

Compressing Large Language Models (LLMs) using Knowledge Distillation for Optimizing Inference Time and Model Size

Rachmad Imam Tarecha^{1✉}, Priska Choirina², Amalia Agung Septarina³

^{1, 2, 3} Teknik Informatika, Fakultas Sains dan Teknologi, Universitas Islam Raden Rahmat, Indonesia

✉ **Corresponding Author** : Rachmad Imam Tarecha (e-mail: ri.tarecha@gmail.com)

Article Information	ABSTRACT
Article History Received : March 10, 2025 Revised : April 14, 2025 Published : April 26, 2025	Large Language Models (LLMs) contain a vast number of parameters and are significantly large in size. For instance, the DeepSeek-V3 model consists of approximately 671 billion parameters and has a file size of up to 720GB. The sheer number of parameters in LLMs reflects their high complexity, which can serve as both an advantage and a drawback, particularly when deployed in environments with limited computational resources. This study focuses on compressing a custom-built lightweight model using knowledge distillation techniques applied to LLMs. The results indicate that the model's parameters can be reduced by up to 94.18%, its file size by up to 71.00%, and its inference time by up to 1.13%. Notably, despite these reductions, the model remains capable of performing specialized tasks with satisfactory accuracy. This finding underscores the potential of knowledge distillation as an effective method for reducing model size while maintaining operational efficiency, particularly in scenarios where computational constraints lead to mismatched capabilities. Efficiency in knowledge distillation is achieved through a combination of model size reduction and the alignment of computational capacity with task-specific requirements.
Keywords: <i>Knowledge Distillation; Model Compression; Large Language Models.</i>	

INTRODUCTION

Over the past two decades, language models have advanced significantly in both language understanding and generation, leading to major breakthroughs in natural language processing (Zhao et al., 2023). According to Zhao et al. (2023), the development of language models has been continuously evolving since 2018, with the introduction of notable models such as GPT-1, BERT (Devlin et al., 2019), GPT-2, T5, GPT-3, Codex (Chen et al., 2021), InstructGPT, ChatGPT, LLaMa (Touvron et al., 2023), and also GPT-4 (Achiam et al., 2023). These models goods in text generation task, translation, summarization, and code completion. More recently, DeepSeek (Guo et al., 2025) and Qwen LLM (Bai et al., 2023) have emerged.

The increasing trend in Large Language Models (LLMs) because their success in learning universal representations, allowing them to generalize across various language-related tasks (Wu et al., 2023). Recent studies have highlighted the extraordinary capabilities of LLMs in generating coherent and contextually relevant text, solving complex problems, and assisting in various applications such as content creation, programming, and research (Naveed et al., 2023). However, despite their impressive advancements, LLMs also face significant challenges, particularly concerning their size and computational requirements. Yang et al. (2024) noted that while large-scale language models exhibit outstanding performance, their sheer size makes them difficult to deploy in resource-constrained environments, such as mobile devices and embedded systems (Chen et al., 2021). Additionally, bigger models do not always guarantee better performance, as

they may suffer from inefficiencies due to capacity mismatches, leading to unnecessary computational overhead without proportional improvements in accuracy (Cho et al., 2019). In real-world applications, achieving a balance between model size, inference speed, and computational complexity is crucial to ensure efficiency, particularly in scenarios that demand real-time processing (Yu et al., 2024).

To address these problems, researchers have explored various methods for reducing the size of LLMs while maintaining their performance. One widely studied approach is knowledge distillation, a compression technique that transfers knowledge from a large model to a smaller model, thereby improving efficiency and reducing computational costs. Wang et al. (2023) introduced knowledge distillation (KD), a method which could optimize model size while retaining accuracy and inference speed. This approach has been extensively used in deep learning research for compressing complex models into smaller, more manageable versions (Tang et al., 2020). However, despite its advantages, knowledge distillation has its limitations. One of the main challenges is that the distilled Student model may not fully retain the capacity of the original Teacher model, leading to potential performance trade-offs (Stanton et al., 2021).

In response to these challenges, our study aims to apply model compression using knowledge distillation to create a lightweight yet efficient model. Our primary objective is to develop a model that performs well in environments with limited computational resources. The key focus of our research is to align the capacity of the distilled model.

RESEARCH METHODS

The method we used to reduce the model size is knowledge distillation. Gou et al. (2021) explained that knowledge distillation is a process where a smaller model learns from a larger model (Hinton et al., 2015).

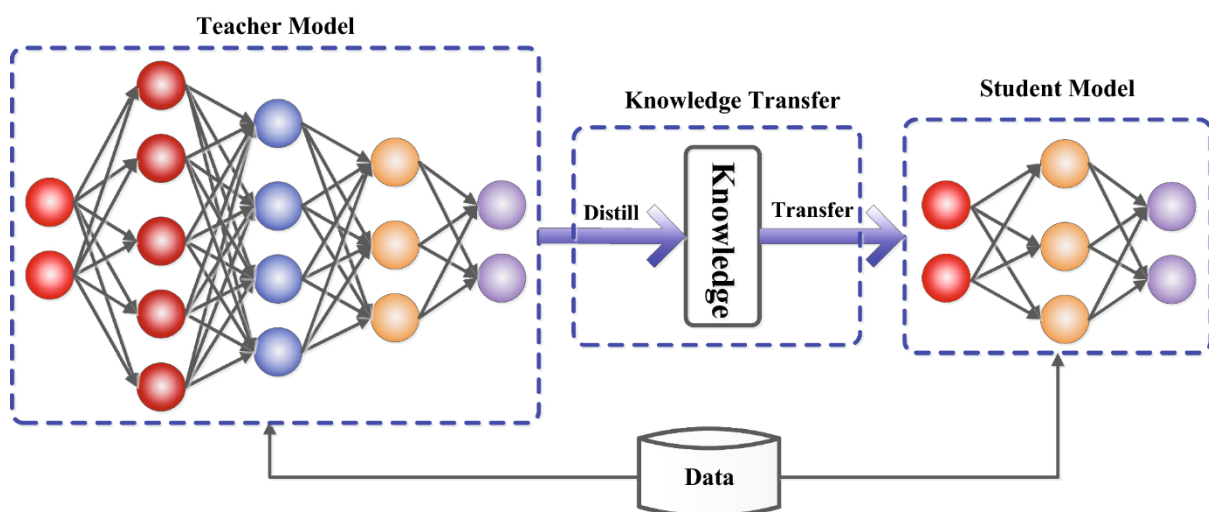


Figure 1 Knowledge Distillation Illustration (Gou et al, 2021)

As illustrated in Figure 1, the larger model is called Teacher, while the smaller model is known as the Student. The knowledge distillation enables the Student model to learn from the Teacher model by transferring essential knowledge (Tao et al., 2023). This approach allows the Student model to retain much of the Teacher model's performance while significantly reducing computational requirements, making it more suitable for environments with limited resources.

The knowledge distillation function combines two main components: the cross-entropy loss with the true labels and the Kullback-Leibler (KL) divergence loss (Cui et al., 2023) between the student and teacher model's predictions. The mathematical notation for this is as follows:

$$L_{\text{distill}} = H\left(\frac{q_T}{T}, \frac{q_S}{T}\right) \quad (1)$$

- H is represent the cross-entropy loss
- q_T is represent the prediction of the teacher model.
- q_S is represent the prediction of the student model.
- T is represent the temperature parameter.
- H is represent the cross-entropy loss

Soft targets (2) and soft predictions (3) are calculated using softmax with a temperature parameter:

$$q_T^{\text{soft}} = \text{softmax}\left(\frac{q_T}{T}\right) \quad (2)$$

$$q_S^{\text{soft}} = \text{softmax}\left(\frac{q_S}{T}\right) \quad (3)$$

The cross-entropy loss notation (4) between soft targets and soft predictions is:

$$H\left(\frac{q_T}{T}, \frac{q_S}{T}\right) = - \sum_i q_T^{\text{soft}} \log(q_S^{\text{soft}}) \quad (4)$$

In this study, we did not actually use large-scale LLMs. Instead, we created our own tiny Large Language Model (LLM) with a size of just 0.13 MB specifically to test the distillation process. This decision was made due to limitations in accessing and utilizing real large LLMs, which require significant computational resources. By designing a small-scale model, we were able to experiment with knowledge distillation in a controlled environment while still observing its effects on model compression and performance.

After constructing our tiny LLM, we conducted a series of comparisons between the original model (Teacher) and the distilled model (Student). One key aspect of our analysis was evaluating the reduction in model size after distillation. This compression is particularly useful for deploying language models in low computing resource.

In addition to model size, we also compared the inference time of both the Teacher and Student models. Faster inference speed is a crucial factor in real-world applications, as it determines how quickly a model can generate responses. By measuring inference time before and after distillation, we assessed whether the compression process contributed to an improvement in processing efficiency. Even a small reduction in inference time can be beneficial, especially when scaling models for practical deployment.

Furthermore, we examined the reduction in parameter count between the two models. A lower parameter count generally translates to lower memory usage and computational demands, making the model more lightweight and efficient. However, it is important to ensure that reducing parameters does not significantly impact the model's ability to generate accurate responses.

Lastly, we compared the output generated by the Teacher and Student models when given the same prompts. This step was crucial in determining whether the distilled model retained the essential knowledge and functionality of the original model. Despite the compression, we aimed to ensure that the distilled model could still perform its intended task effectively.

Through these comparisons, we gained valuable insights into how knowledge distillation affects model size, inference speed, parameter efficiency, and output quality. While our experiment was conducted on a small-scale LLM, the findings highlight the potential of knowledge distillation as a technique for optimizing larger models in the future.

RESULTS AND DISCUSSION

The results of our study are quite promising, demonstrating the effectiveness of knowledge distillation in optimizing model efficiency. After the distillation process, we observed a significant reduction in inference time, allowing the Student model to generate outputs more quickly. Additionally, the parameter count was drastically reduced, leading to lower memory usage and computational requirements. The overall model size also decreased substantially. These findings highlight the potential of knowledge distillation as a practical approach to compressing models while maintaining their essential functionality. The distilled model successfully show accurately result in our specific task.

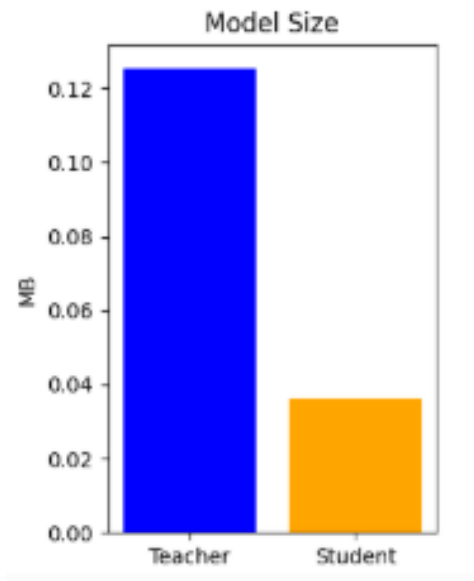


Figure 2 Regular Model (Teacher) and Distilled Model (Student) Size Comparison

The initial model size, as shown in Figure 2, was 0.13 MB before distillation (Teacher). After applying knowledge distillation, the model size was reduced by 71%, shrinking to just 0.04 MB in the distilled model (Student). This significant reduction in size makes the model more suitable for environments with limited storage and computational resources.

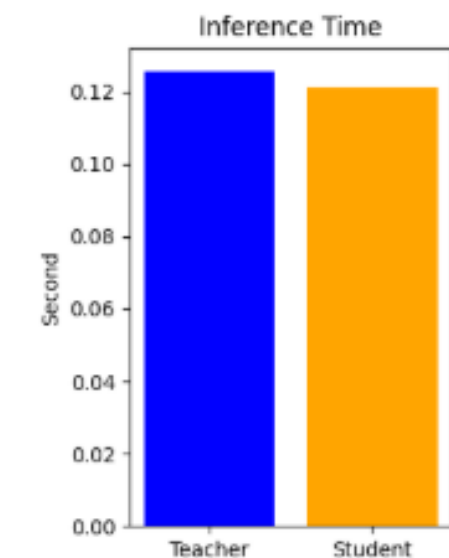


Figure 3 Regular Model (Teacher) and Distilled Model (Student) Inference Time Comparison

The inference time for the Teacher model was 0.110991 seconds, as shown in Figure 3. After applying knowledge distillation, the inference speed improved by 1.13%, reducing the time to 0.109735 seconds in the distilled Student model. Although the reduction in inference time is relatively small, it still contributes to overall efficiency, especially in resource-constrained environments. This improvement demonstrates that knowledge distillation can enhance not only model size but also computational performance. Further optimizations could potentially lead to even greater reductions in inference time, making the model more fast.

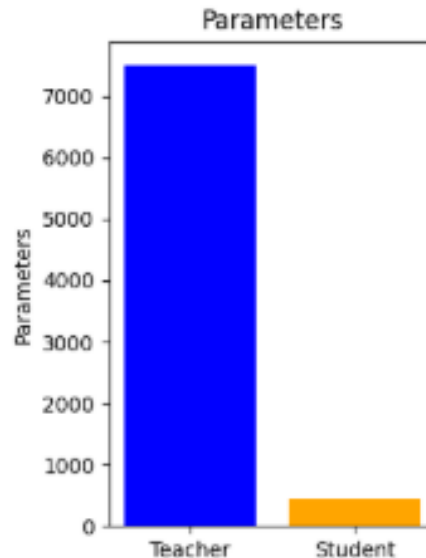


Figure 4 Regular Model (Teacher) and Distilled Model (Student) Parameter Comparison

Additionally, the number of parameters in the model was significantly reduced from 7,509 to 437, as shown in Figure 4. This reduction amounts to approximately 94.18%, demonstrating the effectiveness of knowledge distillation in compressing the model. A smaller number of parameters not only decreases memory usage but also improves computational efficiency, making the model more suitable for low-resource environments.



Figure 5 Regular Model (Teacher) and Distilled Model (Student) Results Comparison

As shown in Figure 5, the output results of both the Teacher and Student models remained identical, indicating that the distillation process preserved the model's ability to generate accurate responses. In this case, we tested the models by prompting them to determine the color of the sun, and both models successfully provided the correct response: yellow. This consistency in output

suggests that despite the significant compression, the distilled model retained essential knowledge from the original Teacher model.

It is important to highlight that this experiment was conducted on a small, custom-built model developed specifically for this research. The Teacher model used in this study was not a large-scale LLM but rather a simplified version designed to test the effectiveness of knowledge distillation on a minimal scale. Among the various reductions achieved, the most significant improvements were in the number of parameters and overall model size. With a 94.18% reduction in parameters and a 71% decrease in model size, the compressed Student model became significantly more efficient while maintaining performance.

CONCLUSION

This research successfully demonstrated that compressing large language models (LLMs) using knowledge distillation good way to reduce model size. By applying knowledge distillation to our own small, custom-built sample model, we achieved significant reductions, including a 94.18% decrease in the number of parameters, a 71.00% reduction in model size, and a 1.13% improvement in inference time. Despite these reductions, the distilled model was still able to perform its intended task—generating the color of an object from a prompt—effectively, indicating that the essential knowledge from the Teacher model was successfully transferred.

However, one limitation of this study is that the Teacher model used for distillation was relatively small. If similar research were conducted using a larger and more complex model, the performance gap between the original model (Teacher) and the distilled model (Student) might be more noticeable. This suggests that while knowledge distillation is effective for model compression, its impact may vary depending on the scale and complexity of the original model.

Future research could explore the application of knowledge distillation on larger LLMs to assess whether similar efficiency gains can be achieved without compromising output quality. Additionally, investigating alternative approaches, such as collaborative teaching—where a Student model learns from multiple Teacher models—could offer new insights into improving the efficiency and accuracy of distilled models.

ACKNOWLEDGEMENTS

We want to say thanks to the developers of Google Colab, ChatGPT, Kimi AI, and the Consensus App for creating advanced tools that have greatly contributed to improving and enhancing the readability of this article during the publication process. Without these valuable tools, we might not have been able to complete this research as effectively.

In detail, we used Google Colab to run the knowledge distillation Python code generated by ChatGPT. Additionally, we used the Consensus App to find relevant quotes and references for our research. Kimi AI was also utilized to generate mathematical notations included in this study. Furthermore, both ChatGPT and Kimi AI were used to expand paragraphs, improve grammar, and enhance readability.

REFERENCES

- Achiam, J., Adler, S., Agarwal, S., Ahmad, L., Akkaya, I., Aleman, F. L., ... & McGrew, B. (2023). GPT-4 technical report. arXiv preprint arXiv:2303.08774.
- Bai, J., Bai, S., Chu, Y., Cui, Z., Dang, K., Deng, X., ... & Zhu, T. (2023). Qwen technical report. arXiv preprint arXiv:2309.16609.
- Chen, M., Tworek, J., Jun, H., Yuan, Q., Pinto, H. P. D. O., Kaplan, J., ... & Zaremba, W. (2021). Evaluating large language models trained on code. arXiv preprint arXiv:2107.03374.

- Chen, W., Peng, L., Huang, Y., Jing, M., & Zeng, X. (2021). Knowledge Distillation for U-Net Based Image Denoising. 2021 IEEE 14th International Conference on ASIC (ASICON), 1–4.
- Cho, J., & Hariharan, B. (2019). On the Efficacy of Knowledge Distillation. 2019 IEEE/CVF International Conference on Computer Vision (ICCV), 4793–4801.
- Cui, J., Tian, Z., Zhong, Z., Qi, X., Yu, B., & Zhang, H. (2023). Decoupled Kullback-Leibler Divergence Loss. ArXiv, abs/2305.13948.
- Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2019, June). BERT: Pre-training of deep bidirectional transformers for language understanding. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers) (pp. 4171–4186).
- Gou, J., Yu, B., Maybank, S. J., & Tao, D. (2021). Knowledge distillation: A survey. International Journal of Computer Vision, 129(6), 1789–1819.
- Guo, D., Yang, D., Zhang, H., Song, J., Zhang, R., Xu, R., ... & He, Y. (2025). Deepseek-r1: Incentivizing reasoning capability in LLMs via reinforcement learning. arXiv preprint arXiv:2501.12948.
- Hinton, G., Vinyals, O., & Dean, J. (2015). Distilling the knowledge in a neural network. arXiv preprint arXiv:1503.02531.
- L. Wang and K.-J. Yoon. (2022). Knowledge Distillation and Student-Teacher Learning for Visual Intelligence: A Review and New Outlooks. IEEE Transactions on Pattern Analysis and Machine Intelligence, 44(6), 3048–3068.
- Naveed, H., Khan, A., Qiu, S., Saqib, M., Anwar, S., Usman, M., Barnes, N., & Mian, A. (2023). A Comprehensive Overview of Large Language Models. ArXiv, abs/2307.06435.
- Stanton, S., Izmailov, P., Kirichenko, P., Alemi, A., & Wilson, A. (2021). Does Knowledge Distillation Really Work?. ArXiv, abs/2106.05945.
- Tang, J., Shivanna, R., Zhao, Z., Lin, D., Singh, A., Chi, E., & Jain, S. (2020). Understanding and Improving Knowledge Distillation. ArXiv, abs/2002.03532.
- Tao, Z., Xia, Q., Cheng, S., & Li, Q. (2023). An Efficient and Robust Cloud-Based Deep Learning With Knowledge Distillation. IEEE Transactions on Cloud Computing, 11, 1733–1745.
- Touvron, H., Lavril, T., Izacard, G., Martinet, X., Lachaux, M. A., Lacroix, T., ... & Lample, G. (2023). LLaMA: Open and efficient foundation language models. arXiv preprint arXiv:2302.13971.
- Wu, L., Zheng, Z., Qiu, Z., Wang, H., Gu, H., Shen, T., Qin, C., Zhu, C., Zhu, H., Liu, Q., Xiong, H., & Chen, E. (2023). A Survey on Large Language Models for Recommendation. ArXiv, abs/2305.19860.
- Yang, C., Zhu, Y., Lu, W., Wang, Y., Chen, Q., Gao, C., ... & Chen, Y. (2024). Survey on knowledge distillation for large language models: methods, evaluation, and application. ACM Transactions on Intelligent Systems and Technology.
- Yu, Y., & Kim, N. (2024). Heterogeneous Knowledge Distillation Using Conceptual Learning. IEEE Access, 12, 52803–52814.
- Zhao, W. X., Zhou, K., Li, J., Tang, T., Wang, X., Hou, Y., ... & Wen, J. R. (2023). A survey of large language models. arXiv preprint arXiv:2303.18223, 1(2).