lewis, Y. Liu, N. Goyal, M. Ghazvininejad, A. Mohamed, O. Levy, V. Stoyanov, L. Zettlemoyer, BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension, arXiv preprint arXiv:1910.13461 (2019). URL: https://arxiv.org/abs/1910.13461.

[14] M. La Quatra, M. Caselle, T. Oberhauser, THExt: A Transformer-based Highlights Extractor for Scientific Papers, in: Proceedings of the 29th International Conference on Computational Linguistics (COLING), 2022, pp. 4932–4944. doi:10.18653/v1/2022.coling-1.435.

[15] A. Goyal, R. Sharma, H. Patel, Prompting Large Language Models for Scientific Highlight Generation, arXiv preprint arXiv:2311.04567 (2023) 1–8.