ATQ: Activation Transformation for Weight-Activation Quantization of Large Language Models

Yundong Gai
Huawei Cloud
gaiyundong@huawei.com

Ping Li
Huawei Cloud
liping61@huawei.com

Abstract

There are many emerging quantization methods to resolve the problem that the huge demand on computational and storage costs hinders the deployment of Large language models (LLMs). However, their accuracy performance still can not satisfy the entire academic and industry community. In this work, we propose ATQ, an INT8 weight-activation quantization of LLMs, that can achieve almost lossless accuracy. We employ a mathematically equivalent transformation and a triangle inequality to constrain weight-activation quantization error to the sum of a weight quantization error and an activation quantization error. For the weight part, transformed weights are quantized along the `in-feature` dimension and the quantization error is compensated by optimizing following in-features. For the activation part, transformed activations are in the normal range and can be quantized easily. We provide comparison experiments to demonstrate that our ATQ method can achieve almost lossless in accuracy on OPT and LLaMA families in W8A8 quantization settings. The increase of perplexity is within 1 and the accuracy degradation is within 0.5 percent even in the worst case.

1 Introduction

Large language models (LLMs) exhibit remarkable performance across various tasks. Many different LLMs were proposed, such as GPT(Brown et al., 2020), OPT(Zhang et al., 2022), LLaMA(Touvron et al., 2023a,b), BLOOM(Le Scao et al., 2023) and so on. However, the large model size and the huge computation cost prevents their deployment in production. Quantization is considered as a promising technique for model compression and inference acceleration, and it can be categorized into two main approaches: quantization-aware training(QAT) and post-training quantization(PTQ).

QAT(Liu et al., 2023b; Dettmers et al., 2024) can achieve comparable performance with the original model. However, QAT is not practical due to the huge training cost and the unavailability of training data. Researchers prefer to use PTQ to quantize LLMs at the cost of some accuracy degradation. In recent few years, a great number of PTQ methods(Frantar et al., 2022a,b; Lin et al., 2023; Xiao et al., 2023; Wei et al., 2023; Shao et al., 2023) for LLMs spring up. They are categorized into two classes: weight-only quantization and weight-activation quantization.

Actually, the inference process of LLMs composes of two stages: Prefilling and Decoding. During the prefilling stage, the huge cost is caused by high-precision matrix-matrix multiplication, which can be alleviated by weight-activation quantization. During the decoding stage, it generates only one token using general matrix-vector multiplication. The decoding stage is memory-bound, which means the decoding latency is constrained by the movement of weights between memories. Therefore, weight-only quantization methods can reduce the weight movement cost.

The number of bits in quantization of LLMs tends to be lower and lower. For examples, LLM.int8() (Dettmers et al., 2022) quantizes the non-outlier columns with 8-bit, and SmoothQuant (Xiao et al., 2023) quantizes weight and activation to INT8 datatype. GPTQ quantizes weight to 3 or 4 bits, and OmniQuant can quantize weight down to 2 bits or quantize weight and activation to 4 bits. QUIK (Ashkboos et al., 2023) and Atom (Zhao et al., 2024) adopt hybrid-precision quantization, where most activations and weights are quantied into INT4 keeping only a small part of activations and weights in high precision. LLM-FP4 (Liu et al., 2023a) quantizes both weights and activations down to FP4. DecoupleQ (Guo et al., 2024) achieves 2-bit uniform PTQ, and OneBit (Ma et al., 2024) introduces a 1-bit quantization framework. Although these methods can achieve an efficient post-training quantization solution for LLMs, their

Findings of the Association for Computational Linguistics: EMNLP 2024, pages 17187–17194
November 12-16, 2024 ©2024 Association for Computational Linguistics

accuracy degradation can not satisfy the practical applications.

In this paper, we propose ATQ, a novel weight-activation post-training quantization pipeline. The main contribution can be summarized as follows.

- A triangle inequality is employed to constrain the weight-activation quantization error to the sum of a weight quantization error and an activation quantization error.
- A mathematically equivalent transformation is applied to activations, so that activations are in normal range and can be quantized easily, and weight quantization error can be compensated.
- ATQ method can achieve almost lossless in accuracy under the W8A8 quantization setting. The perplexity increase is within 1 and the accuracy degradation on zero-shot tasks is within 0.5 percent.

2 Related work and motivation

In this section, we review some related works and present our research motivation.

2.1 Weight-only quantization

As the term implies, the weight-only quantization method only quantize LLMs' weight. Therefore, the size of LLMs and the time of weight movements between memories can be decreased. GPTQ (also OPTQ) is a typical weight-only quantization method, which is build on the traditional OBQ algorithm (Frantar et al., 2023). They quantize one or several weight rows, and compensate the quantization error by optimizing following weight rows. OBQ quantizes the weight row (along the `in-feature` dimension) in a greedy order, whereas GPTQ quantizes weight rows in the uniform left-to-right order without update of the Hessian matrix $\boldsymbol{H}$, which can reduce the computation cost substantially. Besides, GPTQ and SparseGPT(Frantar and Alistarh, 2023) share the same Hessian matrix $\boldsymbol{H}$ and Cholesky decomposition, so the combination of quantization and sparsification is feasible.

2.2 Weight-activation quantization

Activation outliers with wider distribution ranges in LLMs make traditional quantization methods can not be directly applied into LLMs. SmoothQuant introduce a scaling factor to migrate part of the quantization difficulty from activations to weight, but the migration extent is controled by a handcraft hyper-parameter α. It is a tradeoff between activation and weight. Besides activation outliers on magnitude, Outlier Suppression+ find that the asymmetry of activation outliers between different channels make the activation range to be quantized larger and introduce the channel-wise shifting parameter to suppress outliers furtherly. OmniQuant develop two components, LWC and LET, to learn quantization parameters on a relatively small calibration dataset. However, the learning process of OmniQuant is time-consuming.

2.3 Motivation

Despite above weight-activation quantization methods have their respective rationalities, their accuracy performance can not satisfy the entire academic and industry community. In other words, the accuracy degradation after quantization is too large to be commercial deployment. The goal of this paper is to propose a lossless W8A8 quantization solution for LLMs whose accuracy degradation is unprecedentedly small.

3 ATQ Method

Traditional PTQ methods with gradient optimization is hard to be applied into modern LLMs due to the huge solution space. In our ATQ method, we consider the block-wise quantization error minimization problem for each linear layer. The weight-activation quantization problem can be formulated as follows,

$$\min \|\boldsymbol{X}\boldsymbol{W}^T - Q_x(\boldsymbol{X})Q_w(\boldsymbol{W})^T\| \tag{1}$$

where $\boldsymbol{W}$ and $\boldsymbol{X}$ are weight and activation, $Q_w(\cdot)$ and $Q_x(\cdot)$ are weight and activation quantizers respectively. According to the triangle inequality, the objection function can be written as

$$\begin{aligned} &\|\boldsymbol{X}\boldsymbol{W}^T - Q_x(\boldsymbol{X})Q_w(\boldsymbol{W})^T\| \\ \leq&\|\boldsymbol{X}\boldsymbol{W}^T - \boldsymbol{X}Q_w(\boldsymbol{W})^T\| \\ &+ \|(\boldsymbol{X} - Q_x(\boldsymbol{X}))Q_w(\boldsymbol{W})^T\|. \end{aligned} \tag{2}$$

The first term on the right side of Eq.(2), $\|\boldsymbol{X}\boldsymbol{W} - \boldsymbol{X}Q_w(\boldsymbol{W})\|$, is a weight-only quantization problem and can be resolved by GPTQ. The second term represents an activation quantization problem for given quantized weights. This error is very large due to activation outliers observed in SmoothQuant and Outlier Suppression+. To reduce the activation quantization error, We perform a mathematically equivalent transformation on the linear layer, which can be written as

$$\begin{aligned}\boldsymbol{Y} &= \boldsymbol{X}\boldsymbol{W}^T + \boldsymbol{B} \\ &= [(\boldsymbol{X} - \delta) \oslash s] \cdot [s \odot \boldsymbol{W}^T] + [\boldsymbol{B} + \delta \boldsymbol{W}^T] \\ &= \widetilde{\boldsymbol{X}}\widetilde{\boldsymbol{W}}^T + \widetilde{\boldsymbol{B}}\end{aligned} \tag{3}$$

where $\boldsymbol{Y}$ represents the output of a linear layer, $\delta \in \mathbb{R}^{1\times C_{in}}$ and $s \in \mathbb{R}^{1\times C_{in}}$ are channel-wise shifting and scaling parameters. $\widetilde{\boldsymbol{X}}$, $\widetilde{\boldsymbol{W}}$ and $\widetilde{\boldsymbol{B}}$ are transformed and equivalent activation, weight and bias. '$\oslash$' and '$\odot$' means division and multiplication along the `in_feature` dimension. The transformation of activations can be implemented by merging $(\boldsymbol{X} - \delta) \oslash s$ into the layernorm before the linear layer to be quantized.

Now the objective function can be rewritten as

$$\begin{aligned}&\|\widetilde{\boldsymbol{X}}\widetilde{\boldsymbol{W}}^T - \widetilde{\boldsymbol{X}}Q_w(\widetilde{\boldsymbol{W}})^T + \widetilde{\boldsymbol{X}}Q_w(\widetilde{\boldsymbol{W}})^T \\ &\quad - Q_x(\widetilde{\boldsymbol{X}})Q_w(\widetilde{\boldsymbol{W}})^T\| \\ \leq &\|\widetilde{\boldsymbol{X}}\widetilde{\boldsymbol{W}}^T - \widetilde{\boldsymbol{X}}Q_w(\widetilde{\boldsymbol{W}})^T\| \\ &\quad + \|(\widetilde{\boldsymbol{X}} - Q_x(\widetilde{\boldsymbol{X}}))Q_w(\widetilde{\boldsymbol{W}})^T\|.\end{aligned} \tag{4}$$

The first term on the right side is still a weight-only quantization problem and can be resolved by the GPTQ method. The only difference is that the activation used for calculating the Hessian Matrix $\widetilde{\boldsymbol{H}}$ is the transformed $\widetilde{\boldsymbol{X}}$ rather than the original $\boldsymbol{X}$. In the second term, for the given weight $Q_w(\widetilde{\boldsymbol{W}})$, if we transformed activation outliers to be in the normal range of $[-1, 1]$, the quantization error of transformed activations $\widetilde{\boldsymbol{X}}$ should be small enough.

The problem becomes how to find appropriate shifting and scaling factors, δ and s, to elimate activation outliers. In LLMs, activation outliers are asymmetric and have large magnitude on certain channels. We first find the center of activations per-channel on a small calibration dataset, which is the shifting factor δ, and move it to 0. Mathematically, we have

$$\delta_j = \frac{\max(\boldsymbol{X}_{:,j}) + \min(\boldsymbol{X}_{:,j})}{2}. \tag{5}$$

Now, the range of outliers in the j_{th} channel is $(\max(\boldsymbol{X}_{:,j}) - \min(\boldsymbol{X}_{:,j}))/2$. To minimize activation quantization error, we scale all activations into the range of $[-1, 1]$ unless they are already in this range, which means

$$s_j = \max(1.0, \frac{\max(\boldsymbol{X}_{:,j}) - \min(\boldsymbol{X}_{:,j})}{2}). \tag{6}$$

Note that, we do not incorporate weight into shifting and scaling factors, which means we do not need to consider the tradeoff between activation and weight. All of quantization difficulties are migrated from activation to weight, and weight quantization errors can be compensated by optimizing following weight in-features, which is exactly our essential difference from other weight-activation quantization methods.

4 Experiments

In the experiments, we compare our ATQ method with FP16 baselines, GPTQ (Frantar et al., 2022a), SmoothQuant (Xiao et al., 2023), Outlier Suppression+ (Wei et al., 2023) and OmniQuant (Shao et al., 2023).

4.1 Settings

Remember that we focus on accuracy degradation minimization rather than computational efficiency, we keep all experiments in fake INT8 weight and INT8 activation quantization settings. ATQ is generalized from GPTQ, so we set FP16 LLMs and INT8 GPTQ quantized LLMs as baselines. To be fair, all of these methods are calibrated or trained on a small calibration dataset composed of 128 randomly selected 2048 tokens from the WikiText2. All of these methods are tested on two families of LLMs, OPT (Zhang et al., 2022) and Llama (Touvron et al., 2023a,b). Following the previous work, we first evaluate the language generation performance by the perplexity of quantized models on WikiText2 (Merity et al., 2016), PTB (Marcus et al., 1994) and C4 (Raffel et al., 2020). Then, we evaluate quantized models' performance on zero-shot tasks including PIQA (Tata and Patel, 2003), ARC (Boratko et al., 2018), BoolQ (Clark et al., 2019), HellaSwag (Zellers et al., 2019), Winogrande (Sakaguchi et al., 2021), which are executed by using the `lm-eval-harness` (Gao et al., 2023). Experiments are completed on 40GB Nvidia A100 GPUs.

4.2 OPT W8A8 Quantization

OPT, presented by MetaAI, is composed of a Multi-Head Attention(MHA) module with pre-layernorm and a Feed-Forward Network(FFN) with post-layernorm. We employ the mathematically equivalent transformation and quantization on all linear layers except for the second linear layer of FFN, fc2, as OmniQuant. Table 1 shows the perplexity of W8A8 quantization OPT models on the WikiText2 test dataset. Our ATQ method outperforms other

OPT-PPL-WikiText2 ↓	#Bits	125m	1.3B	2.7B	6.7B	13B	30B	66B
Baseline	W16A16	27.655	14.623	12.471	10.861	10.128	9.559	9.339
GPTQ	W8A16	27.655	14.612	12.481	10.859	10.129	9.559	9.344
SmoothQuant	W8A8	27.771	14.697	**12.468**	10.888	10.368	OOM	OOM
Outlier Suppression+	W8A8	34.914	15.541	12.781	11.127	10.928	10.193	OOM
OmniQuant	W8A8	27.699	14.666	12.483	**10.861**	10.138	OOM	OOM
ATQ (ours)	W8A8	**27.677**	**14.640**	12.473	10.863	**10.127**	**9.559**	9.341

Table 1: WikiText2 perplexity of weight-activation quantization results on OPT models

LLaMA-PPL-WikiText2 ↓	#Bits	1-7B	1-13B	2-7B	2-13B	2-70B	3-8B	3-70B
Baseline	W16A16	5.677	5.091	5.472	4.884	3.319	6.135	2.856
GPTQ	W8A16	5.679	5.091	5.474	4.884	3.320	6.140	2.856
SmoothQuant	W8A8	5.712	5.125	5.510	4.927	OOM	6.252	OOM
OmniQuant	W8A8	5.692	5.099	5.490	4.897	OOM	-	-
ATQ	W8A8	**5.678**	**5.091**	**5.475**	**4.889**	3.321	**6.145**	2.891

Table 2: WikiText2 perplexity of weight-activation quantization results on LLaMA models

methods in most cases of OPT. In the OPT-6.7B case, ATQ is worse than OmniQuant, but the PPL increases by only 0.005. Additional perplexity on the PTB validation dataset and the C4 validation dataset and zero-shot experiments in appendix can also demontrate this conclusion, see Tables A1, A2 and A3. OmniQuant wins our ATQ method in some zero-shot tasks, but the computation time of OmniQuant is far longer than ours.

4.3 LLaMA W8A8 Quantization

LLaMA models are new open languange models with superior performance. Each decoderlayer of LLaMA consists of a Multi-Head Attention module and a MLP Module. Following OmniQuant, we apply the activation transformation and quantization into all linear layers except for the gate projection and the down projection in the MLP. Table 2 presents the perplexity of W8A8 quantization LLaMA-1/2/3 models on the WikiText2 test dataset. We can also see that our ATQ method outperforms SmoothQuant and OmniQuant. Corresponding perplexity results on the PTB validation dataset and the C4 validation dataset and zero-shot accuracy results are presented in Appendix Tables. A4, A5 and A6, which can also demonstrate the conclusion. Note that, OmniQuant is not supported LLaMA-3.

4.4 Ablation study

We provide some comparison experiments on OPT-13B and Llama-3-8B to show the effectiveness of diverse parts on accuracy. Firstly, we apply the activation transformation and quantize transformed activations and weights without error compensation (ATQ w/o EC). Secondly, we quantize original activations directly without activation and weights with error compensation (ATQ w/o AT). Experiment results are compared with GPTQ and ATQ in Table 3. We can see that activation transformation and quantization error compensation are contributive to the reduction of accuracy degradation and employing them together can achieve better accuracy.

PPL-WikiText2 ↓	OPT-13B	Llama-3-8B
Baseline	10.128	6.135
GPTQ	10.129	6.140
ATQ (w/o EC)	10.134	6.182
ATQ (w/o AT)	11.843	6.180
ATQ	**10.127**	**6.145**

Table 3: WikiText2 perplexity of ablation experiments

5 Conclusion and future work

We propose an advanced and accurate post-training quantization scheme called ATQ. In the scenario of W8A8 quantization, ATQ exceeds other existing weight-activation quantization methods on accuracy degradation in most cases. Considering the limitation discussed in the next section, we plan to incorporate sparsification into this scheme for higher compression ratio, and implement the CUDA kernel and operator for computational efficiency and memory saving.

Limitations

In this section, we briefly discuss some limitations.

More models

We only evaluate the performance of ATQ on OPT and Llama families. The accuracy performance of ATQ will be evaluated on more models like BLOOM, Falcon, in the short future.

Real quantization

We focus on the minimization of accuracy degradation of W8A8 quantization of LLMs, so we use fake quantization in this paper. However, its corresponding CUDA kernel and operator for real quantization should be implemented before deployment.

Higher compression ratio

We only consider 8-bit weight-activation quantization in this paper. It is interesting and challenging to explore lossless compression methods with a higher compression ratio, such as lower-bit weight-activation quantization and combination of 8-bit weight-activation quantization and sparsification.

References

Saleh Ashkboos, Ilia Markov, Elias Frantar, Tingxuan Zhong, Xincheng Wang, Jie Ren, Torsten Hoefler, and Dan Alistarh. 2023. Towards end-to-end 4-bit inference on generative large language models. *arXiv preprint arXiv:2310.09259*.

Michael Boratko, Harshit Padigela, Divyendra Mikkilineni, Pritish Yuvraj, Rajarshi Das, Andrew McCallum, Maria Chang, Achille Fokoue-Nkoutche, Pavan Kapanipathi, Nicholas Mattei, et al. 2018. A systematic classification of knowledge, reasoning, and context within the arc dataset. *arXiv preprint arXiv:1806.00358*.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Christopher Clark, Kenton Lee, Ming-Wei Chang, Tom Kwiatkowski, Michael Collins, and Kristina Toutanova. 2019. Boolq: Exploring the surprising difficulty of natural yes/no questions. *arXiv preprint arXiv:1905.10044*.

Tim Dettmers, Mike Lewis, Younes Belkada, and Luke Zettlemoyer. 2022. Gpt3. int8 (): 8-bit matrix multiplication for transformers at scale. *Advances in Neural Information Processing Systems*, 35:30318–30332.

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. 2024. Qlora: Efficient finetuning of quantized llms. *Advances in Neural Information Processing Systems*, 36.

Elias Frantar and Dan Alistarh. 2023. Sparsegpt: Massive language models can be accurately pruned in one-shot. In *International Conference on Machine Learning*, pages 10323–10337. PMLR.

Elias Frantar, Saleh Ashkboos, Torsten Hoefler, and Dan Alistarh. 2022a. Gptq: Accurate post-training quantization for generative pre-trained transformers. *arXiv preprint arXiv:2210.17323*.

Elias Frantar, Saleh Ashkboos, Torsten Hoefler, and Dan Alistarh. 2022b. Optq: Accurate quantization for generative pre-trained transformers. In *The Eleventh International Conference on Learning Representations*.

Elias Frantar, Sidak Pal Singh, and Dan Alistarh. 2023. Optimal brain compression: A framework for accurate post-training quantization and pruning.

Leo Gao, Jonathan Tow, Baber Abbasi, Stella Biderman, Sid Black, Anthony DiPofi, Charles Foster, Laurence Golding, Jeffrey Hsu, Alain Le Noac'h, Haonan Li, Kyle McDonell, Niklas Muennighoff, Chris Ociepa, Jason Phang, Laria Reynolds, Hailey Schoelkopf, Aviya Skowron, Lintang Sutawika, Eric Tang, Anish Thite, Ben Wang, Kevin Wang, and Andy Zou. 2023. A framework for few-shot language model evaluation.

Yi Guo, Fanliu Kong, Xiaoyang Li, Hui Li, Wei Chen, Xiaogang Tian, Jinping Cai, Yang Zhang, and Shouda Liu. 2024. decoupleq: Towards 2-bit post-training uniform quantization via decoupling parameters into integer and floating points. *arXiv preprint arXiv:2404.12759*.

Teven Le Scao, Angela Fan, Christopher Akiki, Ellie Pavlick, Suzana Ilić, Daniel Hesslow, Roman Castagné, Alexandra Sasha Luccioni, François Yvon, Matthias Gallé, et al. 2023. Bloom: A 176b-parameter open-access multilingual language model.

Ji Lin, Jiaming Tang, Haotian Tang, Shang Yang, Wei-Ming Chen, Wei-Chen Wang, Guangxuan Xiao, Xingyu Dang, Chuang Gan, and Song Han. 2023. Awq: Activation-aware weight quantization for llm compression and acceleration. *arXiv preprint arXiv:2306.00978*.

Shih-yang Liu, Zechun Liu, Xijie Huang, Pingcheng Dong, and Kwang-Ting Cheng. 2023a. Llm-fp4: 4-bit floating-point quantized transformers. *arXiv preprint arXiv:2310.16836*.

Zechun Liu, Barlas Oguz, Changsheng Zhao, Ernie Chang, Pierre Stock, Yashar Mehdad, Yangyang Shi, Raghuraman Krishnamoorthi, and Vikas Chandra. 2023b. Llm-qat: Data-free quantization aware training for large language models. *arXiv preprint arXiv:2305.17888*.

Shuming Ma, Hongyu Wang, Lingxiao Ma, Lei Wang, Wenhui Wang, Shaohan Huang, Li Dong, Ruiping Wang, Jilong Xue, and Furu Wei. 2024. The era of 1-bit llms: All large language models are in 1.58 bits. *arXiv preprint arXiv:2402.17764*.

Mitch Marcus, Grace Kim, Mary Ann Marcinkiewicz, Robert MacIntyre, Ann Bies, Mark Ferguson, Karen Katz, and Britta Schasberger. 1994. The penn treebank: Annotating predicate argument structure. In *Human Language Technology: Proceedings of a Workshop held at Plainsboro, New Jersey, March 8-11, 1994*.

Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. 2016. Pointer sentinel mixture models. *arXiv preprint arXiv:1609.07843*.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of machine learning research*, 21(140):1–67.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2021. Winogrande: An adversarial winograd schema challenge at scale. *Communications of the ACM*, 64(9):99–106.

Wenqi Shao, Mengzhao Chen, Zhaoyang Zhang, Peng Xu, Lirui Zhao, Zhiqian Li, Kaipeng Zhang, Peng Gao, Yu Qiao, and Ping Luo. 2023. Omniquant: Omnidirectionally calibrated quantization for large language models. *arXiv preprint arXiv:2308.13137*.

Sandeep Tata and Jignesh M Patel. 2003. Piqa: An algebra for querying protein data sets. In *15th International Conference on Scientific and Statistical Database Management, 2003.*, pages 141–150. IEEE.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. 2023a. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023b. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Xiuying Wei, Yunchen Zhang, Yuhang Li, Xiangguo Zhang, Ruihao Gong, Jinyang Guo, and Xianglong Liu. 2023. Outlier suppression+: Accurate quantization of large language models by equivalent and optimal shifting and scaling. *arXiv preprint arXiv:2304.09145*.

Guangxuan Xiao, Ji Lin, Mickael Seznec, Hao Wu, Julien Demouth, and Song Han. 2023. Smoothquant: Accurate and efficient post-training quantization for large language models. In *International Conference on Machine Learning*, pages 38087–38099. PMLR.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. 2019. Hellaswag: Can a machine really finish your sentence? *arXiv preprint arXiv:1905.07830*.

Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, et al. 2022. Opt: Open pre-trained transformer language models. *arXiv preprint arXiv:2205.01068*.

Yilong Zhao, Chien-Yu Lin, Kan Zhu, Zihao Ye, Lequn Chen, Size Zheng, Luis Ceze, Arvind Krishnamurthy, Tianqi Chen, and Baris Kasikci. 2024. Atom: Low-bit quantization for efficient and accurate llm serving. *Proceedings of Machine Learning and Systems*, 6:196–209.

A Example Appendix

In this section, we provide a comprehensive presentation of all experiment results across various methods, models and datasets to complement the main content.

- PTB perplexity of weight-activation quantization results on OPT models
- C4 perplexity of weight-activation quantization results on OPT models
- Accuracy of 6 zero-shot tasks of weight-activation quantization results on OPT models
- PTB perplexity of weight-activation quantization results on LLaMA models
- C4 perplexity of weight-activation quantization results on LLaMA models
- Accuracy of 6 zero-shot tasks of weight-activation quantization results on LLaMA models

OPT-PPL-PTB ↓	#Bits	125m	1.3B	2.7B	6.7B	13B	30B	66B
Baseline	W16A16	32.550	16.964	15.113	13.086	12.341	11.842	11.358
GPTQ	W8A16	32.558	16.975	15.128	13.091	12.339	11.842	11.352
SmoothQuant	W8A8	**32.530**	17.252	15.149	13.152	12.591	OOM	OOM
Outlier Suppression+	W8A8	38.733	17.822	15.418	13.854	12.751	12.040	OOM
OmniQuant	W8A8	32.650	17.029	15.122	13.094	12.348	OOM	OOM
ATQ	W8A8	32.571	**16.992**	**15.121**	**13.089**	**12.342**	**11.848**	11.359

Table A1: PTB perplexity of weight-activation quantization results on OPT models

OPT-PPL-C4 ↓	#Bits	125m	1.3B	2.7B	6.7B	13B	30B	66B
Baseline	W16A16	24.605	14.721	13.165	11.743	11.200	10.694	10.284
GPTQ	W8A16	24.622	14.727	13.167	11.744	11.200	10.694	10.285
SmoothQuant	W8A8	24.641	14.862	13.204	11.803	11.243	OOM	OOM
Outlier Suppression+	W8A8	54.048	30.377	25.993	22.779	22.133	20.330	OOM
OmniQuant	W8A8	24.633	14.754	13.173	11.747	11.203	OOM	OOM
ATQ	W8A8	**24.613**	**14.739**	**13.169**	**11.746**	**11.201**	**10.695**	10.285

Table A2: C4 perplexity of weight-activation quantization results on OPT models

OPT-Acc ↑	Method	#Bits	PIQA	ARC-e	ARC-c	HellaSwag	BoolQ	Winogrande
125M	Baseline	W16A16	62.89	43.56	19.03	29.20	55.47	50.43
	GPTQ	W8A16	63.06	43.43	19.28	29.15	56.36	50.28
	OS+	W8A8	63.55	42.89	19.28	28.87	53.55	51.93
	OmniQuant	W8A8	63.28	**43.69**	**19.54**	29.16	**55.93**	49.88
	ATQ	W8A8	**63.11**	43.35	19.20	**29.18**	55.14	**50.67**
1.3B	Baseline	W16A16	71.60	56.90	23.38	41.49	57.74	59.91
	GPTQ	W8A16	71.65	57.20	23.29	41.56	57.95	59.67
	OS+	W8A8	71.38	56.52	24.40	41.27	56.45	57.85
	OmniQuant	W8A8	**71.55**	57.24	23.46	41.54	57.92	**59.35**
	ATQ	W8A8	71.44	**57.28**	23.46	**41.56**	**58.04**	59.19
2.7B	Baseline	W16A16	73.78	60.77	26.79	45.86	60.37	60.77
	GPTQ	W8A16	73.72	60.94	26.88	45.89	60.61	61.17
	OS+	W8A8	74.37	60.90	27.05	46.04	58.90	61.80
	OmniQuant	W8A8	73.72	**61.03**	26.71	45.82	60.55	60.62
	ATQ	W8A8	**73.99**	60.86	**26.96**	**45.83**	60.55	**60.69**
6.7B	Baseline	W16A16	76.28	65.57	30.46	50.51	66.06	65.19
	GPTQ	W8A16	76.28	65.61	30.55	50.47	66.09	64.88
	OS+	W8A8	76.66	65.11	30.72	50.57	65.05	65.19
	OmniQuant	W8A8	76.22	**65.78**	30.63	50.47	**66.24**	65.04
	ATQ	W8A8	**76.39**	65.70	**30.72**	50.47	66.02	**65.19**
13B	Baseline	W16A16	75.84	67.13	32.94	52.43	65.93	65.04
	GPTQ	W8A16	75.95	67.00	33.19	52.45	65.81	64.96
	OS+	W8A8	76.06	67.38	33.28	52.21	65.90	65.51
	OmniQuant	W8A8	75.79	**67.30**	**33.19**	52.40	65.72	64.64
	ATQ	W8A8	**75.95**	67.00	32.94	**52.44**	**65.75**	**65.35**

Table A3: Accuracy of 6 zero-shot tasks of weight-activation quantization results on OPT models

LLaMA-PPL-PTB ↓	#Bits	1-7B	1-13B	2-7B	2-13B	2-70B	3-8B	3-70B
Baseline	W16A16	27.340	19.225	22.511	28.873	15.647	10.592	8.165
GPTQ	W8A16	27.273	19.238	22.583	28.872	15.648	10.600	8.168
SmoothQuant	W8A8	30.478	22.486	21.152	29.748	OOM	10.735	OOM
OmniQuant	W8A8	27.434	19.293	**20.565**	**28.568**	OOM	-	-
ATQ	W8A8	**27.420**	**19.223**	22.595	28.858	15.634	**10.612**	8.211

Table A4: PTB perplexity of weight-activation quantization results on LLaMA models

LLaMA-PPL-C4 ↓	#Bits	**1-7B**	**1-13B**	**2-7B**	**2-13B**	**2-70B**	**3-8B**	**3-70B**
Baseline	W16A16	7.079	6.611	6.973	6.468	5.521	9.446	7.166
GPTQ	W8A16	7.080	6.612	6.973	6.468	5.522	9.453	7.170
SmoothQuant	W8A8	7.122	6.642	7.022	6.508	OOM	9.650	OOM
OmniQuant	W8A8	7.098	6.626	6.993	6.486	OOM	-	-
ATQ	W8A8	**7.081**	**6.612**	**6.975**	**6.470**	5.522	**9.465**	7.243

Table A5: C4 perplexity of weight-activation quantization results on LLaMA models

LLaMA-Acc ↑	Method	#Bits	**PIQA**	**ARC-e**	**ARC-c**	**HellaSwag**	**BoolQ**	**Winogrande**
1-7B	Baseline	W16A16	78.67	75.29	41.89	56.96	75.08	70.01
	OmniQuant	W8A8	78.35	67.05	38.31	56.41	72.87	66.61
	ATQ	W8A8	**78.62**	**75.59**	**41.89**	**56.96**	**74.80**	**70.01**
1-13B	Baseline	W16A16	79.16	77.36	46.42	59.91	77.89	72.69
	OmniQuant	W8A8	78.84	74.37	44.11	59.05	68.13	69.61
	ATQ	W8A8	**79.11**	**77.27**	**46.59**	**59.94**	**77.98**	**72.77**
2-7B	Baseline	W16A16	78.07	76.35	43.43	57.16	77.74	69.06
	OmniQuant	W8A8	**78.13**	69.40	40.02	56.62	71.38	66.61
	ATQ	W8A8	77.91	**76.30**	**43.17**	**57.12**	**77.65**	**69.14**
2-13B	Baseline	W16A16	79.05	79.38	48.46	60.05	80.58	72.14
	OmniQuant	W8A8	78.84	73.15	45.73	59.64	68.72	69.46
	ATQ	W8A8	**79.05**	**79.55**	**48.63**	**60.11**	**80.67**	**71.82**
3-8B	Baseline	W16A16	79.65	80.26	50.26	60.13	81.13	73.48
	ATQ	W8A8	79.65	80.30	50.85	60.15	81.41	73.09

Table A6: Accuracy of 6 zero-shot tasks of weight-activation quantization results on LLaMA models