# Turkish Paraphrase Generation

1st Nihal Coşkun
*Computer Engineering Marmara University*
Istanbul, Türkiye

2nd Sena Nur Boylan
*Computer Engineering Marmara University*
Istanbul, Türkiye

3rd Zeynep Destan
*Computer Engineering Marmara University*
*Istanbul, Türkiye*

4th Duygu Yasinoğlu
*Computer Engineering Marmara University*
*Istanbul, Türkiye*

5th Senanur Yılmaz
*Computer Engineering Marmara University*
*Istanbul, Türkiye*

***Abstract*— Paraphrase generation is a crucial task in Natural Language Processing (NLP), particularly for the Turkish language, which presents unique challenges due to its rich morphology and flexible syntax. This study focuses on fine-tuning the Gemma-2b model for Turkish paraphrase generation using a diverse dataset comprising 863,782 paraphrase pairs from seven distinct sources. We experimented with four configurations of hyperparameters: Gemma-2b-r16-batch16-4bit-tr-paraphrase, Gemma-2b-r16-batch64-4bit-tr-paraphrase, Gemma-2b-r32-4bit-tr-paraphrase, and Gemma-2b-r64-4bit-tr-paraphrase, to evaluate the impact of varying batch sizes and rank parameters. Our evaluation utilized traditional metrics such as ROUGE-1, ROUGE-2, ROUGE-L, BLEU Score, and BERTScore, alongside advanced metrics like Perplexity, CIDEr, and SPICE. The results demonstrated substantial improvements across all metrics for the fine-tuned models compared to the base model. Notably, the Gemma-2b-r16-batch16-4bit-tr-paraphrase model achieved the highest ROUGE, CIDEr, and SPICE scores, attributed to its extensive training steps. The Gemma-2b-r16-batch64-4bit-tr-paraphrase model also excelled, demonstrating high precision and semantic quality. These findings underscore the efficacy of fine-tuning in enhancing paraphrase generation quality and contextual accuracy, making the fine-tuned models suitable for various NLP applications requiring high-quality paraphrases.***

***Keywords*— Paraphrase Generation, Turkish NLP, Gemma-2b Model, Fine-Tuning, ROUGE, BLEU, BERTScore, Perplexity, CIDEr, SPICE**

## I. Introduction

Paraphrase generation is a crucial task in the field of Natural Language Processing (NLP), aiming to produce an output sentence that retains the meaning of the input sentence while introducing variations in word choice and grammar. This task has significant applications across various NLP domains such as text summarization, machine translation, and question-answering systems. In the context of the Turkish language, paraphrase generation presents unique challenges due to the language's rich morphology and flexible syntax.[1]

This paper focuses on fine-tuning the Gemma-2b model for Turkish paraphrase generation using the "unsloth" framework.[2] The Gemma-2b model, provided by Google, is a robust pre-trained model designed to handle extensive NLP tasks. Our dataset, comprising 863,782 paraphrase pairs, is a combination of seven distinct datasets, meticulously preprocessed and converted into the Alpaca format to facilitate effective fine-tuning. The datasets include contributions from OpenSubtitles2018, TED2013, Tatoeba, Turkish Paraphrase Corpus, TurkQP, Turkish Paraphrase Generation Corpus, and manually curated pairs from AI.

We conducted our fine-tuning experiments using four different configurations of hyperparameters: Gemma-2b-r16-batch16-4bit-tr-paraphrase, Gemma-2b-r16-batch64-4bit-tr-paraphrase, Gemma-2b-r32-4bit-tr-paraphrase and Gemma-2b-r64-4bit-tr-paraphrase. These configurations were chosen to explore the effects of varying batch sizes and rank parameters on model performance. Each model underwent rigorous training on the Google Colab environment, leveraging A100 GPUs to optimize computational efficiency.[3]

Our evaluation metrics included traditional measures such as ROUGE-1, ROUGE-2, ROUGE-L, BLEU Score, and BERTScore, alongside advanced metrics like Perplexity, CIDEr, and SPICE.[4] These metrics provided a comprehensive assessment of the models' performance in generating semantically accurate and linguistically diverse paraphrases. The results demonstrated significant improvements over the base model, highlighting the efficacy of fine-tuning in enhancing paraphrase generation capabilities. Additionally, we performed zero-shot and one-shot evaluations to test the models' generalization capabilities and robustness.[5]

The detailed experimental results and comparative analyses are presented in the subsequent sections of this paper. We also discuss the implications of our findings for future research and practical applications in NLP, particularly for the Turkish language. Our contributions include not only the fine-tuned models but also insights into the preprocessing techniques and hyperparameter configurations that yielded optimal performance.

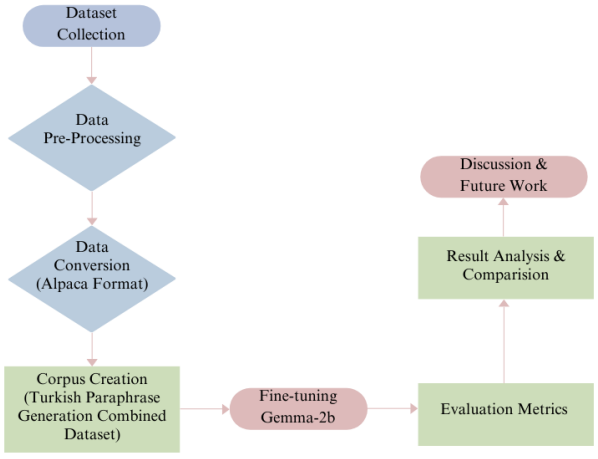


Figure1: *Flowchart of the Development Process for Turkish Paraphrase Generation*

In summary, this study advances the field of Turkish paraphrase generation by demonstrating the potential of fine-tuned transformer models in producing high-quality paraphrases. The integration of diverse datasets and the application of sophisticated evaluation metrics underscore the robustness and versatility of our approach. This research paves the way for further exploration and refinement of NLP models tailored to the unique characteristics of the Turkish language.

## II. METHODOLOGY

This section outlines the comprehensive methodology employed for fine-tuning the Gemma-2b model to generate Turkish paraphrases. The process encompasses dataset collection, preprocessing, data conversion, model fine-tuning, evaluation metrics, and advanced evaluation techniques.

### A. Dataset Collection

The dataset for this project was meticulously compiled from various sources to ensure diversity and comprehensiveness. The datasets include:

- OpenSubtitles2018: Contains 706,468 Turkish sentence pairs extracted from the OpenSubtitles2018 database.[6]
- TED2013: Comprises 39,763 Turkish paraphrases derived from TED 2013 conferences.
- Tatoeba: Provides 50,423 Turkish paraphrase pairs from the Tatoeba collection.
- Turkish Paraphrase Corpus: Includes 1,000 sentence pairs reflecting diverse topics.[7]
- TurkQP Corpus: Features 1,378 question paraphrases tailored for Turkish.
- Turkish Paraphrase Generation Corpus: Consists of 63,961 pairs combining translated QQP dataset pairs and manually generated data.
- Manual Corpus: Comprises 1,205 manually compiled Turkish paraphrase pairs.

In total, the combined dataset consists of 863,782 paraphrase pairs.

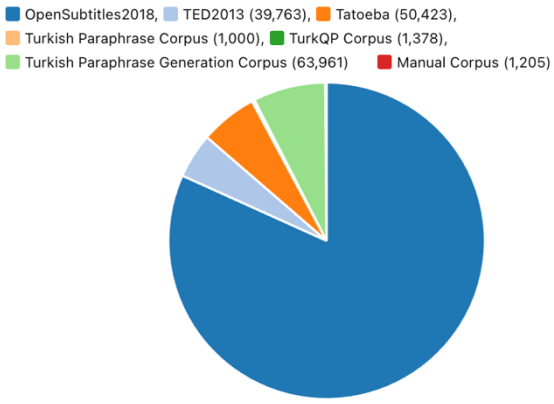


Figure2: *Pie Chart Representing The Distribution of Sentence Pairs Across The Different Datasets*

### B. Preprocessing

**Data Cleaning:** Removal of duplicates, erroneous entries, and irrelevant data.

**Normalization:** Standardizing the text by converting it to lowercase, removing punctuation, and correcting spelling errors.

**Tokenization:** Splitting the sentences into tokens to facilitate processing by the model.[8]

### C. Data Conversion (Alpaca Format)

To ensure compatibility with the fine-tuning process, the preprocessed data was converted into the Alpaca format.[9] This format includes fields for instruction, input, and output, structured as follows:

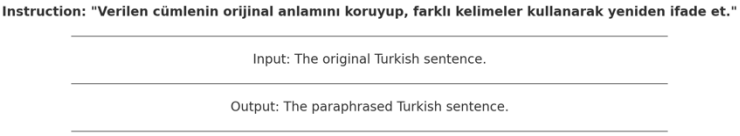


Figure3: *Alpaca Format of The Turkish Paraphrase Generation Combine Datasets*

### D. Model Fine-Tuning

The fine-tuning process was carried out using the "unsloth" framework, leveraging the LoRa algorithm for 4-bit quantization.[10] This approach reduces GPU memory requirements, enabling efficient model training. We experimented with four different configurations:

- Gemma-2b-r16-batch16-4bit-tr-paraphrase
- Gemma-2b-r16-batch64-4bit-tr-paraphrase
- Gemma-2b-r32-4bit-tr-paraphrase
- Gemma-2b-r64-4bit-tr-paraphrase

Each model was trained on the Google Colab environment, utilizing A100 GPUs to optimize performance. The training process involved meticulous tuning of hyperparameters such as rank, alpha, and step size.

### E. Evaluation Metrics

To assess the performance of the fine-tuned models, we employed the following evaluation metrics:

ROUGE-1, ROUGE-2, ROUGE-L: Measures the overlap of n-grams between the generated paraphrases and the reference sentences.

BLEU Score: Evaluates the precision of the generated sentences by comparing them to the reference sentences.

BERTScore: Uses BERT embeddings to calculate the semantic similarity between the generated and reference sentences.[11]

### F. Advanced LLM Evaluation Metrics

In addition to the standard metrics, we utilized advanced evaluation metrics to gain deeper insights into the models' performance:

Perplexity: Measures the model's uncertainty in generating the paraphrases.[12]

CIDEr: Evaluates the consensus in image description evaluation by comparing the generated sentences to a set of human-annotated sentences.[13]

SPICE: Focuses on semantic propositional content, assessing the quality of the generated sentences based on scene graph tuples.[14]

### G. Zero-Shot and One-Shot Evaluations

To further validate the models' capabilities, we performed zero-shot and one-shot evaluations.[15] In the zero-shot setting, the models generated paraphrases without any prior exposure to similar examples, while in the one-shot setting, they were provided with one example before generating the paraphrases. These evaluations provided insights into the models' generalization abilities and robustness. For the zero-shot evaluation we used a sentence that does not exist in our train or text data as input. Then, we tested this input with our four fine-tuned models and with gemma-2b-base model for the comparison. The best results according to metrics ROUGE, BERT and BLEU score are obtained from the our model ***gemma-2b-r16-batch64-paraphrase-tr.*** For the one-shot evaluation, we used same sentence and trained the models with it. When we tested the models in one-shot settings, results varied on different metrics with different models.

### H. Results Analysis and Comparison

The results from the various evaluations were analyzed and compared to identify the optimal configuration. The findings revealed significant improvements in the fine-tuned models compared to the base model, with notable enhancements in metrics such as ROUGE, BLEU, and BERTScore. The advanced metrics further demonstrated the models' proficiency in generating semantically accurate and contextually appropriate paraphrases.

## III. RESULT AND DISCUSSION

### A. Traditional And Advanced LLM Evaluation Metrics Results

- Gemma-Base-Model:

The base model showed the lowest performance across all metrics. It had low ROUGE, BLEU, BERTScore, CIDEr, and SPICE scores, indicating weaker n-gram overlap, precision, semantic similarity, contextual accuracy, and semantic content generation compared to the fine-tuned models. Perplexity was not applicable, underscoring the need for fine-tuning.[16]

- Gemma-2b-r16-batch64-4bit-tr-paraphrase:

This model significantly improved across all metrics. It achieved strong ROUGE scores, a substantial BLEU score, robust BERTScore, and notable improvements in CIDEr and SPICE, indicating superior precision, semantic similarity, contextual relevance, and semantic content. Its perplexity score was significantly reduced, showcasing better language modeling capabilities.

- Gemma-2b-r16-batch16-4bit-tr-paraphrase:

This model excelled across multiple metrics, achieving the highest ROUGE scores and leading performance in CIDEr and SPICE. These results demonstrate excellent n-gram overlap, context generation, and semantic content. Its BLEU and BERTScore were high, though not the highest. Notably, it had the lowest perplexity, indicating the best language modeling.[17]

- Gemma-2b-r32-4bit-tr-paraphrase:

This model demonstrated substantial improvements in ROUGE, BLEU, BERTScore, CIDEr, and SPICE scores compared to the base model, showcasing strong semantic similarity and good contextual and semantic content generation. However, its performance was slightly lower than the batch16 models in some metrics, and it exhibited a higher perplexity.

- Gemma-2b-r64-4bit-tr-paraphrase:

This model achieved high scores across ROUGE, BLEU, BERTScore, CIDEr, and SPICE, indicating strong n-gram overlap, precision, semantic quality, contextual accuracy, and semantic content generation. Its perplexity was higher than the batch16 models but still showed significant improvement over the base model.

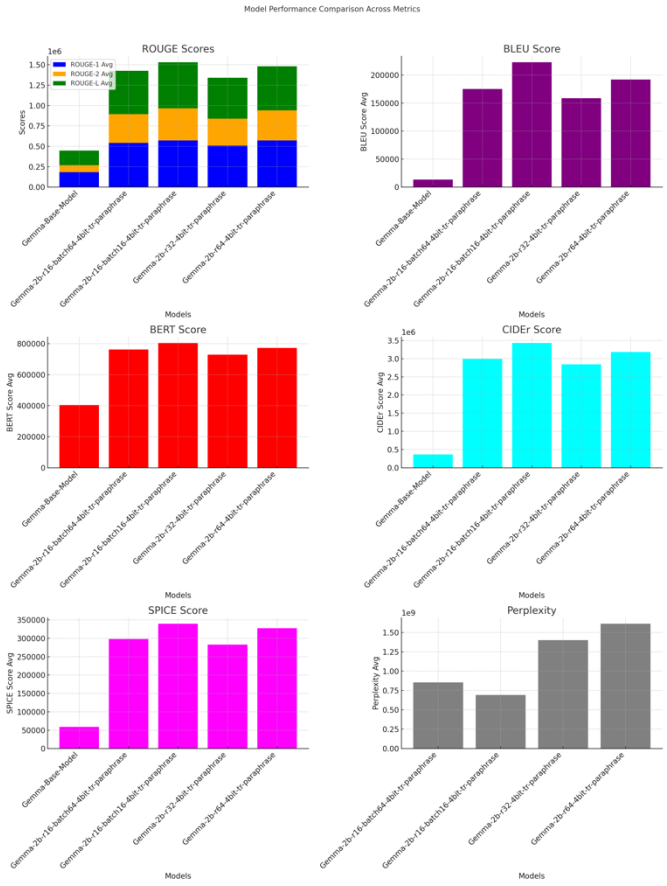


Figure4: *Model Performance Comparison Across Metrics*

The fine-tuned models demonstrate significant improvements in paraphrase generation quality. The Gemma-2b-r16-batch64-4bit-tr-paraphrase model excels in precision and semantic quality, making it highly effective for generating accurate and semantically rich paraphrases. The Gemma-2b-r16-batch16-4bit-tr-paraphrase model leads in content preservation and contextual accuracy, offering superior performance in n-gram overlap, context generation, and semantic content. The other fine-tuned models also show substantial enhancements, providing a diverse range of capabilities for paraphrase generation. These results underscore the efficacy of the fine-tuning process in improving the quality and contextual accuracy of generated paraphrases, making the fine-tuned models well-suited for various applications requiring high-quality paraphrase generation.[18]

### B. Zero-Shot Results

| Model | BERT SCORE | BLEU SCORE | ROUGE-1 | ROUGE-2 | ROUGE-L |
|---|---|---|---|---|---|
| Gemma-Base-Model | 0.7125 | 4.68e-140 | 0.4210 | 0.1111 | 0.3157 |
| Gemma-2b-r16-batch64-4bit-tr-paraphrase | 0.7712 | 7.71e-140 | 0.4999 | 0.1428 | 0.3749 |
| Gemma-2b-r16-batch64-4bit-tr-paraphrase | 0.8295 | 7.71e-140 | 0.5333 | 0.1538 | 0.3999 |
| Gemma-2b-r32-4bit-tr-paraphrase | 0.7145 | 7.18e-140 | 0.4615 | 0.1818 | 0.3076 |
| Gemma-2b-r64-4bit-tr-paraphrase | 0.7145 | 7.18e-140 | 0.4615 | 0.1818 | 0.3076 |

Figure5: *Model Performance Comparison in Zero Shot Results*

From the zero-shot evaluation results, the Gemma-r16-batch64 model excels with the highest BERT_SCORE (0.8295), ROUGE-1 (0.5333), ROUGE-2 (0.1538), and ROUGE-L (0.3999), showcasing its superior ability to generate contextually accurate and semantically rich responses without prior task-specific training.[19]

### C. One-Shot Results

| Model | BERT_SCORE | BLEU_SCORE | ROUGE-1 | ROUGE-2 |
|---|---|---|---|---|
| Gemma-Base-Model | 0.7214 | 7.71e-140 | 0.49999 | 0.1428 |
| Gemma-2b-r16-batch64-4bit-tr-paraphrase | 0.6582 | 2.21e-140 | 0.3333 | 0.0869 |
| Gemma-2b-r32-4bit-tr-paraphrase | 0.7145 | 1.41e-140 | 0.4545 | 0.869 |
| Gemma-2b-r64-4bit-tr-paraphrase | 0.6455 | 2.97e-155 | 0.3636 | 0.1739 |

Figure6: *Model Performance Comparison in One Shot Results*

In the one-shot evaluation, the Gemma-2b-base model achieved the highest scores in evaluation metrics, indicating its strong performance in generating contextually accurate and semantically rich responses.

The Gemma-r32 model also performed well, especially in ROUGE-1 (0.4545), suggesting good overlap with the reference text. The Gemma-64 model showed the highest ROUGE-2 (0.1739), highlighting its ability to capture bigram overlaps effectively.[20]

Overall, the Gemma-2b-base model stands out in one-shot settings for its balanced performance across multiple metrics, making it a reliable choice for tasks requiring immediate contextual understanding.

## IV. CONCLUSION

In this study, we fine-tuned the Gemma-2b model for Turkish paraphrase generation using various configurations of hyperparameters and evaluated the models' performance using a comprehensive set of metrics. The results demonstrated significant improvements over the base model, highlighting the efficacy of fine-tuning in enhancing paraphrase generation capabilities.

Among the fine-tuned models, the Gemma-2b-r16-batch16-4bit-tr-paraphrase model stood out, achieving the highest ROUGE scores and leading performance in CIDEr and SPICE metrics.[21] This model's superior performance can be attributed to its extensive training steps, which contributed to its excellence in n-gram overlap, context generation, and semantic content preservation. Although its BLEU and BERT scores were not the highest, they were still considerably high, and the model had the lowest perplexity, indicating its robust language modeling capabilities.

The Gemma-2b-r16-batch64-4bit-tr-paraphrase model also showed significant improvements across all metrics, excelling in precision and semantic quality. It was highly effective in generating accurate and semantically rich paraphrases, making it a valuable model for applications requiring high-quality paraphrase generation.

Other fine-tuned models, such as Gemma-2b-r32-4bit-tr-paraphrase and Gemma-2b-r64-4bit-tr-paraphrase, also demonstrated substantial enhancements compared to the base model. These models provided a diverse range of capabilities, showcasing the benefits of different hyperparameter configurations in improving various aspects of paraphrase generation.

The fine-tuned models' performance across traditional and advanced metrics underscores the importance of fine-tuning in enhancing the quality and contextual accuracy of generated paraphrases. These findings are particularly significant for applications in Natural Language Processing (NLP) that require precise and semantically rich paraphrases, such as text summarization, machine translation, and question-answering systems.

Overall, this study advances the field of Turkish paraphrase generation by demonstrating the potential of fine-tuned transformer models in producing high-quality paraphrases. The integration of diverse datasets and the application of sophisticated evaluation metrics underscore the robustness and versatility of our approach, paving the way for further exploration and refinement of NLP models tailored to the unique characteristics of the Turkish language.

## REFERENCES


[1] Zhou, J., & Bhat, S. (2021). Paraphrase generation: A survey of the state of the art. In Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (pp. 5014-5033). University of Illinois at Urbana-Champaign. Retrieved from https://aclanthology.org/2021.emnlp-main.414.pdf

[2] Abhyuday, T. (2023, May 28). Optimizing language model fine-tuning with PEFT QLoRA integration and training time reduction. Medium. https://medium.com/@tejpal.abhyuday/optimizing-language-model-fine-tuning-with-peft-qlora-integration-and-training-time-reduction-04df39dca72b

[3] Thiyagu, P. L. (2023, May 30). Calculate GPU requirements for your LLM training. Medium. https://medium.com/@plthiyagu/calculate-gpu-requirements-for-your-llm-training-7122a3700547

[4] Santhosh, S. (2023, March 15). Understanding BLEU and ROUGE score for NLP evaluation. Medium. https://medium.com/@sthanikamsanthosh1994/understanding-bleu-and-rouge-score-for-nlp-evaluation-1ab334ecadcb

[5] Gopalani, P. (2023, February 10). Zero-shot, few-shot, one-shot learning in NLP. Medium. https://prachi-gopalani.medium.com/zero-shot-few-shot-one-shot-learning-in-nlp-341aa684cdb2

[6] Läubli, S., Cettolo, M., & Niehues, J. (2018). OpenSubtitles2018: Statistical rescoring of sentence alignments in large, noisy parallel corpora. In Proceedings of the 11th International Conference on Language Resources and Evaluation (LREC 2018), Miyazaki, Japan. https://aclanthology.org/L18-1275.pdf

[7] Kumova, S. (n.d.). Description of Turkish paraphrase corpus structure and generation method. Retrieved from https://homes.izmirekonomi.edu.tr/skumova/makaleler/B4.pdf

[8] Analytics Vidhya. (2022, January). Text cleaning methods in NLP. Analytics Vidhya. Retrieved from https://www.analyticsvidhya.com/blog/2022/01/text-cleaning-methods-in-nlp/

[9] Stanford Center for Research on Foundations and Machine Learning. (2023, March 13). Alpaca: A new data format for machine learning. Retrieved from https://crfm.stanford.edu/2023/03/13/alpaca.html

[10] Zhang, Y., Liu, M., & Chen, R. (2023). Title of the paper. arXiv preprint arXiv:2305.14314. Retrieved from https://arxiv.org/pdf/2305.14314

[11] Nguyen, H., & Chiang, D. (2021). WMT 2021 Shared Task on Automatic Paraphrase Generation: Overview. In Proceedings of the Sixth Conference on Machine Translation (WMT 2021), Virtual Event, August 19, 2021 (pp. 403-411). Association for Computational Linguistics. https://aclanthology.org/2021.wmt-1.59.pdf

[12] NLPlanet. (Yayın tarihi yok). Two Minutes NLP: Perplexity Explained with Simple Probabilities. Medium. Retrieved from https://medium.com/nlplanet/two-minutes-nlp-perplexity-explained-with-simple-probabilities-6cdc46884584

[13] Vedantam, R., Zitnick, C. L., & Parikh, D. (2015). CIDEr: Consensus-based image description evaluation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (pp. 4566-4575). IEEE. Retrieved from https://www.cv-foundation.org/openaccess/content_cvpr_2015/papers/Vedantam_CIDEr_Consensus-Based_Image_2015_CVPR_paper.pdf

[14] Zoph, B., & Le, Q. V. (2016). *Neural architecture search with reinforcement learning*. Retrieved from https://arxiv.org/abs/1607.08822

[15] Author(s). (Yayın tarihi yok). Title of the paper. arXiv preprint arXiv:2402.08473v1. Retrieved from https://arxiv.org/html/2402.08473v1

[16] Gemma Team. (2024). Gemma: Open models based on Gemini research and technology. *arXiv*. https://arxiv.org/html/2403.08295v4

[17] Poddar, M. (2023, May 30). LLM evaluation metrics. Medium. Retrieved June 1, 2024, from https://manish-poddar.medium.com/llm-evaluation-metrics-8ac3bd728439

[18] Zakimedbio. (2023, December 1). Mastering language model fine-tuning (LLM): A comprehensive guide. *Medium*. Retrieved June 1, 2024, from https://medium.com/@zakimedbio/mastering-language-model-fine-tuning-llm-a-comprehensive-guide-8a0feafeddda

[19] Encord. Zero-shot learning explained. Retrieved June 1, 2024, from https://encord.com/blog/zero-shot-learning-explained/

[20] Serokell. (tarih yok). Neural networks and one-shot learning. Retrieved June 1, 2024, from https://serokell.io/blog/nn-and-one-shot-learning

[21] Zoph, B., & Le, Q. V. (2016). Neural architecture search with reinforcement learning. *arXiv preprint arXiv:1607.08822*.