62FIT4ATI - Deep Learning Applications

Final Project Report: Scientific Paper Title Generation
Topic 5: Abstractive Text Summarization with T5

Group 14

Vu Minh Hieu - 2201140030
Tran Ngoc Tu - 2201140080
Ngo Hoang Anh - 2201140004

December 16, 2025

Contents

1 Introduction **2**
1.1 Problem Formulation 2
1.2 Project Goal 2

2 Data Preparation and Analysis **2**
2.1 Dataset Description 2
2.2 Data Inspection and Visualization 2

3 Model Architecture **3**
3.1 Why T5? 3

4 Optimization Techniques **3**
4.1 1. Weight Decay (Regularization) 3
4.2 2. Learning Rate Warmup 4

5 Experimental Results **4**
5.1 Training Performance 4
5.2 Quantitative Evaluation (ROUGE Scores) 4
5.3 Qualitative Analysis (Inference) 4

6 Conclusion **5**

1 Introduction

1.1 Problem Formulation

In the digital era, the exponential growth of scientific publications creates a significant challenge for efficient information retrieval and categorization. A compelling title must effectively summarize the core content while remaining concise. The problem addressed in this project is to construct a Deep Learning model capable of automatically generating scientific paper titles based on their abstracts. This task falls under the category of **Abstractive Text Summarization**, where the model is required to "understand" the context of a long input text (abstract) to synthesize a concise summary (title), rather than merely extracting existing phrases.

1.2 Project Goal

Our primary objective is to apply **Transfer Learning** techniques to fine-tune the **T5 (Text-to-Text Transfer Transformer)** large language model. We aim to train this model on the arXiv dataset to solve the aforementioned sequence-to-sequence task and evaluate its performance using standard NLP metrics such as ROUGE.

2 Data Preparation and Analysis

2.1 Dataset Description

We utilized the **arXiv dataset** provided for Topic 5. The dataset comprises metadata for approximately 136,238 scientific papers across various domains. The data structure for this supervised learning task consists of:

- **Input:** The paper's summary (`abstract`).
- **Target:** The paper's original title (`title`).

2.2 Data Inspection and Visualization

Prior to training, we analyzed the length distribution of the data to establish appropriate preprocessing parameters. We visualized the word count distribution for both Abstracts and Titles to handle padding and truncation effectively.

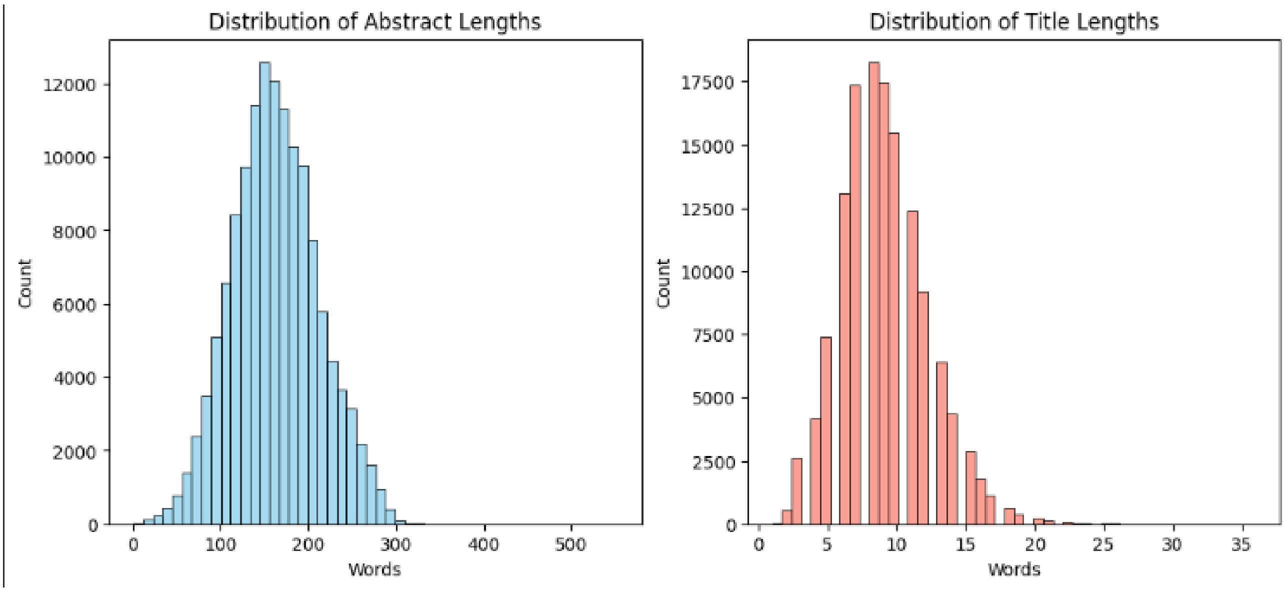

Figure 1: Distribution of Abstract and Title Lengths used to determine tokenization limits.

Based on the visualization (Figure 1), we observed that most abstracts contain fewer than 400 words. Consequently, to optimize memory usage while ensuring full context coverage, we configured the tokenization limits as follows:

- **Max Input Length:** 512 tokens.
- **Max Target Length:** 64 tokens.

3 Model Architecture

We selected the **T5 (Text-to-Text Transfer Transformer)** model architecture for this project.

3.1 Why T5?

T5 treats every Natural Language Processing problem as a text-to-text task. By adding the task-specific prefix `"summarize: "` to the input abstract, the model leverages its pre-trained knowledge to generate the target title. The architecture follows the standard Transformer Encoder-Decoder structure, which is the state-of-the-art approach for sequence generation tasks.

4 Optimization Techniques

To ensure efficient training and prevent overfitting on the limited subset of data, we researched and applied the following optimization techniques:

4.1 1. Weight Decay (Regularization)

- **Configuration:** `weight_decay = 0.01`
- **Analysis:** Deep Transformer models are prone to overfitting, especially when fine-tuning on specific domains like scientific texts. Weight decay adds a regularization term to the

loss function (L_2 penalty), penalizing large weights. This forces the model to learn more robust features rather than memorizing the training data.

4.2 2. Learning Rate Warmup

- **Configuration:** `warmup_steps = 500`
- **Analysis:** Transformer models are sensitive to the initial learning rate. A "Warmup" scheduler starts the learning rate at 0 and linearly increases it to the target value ($3e^{-4}$) over the first 500 steps. This stabilizes the gradients early in the training phase, preventing the model from diverging before it settles into a stable optimization path.

5 Experimental Results

5.1 Training Performance

We fine-tuned the `t5-small` model for 3 epochs. The training loss consistently decreased from approximately 2.45 to 1.89, indicating effective learning convergence. The detailed training configuration is summarized in Table 1.

Hyperparameter	**Value**
Base Model	T5-small
Batch Size	8
Learning Rate	$3e^{-4}$
Epochs	3
Optimizer	AdamW
Weight Decay	0.01
Warmup Steps	500

Table 1: Hyperparameters and Training Configuration

5.2 Quantitative Evaluation (ROUGE Scores)

We evaluated the model using the **ROUGE** metric (Recall-Oriented Understudy for Gisting Evaluation). The final results on the validation set are presented below:

Metric	**Score (%)**	**Interpretation**
ROUGE-1	**42.20**	High overlap of individual words
ROUGE-2	**23.13**	Good fluency (bigram overlap)
ROUGE-L	**38.03**	Accurate sentence structure

Table 2: Final Evaluation Metrics

5.3 Qualitative Analysis (Inference)

We tested the model on unseen data to assess its real-world performance.

Example 1 (Success Case):

- **Input Abstract:** "An onboard prediction of dynamic parameters (e.g. Aerodynamic drag, rolling resistance) enables accurate path planning for EVs. This paper presents EV-PINN, a Physics-Informed Neural Network approach..."
- **Ground Truth Title:** "EV-PINN: A Physics-Informed Neural Network for Predicting Electric Vehicle Dynamics"
- **Generated Title:** "EV-PINN: Physics-Informed Neural Network for Onboard Prediction of Dynamic Parameters"

Analysis: The model successfully identified the specific method name ("EV-PINN") and the core technology. Interestingly, the generated title used the phrase "Onboard Prediction", which was explicitly present in the abstract, making the generated title highly descriptive and grammatically correct.

Limitations: While the model performs well generally, we observed minor issues in highly technical abstracts where the model occasionally generated generic titles or missed specific mathematical notations. This indicates that while T5 captures the general context well, handling extremely domain-specific jargon remains a challenge for the `t5-small` variant without larger-scale pre-training.

6 Conclusion

In this project, we successfully implemented a deep learning workflow for scientific paper title generation. By leveraging the T5 architecture and applying optimization techniques like Weight Decay and Learning Rate Warmup, we achieved a ROUGE-1 score of **42.20%**. The qualitative analysis confirms that the model can synthesize coherent and relevant titles, fulfilling the project requirements.