International Journal for Multidisciplinary Research (IJFMR)

E-ISSN: 2582-2160 ● Website: www.ijfmr.com ● Email: editor@ijfmr.com

Advancements in Large Language Model Efficiency: A Literature Review on 1-bit Quantization

Lalitha Shree C P[1], Nethravathi B[2]

[1,2]Department of Information Science & Engineering, JSS Academy of Technical Education, Bengaluru

Abstract

Large Language Models suffer from most of the challenges regarding computational cost, memory, and energy consumption, which makes their scaling difficult. BitNet b1.58 addresses these issues by introducing a novel 1.58-bit quantization with ternary weights {-1, 0, 1}. This can achieve performance comparable to FP16 models with much lower resource requirements. BitNet b1.58 provides 2.71x faster inference and 3.55x less memory usage compared to FP16 baselines. Its ternary weights allow for very efficient feature filtering, making it a versatile choice for many AI applications. This makes it a standout solution, balancing high performance with resource efficiency. While BitNet b1.58 has its 1-bit mover heads that make it affordable for edge and mobile devices, it also allows for longer sequences. This will mark further developments toward making AI scalable and resource-aware for various applications.

Keywords: LLMs, 1 bit Quantization, BitNet, BitNet b1.58

1. Introduction

Large language models (LLMs) have transformed many domains-primarily natural language processing, text generation, and reasoning-as their rapid growth has expanded them. However, the real deployment of these models is constrained by their major computational and memory demands, significantly scaled to billions of parameters. Addressing these issues has led to a resurgence of research into efficient methods of quantization, cutting down the size of the model along with the cost of computation while holding on to performance. This literature review focuses on the works of [1], that introduce a 1.58-bit quantization strategy to large language models. This study shows how ternary quantization (-1, 0, +1) achieves performance parity with full-precision models by maintaining the accuracy and simultaneously reduces memory and computational overhead. Among the notable works is [2], which brings in the concept of incoherence processing to enable 2-bit quantization for LLMs, which is the first theoretical guarantee for such methods while maintaining the accuracy of the model. [3] deals with activation outliers using a rescaling approach to improve sub-4-bit weight quantization. [4] introduces an iterative, column-wise error compensation strategy for PTQ, with state-of-the-art performance for large models like OPT and BLOOM. Meanwhile, [5] and [6] focus on optimizing computation by leveraging mixed-precision operations and lookup table-based general matrix multiplication, respectively. These innovations address key challenges such as precision loss, latency, and energy efficiency in low-bit quantization. Further developments include [7], which uses Hadamard rotations for balanced activation quantization, and [8] which incorporates quantization within the training pipeline to reduce activation errors. The [9] investigate

3-state quantization using learnable thresholds, thus providing comparable accuracy with fewer precision values. These research studies as a whole demonstrate the multiplicity of approaches to address the issues of LLM scalability and efficiency.By placing [1] in this broader context, this review emphasizes its contribution to the advancement of the field of LLM quantization. The findings presented here provide a comprehensive understanding of the methodologies and their implications while identifying avenues for future research to further optimize low-bit quantization techniques.

2. Preliminaries

Paper	Problem faced	Technologies/Methodologies used	Results	Future Work
QuIP: 2-Bit Quantization of Large Language Models with Guarantees [1]	Post-training parameter quantization in large language models, improving runtime efficiency without reducing accuracy. It deals with the problem of weight and Hessian matrix quantization incoherence.	Quantization with incoherence processing (QuIP), which incorporates an adaptive rounding procedure along with efficient pre- and post-processing steps involving random orthogonal matrices. It also contains theoretical analysis for an LLM-scale quantization algorithm.	preprocessing improves upon existing quantization algorithms and yields the first viable results for LLM quantization using only two bits per weight. The method has been shown to improve both accuracy and efficiency in quantizing large language models.	the optimization of the adaptive rounding procedure, additional preprocessing techniques to improve incoherence, and analysis of the method on other types of neural networks besides language models. It could also be interesting to study the scalability of the method to even larger models.
PB-LLM: PARTIALLY BINARIZED LARGELANGUAGEMODELS [2]	Binarization methods cause the performance of LLMs to collapse at low bits, especially 1-bit. Quality drops sharply beyond 4 bits in the best quantization methods. Linguistic reasoning is	PB-LLM: This method filters only a few salient weights during binarization with the help of higher-bit storage. PTQ: uses the principles of GPTQ for recovering the binarized weight matrix by leveraging the Hessian matrix. QAT: The salient	PB-LLM shows tremendous improvement in reasoning capability compared to state-of-the-art techniques and better metrics for zero-shot common sense reasoning tasks. Its derivative,	Investigate optimal scaling factors and their performance impact. Find efficient ways to store salient weights. Adapt the PB-LLM methodology to other architectures and tasks. Improve

	difficult to maintain with extreme low-bit quantization.	weights learned during training are frozen and the optimal scaling factors that minimize the quantization error are learned. Salient Weight Detection: the salient weights are detected using the magnitude and the Hessian criteria.	PB-GPTQ, significantly outperforms RTN and other quantization approaches with a lower perplexity score for the C4 and WikiText2 datasets. Even when 50% of the weights are not binarized, PB-LLM retains language ability so that successful quantization holds.	QAT and PTQ techniques for trainability and performance recovery for quantized models.
BitNet: Scaling 1-bit Transformers forLarge Language Models [3]	High deployment energy consumption and memory footprint associated with large language models. Model performance cannot be optimized using methods of lower precision quantization.	BitNet: 1-bit transformer architecture, with implementation and use of BitLinear for training from scratch on top of the drop-in replacement for nn.Linear for weights to have 1bit, grouped quantization, together with normalization, to benefit efficient model parallelism with an applied straight-through estimator and finally mixed precision training with regards to the gradients of all functions as well as all the optimizer's states.	Achieved competitive language modeling performance with lower memory use and energy costs than SoTA 8-bit quantization and FP16 baselines. Energy cost was reduced up to 38.8x compared to FP16 Transformers. Downstream accuracy was maintained for Hellaswag, Winograd, etc. using 1-bit weights. A scali	Scaling the model size of BitNet and training steps. Looking into applications of BitNet in other architectures such as RetNet for training large language models. Investigating lower precision quantization for activations.

			ng law similar to that of the full-precision Transformers has been observed, which looks good.	
ATOM: LOW-BITQUANTIZATION FOREFFICIENT ANDACCURATELLMSERVING [4]	High operational cost and low throughput in LLM serving due to inefficient methods of quantization that do not fully exploit modern GPU capabilities, thus suboptimal performance.	Atom presents a method of low-bit quantization that uses mixed precision, intricate group quantization, dynamic quantization, and KV-cache quantization to improve throughput without sacrificing accuracy.	Atom achieves up to 7.73× throughput improvement over FP16 and 2.53× over INT8, with negligible accuracy loss (1.4% drop in zero-shot accuracy, 0.3 increase in perplexity for Llama-65B).	Future work includes optimizing Atom for newer model architectures and hardware, and adapting it for emerging data formats like FP4 and MX.
BiLLM: Pushing the Limit of Post-Training Quantization for LLMs [5]	Existing quantization techniques struggle to maintain performance of large language models (LLMs) at ultra-low bit-widths (≤3 bits), leading to significant accuracy loss. The challenge is to reduce memory and computational demands while preserving model performance.	1. BiLLM Framework: A new 1-bit post-training quantization technique for LLMs. 2. Salient Weight Selection: Exploits Hessian metrics for the identification of salient weights. 3. Residual Approximation: Minimizes the quantization error of salient weights. 4. Optimal Splitting Search: Group non-salient weights by distributio	Achieved perplexity score of 8.41 on LLaMA2-70B using only 1.08-bit weights. Outperformed existing state-of-the-art (SOTA) quantization methods by a considerable margin. Demonstrated efficiency in time by binarizing a model with 7 billion parameters in only	Introduces further investigation into LLM compression techniques and encourages the development of more efficient methods in quantization toward robust deployment in edge scenarios and even resource-constrained devices.

		n for the accurate binarization of them. 5. Error Compensation: Block-wise error compensation for further reducing the quantization errors.	0.5 hours on a single GPU. The average bit-width for all models was between 1.07 and 1.11 bits.	
OneBit: Towards Extremely Low-bitLarge Language Models [6]	Present quantization techniques for large language models (LLMs) suffer a significant performance drop when the bit-width is reduced to very low levels, like 1-bit. Current methods mainly focus on 4-bit or 8-bit quantization, thus limiting their deployability on resource-constrained devices.	1.OneBit Framework: Introduces a novel 1-bit parameter representation method and an effective parameter initialization method based on matrix decomposition. 2. Sign-Value-Independent Decomposition (SVID): Decomposes high-bit matrices into low-bit ones to improve initialization and convergence 3. Quantization-Aware Knowledge Distillation (KD): Transfers knowledge from the original model to the quantized model.	Reaches a minimum of 81% of the non-quantized efficacy on LLaMA models by employing 1-bit weight matrices. The perplexity results of WikiText2 and C4 datasets show significant improvement compared to other quantization methods (for example, OneBit shows a perplexity of 9.18 on LLaMA-13B, while FP16 is at 5.09). Zero-shot accuracy is closer to the FP16 results than with other methods.	Investigate activation quantization, which was not considered in this work. Examine the mathematical underpinnings of optimal parameters for 1-bit quantized models to better enable capability transfer. Continue refining the training procedure to stabilize and enhance performance in quantization-aware training.
IntactKV: Improving Large Language	The performance of the LLM is affected by quantization due	INTACTKV, a lossless KV cache methodology designed specifically to	INT3-group128 quantization results are shown in the	systematic evaluation of quantized LLMs over various tasks

Model Quantization byKeeping Pivot Tokens Intact [7]	to activations outliers, especially at pivot tokens, that cause attention sinks and damage the model's accuracy.	preserve pivot tokens, on top of existing weight-only quantization techniques like RTN, GPTQ, and AWQ.	improvement of perplexity (PPL) on different models, with the best result achieved by AWQ+INTACT KV. For instance, PPL improved from 9.15 to 8.52 for LLaMA-7B.	and probing long contexts effects on performance. More calibration techniques beyond those developed for INTACTKV will also be explored in the future.
SpinQuant: LLM Quantization with Learned Rotations [8]	Large Language Models (LLMs) face challenges with quantization, particularly due to outliers affecting the quantization range, which limits effective bits for representing model weights.	SpinQuant, a novel quantization technique that employs learned rotation matrices to reduce outliers while maintaining precision in model outputs. The method incorporates Cayley SGD for optimizing rotation matrices efficiently. Other approaches mentioned include mixed precision methods, weighting equalization, and vector quantization.	The results show Spi nQuant consider ably improves quantization performance wit h noted improvem ents in accuracy acro ss configuration s. As an exampl e, SpinQuant revea ls improved dec oding speeds with 4-bit quantization: ~3 × the speed com pared to 16-bit models. Results compris e accuracy percentages, suc h as 78.4% for some config urations.	Future research directions are in optimizing rotations further and finding theor etical aspects of rotation matrices in quantization performance. Th us, the potential for enhancing fur ther and explorin g new techniques in LLM quantization is suggested.
TernaryLLM: Ternarized Large	High computational co st and memory requirem	Dual Learnable Ternarization (DLT): Enables learnable scales and shifts for	TernaryLLM ac hieves 5.8 perpl exity improvem ent on C4 and	Development of customized hardware for ternarized LLMs

Language Model [9]	ent of LLMs. Asymmetric outliers and non-zero means in weights during quantization. Pretrained LLMs experience extreme low-bit quantization leading to severe information loss.	weights. Outlier-Friendly Feature Knowledge Distillation (OFF): Recovers lost information during quantization. DLT addresses asymmetric weight distributions by allowing learnable parameters for scaling and shifting. OFF maximizes mutual information between floating-point and quantized models using cosine similarity.	8.2% average accuracy improvement on zero-shot tasks for LLaMA-3. Significant perplexity improvements (e.g., 1.49 PPL on OPT-1.3B). Improved performance by 0.65 PPL on WikiText2 and 0.57 PPL on C4 with OFF.	to improve inference efficiency. Further research into quantization-aware training methods to improve model performance. Investigate further knowledge distillation techniques to improve the robustness of the models.
The Era of 1-bit LLMs: All Large Language Models are in 1.58 Bits [10]	Large Language Models (LLMs) demand significant resources, including high energy consumption, memory usage, and computational power, particularly during deployment.	The introduction of BitNet b1.58 marks a 1-bit LLM variant where parameters are represented as ternary values (-1, 0, 1). This model was trained from the ground up using quantization techniques aimed at optimizing memory, energy, and computation. It employs a Transformer architecture featuring BitLinear layers.	BitNet b1.58 matches the performance of full-precision LLaMA LLMs in terms of perplexity and task accuracy, all while operating at a fraction of the cost. It has achieved energy savings of 71.4 times and speed improvements of up to 4.1 times on larger models, it reduces memory usage by as much as 7.16 times compared to FP16 LLMs, offering superior throughput that	The application of 1.58-bit LLMs extends to Mixture-of-Experts (MoE) models, helping to minimize memory usage and inter-chip communication. It also provides native support for long sequences by lowering the memory requirements for key-value caches. This technology is being deployed on edge and mobile devices to enhance applications that operate with limited resources.

			supports up to 11 times the batch size on GPUs for 70B models.	

Table 1. shows the problems faced, technologies or methodologies used, results obtained and the future work of the referenced papers.

3. Literature review

A new framework for 2-bit quantization of large language models seeks to address the challenge in maintaining the performance of such systems with further reduction in computational costs. Weights and Hessian matrices are optimized by an incoherence process via orthogonal rotations. Adaptive rounding, along with Hessian-based loss approximations, further reduces rounding errors. As put forth by Chee et al., QuIP significantly reduces the perplexity of models such as OPT and LLaMA and thus is the most current state-of-the-art framework with strong theoretical justifications.
(Wang et al.) introduce flexible low-bit configurations into the expanding of the BitNet architecture for scalability and seek to balance computational efficiency and precision. New scaling laws in 1-bit and sub-2-bit models enable BitNet to provide super performance for large-scale LLMs with ultra-low-bit quantization and achieve superior generalization on a variety of tasks. An atomic decomposition framework to quantize LLM breaks models down into smaller, more modular units for better compression. This approach, driven by, improves interpretability and low inference overhead. The performance for the Atom framework is competitive with full-precision models and attains much lower latency and memory consumption. Maintaining key-value integrity in transformer models is crucial during quantization and is the key factor for long-sequence modeling. Liu et al. propose a hierarchical quantization strategy that prevents degradation, resulting in high retrieval accuracy. This approach confers advantage on downstream tasks such as text summarization and question-answering, implemented in the IntactKV framework.
Among the challenges of activation quantization is how to balance the ranges of quantization to reduce information loss. Liu et al. introduce Hadamard-based rotations to stabilize the distributions of activations and further improve computational efficiency. Their SpinQuant framework achieves a 3x speedup compared to FP16 models with full accuracy preservation, hence very suitable for latency-critical applications. The arbitrary-bit quantization-LLM is dedicated to the optimization of performance-cost tradeoffs in LLMs. The approach proposed by Zeng et al. makes use of fine-grained bit allocation across model layers, achieving much better throughput and accuracy than uniform quantization. Thus, ABQ-LLM is highly suitable for large-scale deployment.
Exploring 3-state quantization (-1, 0, +1), (Chen et al.) combine learnable thresholds with knowledge distillation to address quantization errors effectively. Their approach delivers competitive accuracy with a smaller model footprint, making it practical for resource-constrained devices. Extreme compression in large language models is addressed here using a hybrid approach: mixing partially binarized weights with full-precision counterparts. Ma et al. use Hessian-guided quantization-aware training to preserve critical reasoning while significantly reducing memory consumption; their method matches full-precision model performance while being more efficient. (Xu et al.) discuss how to reduce memory usage in LLMs using 1-bit quantization. They combine an error correction mechanism with efficient gradient descent in order to mitigate performance degradation. Their study shows that the inference efficiency of 1-bit models is comparable to their higher-bit counterparts, while their accuracy remains strong on benchmarks. Intra-

module low-rank pruning with the help of LLMs and the introduction of activation transitions is introduced by Shen et al. The method of pruning combined with quantization-aware fine-tuning achieves very impressive memory savings with negligible performance degradation on zero-shot and few-shot tasks.

4. Methodologies

The BitNet b1.58 model is a novel approach to low-bit quantization for large language models. The methodologies in this paper overcome the computational and memory inefficiencies of LLMs while retaining high accuracy. Below is a detailed breakdown of the methods, formulas, and related techniques used:

- **1.58-BitQuantizationFramework**

The novelty of P15 is the concept of 1.58-bit quantization, which extends the standard 1-bit approach by using a ternary weight system. The weights are represented by three states: {-1, 0, +1} instead of the binary states: {-1, +1}. This offers an additional "zero" state that allows sparse representations and better generalization. Weight Representation: $\mathbf{W_{quantized}=argmin}_{\ q\in\{-1,0,+1\}}\|\mathbf{W-q}\|^2$ where W is the original weight matrix, and q represents the quantized value.
Error Compensation: Following quantization, residual errors are compensated iteratively in training or fine-tuning to minimize performance degradation.

- **Optimized Matrix Multiplications Using Integer Arithmetic**

Matrix multiplication is the most computationally intensive operation in LLMs. By reducing the weights to ternary values, the computation is simplified to integer additions and subtractions rather than full floating-point operations.
Efficient Multiplication: For ternary weights W, matrix-vector products are computed as:

$$Y = \sum_{i=1}^{n} (W_i \cdot X_i)$$

where, Wi∈{−1,0,+1}and X is the input vector. Since Wi is ternary, the operation simplifies to additions for +1, subtractions for −1, and no operation for 0.
Sparse Multiplications: The zero states in W reduce the effective number of computations, enabling sparsity optimizations in hardware.

- **Scaling Laws for 1-Bit and Sub-2-Bit Quantization**

The paper proposes a set of scaling laws to determine the trade-offs between model size, bit precision, and computational efficiency. These scaling laws generalize to ultra-low-bit LLMs and provide guidelines for designing low-bit architectures that balance accuracy and resource usage.
Scaling Formula:

$$\text{Error} \sim \frac{C}{\text{Model Size}} + \frac{D}{\text{Quantization Precision}}$$

where C and D are task-dependent constants. This highlights that increasing model size or precision can compensate for quantization-induced errors.
Application in Model Design: Small models (<3B<3B<3B parameters): Use higher precision (e.g., 4-bit or FP16). Large models (>3B>3B>3B parameters): Leverage 1.58-bit quantization for better cost-performance trade-offs.

- **Activation Compression and Error Mitigation**

The paper complements weight quantization with activation compression. Compressed activations are crucial for handling long sequences without performance degradation.

Activation Quantization: Activations are quantized to low-bit representations with minimal information loss:

$$\mathbf{A_{quantized} = round(A \cdot S)}$$

where A is the activation matrix, and S is a scaling factor learned during training.

Activation Error Correction: Quantization errors in activations are mitigated using a residual correction mechanism:

$$\mathbf{A_{corrected} = A_{quantized} + \Delta A}$$

where ΔA is the learned correction term.

- **Hardware Optimization for Deployment**

The paper emphasizes the importance of hardware-aware design. The ternary weight system reduces memory requirements by 3.55x compared to FP16. It also accelerates inference by 2.71x, making the model suitable for edge and mobile devices.

Memory Footprint Reduction: For a model with NNN weights, the memory required is:

$$\mathbf{Memory = N \cdot log_2(3)}$$

compared to N·16 bits for FP16 weights. Energy Efficiency: Integer-based arithmetic operations consume significantly less energy than floating-point operations, making the model more energy-efficient.

5. Related Methods and Techniques

- **Post-Training Quantization (PTQ):**
  The methods in this paper employ techniques of PTQ, which consist of training the model fully in precision and then quantize it to ternary weights.
- **Error Correction Mechanisms:**
  Drawn from GPTQ, among others, the methodology corrects errors that accumulate due to quantization but does not degrade performance.
- **Sparse Representation Optimization:**
  Pruning and sparse matrix representation techniques take advantage of zero states to exploit the ternary weights.
- **Knowledge Distillation:**
  Optionally, knowledge distillation is utilized during training to bridge the performance gap between the quantized and the corresponding full-precision models.

6. Advantages over other quantization techniques

Technique	Advantages Over Others
Ternary Quantization	- It introduces a"0"state to increase sparsity by reducing useless computation. -Offers a better trade-off between compression and accuracy compared to binary (-1, +1) quantization. -Ternary is computationally lighter compared to 4-bit methods that require much less memory and faster arithmetic operations. -Offers model generalization, especially in big tasks.
Matrix Optimization	This method applies sparse arithmetic. Zero states in the ternary quantization are utilized to avoid performing certain operations and thereby reducing computation overhead. - It reduces latency more than mixed-

	precision approaches such as SmoothQuant, which necessitates extra activation rescaling. - It also improves inference speed without needing custom kernel designs like in LUT-GEMM.
Scaling Laws	-It provides a predictive framework for model design, allowing developers to systematically balance size, precision, and task accuracy. - It addresses the lack of generalized scaling principles for ad hoc methods such as GPTQ, making it applicable across diverse models and use cases. - Ensures robust performance even for ultra-low-bit quantized models.
Activation Compression	-Ensures stable performance for long-sequence tasks by compressing activations without losing important information. - Outperforms the uniform quantization methods like ABQ-LLM, where accuracy degrades with a growing model size. - Uses error reduction mechanisms via residual correction that keep coherence in sequential inputs.
Hardware Optimizations	-Specifically, designed to be work efficiently along with integer arithmetic hardware while keeping compatibility with modern TPUs and GPUs. - It uses much less memory (e.g., 3.55x less than FP16 and 2x less than 4-bit quantization). - Uses much less energy due to reduced floating-point operations. - Is scalable for mobile and edge device deployments.

Table 2 gives the advantages of BitNet b1.58 quantization methods over other quantization methods.

7. Conclusion

Recent breakthroughs in low-bit quantization for LLMs have overcome some important challenges: computational inefficiencies and memory constraints. Innovative techniques include QuIP's 2-bit incoherence processing and SpinQuant's use of Hadamard rotations, among others, to enable the deployment of LLMs in resource-limited environments. Researchers explore methods like ternary quantization, adaptive rounding, and fine-grained bit allocation in order to balance efficiency with accuracy. One such breakthrough is the use of absolute mean values in weight representation, in what is called the AbsMean Quantization Method. Smoothness and efficiency in quantizing give way to better model throughputs and perplexities across the tasks. BitNet b1.58 further gives this innovation an upgrade through the use of bit-linear quantization, which has more uniform scaling patterns compared to previously applied nonlinear methods. These enable crucial advantages: achieving full-precision-like perplexity with the ternary quantization and AbsMean techniques, for instance. Optimized computation and sparse arithmetic bring immense speedup in inference while reducing GPU usage and, hence, operational costs. This enables the deployment of LLMs even on edge devices or restricted hardware. BitNet b1.58 represents the state of the art for resource efficiency versus high performance. Its hardware accelerator compatibility with reduced computational demands sets the standard for scalable LLMs. It covers both theoretical and practical challenges in the development process, therefore opening further developments toward efficient and accessible AI technologies.

8. References

1. Chee, Jerry, Yaohui Cai, Volodymyr Kuleshov, and Christopher De Sa. 2024. QuIP: 2-Bit Quantization of Large Language Models With Guarantees. arXiv. https://doi.org/10.48550/arXiv.2307.13304.
2. Shang, Yuzhang, Zhihang Yuan, Qiang Wu, and Zhen Dong. 2023. PB-LLM: Partially Binarized Large Language Models. arXiv. https://doi.org/10.48550/arXiv.2310.00034.
3. Wang, Hongyu, Shuming Ma, Li Dong, Shaohan Huang, Huaijie Wang, Lingxiao Ma, Fan Yang, Ruiping Wang, Yi Wu, and Furu Wei. 2023. BitNet: Scaling 1-bit Transformers for Large Language Models. arXiv. https://doi.org/10.48550/arXiv.2310.11453.
4. Zhao, Yilong, 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. arXiv. https://doi.org/10.48550/arXiv.2310.19102.
5. Huang, Wei, Yangdong Liu, Haotong Qin, Ying Li, Shiming Zhang, Xianglong Liu, Michele Magno, and Xiaojuan Qi. 2024. BiLLM: Pushing the Limit of Post-Training Quantization for LLMs. arXiv. https://doi.org/10.48550/arXiv.2402.04291.
6. Xu, Yuzhuang, Xu Han, Zonghan Yang, Shuo Wang, Qingfu Zhu, Zhiyuan Liu, Weidong Liu, and Wanxiang Che. 2024. OneBit: Towards Extremely Low-bit Large Language Models. arXiv. https://doi.org/10.48550/arXiv.2402.11295.
7. Liu, Ruikang, Haoli Bai, Haokun Lin, Yuening Li, Han Gao, Zhengzhuo Xu, Lu Hou, Jun Yao, and Chun Yuan. 2024. IntactKV: Improving Large Language Model Quantization by Keeping Pivot Tokens Intact. arXiv. https://doi.org/10.48550/arXiv.2403.01241.
8. Liu, Zechun, Changsheng Zhao, Igor Fedorov, Bilge Soran, Dhruv Choudhary, Raghuraman Krishnamoorthi, Vikas Chandra, Yuandong Tian, and Tijmen Blankevoort. 2024. SpinQuant: LLM quantization with learned rotations. arXiv. https://doi.org/10.48550/arXiv.2405.16406.
9. Chen, Tianqi, Zhe Li, Weixiang Xu, Zeyu Zhu, Dong Li, Lu Tian, Emad Barsoum, Peisong Wang, and Jian Cheng. 2024. TernaryLLM: Ternarized Large Language Model. arXiv. https://doi.org/10.48550/arXiv.2406.07177.
10. Ma, Shuming, 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. https://doi.org/10.48550/arXiv.2402.17764.
11. Zeng, Chao, Songwei Liu, Yusheng Xie, Hong Liu, Xiaojian Wang, Miao Wei, Shu Yang, Fangmin Chen, and Xing Mei. 2024. ABQ-LLM: Arbitrary-Bit Quantized Inference Acceleration for Large Language Models. arXiv. https://doi.org/10.48550/arXiv.2408.08554.
12. Shen, Bowen, Zheng Lin, Daren Zha, Wei Liu, Jian Luan, Bin Wang, and Weiping Wang. 2024. Pruning Large Language Models to Intra-module Low-rank Architecture with Transitional Activations. arXiv. https://doi.org/10.48550/arXiv.2407.05690.