Comparing Fine-tuned T5 and Pegasus Models on News Summarization Task

Hira Nur Morca
Computer Engineering Department
Yildiz Technical University, 34220 Istanbul, Turkey
nur.morca@yildiz.edu.tr

***Özetçe* —Bu çalışmada, Transformers tabanlı modeller olan T5 ve PEGASUS'un haber özetleme görevi için fine-tune işleminden geçirilmesi ve performansının ölçülmesi işlemi yapılmıştır. Özetleme görevi için hazırlanmış verisetleri üzerindeki performansları değerlendirilmiş ve başarıları ölçülmüştür. Sonuçlar, Pegasus'un T5'ten daha iyi sonuçlar vermesiyle beraber her iki modelin de iyi ölçüde başarılı sonuç verdiğinigöstermektedir.**

Anahtar Kelimeler—Soyut Özetleme, Doğal Dil İşleme, Büyük Dil Modelleri, Haber Özetleme

***Abstract*—This paper explores the fine-tuning of two state-of-the-art transformer-based models, T5 (Text-to-Text Transfer Transformer) and PEGASUS (Pre-training with Extracted Gap-sentences for Abstractive Summarization Sequence-to-Sequence models), specifically for the task of news summarization. Their performance on summarization datasets is evaluated, examining metrics to assess improvements. Results indicate that both models achieve significant improvements in summarization quality, with PEGASUS outperforming T5 in terms of ROUGE scores.**

Keywords—Abstractive summarization, Natural Language Processing, Large Language Models, News Summarization

I. Introduction

Large Language Models are advanced artificial intelligence systems designed to understand texts written in natural languages and produce similar texts. These systems with complex architectures are trained on large-scale data and are capable of performing a number of natural language processing tasks, from question answering to content generation. One of these tasks, text summarization, aims to make a text shorter by focusing on the important and necessary points of an existing text [1]. Large Language Models have achieved almost human-level success in this process [2].

In this study, we aim to create a system that can summarize live news by integrating fine-tuned T5 and PEGASUS models from large language models. For the fine-tuning process, CNN/DailyMail, a dataset which is used frequently on the similar works, and open source datasets present on the HuggingFace environment were utilized. CNN/DailyMail is a dataset of news summaries extracted from CNN and DailyMail news sites [3].

The main motivation behind this work is to understand the architecture of deep learning, artificial neural networks and big language models by reviewing the literature and to implement the knowledge gained in order to produce a usable and accessible product. In this way, it is expected to contribute to the work done in this field, to understand it and to make its use widespread.

The T5 and PEGASUS models to be implemented in this study are the state-of-the-art models in the field of natural language processing. Both models are built on the Transformers architecture and are trained on large-scale data in accordance with their purposes. The T5 model performs many natural language processing tasks other than summarization, and was trained to perform summarization during training. PEGASUS is a model trained only on text summarization and is considered the best model in this field [1] [4].

The summarization work of LLMs is not only based on news summarization. LLMs are trained to summarize a variety of texts such as articles, dialogues, official documents, etc. In addition, LLMs follow two different ways of summarizing texts. These are 'extractive' summarization and 'abstractive' summarization. 'Extractive' summarization finds the most important sentences as a result of computations across the relevant text and extracts them without making any changes. 'Abstractive' summarization is a technique where the text is 'analyzed' by the large language model and summarized with 'its own sentences'[3]. The T5 and Pegasus models integrated in this study adopt the abstractive summarization technique. After the fine-tuning process, the performances of the models were measured with the ROUGE scores.

II. The Models and Their Architectures

A. The T5-Base Model

Developed by Google Research, T5, which stands for Text-to-Text Transfer Transformer, is a language model that can perform tasks specific to natural language processing, built on the Transformers architecture, as the name suggests. In this model, it has the capacity to perform many tasks such as translation between natural languages, text generation, determining the similarity of sentences with each other. Again, as the name suggests, it has text-to-text feature, which means that it takes text data as input and produces text as output[4].

T5 is based on Transformer architecture. In this architecture, the self-attention mechanism is used to 'pay attention' to different points of the input. With this mechanism, artificial intelligence models can process sequential data such as text in a healthy way. When we

talk about the main outlines of the T5 architecture, the first one is the Encoder-Decoder structure. Here, the encoder processes the input text into a 'hidden state' sequence. There are multiple Transformers blocks and each block has a multi-head self-attention layer, a feed-forward layer, a residual connection and layer normalization. The Decoder side uses these 'hidden states' to generate the output text. Again, multiple Transformers blocks are involved on this side, as well as a multi-headed cross-attention mechanism that analyzes the output generated by the encoder in accordance with the input [4][5]. A representative visualization of the architecture is shown in Figure 1.

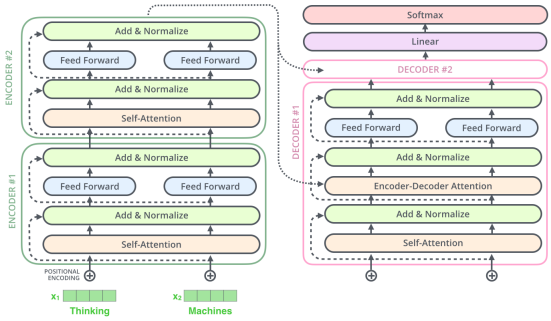

Figure 1 The Transformers Architecture

Like any LLM, T5 cannot process text data directly. In this sense, various operations need to be performed on the text data. The first of these operations is 'tokenization'. For T5, this is done using the SentencePiece library, which is an unsupervised text tokenizer. Unlike common tokenizers, where text is separated by spaces or predefined rules, SentencePiece treats text as a string of characters and separates it into complete words or word fragments [6]. An example is given in the Figure 2.

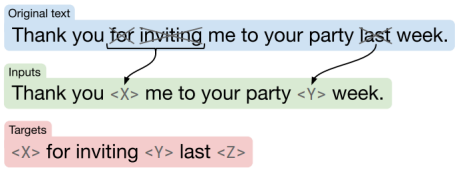

Figure 2 A representative of Sentence Piece Technique

In the training of the T5 model, 'Span Corruption' and 'Text Infilling' were used as 'unsupervised objectives'. Span Corruption is a technique based on hiding specific tokens in succession and then having the model predict or generate the hidden parts. In order to apply this technique, the data in the input text is first hidden to a certain extent by special tokens during training. The model tries to predict the hidden parts by examining the unhidden parts, and through this step, it learns to understand the context and produce appropriate outputs.

B. The Pegasus Model

PEGASUS, Pre-training with Extracted Gap-sentences for Abstractive Summarization, developed by Google Research, is a big language model specifically designed for text summarization. Like the T5 model, PEGASUS is based on the standard Encoder-Decoder Transformers architecture. Since it is a model that focuses on summarization, it naturally works in text-to-text format. Where PEGASUS differs from T5 and other major language models is that it is directly trained for summarization in the first training phase. This is achieved by applying GSG, Gap Sentences Generation, and MLM, Masked Language Modeling, techniques developed specifically for its training [1].

Gap Sentences Generation is a concept that 'mimics' summarization. It does this by hiding the most important sentences in the input text data. This text input with gaps is given to the model and the model is trained to predict these gaps. The aim here is to ensure that the model captures the context of the text and produces content accordingly. Masked Language Modeling, another method used to train PEGASUS, is a widely used model. Similar to GSG, it is realized by replacing the tokens in the texts given as input with random [MASK] tokens and giving them to the model. With this method, the model understands the relationships between words and can make sense of the text from a broader perspective [1][7]. A representative visualization of Gap Sentences technique and the model's architecture is given in the Figure 3.

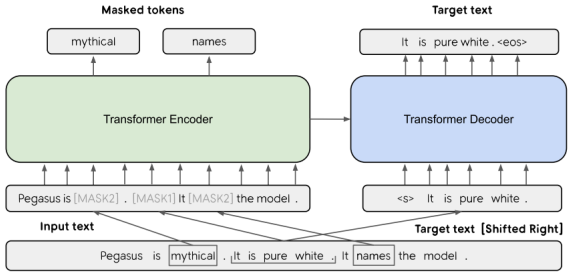

Figure 3 The Transformers Architecture

III. The Fine-tuning Process

The versions of the models to be fine tuned are T5 Base for T5 and PEGASUS-Large for PEGASUS. The datasets to be given to the models are 'CNN/DailyMail' for T5 and 'Summary Of News Articles' for PEGASUS. The CNN/DailyMail dataset is a dataset of 286,817 training, 13,368 validation and 11,487 test pairs of news articles and summaries from CNN and DailyMail websites used in abstract text summarization tasks [3]. This dataset contains an 'id' for each data item, an 'article' variable that holds the main text and a 'highlight' variable that holds the summary. The summary part contains a summary of the text given directly. Summary Of News Articles is a dataset published as open source on the internet. There are 56,216 pieces of data in this dataset. The column containing the main text of the news article is labeled as 'document' and the column containing the summaries of these news articles is labeled as 'summary'.

In order for the fine-tuning process to be carried out properly, the texts given as input to the model must go

through certain processes. These processes can be divided into preprocessing and tokenization. In this step, the text is edited by removing gaps and empty sentences. In the tokenization step, the texts are vectorized using the tokenizer of the model. Here, while the 'max-input-length' value is 1024, the 'max-output-size' value is set to 128. In the T5 model, 'summarize:' is also added at the beginning of the text for summarization. After these preparations, the models were fine-tuned in a high processor and large disk environment.

A. Fine-tuning Of the T5 Model

The fine-tuning process largely involves the values entered as hyperparameters in addition to the dataset given to the model. At this point, in the training of the T5 model, various hyperparameters were assigned values deemed appropriate. Batch size was chosen as 4 and learning rate was 3e-4. AdamW was used for optimization. Gradient accumulation steps are 4 and weight decay is 0.01. The model entered the evaluation step every 100 batch steps and the best model is determined according to the ROUGE1 score. The model was trained on 18,750 batches and fine-tuning was completed.

B. Fine-tuning of the PEGASUS Model

Since the PEGASUS model was previously fine-tuned with CNN/DailyMail data, a different dataset was used. As hyperparameters, batch size was again chosen as 4 and learning rate was 3e-4. AdamW was used for optimization. Gradient accumulation steps are 4 and weight decay is 0.01. The model entered the evaluation step at every 500 batch steps and the best model was determined according to the ROUGE1 score. The training of Pegasus took 8430 batches and the fine-tuning process was completed.

IV. Results

In this section, the results and the performances of the models are presented and compared.

Model sizes and number of parameters are given in the table 1. According to this table, it is seen that the PEGASUS model is larger in terms of both size and number of parameters.

Model Name	Size	Parameter Count
T5 Base	892MB	222M
Pegasus Large	2.28GB	570M

Table 1 Model Sizes and Parameter Counts

When we examine the training and validation loss results on Table 2, we can say that the T5 model gives better results than PEGASUS. It is difficult to know whether this is due to overfit of the model to the dataset.

When it comes to the summarization process, the success of the models is measured by the ROUGE score. In this study, ROUGE is used to calculate the performance. The ROUGE score is used for summarizing and comparing AI generated texts with human texts. There are multiple

Model Name	Training	Validation
T5 Base	1.401600	1.391697
Pegasus Large	1.795200	1.973148

Table 2 Training and Validation Loss Results for Models

variants of the ROUGE score and the metrics 'ROUGE-1', 'ROUGE-2', 'ROUGE-L', and 'ROUGE-LSUM' were used during training. ROUGE-1 and ROUGE-2 compare n-grams (n words together) between two given texts. In the calculation of these metrics, n-gram values are 1 (unigram) and 2 (bigram) respectively. ROUGE-L calculates the 'longest common subsequence' between the generated text and the reference text [8]. In the evaluations, it has been observed that the ROUGE-L score is closer to the human summary evaluation criteria, including other metrics other than the ROUGE score [2]. However, it is important to note here that whether the ROUGE score is a successful metric depends on the quality of the dataset. If the summaries in the dataset do not have the characteristics required for the desired output, the ROUGE score may be misleading.

The ROUGE scores obtained for the trained models are given in the Table 3. The higher the ROUGE score, the more successful the model is in the summarization task.

Model Name	Rouge1	Rouge2	Rougel
T5 Base	25.018700	11.829600	20.413300
Pegasus Large	42.938800	16.573600	24.844800

Table 3 ROUGE Scores for the Models

However, it should also be taken into consideration during the evaluation that the highest ROUGE score value obtained to date is 51.06[9]. In this perspective, it is possible to say that the Pegasus model is more successful in summarization. Especially when we look at the ROUGE1 score, there is a big difference between them. Nevertheless, this does not mean that the T5 model performs poorly. It is possible to say that both models performed at a good-medium level.

V. Conclusion

The findings of the study show that LLMs designed to perform natural language processing tasks can provide highly successful results in the summarization task. The fact that the LLM models used are pre-trained models and fine-tuned on news summarization datasets is an important factor in achieving successful results. The attention mechanisms implemented in the models and the Transformers architecture on which they are based make it possible for them to establish relationships in the text data fed to them and to recognize important points and extract summary information from these points. After the research and literature review on LLMs and especially the T5 and Pegasus models, it has been revealed that these models can be used for summarization processes. In particular, it has been proved that the Pegasus model is very successful in summarization and that the T5 model can give very good results with fine-tuning.

References

[1] J. Zhang, Y. Zhao, M. Saleh, and P. J. Liu, "PEGASUS: pre-training with extracted gap-sentences for abstractive summarization," *CoRR*, vol. abs/1912.08777, 2019. [Online]. Available: http://arxiv.org/abs/1912.08777

[2] T. Zhang, F. Ladhak, E. Durmus, P. Liang, K. McKeown, and T. Hashimoto, "Benchmarking large language models for news summarization," 01 2023.

[3] R. Nallapati, B. Zhou, C. dos Santos, Ç. Gulçehre, and B. Xiang, "Abstractive text summarization using sequence-to-sequence RNNs and beyond," in *Proceedings of the 20th SIGNLL Conference on Computational Natural Language Learning*, S. Riezler and Y. Goldberg, Eds. Berlin, Germany: Association for Computational Linguistics, Aug. 2016, pp. 280–290. [Online]. Available: https://aclanthology.org/K16-1028

[4] C. Raffel, N. M. 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," *J. Mach. Learn. Res.*, vol. 21, pp. 140:1–140:67, 2019. [Online]. Available: https://api.semanticscholar.org/CorpusID:204838007

[5] A. Vaswani, N. M. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *Neural Information Processing Systems*, 2017. [Online]. Available: https://api.semanticscholar.org/CorpusID:13756489

[6] T. Kudo and J. Richardson, "SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing," in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, E. Blanco and W. Lu, Eds. Brussels, Belgium: Association for Computational Linguistics, Nov. 2018, pp. 66–71. [Online]. Available: https://aclanthology.org/D18-2012

[7] K. Sinha, R. Jia, D. Hupkes, J. Pineau, A. Williams, and D. Kiela, "Masked language modeling and the distributional hypothesis: Order word matters pre-training for little," in *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, M.-F. Moens, X. Huang, L. Specia, and S. W.-t. Yih, Eds. Online and Punta Cana, Dominican Republic: Association for Computational Linguistics, Nov. 2021, pp. 2888–2913. [Online]. Available: https://aclanthology.org/2021.emnlp-main.230

[8] 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://aclanthology.org/W04-1013

[9] P. W. Code, "State-of-the-art text summarization on pubmed," 2024. [Online]. Available: https://paperswithcode.com/sota/text-summarization-on-pubmed-1