Iterative Layer Pruning for Efficient Translation Inference

Yasmin Moslem*
ADAPT Centre
Trinity College Dublin
Dublin, Ireland
yasmin.moslem@adaptcentre.ie

Muhammad Hazim Al Farouq*
Kreasof AI
Research Labs
Jakarta, Indonesia
muhammad.hazim@kreasof.my.id

John D. Kelleher
ADAPT Centre
Trinity College Dublin
Dublin, Ireland
john.kelleher@adaptcentre.ie

Abstract

Large language models (LLMs) have transformed many areas of natural language processing, including machine translation. However, efficient deployment of LLMs remains challenging due to their intensive computational requirements. In this paper, we address this challenge and present our submissions to the *Model Compression* track at the Conference on Machine Translation (WMT 2025). In our experiments, we investigate iterative layer pruning guided by layer importance analysis. We evaluate this method using the Aya-Expanse-8B model for translation from Czech to German, and from English to Egyptian Arabic. Our approach achieves substantial reductions in model size and inference time, while maintaining the translation quality of the baseline models.

1 Introduction

Large language models (LLMs) have demonstrated powerful capabilities in diverse natural language processing tasks, including translation. However, LLMs are often computationally intensive, making them impractical to deploy in real-world settings with limited resources. To enhance the efficiency of these models, researchers have explored various model compression techniques, aiming to reduce their computational requirements while preserving quality (Gandhi et al., 2023; Sajjad et al., 2023; Treviso et al., 2023; Sreenivas et al., 2024; Gu et al., 2025; Moslem, 2025).

Aya Expanse is an open-weight large language model with multilingual capabilities. The WMT 2025 Model Compression track (Gaido et al., 2025) required all submissions to be derived from the Aya-Expanse-8B model. This work focuses on translation from Czech to German and from English to Egyptian Arabic.

Our experiments build on established work on iterative layer pruning guided by layer importance evaluation (Peer et al., 2022; Moslem, 2025). We apply iterative layer pruning to the baseline model Aya-Expanse-8B[1] which originally consists of 32 layers and 8.03B parameters. This approach incrementally identifies and removes layers with minimal contribution to translation quality, one layer at a time. To this end, we conduct layer importance evaluation by measuring translation performance without each layer. After identifying and removing the least critical layer, we repeat the layer importance evaluation on the remaining layers until reaching our pruning target. The pruned model resulting from this process is then fine-tuned on the News Commentary dataset. We have made three submissions; the primary submission is a 24-layer model with 6.28B parameters, and the two contrastive submissions are 20-layer and 16-layer models, with 5.41B and 4.54B parameters, respectively.

2 Data

After layer pruning of the *Aya-Expanse-8B* model (cf. Section 3), we need to fine-tune the pruned model on medium-sized training data to restore the translation quality of the baseline model. To this end, we use the News Commentary dataset[2] which consists of news articles and their corresponding translations in several languages, including Arabic, English, German, and Czech.

We start by rule-based filtering of the Czech-to-German (CES-DEU) News Commentary dataset by removing duplicates, segments longer than 200 words, and those whose source/target length ratio is larger than 1.5 times. We also apply language detection with fastText[3] (Joulin et al., 2017) with a 0.9 threshold. Finally, we conduct semantic filtering using the *mUSE* model (Yang et al.,

*These authors contributed equally to this work.

[1] https://hf.co/CohereLabs/aya-expanse-8b

[2] https://data.statmt.org/news-commentary/v18/training/

[3] In particular, we used the fastText *"lid.176.bin"* model.

Proceedings of the Tenth Workshop on Machine Translation (WMT), pages 1022–1027
November 8–9, 2025.

2020) and *Sentence-Transformers* (Reimers and Gurevych, 2019) with a 0.7 threshold of semantic similarity between the source and target. The CES-DEU News Commentary dataset includes 250.4K segments before filtering, and 201.3K segments after filtering.[4] Eventually, we split the dataset into train and test splits, where the test set includes 500 segments used for both testing and layer importance evaluation. Then, we sample 100K of the training data, using 0 as the random seed in both cases.

As the English-to-Arabic News Commentary dataset uses Standard Arabic,[5] we first apply the same rule-based and semantic filtering steps as those we employ while processing the Czech-to-German dataset, which result in 84.3K segments. Afterwards, we convert Standard Arabic text segments into Egyptian Arabic (ARZ) with GPT-4.1-Mini, using the prompt in Appendix A, providing a fixed verified example that includes the English source as well as both the Standard Arabic and Egyptian Arabic translations. For parameters, we use temperature 0.3 and top-p 1. After completing the generation of the synthetic Egyptian Arabic translations, we apply rule-based filtering, comparing the generated Egyptian Arabic text segments to the original English source. Finally, we calculate the semantic similarity between the English source and the Egyptian Arabic target and select the 500 segments with the highest scores (0.91-0.98) for the "test" split, while the "train" split comprises the remaining 83.2K segments.[6] Using this synthetic dataset to fine-tune our models yields clear quality gains compared to the baseline models, when evaluated on both the in-domain holdout test dataset (cf. Table 1) and the WMT24++ benchmark[7] (Deutsch et al., 2025) which includes 998 segments (cf. Table 3).

3 Iterative Layer Pruning

As previous research demonstrates, iterative layer pruning achieves better quality than middle layer pruning (Moslem, 2025). In this experimental setup, we apply iterative layer pruning to the *Aya-Expanse-8B* baseline model. This approach incrementally identifies and removes layers with minimal contribution to translation quality, one layer at a time. The pruned models resulting from this process are then fine-tuned on the training dataset. Furthermore, knowledge distillation data from the teacher model can be added. Fine-tuning the pruned model restores most of the baseline model's translation quality. The following points elaborate on the process.

Layer importance evaluation: We conduct layer importance evaluation by measuring translation performance without each layer. In this greedy layer pruning approach (Peer et al., 2022; Rostami and Dousti, 2024; Moslem, 2025), to prune $n + 1$ layers, only a single optimal layer to prune must be added to the already known solution for pruning n layers. After identifying and removing the least critical layer, we repeat the layer importance evaluation on the remaining layers until reaching our n pruning target. We observe that while removing certain layers of the model (e.g. the first or last layer) substantially degrades translation performance, others result in minimal performance drops. Following Moslem (2025), we use the chrF++ metric for layer importance evaluation for both better efficiency and quality.

Layer pruning: We iteratively prune one decoder layer at a time, selecting the layer whose removal has the least negative impact on translation quality, measured by chrF++ scores. At each iteration, we evaluate the translation performance of the pruned model on the test split of the News Commentary dataset, after removing each candidate layer. The layer whose removal yields the best performance is eventually pruned. This process continues until a predefined number of layers (8, 12, and 16 layers) have been removed. By iteratively removing the least important layers, this performance-guided method produces a more compact model that can be fine-tuned further to recover the translation quality of the original model. We observe that the performance of the CES-DEU model is more impacted by pruning than the ENG-ARZ model, which might be attributed to the pre-training process (cf. Table 1). In other words, the evaluation of the baseline for CES-DEU translation achieves better results than that for ENG-ARZ translation; hence, it seems that fine-tuning the pruned ENG-ARZ models has helped with improving the translation quality of this language pair.

Fine-tuning: The pruning step is followed by fine-tuning the pruned model for 1 epoch using the News Commentary dataset (cf. Section 2). The

[4] https://hf.co/datasets/ymoslem/news-commentary-cs-de
[5] https://hf.co/datasets/ymoslem/news-commentary-en-ar
[6] https://hf.co/datasets/ymoslem/news-commentary-eng-arz
[7] https://hf.co/datasets/google/wmt24pp

Language	Model	Layers	chrF++ ↑	COMET ↑	Params (B) ↓	Speed (mm:ss) ↓
CES-DEU	Baseline	32	**52.79**	**87.18**	8.03	00:47
	Pruned + FT	24	51.35	85.70	6.28	00:34
		20	49.45	83.95	5.41	**00:27**
		16	45.79	79.39	**4.54**	**00:27**
ENG-ARZ	Baseline	32	42.03	81.45	8.03	01:22
	Pruned + FT	24	**58.38**	**85.74**	6.28	00:54
		20	55.69	84.50	5.41	00:51
		16	51.17	82.10	**4.54**	**00:42**

Table 1: Evaluation of layer pruning experiments. For translation from Czech to German (CES-DEU), pruning 8 layers and then fine-tuning the resulting model retains 98% of the translation quality (as measured by COMET). Interestingly, for translation from English to Egyptian Arabic (ENG-ARZ), the model resulting from pruning up to 16 layers and then fine-tuning outperforms the Aya-Expanse-8B baseline for this language pair.

training uses a learning rate of 2e-5, a batch size of 8, and early stopping with a patience value of 5 evaluation runs, and it is conducted on one A100 80GB GPU. This fine-tuning step recovers most of the translation quality of the baseline model.

Model	Layers	KD	chrF++ ↑	COMET ↑
Baseline 32B	40	-	54.57	87.76
Baseline 8B	32	-	52.79	87.18
Pruned + FT	24	⊗	51.35	85.70
		✓	52.68	86.50
	20	⊗	49.45	83.95
		✓	51.25	85.19
	16	⊗	45.79	79.39
		✓	48.60	81.39

Table 2: Evaluation of knowledge distillation (KD). Fine-tuning pruned models on a combination of authentic and synthetic data (generated by Aya-Expanse-32B) improved the CES-DEU translation quality, with the 24-layer pruned model nearly matching the performance of the Aya-Expanse-8B baseline.

Knowledge distillation: To improve the quality of the CES-DEU models, we employed sequence-level knowledge distillation, where the student model is fine-tuned on a combination of authentic data and synthetic data generated by the teacher model for the same training dataset. In this case, the teacher model is the Aya-Expanse-32B while the students are the pruned models. After generating the data, we filter it by removing duplicates (exact matches in the target side of the authentic data), and translations with less than 70% COMET scores, resulting in extra 98.6K segments of training data (cf. Section 2). As Table 2 demonstrates, fine-tuning the pruned models with a combination of both the authentic and knowledge distillation data has improved their translation quality, and helped close the performance gap between the 24-layer pruned model and the Aya-Expanse-8B baseline. Similarly, the 20-layer and 16-layer models show 2-3 points of improvement in terms of chrF++ and COMET metrics.

4 Inference and Evaluation

For inference, we use greedy generation by disabling the sampling options, and setting the temperature argument to 0. We apply a simple translation prompt: "Translate the following text from {source_language} to {target_language}:"

To evaluate our systems, we calculated BLEU (Papineni et al., 2002), chrF++ (Popović, 2017), as implemented in the sacreBLEU library[8] (Post, 2018). For semantic evaluation, we use COMET (Rei et al., 2020).[9] Table 1 reports the results of the main experiments using the *Transformers* framework[10] (Wolf et al., 2020) for inference.

5 Results

The process of iterative layer pruning has achieved model compression from 8.03B parameters to 6.28B, 5.41B, and 4.54B parameters, after removing 8, 12, and 16 layers, respectively. Moreover, the quality degradation caused by pruning has been mitigated through fine-tuning on medium-sized data

[8] https://github.com/mjpost/sacrebleu

[9] In particular, we used the *"wmt22-comet-da"* model.

[10] https://github.com/huggingface/transformers

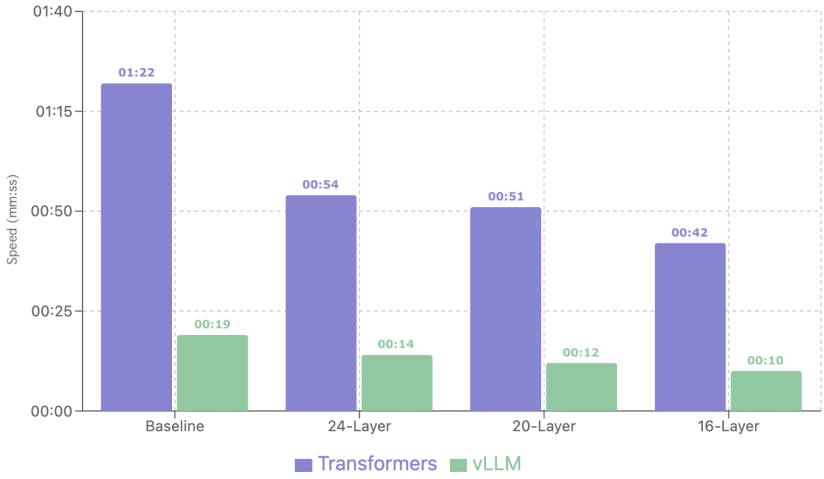

Figure 1: Inference speed comparison between *Transformers* and *vLLM*, using the Aya-Expanse-8B model for ENG-ARZ translation. *vLLM* consistently outperforms *Transformers* across all model sizes. Speedup ranges from 4.2x (16-layer) to 4.3x (baseline model). Both frameworks show improved performance with layer pruning. The 16-layer model achieves the fastest inference times overall.

(80K-100K) and knowledge distillation. As demonstrated by Table 1, by the end of the process, the pruned model could recover most of the translation quality of the baseline model. For translation from Czech to German (CES-DEU), pruning 8 layers and then fine-tuning the resulting model retains 98% of the translation quality (as measured by COMET) before knowledge distillation and 99% after knowledge distillation. Interestingly, for translation from English to Egyptian Arabic (ENG-ARZ), the model resulting from pruning up to 16 layers and then fine-tuning outperforms the baseline model. This can be attributed to the initial quality of the baseline model for this language pair.

Moreover, we experimented with immediate recovery through fine-tuning the model after each pruning phase (i.e. pruning the fine-tuned 24-layer model into 20 layers instead of pruning the baseline model directly), and noticed that the final quality was similar to pruning the baseline directly and then only fine-tuning the pruned model. This matches the results demonstrated by Moslem (2025) who experimented with immediate fine-tuning after pruning each layer, and observed that this could lead to overfitting. In other words, it is sufficient to fine-tune the final pruned model.

In terms of inference performance, we observe that using *vLLM* [11] (Kwon et al., 2023) as an inference engine instead of *Transformers* increases the inference speed by more than four times when conducting the evaluation on one A40 48GB GPU (cf. Figure 1). Moreover, while 4-bit quantization using *bitsandbytes* (Dettmers et al., 2023) reduces the memory footprint, pruning results in higher inference speed and throughput (cf. Table 4).

6 Conclusions and Future Work

In this work, we demonstrated that iterative layer pruning is an effective approach for compressing LLMs while retaining translation quality. The method relies on layer importance evaluation, followed by fine-tuning on a medium-sized dataset. This iterative layer pruning process reduces the model size and accelerates inference. To ensure reproducibility, we have made our code publicly available.[12]

Future research directions include investigating adaptive compression approaches that dynamically select appropriate model configurations based on real-time deployment constraints such as memory limits and latency requirements. Moreover, we plan to assess our compression methods on a broader range of datasets, including both sentence-level and document-level data. Since Aya-Expanse is designed to follow textual instructions, exploring retrieval-augmented generation combined with few-shot prompting presents a promising opportunity for enhancing translation performance in compressed models.

[11] https://github.com/vllm-project/vllm

[12] https://github.com/ymoslem/Model-Compression

Acknowledgements

We sincerely thank the ADAPT Centre (Ireland) and Kreasof AI (Indonesia) for providing the resources and support that made this work possible.

References

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. 2023. QLoRA: Efficient Finetuning of Quantized LLMs. *arXiv [cs.LG]*.

Daniel Deutsch, Eleftheria Briakou, Isaac Caswell, and 14 others. 2025. WMT24++: Expanding the Language Coverage of WMT24 to 55 Languages & Dialects. *arXiv preprint arXiv:2502.12404*.

Marco Gaido, Thamme Gowda, Roman Grundkiewicz, and Matteo Negri. 2025. Findings of the WMT25 Model Compression Shared Task: Early Insights on Compressing LLMs for Machine Translation. In *Proceedings of the Tenth Conference on Machine Translation*, Suzhou, China.

Sanchit Gandhi, Patrick von Platen, and Alexander M Rush. 2023. Distil-Whisper: Robust knowledge distillation via large-scale pseudo labelling. *arXiv [cs.CL]*.

Yuxian Gu, Qinghao Hu, Shang Yang, and 4 others. 2025. Jet-Nemotron: Efficient language model with Post Neural Architecture Search. *arXiv [cs.CL]*.

Armand Joulin, Edouard Grave, Piotr Bojanowski, and Tomas Mikolov. 2017. Bag of Tricks for Efficient Text Classification. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 427–431, Valencia, Spain. Association for Computational Linguistics.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, and 6 others. 2023. Efficient memory management for large language model serving with PagedAttention. In *Proceedings of the 29th Symposium on Operating Systems Principles*, pages 611–626, New York, NY, USA. ACM.

Yasmin Moslem. 2025. Efficient speech translation through model compression and knowledge distillation. In *Proceedings of the 22nd International Conference on Spoken Language Translation (IWSLT 2025)*, pages 379–388, Stroudsburg, PA, USA. Association for Computational Linguistics.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a Method for Automatic Evaluation of Machine Translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

David Peer, Sebastian Stabinger, Stefan Engl, and Antonio Rodríguez-Sánchez. 2022. Greedy-layer pruning: Speeding up transformer models for natural language processing. *Pattern Recognit. Lett.*, 157:76–82.

Maja Popović. 2017. chrF++: words helping character n-grams. In *Proceedings of the Second Conference on Machine Translation*, pages 612–618, Copenhagen, Denmark. Association for Computational Linguistics.

Matt Post. 2018. A Call for Clarity in Reporting BLEU Scores. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 186–191, Brussels, Belgium. Association for Computational Linguistics.

Ricardo Rei, Craig Stewart, Ana C Farinha, and Alon Lavie. 2020. COMET: A Neural Framework for MT Evaluation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2685–2702, Online. Association for Computational Linguistics.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992, Hong Kong, China. Association for Computational Linguistics.

Pedram Rostami and Mohammad Javad Dousti. 2024. CULL-MT: Compression using language and layer pruning for machine translation. *arXiv [cs.CL]*.

Hassan Sajjad, Fahim Dalvi, Nadir Durrani, and Preslav Nakov. 2023. On the effect of dropping layers of pre-trained transformer models. *Comput. Speech Lang.*, 77(101429):101429.

Sharath Turuvekere Sreenivas, Saurav Muralidharan, Raviraj Joshi, and 19 others. 2024. LLM pruning and distillation in practice: The Minitron approach. *arXiv [cs.CL]*.

Marcos Treviso, Ji-Ung Lee, Tianchu Ji, and 19 others. 2023. Efficient methods for natural language processing: A survey. *Transactions of the Association for Computational Linguistics*, 11:826–860.

Thomas Wolf, Lysandre Debut, Victor Sanh, and 19 others. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Stroudsburg, PA, USA. Association for Computational Linguistics.

Yinfei Yang, Daniel Cer, Amin Ahmad, and 9 others. 2020. Multilingual Universal Sentence Encoder for Semantic Retrieval. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 87–94, Online. Association for Computational Linguistics.

A Prompt for Synthetic Data Generation for Egyptian Arabic

```
I would like to convert a Standard Arabic text into Egyptian Arabic.
Please generate the Egyptian Arabic version using a neutral, informative
tone with slightly conversational phrasing, similar to the example below.
The output should feel natural, like it's written for a general Egyptian
audience but still accurate and clear. Do not add any commentary; just
return the Egyptian Arabic version.

English:
<english_example>

Standard Arabic:
<standard_arabic_example>

Egyptian Arabic:
<egyptian_arabic_example>

English:
{new_source_text}

Standard Arabic:
{new_target_text}

Egyptian Arabic:
```

B Evaluation of Egyptian Arabic Translation on WMT24++

Model	Layers	chrF++ ↑	COMET ↑
Baseline 32B	40	33.89	75.55
Baseline 8B	32	30.62	74.50
Pruned + FT	24	**37.01**	**76.86**
	20	34.24	74.95
	16	29.32	68.70

Table 3: Evaluation of the ENG-ARZ models fine-tuned with target-side synthetic data. The evaluation uses the WMT24++ benchmark and shows quality improvement compared to the baseline models.

C Quantization Speed and Throughput

Model	Layers	4-bit	Memory ↓	Speed ↓	Throughput ↑
Baseline 8B	32	no	14.96	00:19	2275
		yes	5.61	00:42	1053
Pruned + FT	24	no	11.71	00:14	3008
		yes	4.70	00:22	2004
	20	no	10.08	00:12	3484
		yes	4.24	00:18	2367
	16	no	8.46	00:10	4192
		yes	3.78	00:15	2908

Table 4: Performance comparison of Aya-Expanse-8B baseline and the pruned models with and without 4-bit quantization, in terms of memory (GiB), speed (mm:ss), and output throughput (tokens/sec). The evaluation uses the holdout ENG-ARZ News Commentary test dataset, on one A40 48GB GPU. While 4-bit quantization reduces the memory footprint, layer pruning achieves both higher inference speed and throughput.