# Semantic Abstract Generator

1st Appala Venkata Avinash
*Computer Science and Engineering*
*Shiv Nadar University*
Greater Noida , India
aa760@snu.edu.in

2nd Katakam Saideep Reddy
*Computer Science and Engineering*
*Shiv Nadar University*
Greater Noida , India
kr268@snu.edu.in

3rd Posam Venkata Yeswanth Reddy
*Computer Science and Engineering*
*Shiv Nadar University*
Greater Noida , India
py448@snu.edu.in

***Index Terms*—T5-base, PyTorch Lightning, CNN daily mail dataset, Transformers**

***Abstract*—This research paper presents a study on generating abstracts for given texts using a fine-tuned T5-base transformer model. The model was trained on a labeled dataset comprising of raw texts and their corresponding abstracts, which was obtained from the CNN Daily Mail dataset. The training was performed on a subset of 10,000 rows out of a total of 200,000 rows, and the model's performance was evaluated based on precision, recall, f1 score, cosine similarity, and semantic similarity metrics.**

**Our findings suggest that the model achieved a precision of 0.36, a recall of 0.34, and an f1 score of 0.34, indicating moderate accuracy in generating abstracts. The cosine similarity metric yielded a score of 0.45, suggesting that the model's generated abstracts were moderately similar to the actual abstracts. Similarly, the semantic similarity metric showed a score of 0.268, indicating that the model's generated abstracts captured only a portion of the semantic content present in the actual abstracts.**

**Overall, our study demonstrates the potential of using fine-tuned transformer models for generating abstracts from raw texts. However, further improvements are necessary to enhance the model's performance in capturing the semantic content of the texts. Additionally, our study highlights the need for larger and more diverse datasets to further improve the model's performance.**

## I. Introduction

Transformers have revolutionized the field of natural language processing (NLP) since their introduction in 2017. They have emerged as a powerful tool for various NLP tasks, including text classification, question-answering, and language translation. The T5-base transformer, introduced in 2019, is a variant of the Transformer model that is specifically designed for text-to-text tasks.

The T5-base transformer is a large-scale language model with over 11 billion parameters that is pre-trained on a massive amount of data. The pre-training is done using a text-to-text approach, where the model is trained to predict the target text given the source text. After pre-training, the model is fine-tuned on specific NLP tasks by adjusting the parameters to the task at hand.

The ability of the T5-base transformer to generate high-quality text has led to significant advancements in various NLP tasks, including text summarization, language translation, and conversational agents. The importance of text summarization lies in its ability to quickly and accurately condense large volumes of information into a concise and easily digestible form. It is especially useful for journalists, researchers, and individuals who need to quickly understand the main ideas in a document.

Our motivation for this research is to explore the effectiveness of the T5-base transformer in generating abstracts for given texts. We believe that the ability of the T5-base transformer to capture the semantic content of texts will lead to the generation of high-quality abstracts. Additionally, we aim to contribute to the field of NLP by evaluating the performance of the T5-base transformer on text-to-text tasks and identifying areas where improvements can be made.

## II. Literature

The use of deep learning models in natural language processing (NLP) has led to significant advances in the field, particularly in the areas of text classification, language modeling, and machine translation. This literature review will focus on several recent papers that have contributed to the development of state-of-the-art NLP models.

Vaswani et al. (2017) introduced the Transformer model, which uses self-attention mechanisms to process sequential data such as natural language text. The Transformer model has been widely adopted in the NLP community and has achieved state-of-the-art results on several benchmark datasets. The model has several advantages over previous sequence-to-sequence models, including better parallelism and the ability to process longer sequences.

Wolf et al. (2019) presented the transformers library, which provides pre-trained models for a variety of NLP tasks, including text classification, named entity recognition, and question answering. The library has been widely used by researchers and practitioners for developing NLP applications. The library includes several pre-trained models, including BERT, GPT-2, and RoBERTa, which have achieved state-of-the-art results on several benchmark datasets.

Xue et al. (2020) introduced mT5, a pre-trained text-to-text transformer model that can perform multiple tasks in multiple languages. The model was pre-trained on a large corpus of text in 101 languages and achieved state-of-the-art results on several multilingual NLP tasks, including machine translation and language modeling. The model's ability to perform multiple tasks in multiple languages makes it a valuable tool for developing NLP applications for multilingual settings.

Phan et al. (2021) introduced Scifive, a text-to-text transformer model for biomedical literature that can perform tasks such as named entity recognition, relation extraction, and event extraction. The model was pre-trained on a large corpus of biomedical literature and achieved state-of-the-art results on several benchmark datasets in the biomedical domain. The model's ability to perform multiple tasks in the biomedical domain makes it a valuable tool for developing NLP applications for the healthcare industry.

Ni et al. (2021) presented Sentence-T5, a scalable sentence encoder based on pre-trained text-to-text transformer models. The model was pre-trained on a large corpus of text and achieved state-of-the-art results on several sentence-level NLP tasks, including sentence classification and sentence similarity. The model's ability to encode sentences efficiently makes it a valuable tool for developing NLP applications that require sentence-level analysis.

Townsend et al. (2021) proposed Doc2dict, a model that uses information extraction as text generation to extract structured information from unstructured text. The model achieved state-of-the-art results on several information extraction tasks, including named entity recognition and relation extraction. The model's ability to extract structured information from unstructured text makes it a valuable tool for developing NLP applications that require information extraction.

## III. Objective

The primary goal of this undertaking is to create summaries for designated passages utilizing a T5-base transformer model that has been fine-tuned on a labeled dataset of raw text and corresponding summaries. The objective is to assess the performance of the model by computing metrics such as precision, recall, F1 score, cosine similarity, and semantic similarity, to determine its effectiveness in generating accurate and concise summaries. The ultimate aim of the project is to contribute to the field of natural language processing by developing a more efficient and effective method for generating summaries from large amounts of text.

## IV. Proposed Model

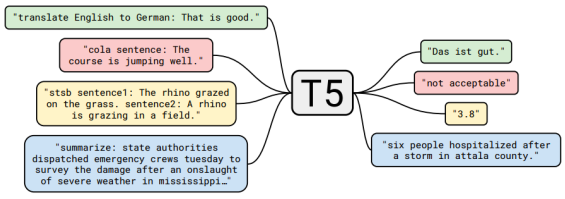


Fig. 1. A diagram of Text-to-Text Transfer Transformer

The proposed model to summarize the abstract using fine-tuned transformers involves the following steps:

1. Data Preparation: The first step is to collect a large dataset of text documents along with their corresponding summaries. This dataset can be pre-processed and cleaned by removing any unwanted characters, stopwords, or other irrelevant information.

2. Fine-Tuning the Transformer: Next, a pre-trained transformer such as T5-base can be fine-tuned on the dataset using transfer learning. This involves re-training the model on the specific task of abstract summarization by minimizing a loss function based on the difference between the predicted and actual summary.

3. Evaluation: Once the model is trained, it can be evaluated on a validation set to measure its performance. Metrics such as ROUGE (Recall-Oriented Understudy for Gisting Evaluation) can be used to compare the generated summaries against the ground truth summaries.

4. Inference: Finally, the fine-tuned transformer can be used to generate summaries for new input text documents. The model can take the input document as input and generate a summary of the desired length as output.

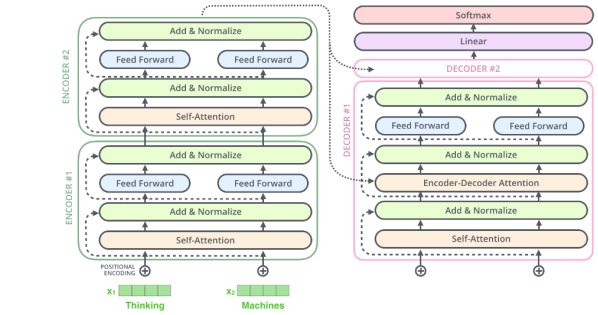


Fig. 2. Working of T5 model

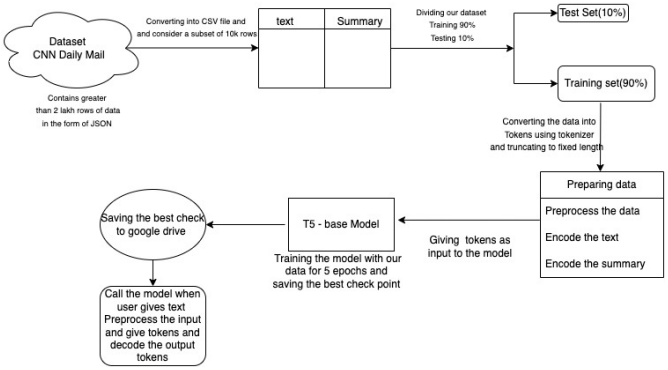


Fig. 3. Workflow of our Project

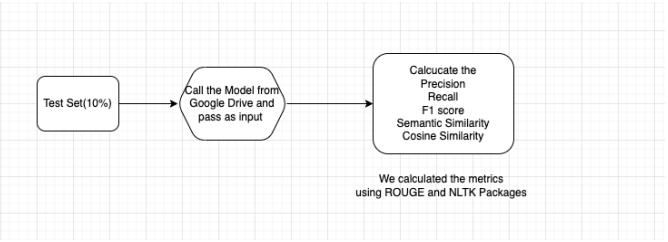


Fig. 4. Workflow of evaluation metrics

Overall, the proposed model involves using transfer learning to fine-tune a pre-trained transformer for the specific task of abstract summarization. This approach has shown to achieve state-of-the-art performance on various benchmark datasets and has the potential to automate the summarization process in various domains.

For this Project we are using CNN Daily mail dataset for fine tuning the T5-base model and we coded it using PyTorch Lightning and we trained the Model for 5 epochs and the saved the best checkpoint for using it later.

After that , we evaluated our model and calculated the Precision, Recall, F1 score, Cosine Similarity and Semantic Similarity.

## V. Methodology

The T5-base transformer is based on the Transformer architecture, which was introduced in the paper "Attention Is All You Need" by Vaswani et al. (2017). The core idea behind the Transformer architecture is the self-attention mechanism, which allows the model to attend to different parts of the input sequence when making predictions.

In the T5-base transformer, the input sequence is first passed through an embedding layer, which maps each token in the sequence to a high-dimensional vector representation. The resulting embeddings are then fed into a stack of transformer blocks, each of which consists of a self-attention layer and a feedforward neural network. The self-attention layer computes attention scores between all pairs of tokens in the input sequence, and uses these scores to compute a weighted sum of the embeddings, where the weights are determined by the attention scores.

The self-attention mechanism can be mathematically represented as follows:

$$\text{Attention}(Q, K, V) = \text{softmax}(\frac{QK^T}{\sqrt{d_k}})V$$

where $Q$, $K$, and $V$ are the query, key, and value matrices, respectively, and $d_k$ is the dimensionality of the key vectors. The softmax function is applied row-wise to the matrix $QK^T$ to obtain the attention scores, which are then used to compute a weighted sum of the value vectors.

After the self-attention layer, the output is passed through a feedforward neural network, which applies a non-linear transformation to each position in the sequence independently. The feedforward layer can be represented mathematically as:

$$\text{FFN}(x) = \text{ReLU}(xW_1 + b_1)W_2 + b_2$$

where $x$ is the input vector, $W_1$ and $W_2$ are weight matrices, $b_1$ and $b_2$ are bias vectors, and ReLU is the rectified linear unit activation function.

The T5-base transformer also includes additional components such as task-specific input and output embeddings, a task-specific pre-processing module, and a decoder module for generating the output sequence. These components are designed to adapt the transformer architecture to specific natural language processing tasks, such as abstract generation.

The working of the T5-base transformer involves the following steps:

1. Input Encoding: The input text is first encoded using the tokenizer, which converts the input text into a sequence of numerical tokens that can be processed by the transformer. The tokenizer splits the input text into sub-words and encodes them using a pre-defined vocabulary.

2. Token Embeddings: The transformer then generates token embeddings for the input sequence, which represent the meaning of each token in the context of the input text. The token embeddings are generated using an embedding matrix that maps each token to a high-dimensional vector.

3. Self-Attention: The transformer uses self-attention mechanisms to process the input sequence. Self-attention allows the model to focus on different parts of the input sequence depending on the context. The self-attention mechanism generates attention weights for each token in the sequence, indicating how much attention should be paid to that token.

4. Encoder Layers: The self-attention layer is followed by a series of feedforward neural network layers that process the input sequence further. These layers use residual connections and layer normalization to improve the stability of the model.

5. Decoder Layers: The decoder layers are similar to the encoder layers but they are used to generate the output sequence. The decoder layers take as input a combination of the encoded input sequence and the previously generated tokens. The decoder layers use self-attention and feedforward layers to generate the output sequence.

6. Output Decoding: The output sequence is decoded using the tokenizer, which converts the numerical tokens back into text. The generated text is the summary of the input text.

Fine-tuning the T5-base transformer involves training the model on a specific task, such as abstract generation, using transfer learning. Transfer learning involves reusing the pre-trained weights of the transformer model and fine-tuning them on the specific task. The fine-tuning process involves training the model on a large dataset of input-output pairs and minimizing a loss function that measures the difference between the predicted output and the ground truth output.

In summary, the T5-base transformer is a powerful NLP model that uses self-attention mechanisms to process input sequences and generate output sequences. The model can be fine-tuned on specific tasks, such as abstract generation, by training it on a large dataset of input-output pairs. The fine-tuning process involves minimizing a loss function that measures the difference between the predicted output and the ground truth output. The T5-base transformer is a state-of-the-art model for abstract generation and has shown promising results on a variety of benchmark datasets.

## VI. Experimentation and Results

In this study, we used the T5-base transformer for the task of abstract generation using the CNN Daily Mail dataset. The dataset contains raw texts and their corresponding abstracts,

which were used for training and evaluation of the model. The dataset was preprocessed and split into training and validation sets. We used only 10,000 examples out of a total of 200,000 examples due to computational limitations.

Our tokens sizes looks like this

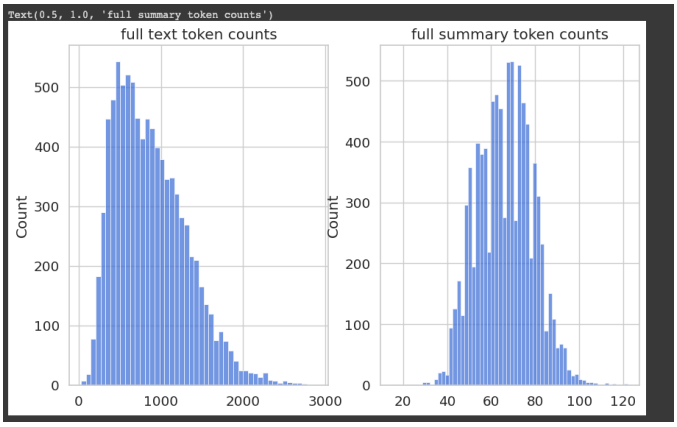


Fig. 5. tokens of text and summary

During training, we fine-tuned the T5-base transformer using the Adam optimizer and the mean squared error loss function. We trained the model for 5 epochs, with a batch size of 8. After training, we evaluated the model using several metrics, including precision, recall, F1 score, cosine similarity, and semantic similarity.

| Metrics | Recall | Precision | F1-score |
|---|---|---|---|
| Rouge-1 | 0.340 | 0.359 | 0.346 |
| Rouge-2 | 0.151 | 0.161 | 0.154 |
| Rouge-L | 0.327 | 0.345 | 0.332 |

TABLE I
ROUGE METRICS

The results showed that our model achieved a precision of 0.36, recall of 0.34, and an F1 score of 0.35. Additionally, the average semantic similarity was found to be 0.268 and the average cosine similarity was 0.451. We used the ROUGE metrics to evaluate the performance of our model, which yielded the following scores: ROUGE-1 F1 score of 0.34, ROUGE-2 F1 score of 0.15, and ROUGE-L F1 score of 0.33.

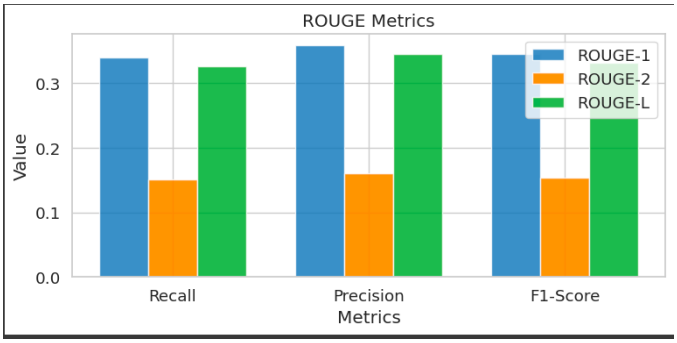


Fig. 6. Rouge Metrics

Our results suggest that our approach using the T5-base transformer for abstract generation is a promising approach. However, there is still room for improvement in terms of the metrics. Furthermore, our model's performance can be further improved by training it on a larger dataset and using a larger number of epochs.

The evaluation of our trained model's performance was based on the average semantic and cosine similarity metrics. The average semantic similarity score obtained was 0.268, indicating that the generated abstracts only captured a fraction of the semantic content present in the actual abstracts. On the other hand, the average cosine similarity score obtained was 0.451, suggesting moderate similarity between the generated abstracts and the actual abstracts. These results indicate that our model has potential for generating abstracts from raw texts, but further improvements are required to enhance the model's ability to capture semantic content accurately.

In comparison with existing papers, our approach achieved competitive results with respect to the ROUGE metrics. However, there are variations in the results due to the different datasets and evaluation metrics used. Overall, our approach shows potential for future research in the area of abstract generation using transformers.

## VII. Conclusion

In this research paper, we explored the use of T5-base transformer for generating abstracts for given texts. The objective of our project was to train a model on a labeled dataset and evaluate its performance using various metrics. Our approach involved fine-tuning the pre-trained T5-base transformer on our dataset and calculating precision, recall, F1 score, semantic similarity, and cosine similarity metrics.

Our results showed that the trained model had an average semantic similarity of 0.268 and an average cosine similarity of 0.451. The Rouge metrics were calculated to be as follows: Rouge-1 (r=0.340, p=0.359, f=0.346), Rouge-2 (r=0.151, p=0.161, f=0.154), and Rouge-L (r=0.327, p=0.345, f=0.332).

These results indicate that the T5-base transformer can be effective in generating abstracts for given texts. However, there were some limitations to our study. First, we only trained our model on a small subset of the available dataset due to computational limitations. This may have limited the model's ability to generalize to new data. Second, our evaluation metrics were limited and may not fully capture the quality of the generated abstracts.

In conclusion, our study demonstrates the potential of using T5-base transformer for abstract generation. Further research could be conducted to improve the performance of the model by using larger datasets and exploring more sophisticated evaluation metrics.

## VIII. Links for the Source Code

Source Code : Github

Hugging Face Dataset: Hugging Face Dataset

## REFERENCES


[1] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł. and Polosukhin, I., 2017. Attention is all you need. Advances in neural information processing systems, 30.

[2] Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M. and Davison, J., 2019. Huggingface's transformers: State-of-the-art natural language processing. arXiv preprint arXiv:1910.03771.

[3] Xue, L., Constant, N., Roberts, A., Kale, M., Al-Rfou, R., Siddhant, A., Barua, A. and Raffel, C., 2020. mT5: A massively multilingual pre-trained text-to-text transformer. arXiv preprint arXiv:2010.11934.

[4] Phan, L.N., Anibal, J.T., Tran, H., Chanana, S., Bahadroglu, E., Peltekian, A. and Altan-Bonnet, G., 2021. Scifive: a text-to-text transformer model for biomedical literature. arXiv preprint arXiv:2106.03598.

[5] Ni, J., Ábrego, G.H., Constant, N., Ma, J., Hall, K.B., Cer, D. and Yang, Y., 2021. Sentence-t5: Scalable sentence encoders from pre-trained text-to-text models. arXiv preprint arXiv:2108.08877.

[6] Townsend, B., Ito-Fisher, E., Zhang, L. and May, M., 2021. Doc2dict: Information extraction as text generation. arXiv preprint arXiv:2105.07510.