AWQ Support on Obsolete Server GPUs

Bowen Gu
Harvard University
bowen_gu@dfci.harvard.edu

Xiaomin Li
Harvard University
xiaominli@g.harvard.edu

Moulinrouge Kaspar
MIT
mfkaspar@mit.edu

Andrew Palacci
Harvard University
andrewpalacci@college.harvard.edu

Abstract

The scalability of Large Language Models (LLMs) in resource-limited environments, particularly on obsolete server GPUs, presents a significant challenge due to the immense model sizes and intensive computational requirements. This study leverages the principles of Activation-aware Weight Quantization (AWQ) to address these challenges. We used the llm-awq GitHub repository from MIT HAN LAB as the codebase, modified it to replace the GEMM kernel with the GEMV kernel to enable support for compute capabilities 7.0 GPUs, and added the AWQ support for the Mistral model family to expand the AWQ application. To test the performance of the AWQ INT4 quantized LLMs, we used a server from the Brigham and Women's Hospital (BWH) Department of Medicine, which only has a single NVIDIA V100 GPU, selected eight popular LLMs, and compared the full-precision and quantized model VRAM usage and performance on three different accuracies using a private EHR dataset that includes patients' Social Determinants of Health (SDoH). We found that while only requiring about 1/6 of the VRAM, the AWQ INT4 quantized models can achieve comparable performance to the full-precision counterparts on all three accuracies even on obsolete GPUs, which further proved the effectiveness of AWQ on LLM compression.

1 Background

The rapid evolution of Large Language Models (LLMs) such as Llama and ChatGPT has significantly advanced the field of natural language processing. Nonetheless, these models' expansive sizes impose considerable demands on computational hardware, particularly in terms of Graphics Processing Unit (GPU) Video RAM (VRAM) requirements. In response to these challenges, various quantization strategies, including Gradient-based Partial Quantization (GPTQ)[FAHA23] and Round-to-Nearest Quantization (RTN)[NAVB$^+$20], have been developed. GPTQ, however, necessitates a substantial volume of examples for effective quantization, leading to performance degradation when examples are insufficient. Conversely, RTN is hindered by its sensitivity to initial weight distributions in the model, which can result in instability in smaller, yet critical, weight values that are essential for model performance.

In an effort to mitigate the limitations of these existing quantization approaches, Activation-aware Weight Quantization (AWQ)[LTT$^+$23] has been introduced as an innovative and effective alternative. AWQ primarily concentrates on the model's most salient weights, implementing per-channel scaling to reduce quantization errors significantly. The underlying principle of AWQ is the acknowledgment that weights in an LLM do not contribute equally to its performance. By selectively focusing on a small, but pivotal, subset of weights—typically ranging from 0.1% to 1% of the total—AWQ achieves a substantial reduction in quantization errors. This reduction is facilitated through the analysis of activation patterns rather than the weights themselves, thereby identifying the weights that most substantially influence model performance. Unlike GPTQ and RTN, AWQ does not require input examples for quantization and is not affected by the instabilities associated with weight distribution. Furthermore, empirical evidence demonstrates that AWQ consistently surpasses these methods in terms of performance across various model scales, task types, and testing scenarios.

AWQ has demonstrated notable efficacy in language modeling tasks such as WikiText-2 and in common sense QA benchmarks including PIQA [BZB+19], WinoGrande [SBBC19], and ARC [CCE+18]. However, the application of AWQ-quantized LLMs in understanding Electronic Health Record (EHR) notes remains limited due to HIPAA and privacy concerns which restrict the utilization of clinical data. Additionally, most widely-adopted AWQ packages, such as MIT HAN LAB's llm-awq[LTT+23] and AutoAWQ, lack support for GPUs with compute capabilities below 7.5 (Turing architecture), thereby posing challenges in deploying AWQ on older GPU models. Furthermore, there is a notable absence of support for the recently introduced Mistral model family in many established AWQ packages, limiting the broader applicability of AWQ.

In this study, we modified the codebase from the MIT HAN LAB llm-awq to implement the support for the compute capability 7.0 GPUs (Volta architecture) and the Mistral model family using A4W16 GEMV CUDA kernel. We also tested the quantized model performance in extracting the Social Determinant of Health (SDoH) from unstructured textual clinical data from the Mass General Brigham Hospital system on a single NVIDIA V100 GPU. Our findings indicate that while requiring 1/6 of the VRAM, the AWQ INT4 quantized LLMs achieved comparable accuracies to their full-precision counterparts even on obsolete GPUs, which shows the effectiveness of AWQ in optimizing VRAM usage and maintaining performance.

2 Methods

2.1 Data Source

The raw data set is from Mass General Brigham (MGB) EHRs, which are readily accessible for research through a centralized clinical data registry, the Research Patient Data Repository (RPDR). The data set contains directly accessible raw data on structured diagnosis codes from inpatient and outpatient visits, medication prescribing and dispensing records, vital signs, lifestyle factors, and laboratory test results for MGB patients from 2007 to 2020. The data set also includes free-text elements such as ambulatory notes, discharge summaries, and specialty reports for the above patients.

We then extracted the free text that is associated with the patients' social documentation from their progress notes using regular expression matching. Since the patients' social documentation is added incrementally over time, we only used the most recent social documentation for each patient. This results in a refined data set that contains 166,523 unique patients.

2.2 SDoH Questions and Human-Evaluated Labels

To decide which SDoH we could extract from the social documentation, we did a manual review of the first 200 patients' social documentation and summarized 9 aspects of the patient's SDoH that appear in at least 5

Due to the size of the LLMs and the corpus that the models are pre-trained on, they do not need to be trained again for specific downstream tasks. As a result, for the 200 reviewed patients, we split the first 100 patients' social documentation as the validation set and the remaining 100 patients' social documentation as the test set.

For each of the 9 SDoH aspects, we designed a question and various corresponding choices to quantify the SDoH aspect. The SDoH questions, together with the choices distribution in the validation and the test set, are shown in Table S.1.

For each of the 200 patients, two human experts manually labeled the 9 SDoH aspects to one of the quantified choices according to the labeling criteria documentation. The labeling was done individually.

After completing the labeling, the two human experts compared their labeling results and eliminated any labeling inconsistencies within the dataset. New criteria were also added to the labeling criteria documentation that addressed the causes of these inconsistencies.

2.3 LLM Choices

We selected 8 LLMs that have shown great performance on the LLM leaderboard hosted by Hugging Face [BFH$^+$23, GTB$^+$21, CCE$^+$18, ZHB$^+$19, HBB$^+$21, LHE22]. The selected LLMs, together with their benchmark on Eleuther AI Language Model Evaluation Harness [GTB$^+$21], provide a unified framework to test generative language models on a large number of different evaluation tasks, are shown in Table 1 below.

Model	Parameter Size	Average	ARC	HellaSwag	MMLU	Truthful QA	Winogrande	GSM8K
openchat_3.5	7B	61.24	63.91	84.79	64.94	46.38	80.58	26.84
zephyr-7b-beta	7B	61.59	62.46	84.35	60.7	57.83	77.11	27.07
vicuna-7b-v1.5	7B	52.06	53.24	77.39	51.04	50.34	72.14	8.19
Llama-2-7b-chat-hf	7B	50.74	52.9	78.55	48.32	45.57	71.74	7.35
vicuna-13b-v1.5	13B	55.41	57.08	81.24	56.67	51.51	74.66	11.3
WizardL_M-13B-V1.2	13B	54.76	59.04	82.21	54.64	47.27	71.9	13.5
Llama-2-13b-chat-hf	13B	54.91	59.04	81.94	54.64	44.12	74.51	15.24
vicuna-33b-v1.3	33B	58.54	62.12	83	59.22	56.16	77.03	13.72

Table 1: The Eleuther AI Language Model Evaluation Harness benchmarks

2.4 Baseline Model

To evaluate the LLM performance, we designed a baseline model that used pattern matching to extract the answers to the SDoH questions from the patients' social documentation. The matching patterns were designed according to the labeling criteria. If a match was found in the patients' social documentation, the output answer was guaranteed to be one of the choices of the SDoH question. If no match was found in the patients' social documentation, the output answer was defaulted to be "Not mentioned". The specific pattern for each SDoH question is shown in Table S.2.

2.5 Full-precision LLMs and AWQ quantized LLMs

For each LLM listed in Table 1, we quantized them using a modified version of the MIT HAN LAB llm-awq codebase. Specifically, to enable support for the V100 GPU,which has Compute Capability 7.0 and is not supported by the GEMM kernel in the codebase, we modified the codebase to remove all references to "gemm_cuda.h" and "gemm_cuda.cu" in the codebase. Moreover, for each reference to "awq_inference_engine.gemm_cuda", we replaced it with a function call to "awq_inference_engine.gemv_cuda".This makes the codebase use the GEMV kernel during AWQ, which supports the V100 GPU. Moreover, to enable support for the Mistral model family, we implemented the integration for MistralDecoderLayer, MistralRMSNorm, and MistralForCausalLM based on the structure of LlamaDecoderLayer, LlamaRMSNorm, and LlamaForCausalLM into the codebase in "auto_scale.py" and "pre_quant.py". The modified codebase is available on GitHub. All the LLMs were then quantized using 4-bit quantization with a group size of 128.

The full-precision models and the corresponding tokenizers are downloaded from Hugging Face. Due to constrained computational resources, we loaded the full-precision models to the CPU, and the quantized models to the GPU to run inference. For the model setup, we set "max_length = 2048" to prevent the model response from being truncated. For the tokenizer setup, we set "skip_special_tokens = True", and "clean_up_tokenization_spaces = False". We used the default values for the rest parameters for the model and the tokenizer. The quantized LLMs were compared side-by-side with the full-precision LLMs on the model performance.

2.6 LLM performance evaluation

To calculate model accuracy, we built an auto-grader by comparing the model response to the ground-truth human labels. The auto-grader used the following grading logic:

- If the response from the model was an exact match to the human label, then GREEN.
- If the response from the model was an invalid answer since it was not one of the choices provided, then RED.
- If the response from the model did not match the human label, then RED.

To evaluate the model performance, we used three metrics: Accuracy A, Accuracy B, and Accuracy C, to evaluate the LLM performance. The three accuracies are defined as follows, where Accuracy A can be considered as a weighted average of Accuracy B and Accuracy C, with the weights depending on the missingness of the SDoH aspects in the text. For the number of questions where the ground truth is labeled or is not labeled as "not mentioned", consult Table 2 in Appendix for details.

$$\text{Accuracy A} = \frac{\text{Total \# of GREEN}}{\text{Total \# of questions}}$$

$$\text{Accuracy B} = \frac{\text{Total \# of GREEN where the ground truth is NOT labeled as ``not mentioned''}}{\text{Total \# of questions where the ground truth is NOT labeled as ''not mentioned''}}$$

$$\text{Accuracy C} = \frac{\text{Total \# of GREEN where the ground truth is labeled as ``not mentioned''}}{\text{Total \# of questions where the ground truth is labeled as ``not mentioned''}}$$

For each LLM and the baseline model, we calculated all three accuracies on all 9 SDoH questions.

3 Results

3.1 Accuracy Comparison between Full Precision and Quantized LLMs

To compare the performance between the LLMs and the baseline, for each LLM, we averaged the model accuracy across the 9 SDoH questions for each LLM and the baseline and did this for the three accuracies defined in the "LLM performance evaluation" section. Figure 1 shows the three accuracies of each LLM together with their quantized variants, with the three horizontal lines being the baseline accuracies.

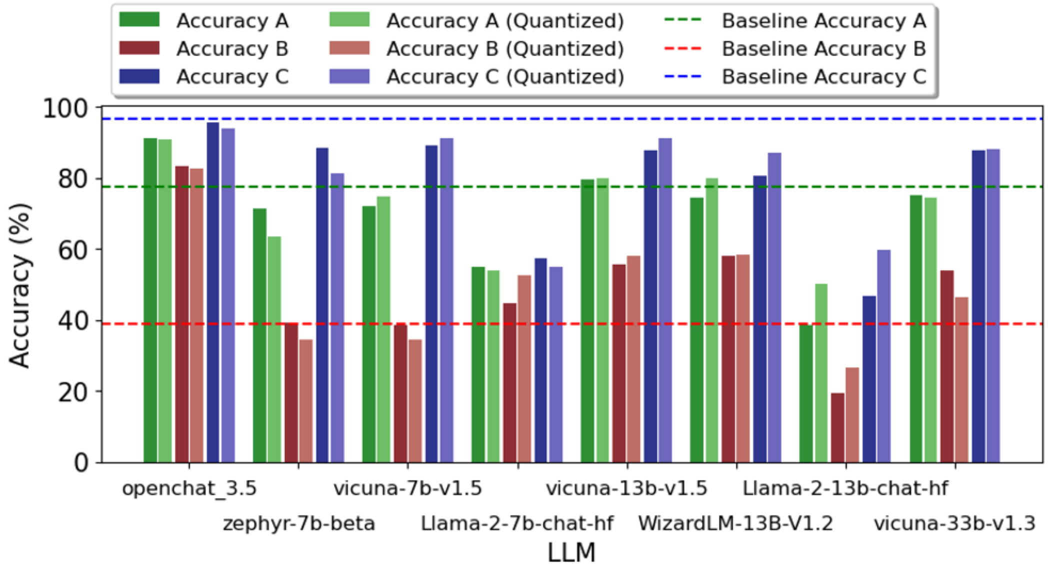

Figure 1: Average accuracy comparison of the LLMs over the baseline

An approximation of the GPU VRAM usage of LLMs with different parameter sizes is shown in Figure 3, where the VRAM usage within the threshold of the V100 GPU is measured using the "nvidia-smi" command on the BWH server, and the VRAM usage above the threshold is interpolated using the fact the LLM VRAM consumption is in direct proportion to its parameter size.

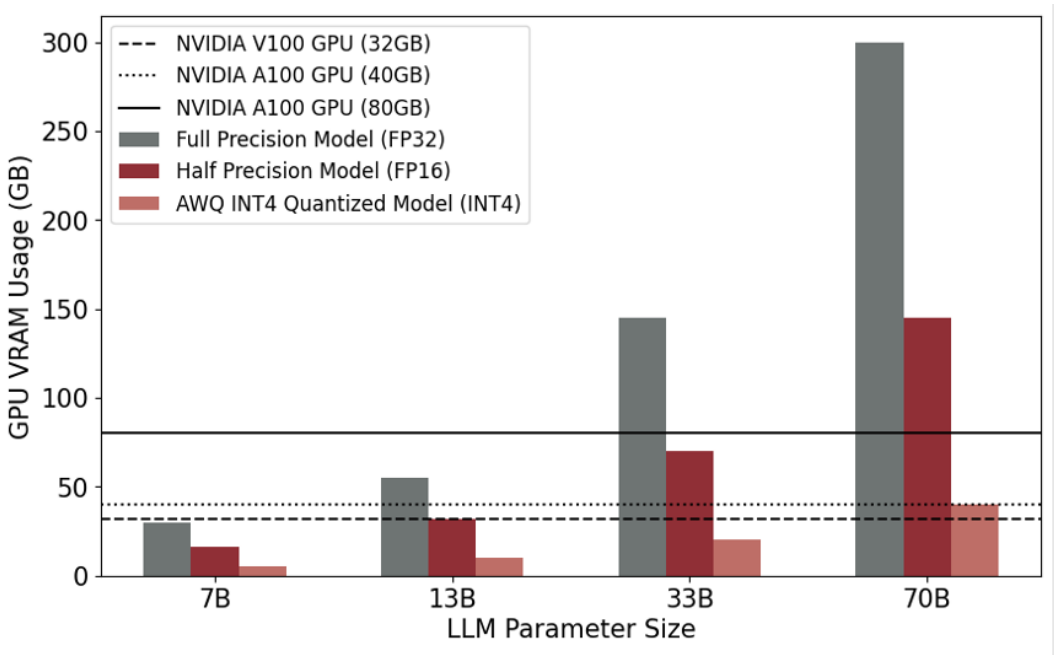

Figure 2: Approximation of GPU VRAM usage on LLMs with different parameter sizes

AWQ has been proven to achieve a great compression rate of LLM VRAM usage yet well preserve the performance on generic benchmarks [LTT$^+$23]. According to Figure 2, we found that when using the private patient social documentation as the test set, for all 8 LLMs tested, the quantized LLMs had comparable performance to their full precision counterparts across all 3 accuracies, which indicated that while requiring about $1/6$ of the VRAM compared to the full precision counterparts, the AWQ quantized LLMs could well maintain their performance in SDoH data extraction.

4 Conclusion

In this work, we refined the codebase from the MIT HAN LAB llm-awq GitHub repository by adding support for Volta architecture GPUs (compute capability 7.0) and the Mistral model family. By testing the LLM's SDoH extraction capability using a private patient EHR data set that contains patients' social documentation. We found that even on obsolete GPUs, AWQ can not only optimize the GPU VRAM usage, but also maintain the performance.

The integration of LLMs into computational research has significantly advanced tasks like information extraction and data analysis. However, the vast size of these models poses a barrier to many institutions, primarily due to high VRAM requirements that older GPUs cannot meet. Most existing quantization methods, aimed at reducing VRAM usage, are incompatible with GPUs having compute capability lower than 7.5 and often exclude the latest LLM structures like the Mistral model family. This limitation not only restricts the use of advanced LLMs in resource-constrained environments but also hinders the adoption of the most current LLM technologies.

The integration of AWQ with obsolete GPUs heralds a new era of accessibility and application potential for LLMs. Firstly, smaller academic institutions and research labs with limited funding can now utilize advanced LLMs for tasks like semantic analysis, predictive modeling, and large-scale data interpretation. This capability can significantly enhance the quality and scope of research conducted, leading to more profound and impactful discoveries. Secondly, in sectors like healthcare, where patient data analysis and predictive diagnostics are crucial, the ability to deploy advanced LLMs on older hardware could substantially improve patient care and outcomes. Moreover, the inclusion of the Mistral model family extends these benefits by enabling institutions to work with the latest LLM structures. This advancement opens avenues for better results in cutting-edge research. Institutions can now explore better-quality models, leading to richer insights and more nuanced understandings of unstructured notes. The support for newer models like Mistral also ensures that research stays current with global technological trends, fostering a more dynamic and innovative research environment.

The primary contribution of this study is the facilitation of AWQ on older GPU models, expanding the accessibility of advanced LLMs to a broader range of users and applications. Additionally, our work's inclusion of the latest models, such as Mistral, ensures that research institutions remain at the cutting edge of LLM technology. However, our study is limited by the lack of diverse test data, which restricts our ability to assess the quantized LLM's performance across various tasks comprehensively. Despite this limitation, the positive outcomes from both generic benchmarks and specific tests using our private EHR dataset provide substantial evidence of the maintained performance of AWQ-quantized LLMs on a range of tasks, even with obsolete GPUs.

5 Future Directions

Our work uses AWQ to compress LLMs in a way that is manageable and compatible with outdated GPUs — this has multiple clearly impactful future directions. First, this work can be expanded to a wide variety of models (beyond the eight that we tested) and also to a wide variety of tasks. In fact, we hoped to test AWQ compression on the WikiText-2 dataset[MXBS16] using a perplexity benchmark, but decided to point our focus to SDoH extraction. Since our current SDoH task involves a limited, discrete set of possible outcomes, we wonder whether this task contributed to a lack of degradation in model performance. It is possible that using AWQ, specifically on outdated GPUs, could result in increased degradation for model perplexity on complex language tasks, while leaving more straightforward accuracy tasks unhindered. This would be valuable to explore further using WikiText-2 and other standard LLM benchmarks.

Our work is also extensible to a wide variety of GPUs — although the original llm-awq codebase is built for GPUs with compute capability 7.5 and 8.0 and our tests were run on a V100 GPU with compatibility 7.0, it would also be intriguing to examine this method's extensibility to GPUs with capabilities 6.5 or lower. When, if at all, does the AWQ process become exceedingly difficult? How do GPUs change structurally over time in a way that affects the ways LLMs can be quantized and can perform? Answering these and similar questions could be particularly useful for organizations like Brigham and Women's Hospital or those with even fewer computational resources, guiding the organizations towards techniques that prevent them from making large, unnecessary expenditures on new GPUs with the help of optimization techniques like AWQ.

Appendix

A.1 Supplementary table for the distribution of the SDoH question choices

Table 2: Distribution of the SDoH question choices

SDoH Question	SDoH Choices	Appearance Count (Validation)	Appearance Count (Test)
Q1. What is the patient's marital status?	Single	28	34
	Widowed	10	7
	Divorced	6	9
	Married	49	41
	Not mentioned	7	9
Q2. How many children does the patient have?	0	3	7
	1	21	22
	2	16	18
	3	7	9
	4	4	0
	5 or more	3	4
	Not mentioned	46	40
Q3. Does the patient currently use tobacco?	Yes	3	1
	No	17	16
	Not mentioned	80	83
Q4. Does the patient currently consume alcohol?	Yes	11	10
	No	9	3
	Not mentioned	80	87
Q5. Does the patient currently use illicit drugs?	Yes	0	1
	No	9	7
	Not mentioned	91	92
Q6. Does the patient live alone?	Yes	8	7
	No	48	54
	Not mentioned	44	39
Q7. What is the patient's employment status?	Employed	34	28
	Jobless	12	14
	Retired	27	26
	Not mentioned	27	32
Q8. What is the patient's highest education level?	Elementary school	1	2
	Middle school	1	0
	High school	3	7
	College	4	12
	Graduate school	4	2
	Not mentioned	87	77
Q9. Does the patient exercise?	Yes	15	10
	No	0	0
	In the past	1	0
	Not mentioned	84	90

A.2 Supplementary table for the matching pattern for the baseline model

Table 3: Distribution of the SDoH question choices

SDoH Question	Matching Pattern (Case Insensitive)	Resulting Choice
Q1. What is the patient's marital status?	married	Married
	widowed	Widowed
	divorced/separated	Divorced
	single/no/spouse/partner/boyfriend/girlfriend/long-term partner	Single
	default	Not mentioned
Q2. How many children does the patient have?	no children/0 children	0
	1 child/1 son/1 daughter	1
	2 children/2 son/2 daughter	2
	3 children/3 son/3 daughter	3
	4 children/4 son/4 daughter	4
	5 children/6 children/7 children/8 children/9 children/10 children	5 or more
	default	Not mentioned
Q3. Does the patient currently use tobacco?	currently smokes/cigarettes/cigars/smokeless tobacco/tobacco use/smoke.	Yes
	never used tobacco/quit smoking/quit tobacco/quit smoke/past tobacco use/never smoke/no smoke	No
	default	Not mentioned
Q4. Does the patient currently consume alcohol?	consumes alcohol/drinks/alcohol/ETOH	Yes
	never consumed alcohol/no alcohol/ETOH	No
	default	Not mentioned
Q5. Does the patient currently use illicit drugs?	illicit drug/uses/drugs/cocaine/marijuana/substance abuse	Yes
	never used drugs/sober from drugs/past drug use/deny illicit drugs	No
	default	Not mentioned
Q6. Does the patient live alone?	alone	Yes
	with husband/with wife/with child/with children/with son/with daughter/with boyfriend/with girlfriend/with grandparents/with uncle/with aunt/with parents/with father/with mother/with dad/with mom	No
	default	Not mentioned
Q7. What is the patient's employment status?	full-time/part-time/employed/work/employ	Employed
	stay-at-home/at home/unemployed/on disability/homemaker/not work/no work/jobless	Jobless
	retired/former employee/worked/used to work default	Retired Not mentioned
Q8. What is the patient's highest education level?	elementary school /1st grade /2nd grade /3rd grade /4th grade /5th grade	Elementary school
	middle school /6th grade /7th grade /8th grade	Middle school
	high school /9th grade /10th grade /11th grade	High school
	college /Associates /Bachelors /BS /BA /freshman /sophomore /junior /senior /post-bacc	College
	graduate school /grad school /Masters /degree /MS /MBA /MPH /MENG /JD /Doctoral degree /PHD /MD	Graduate school
	default	Not mentioned
Q9. Does the patient exercise?	exercise /walk /run /dance /gym /treadmill /yoga	Yes
	does not exercise /never exercise /not exercising	No
	used to exercise /used to walk	In the past
	default	Not mentioned

References

[BFH$^+$23] Edward Beeching, Clémentine Fourrier, Nathan Habib, Sheon Han, Nathan Lambert, Nazneen Rajani, Omar Sanseviero, Lewis Tunstall, and Thomas Wolf. Open llm leaderboard. `https://huggingface.co/spaces/HuggingFaceH4/open_llm_leaderboard`, 2023.

[BZB$^+$19] Yonatan Bisk, Rowan Zellers, Ronan Le Bras, Jianfeng Gao, and Yejin Choi. Piqa: Reasoning about physical commonsense in natural language, 2019.

[CCE$^+$18] Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. Think you have solved question answering? try arc, the ai2 reasoning challenge, 2018.

[FAHA23] Elias Frantar, Saleh Ashkboos, Torsten Hoefler, and Dan Alistarh. Gptq: Accurate post-training quantization for generative pre-trained transformers, 2023.

[GTB$^+$21] Leo Gao, Jonathan Tow, Stella Biderman, Sid Black, Anthony DiPofi, Charles Foster, Laurence Golding, Jeffrey Hsu, Kyle McDonell, Niklas Muennighoff, Jason Phang, Laria Reynolds, Eric Tang, Anish Thite, Ben Wang, Kevin Wang, and Andy Zou. A framework for few-shot language model evaluation, September 2021.

[HBB$^+$21] Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding, 2021.

[LHE22] Stephanie Lin, Jacob Hilton, and Owain Evans. Truthfulqa: Measuring how models mimic human falsehoods, 2022.

[LTT$^+$23] Ji Lin, Jiaming Tang, Haotian Tang, Shang Yang, Xingyu Dang, and Song Han. Awq: Activation-aware weight quantization for llm compression and acceleration. *arXiv preprint arXiv:2306.00978*, 2023.

[MXBS16] Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. Pointer sentinel mixture models, 2016.

[NAVB$^+$20] Markus Nagel, Rana Ali Amjad, Mart Van Baalen, Christos Louizos, and Tijmen Blankevoort. Up or down? Adaptive rounding for post-training quantization. In Hal Daumé III and Aarti Singh, editors, *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 7197–7206. PMLR, 13–18 Jul 2020.

[SBBC19] Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. Winogrande: An adversarial winograd schema challenge at scale, 2019.

[ZHB$^+$19] Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. Hellaswag: Can a machine really finish your sentence?, 2019.