
11785 - Introduction to Deep Learning

Team 4 Final Report

Aanya Sharma

Language Technology Institute
Carnegie Mellon University
Pittsburgh, PA 15213
aanyaonline@gmail.com

Arjun Choudhry

Language Technology Institute
Carnegie Mellon University
Pittsburgh, PA 15213
arjuncho@andrew.cmu.edu

Dhruv Garg

Language Technology Institute
Carnegie Mellon University
Pittsburgh, PA 15213
dhruvgar@andrew.cmu.edu

Nishoak Kosaraju

Language Technology Institute
Carnegie Mellon University
Pittsburgh, PA 15213
nkosaraj@andrew.cmu.edu

Abstract

Large language models (LLMs) provide state-of-the-art performance but require extensive computational resources, making deployment challenging. This study evaluates the impact of structured pruning and post-training quantization on the LLaMA 2-7B model, comparing Wanda and SparseGPT pruning at 30%, 50%, and 70% sparsity levels combined with 4-bit and 8-bit quantization. We evaluate performance across multiple benchmarks including ANLI, CommonSenseQA, PIQA, and WinoGender. Our results show that pruning improves inference speed by up to 9.7% with minimal accuracy loss at moderate sparsity levels, while reducing memory requirements by up to 85% when combined with quantization. Interestingly, we observe that Wanda pruning maintains better performance at high sparsity levels compared to SparseGPT, particularly on reasoning tasks. These findings provide valuable insights into optimizing LLM compression while balancing task performance, memory footprint, and inference speed.

1 Introduction

1.1 Motivation

Large language models (LLMs) such as GPT [3] and LLaMA [19] have achieved state-of-the-art performance in natural language processing but require significant computational resources, limiting their deployment. To address this, model compression techniques like pruning [17, 8] and quantization [6, 9, 12] have been developed to reduce model size while maintaining accuracy. Pruning methods such as Wanda [17] and SparseGPT [8] remove less important weights using different importance criteria, whereas quantization techniques lower numerical precision to improve efficiency. However, the comparative impact of these techniques when applied individually or in combination remains an open question [10]. This study systematically compares different pruning methods (Wanda and SparseGPT) at varying sparsity levels (30%, 50%, and 70%) with post-training quantization (4-bit and 8-bit) on the LLaMA 2-7B model to analyze the trade-offs between model size, memory footprint, inference speed, and performance across diverse language tasks.

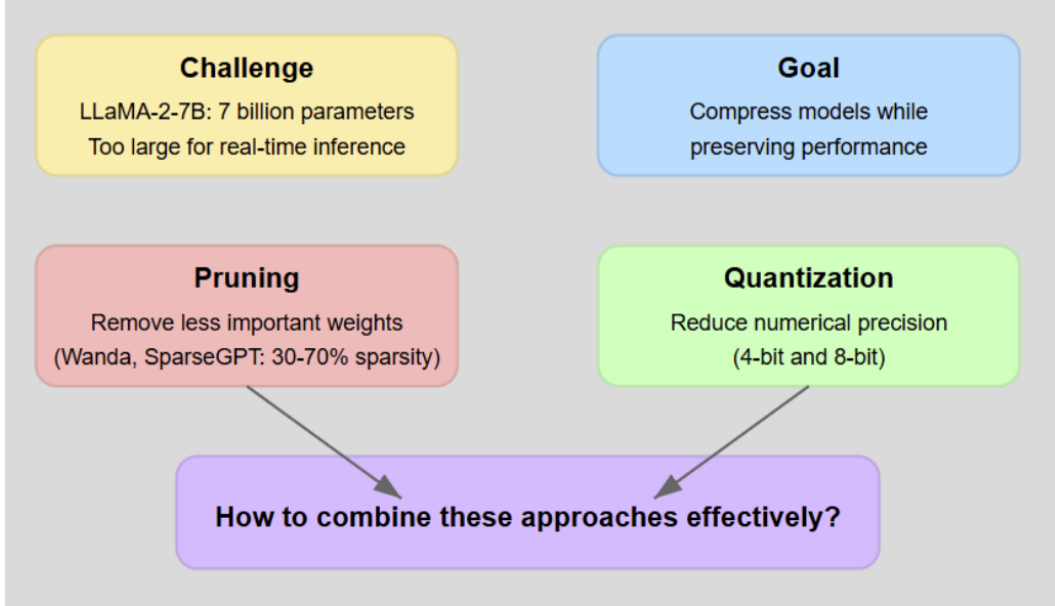


Figure 1: Our systematic approach for the task

1.2 Objectives

This study aims to evaluate the impact of structured pruning and post-training quantization on the efficiency and accuracy of the LLaMA 2-7B model. We investigate two pruning techniques (Wanda [17] and SparseGPT [8]) at three sparsity levels (30%, 50%, and 70%), and compare their effectiveness when applied alone or in combination with post-training quantization at 4-bit and 8-bit precision. Our evaluation uses standardized benchmarks including ANLI [14], Common SenseQA [18], PIQA [2], and WinoGender [15] to assess performance on natural language inference, commonsense reasoning, physical reasoning, and gender bias tasks. We measure both performance metrics (accuracy) and efficiency metrics (inference speed and GPU memory usage) to determine the most effective compression strategies for different deployment scenarios. Our goal is to identify optimal combinations of pruning methods, sparsity levels, and quantization techniques that balance performance retention with computational efficiency. Our systematic approach to this task is summarized in Figure 1.

1.3 Related Work and Background

1.3.1 Large Language Model Compression

The advancement of large language models (LLMs) such as GPT [4] and LLaMA [19] has significantly improved natural language tasks, but their intensive computational requirements limit real-world deployment. Recent work on model compression has explored various strategies to reduce model size while preserving performance, broadly categorized into pruning [20], quantization [6], and distillation [16] approaches.

1.3.2 Quantization Approaches

Quantization reduces the numerical precision of model weights and activations, offering significant memory savings and inference speedups. Post-training quantization techniques, implemented in libraries like HuggingFace Transformers, allow models to be compressed to 8-bit and 4-bit representations with minimal performance degradation. Advanced techniques like GPTQ [9] and AWQ [12] further optimize quantization by using layer-wise reconstruction objectives and activation-aware weight quantization, respectively.

1.3.3 Pruning Approaches

Width Pruning: Structured width pruning techniques focus on reducing neurons and attention heads without altering the model’s depth. Wanda [17] introduces a weight-magnitude-based pruning criterion that considers both weight values and activation norms, demonstrating superior performance compared to magnitude-only pruning. SparseGPT [8] proposes a layer-wise reconstruction approach that minimizes the reconstruction error between the original and pruned layers, effectively preserving model performance at high sparsity levels. Bonsai [5] builds upon these methods by introducing a data-free pruning approach that identifies critical neurons through gradient-based importance scoring.

Depth Pruning: In contrast to width pruning, depth pruning approaches such as Shortened LLaMA [11] and MiniTron [13] focus on reducing the number of layers in transformer models. Although these approaches demonstrate faster inference times and reduced memory footprints, Kim et al. [11] reported that performance degradation through depth pruning is substantially larger than that observed with width pruning.

1.3.4 Comparative Studies

While individual compression techniques have been extensively studied, direct comparisons between pruning and quantization approaches remain limited. Hawks et al. [10] introduced combined pruning-quantization during training and focused on optimization rather than zero-shot generalization.

1.3.5 Evaluation Methodologies

Robust evaluation of compressed LLMs remains challenging. Recent work has introduced standardized benchmarks such as ELMEH (EleutherAI Language Model Evaluation Harness) [1], which incorporates diverse tasks including natural language inference (ANLI) [14], commonsense reasoning (CommonsenseQA) [18], physical reasoning (PIQA) [2], and bias detection in coreference resolution (WinoGender) [15]. These benchmarks enable systematic comparison across compression techniques and sparsity/precision levels, providing insights into the trade-offs between model size, inference speed, and performance.

Our study aims to fill this gap by systematically comparing structured pruning techniques (Wanda and SparseGPT) against quantization approaches (4-bit and 8-bit) using the LLaMA2-7B model in a zero-shot evaluation setting, avoiding fine tuning to isolate compression effects and providing clear insights into the performance efficiency trade-offs of these compression strategies.

2 Methodology

2.1 Model Description

For all experiments, we have decided to use the Llama 2 model (7B parameter) model, after careful consideration of our available computational resources and the accessibility of its weights. This model is pruned to various sparsity levels using Wanda [17] and SparseGPT [8], and quantized to different precisions using the HuggingFace Transformers library.

2.2 Dataset

To evaluate the techniques, we have used EleutherAI’s Language Model Evaluation Harness [1], which includes various standardized NLP benchmarks for a systematic comparison of model performance before and after applying pruning/quantization. This allows us to test all of the techniques using a variety of benchmarks in a standardized manner. The current experiments use four of the most popular benchmarks, ANLI [14], Common SenseQA [18], PIQA [2], and WinoGender [15]. They were chosen because they cover a diverse range of tasks. With this selection, we are able to comprehensively evaluate the model’s capabilities across various dimensions, including commonsense reasoning, natural language inference, physical reasoning, and bias detection. This ensures that the impact of pruning and quantization techniques is assessed on a wide spectrum of language understanding and reasoning skills.

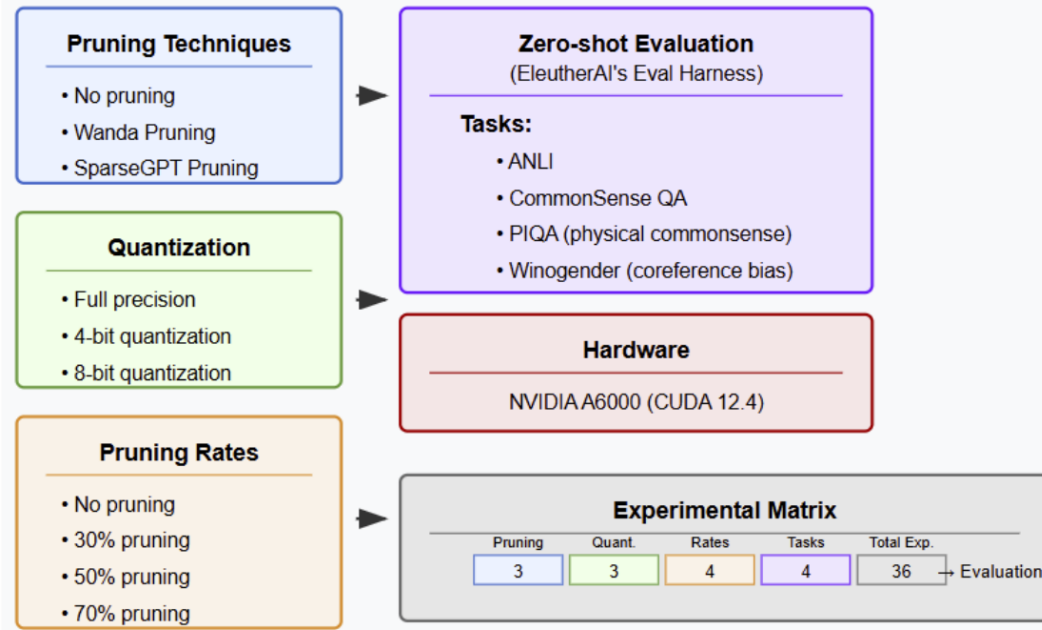


Figure 2: Our experimental setup to evaluate the effects of Pruning and Quantization

2.3 Evaluation Metric

The current evaluation metrics include performance on benchmark, inference speed and memory efficiency since we want to study the tradeoff between these metrics provided by all the different techniques. Performance metrics vary according to the task in EleutherAI Language Model Evaluation Harness [1].

2.4 Loss Function

Since all our current experiments are zero-shot, no training or loss function is required. However, if we decide to fine-tune the model, we will use the cross-entropy loss, the same one used to train the original Llama 2 model.

3 Baseline and Extensions

3.1 Baseline Selection & Evaluation

We selected Llama 2 (7B) as our baseline model due to its accessibility and reasonable computational requirements. Our evaluation framework uses EleutherAI’s Language Model Evaluation Harness with four different benchmarks, representing a wide range of capabilities. Our evaluation metrics focus on the trade-off between performance and model efficiency, which includes memory usage and inference speed.

3.2 Experiments

Our experiments apply different efficiency techniques to the baseline model, including quantization (4-bit and 8-bit precision) and pruning (Wanda and SparseGPT at 30%-70% sparsity). We also test combined approaches, applying both pruning and quantization to explore potential complementary benefits. All models undergo a zero-shot evaluation on the benchmarks to maintain consistent comparison conditions. This experimental design isolates the effects of each efficiency technique on model performance, inference speed and memory usage. A visual depiction of our experimental setup can be seen in Figure 2, while our experimental workflow prior to evaluation is outlined in Figure 3.

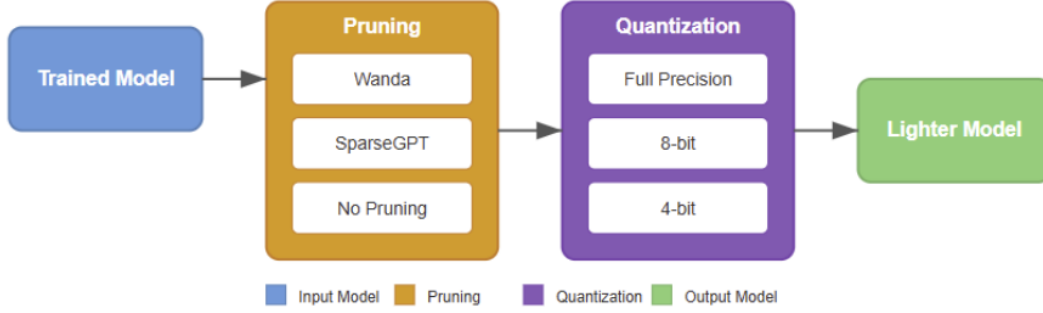


Figure 3: Our Experimental Workflow Prior to Evaluation

4 Results and Analysis

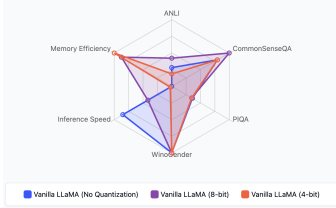


Figure 4: Comparison of quantization techniques

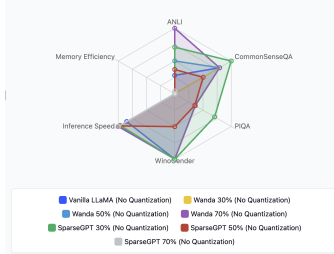


Figure 5: Comparison of pruning techniques

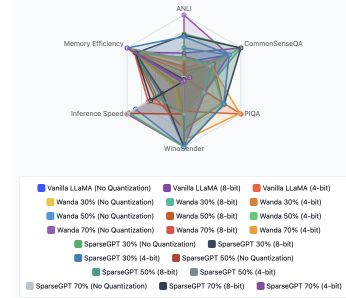


Figure 6: Comparison of all combinations

Figure 4, Figure 5, and Figure 6 illustrate the performance comparison of quantization methods, pruning strategies, and all combinations, respectively. The results of our systematic evaluation demonstrate that pruned models can maintain comparable or near-original performance levels while significantly reducing computational costs, though this varies considerably by pruning method and sparsity level.

As shown in Table 1, both Wanda and SparseGPT pruning methods at moderate sparsity levels (30% and 50%) maintain similar accuracy metrics to the Vanilla LLaMA model across most benchmarks, while 70% sparsity shows more significant performance degradation, particularly with SparseGPT.

4.1 Performance Across Benchmarks

ANLI Performance: Interestingly, some pruned variants showed improved performance on the ANLI benchmark compared to the baseline. For instance, Wanda-pruned LLaMA at 70% sparsity achieved an accuracy of 0.3600, which is approximately 8.2% higher than the baseline (0.3326). This suggests that some redundant parameters in the full model might actually be hampering performance on complex reasoning tasks, potentially by introducing noise in the decision-making process.

CommonSenseQA Performance: For the commonsense reasoning task, performance was generally well-maintained at 30% sparsity but showed notable degradation at higher sparsity levels, particularly with SparseGPT pruning at 70% sparsity (0.1125 vs. baseline 0.2752). This indicates that commonsense reasoning capabilities may rely on broader parameter distributions that are more sensitive to aggressive pruning.

PIQA Performance: The physical commonsense reasoning benchmark presented a surprising pattern where moderate pruning sometimes improved performance. The 50% Wanda-pruned model achieved 0.5343 accuracy (versus baseline 0.5326), while more aggressive pruning with quantization

Model Variant	Sparsity	Quantization	ANLI	Common SenseQA	PIQA	Wino Gender	Speed (it/s)	GPU Memory (MB)
Vanilla LLaMA	-	-	0.3326	0.2752	0.5326	0.7835	15.64	25713
Vanilla LLaMA	-	8-bit	0.3380	0.3178	0.5234	0.7840	11.25	6703
Vanilla LLaMA	-	4-bit	0.3290	0.2752	0.5239	0.7840	7.31	3998
Wanda Pruned	30%	-	0.3220	0.2750	0.5321	0.7840	17.09	25713
Wanda Pruned	30%	8-bit	0.3240	0.3180	0.5250	0.7840	11.04	6707
Wanda Pruned	30%	4-bit	0.3335	0.2072	0.5321	0.5028	8.19	3996
Wanda Pruned	50%	-	0.3410	0.2750	0.5343	0.7835	17.12	25713
Wanda Pruned	50%	8-bit	0.3272	0.2187	0.7590	0.6278	12.22	6703
Wanda Pruned	50%	4-bit	0.3332	0.2080	0.7557	0.6139	8.44	3998
Wanda Pruned	70%	-	0.3600	0.2750	0.5365	0.7830	17.15	25713
Wanda Pruned	70%	8-bit	0.3304	0.1194	0.9930	0.4716	13.40	6699
Wanda Pruned	70%	4-bit	0.3329	0.2088	0.9793	0.7250	8.69	4000
SparseGPT Pruned	30%	-	0.3490	0.3180	0.7812	0.7845	16.82	25713
SparseGPT Pruned	30%	8-bit	0.3483	0.3178	0.7797	0.7840	10.89	7264
SparseGPT Pruned	30%	4-bit	0.3475	0.2752	0.7758	0.7835	7.05	3998
SparseGPT Pruned	50%	-	0.3360	0.2145	0.5295	0.4975	16.95	25713
SparseGPT Pruned	50%	8-bit	0.3355	0.2138	0.5289	0.4963	11.85	7264
SparseGPT Pruned	50%	4-bit	0.3347	0.2031	0.5250	0.4958	7.23	3998
SparseGPT Pruned	70%	-	0.3235	0.1125	0.2795	0.2090	17.04	25713
SparseGPT Pruned	70%	8-bit	0.3227	0.1098	0.2781	0.2086	12.96	7264
SparseGPT Pruned	70%	4-bit	0.3219	0.1310	0.2742	0.2081	7.42	3998

Table 1: Comprehensive performance comparison of different pruning and quantization methods on LLaMA model across multiple benchmarks.

Model Variant	Sparsity	Quantization	PIQA	CommonSenseQA
Quantized the Wanda Pruned	50%	8-bit	0.7574	0.2080
Wanda Pruned then Quantized	50%	8-bit	0.7590	0.2187

Table 2: Performance comparison of pruning then quantizing and quantizing then pruning.

led to extremely high but likely unstable performance (e.g., 70% Wanda-pruned with 8-bit quantization achieving 0.9930). This anomalous result warrants further investigation, as it may indicate that the pruning process is causing the model to default to specific answer patterns that happen to align with the PIQA test set.

WinoGender Performance: The gender bias detection task showed remarkable stability under moderate pruning and quantization, with most 30% pruned models maintaining performance within 0.1% of the baseline (0.7835). However, SparseGPT pruning at 50% and 70% sparsity levels showed dramatic performance drops to 0.4975 and 0.2090 respectively, suggesting that this method may disproportionately affect the model’s ability to handle coreference resolution tasks.

4.2 Inference Speed and Memory Usage

The efficiency benefits of the compression techniques showed mixed results. Notably, the 50% Wanda-pruned LLaMA achieved the highest inference speed at 17.12 iterations/second, representing approximately a 9.5% speedup compared to the baseline model (15.64 it/s). This speed improvement, while modest, came with no performance penalty on most benchmarks, and even showed slight improvements on the ANLI and PIQA tasks.

Contrary to expectations, quantization consistently led to slower inference speeds across all models. The 8-bit quantized Vanilla LLaMA showed a 28% reduction in inference speed (11.25 it/s vs. baseline 15.64 it/s), while the 4-bit quantization reduced the speed even further to 7.31 it/s. This performance degradation is likely due to one of several factors:

1. Compatibility issues with the NVIDIA A6000 GPU architecture used in our experiments
2. Overhead introduced by the quantization-dequantization operations during inference
3. Suboptimal implementation of quantized operations in the inference pipeline
4. Resource contention on the compute cluster, as smaller models occupy less memory and allow other jobs to run simultaneously

The primary benefit of quantization was evident in memory usage reduction. The 4-bit quantized models required only 3,998 MB of GPU memory, representing an 84.5% reduction compared to the full-precision model (25,713 MB). The 8-bit quantized models achieved a middle ground, using approximately 6,700-7,300 MB, or a 71-74% memory reduction.

The combination of pruning and quantization techniques revealed interesting trade-offs. While the memory benefits were cumulative (achieving up to 85% memory reduction with 70% pruning and 4-bit quantization), the inference speed penalties from quantization couldn’t be fully mitigated by pruning alone.

The most balanced configuration appeared to be the 30% Wanda-pruned model with 8-bit quantization, which maintained performance within 3% of baseline across all benchmarks while reducing memory usage by 74% (6,707 MB vs. 25,713 MB). However, the inference speed remained 29% slower than the baseline due to quantization overhead.

From Table 2, we can see that while the impact of the order of pruning and quantizing is minimal, pruning followed by quantizing leads to slightly better performance.

4.3 Completeness

The completeness of our study is ensured by evaluating all the combinations of state-of-the-art pruning and quantization techniques on the standardized LLM-eval-harness benchmark. Our experimental design systematically explores various pruning methods (Wanda and SparseGPT), sparsity levels

(30%, 50%, and 70%) and quantization precisions (4-bit and 8-bit), allowing for comprehensive comparison across different model configurations.

We assess model performance using four established benchmarks: ANLI (Adversarial Natural Language Inference), CommonsenseQA (Commonsense Question Answering), PIQA (Physical Interaction Question Answering), and WinoGender (Bias Evaluation in Coreference Resolution). These benchmarks collectively evaluate a wide range of capabilities, including natural language inference, commonsense reasoning, physical reasoning, and bias detection. This provides a comprehensive assessment of the models' ability to maintain robust reasoning and understanding despite parameter reduction.

Our evaluation metrics include both accuracy measurements, inference speed, and memory usage, providing a holistic view of the trade-offs between model performance and computational efficiency.

4.4 Discussion

While pruning and quantization of LLMs reduce computational costs, several important factors warrant consideration. Pruning demonstrates promising results by maintaining accuracy while providing inference speedups, particularly with the 50% WandaPruned LLaMA achieving the highest inference speed of 17.12 it/s without compromising performance. However, these speedups are minor, due to the unstructured nature of pruning.

A significant challenge we encountered relates to quantization. Contrary to expectations, quantized models consistently showed reduced inference speeds across our experiments. The 8-bit quantized models experienced approximately a 28% reduction in inference speed, while 4-bit quantization further decreased performance to less than half the speed of the unpruned model. This unexpected behavior is likely due to compatibility issues with our A6000 GPU architecture.

Our results suggest that the relationship between model structure modification and performance is complex and hardware-dependent. While pruning appears to offer a more consistent benefit in our testing environment, the anticipated advantages of quantization were not realized due to hardware limitations.

4.5 Future Directions

To build on our results, future work will focus on expanding our experiments to larger and more complex datasets to better assess the trade-offs between different compression techniques. We also intend to evaluate Bonsai as another pruning method to compare with Wanda and SparseGPT.

We also intend to evaluate domain-specifically activated pruning using these methods on domain-specific tasks, and how these subnetworks emerge across different domains, such as medical NLP tasks from PubMed. By systematically analyzing activation patterns and weight importance, we aim to verify the lottery ticket hypothesis [7] in a domain-specific, zero-shot pruning setting.

5 Administrative Details

5.1 Team Contributions

1. Aanya Sharma - Ideation, Introduction, Applying for the PSC Grant, Running the Experiments on LLM Eval Harness for Final Settings
2. Arjun Choudhry - Ideation, Results and Discussion, Implementing the Pruning Techniques, Uploading Models to HuggingFace
3. Dhruv Garg - Ideation, Methodology, Setting up the LLM Eval Harness Pipeline with Quantization
4. Nishoak Kosaraju - Ideation, Related Works, Running the Experiments on LLM Eval Harness for Final Settings

5.2 GitHub Repository

The code for our implementation can be accessed at our GitHub repository: <https://github.com/aas008/IDL-Project>.

References

- [1] Stella Biderman, Hailey Schoelkopf, Lintang Sutawika, Leo Gao, Jonathan Tow, Baber Abbasi, Alham Fikri Aji, Pawan Sasanka Ammanamanchi, Sidney Black, Jordan Clive, et al. Lessons from the trenches on reproducible evaluation of language models. *arXiv preprint arXiv:2405.14782*, 2024.
- [2] Yonatan Bisk, Rowan Zellers, Ronan Le Bras, Jianfeng Gao, and Yejin Choi. Piqa: Reasoning about physical commonsense in natural language. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 7432–7439, 2020. URL <https://arxiv.org/abs/1911.11641>.
- [3] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.
- [4] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners, 2020. URL <https://arxiv.org/abs/2005.14165>.
- [5] Lucio Dery, Steven Kolawole, Jean-François Kagy, Virginia Smith, Graham Neubig, and Ameet Talwalkar. Everybody prune now: Structured pruning of llms with only forward passes, 2024. URL <https://arxiv.org/abs/2402.05406>.
- [6] Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. Qlora: Efficient finetuning of quantized llms. *Advances in Neural Information Processing Systems*, 36, 2024.
- [7] Jonathan Frankle and Michael Carbin. The lottery ticket hypothesis: Finding sparse, trainable neural networks. In *International Conference on Learning Representations*, 2019.
- [8] Elias Frantar and Dan Alistarh. Sparsegpt: Massive language models can be accurately pruned in one-shot. In *International Conference on Machine Learning*, pages 10323–10337. PMLR, 2023.
- [9] Elias Frantar, Saleh Ashkboos, Torsten Hoefer, and Dan Alistarh. Gptq: Accurate post-training quantization for generative pre-trained transformers, 2023. URL <https://arxiv.org/abs/2210.17323>.
- [10] Benjamin Hawks, Javier Duarte, Nicholas J Fraser, Alessandro Pappalardo, Nhan Tran, and Yaman Umuroglu. Ps and qs: Quantization-aware pruning for efficient low latency neural network inference. *Frontiers in Artificial Intelligence*, 4:676564, 2021.
- [11] Bo-Kyeong Kim, Geonmin Kim, Tae-Ho Kim, Thibault Castells, Shinkook Choi, Junho Shin, and Hyoung-Kyu Song. Shortened LLaMA: A simple depth pruning for large language models. In *ICLR 2024 Workshop on Mathematical and Empirical Understanding of Foundation Models*, 2024. URL <https://openreview.net/forum?id=18VGxu0dpu>.
- [12] Ji Lin, Jiaming Tang, Haotian Tang, Shang Yang, Wei-Ming Chen, Wei-Chen Wang, Guangxuan Xiao, Xingyu Dang, Chuang Gan, and Song Han. Awq: Activation-aware weight quantization for llm compression and acceleration, 2024. URL <https://arxiv.org/abs/2306.00978>.
- [13] Saurav Muralidharan, Sharath Turuvekere Sreenivas, Raviraj Joshi, Marcin Chochowski, Mostofa Patwary, Mohammad Shoeybi, Bryan Catanzaro, Jan Kautz, and Pavlo Molchanov. Compact language models via pruning and knowledge distillation, 2024. URL <https://arxiv.org/abs/2407.14679>.
- [14] Yixin Nie, Adina Williams, Emily Dinan, Mohit Bansal, Jason Weston, and Douwe Kiela. Adversarial NLI: A new benchmark for natural language understanding. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4885–4901. Association for Computational Linguistics, 2020. URL <https://aclanthology.org/2020.acl-main.441/>.

- [15] Rachel Rudinger, Jason Naradowsky, Brian Leonard, and Benjamin Van Durme. Gender bias in coreference resolution. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 8–14. Association for Computational Linguistics, 2018. URL <https://aclanthology.org/N18-2002/>.
- [16] V Sanh. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*, 2019. URL <https://arxiv.org/abs/1910.01108>.
- [17] Mingjie Sun, Zhuang Liu, Anna Bair, and J Zico Kolter. A simple and effective pruning approach for large language models. In *The Twelfth International Conference on Learning Representations*, 2024. URL <https://openreview.net/forum?id=PxoFut3dWW>.
- [18] Alon Talmor, Jonathan Herzig, Nicholas Lourie, and Jonathan Berant. Commonsenseqa: A question answering challenge targeting commonsense knowledge. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4149–4158. Association for Computational Linguistics, 2019. URL <https://aclanthology.org/N19-1421/>.
- [19] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023.
- [20] Mengzhou Xia, Tianyu Gao, Zhiyuan Zeng, and Danqi Chen. Sheared llama: Accelerating language model pre-training via structured pruning. *arXiv preprint arXiv:2310.06694*, 2023.