TriLM vs FloatLM: Ternary LLMs are more Performant than Quantized FP16 LLMs

Ayush Kaushal [*1] **Tejas Vaidhya** [*1 2] **Tejas Pandey** [*3] **Aaryan Bhagat** [4] **Irina Rish** [1 2]

Abstract

Ternary LLMs offer significantly better performance for their size (measured in bits) than the models trained and deployed in FP16/BF16. Given the widespread usage of quantization before deployment and advancements in Post Training Quantization of LLMs, a pivotal question arises: do ternary LLMs indeed provide any discernible benefits? To address this, we first build an open family of pre-trained ternary Large Language Models (TriLM). Additionally, we include their counterparts pre-trained in FP16 (FloatLM) and quantized versions of FloatLM (QuantLM) with parameters across almost two orders of magnitude - from 99M to 3.9B parameters. We demonstrate that TriLMs with 3B+ parameters start to offer competitive performance compared to FloatLMs with the same parameter count, while providing significantly better performance for their size. TriLMs also outperform quantized models, with TriLM 3.9B surpassing the larger QuantLM-3bit 3.9B. Furthermore, across knowledge-based benchmarks, TriLM maintains a superiority for its size. To advance research on Ternary LMs, we open source over 500+ checkpoints across the model families at https://github.com/NolanoOrg/SpectraSuite.

1. Introduction

LLMs are notoriously expensive and hard to deploy due to their huge sizes exceeding the memory capacity of hardware accelerators like GPUs, with the largest open source models like Grok-1 314B [1] and LLaMa 3 70B (AI@Meta, 2024) exceeding the DRAM size of data center GPUs (A100s, H100s) and consumer-grade GPUs (RTX 4090s) respectively, even after 4-bit quantization. Moreover, the autoregressive generation phase of LLM inference is usually memory bound (Kim et al., 2024), making the number of tokens generated per second directly proportional to the bits (i.e. model size) that need to be transferred across the memory hierarchy. A reduction in model size (bits) can directly improve generation speed, making deployment easier and more economically feasible.

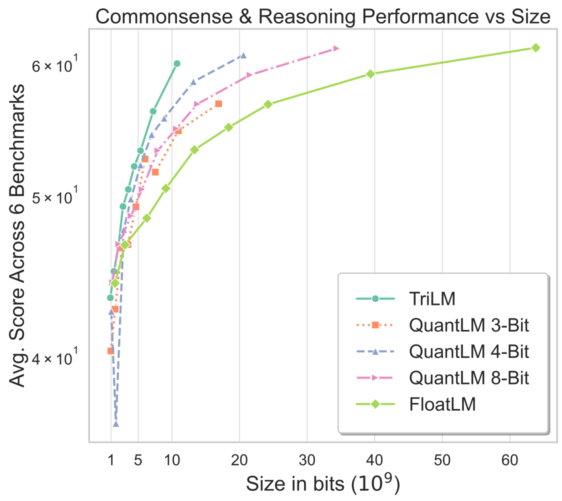

Figure 1. Commonsense & Reasoning score avg. across 6 benchmarks for ternary TriLM, FP16 FloatLM and quantized QuantLM models. At 3B+ scales, TriLMs offers significantly better performance for its size (number of bits) than FloatLM and QuantLM.

Recently, Ma et al. (2024) proposed BitNet b1.58 LLM which represents each parameter in its weight matrices in one of three ternary states -1, 0, 1 along with a single additional floating point scale for the entire matrix. At 3B+ parameter scale, the performance gap between BitNets and FP16 LLMs of similar parameter count narrows. BitNets, with 1.58 bits per parameter can be upto 10x smaller than FP16 LLMs of similar parameter count. However, with the recent progress in Post Training Quantization (Dettmers et al., 2022; Lin et al., 2024; Egiazarian et al., 2024) and fast inference kernels for quantized LLMs (Frantar and Alistarh, 2024), deploying quantized FP16 LLMs have become a prevalent practice. It is unclear how well ternary LLMs fare

[*]Equal contribution [1]Université de Montréal [2]Mila – Québec AI Institute [3]Indian Institute of Technology Kharagpur [4]University of California, Riverside. Correspondence to: Tejas Vaidhya <iamtejasvaidhya@gmail.com>.

Published at ICML 2024 Workshop on Foundation Models in the Wild. Copyright 2024 by the author(s).

[1]https://github.com/xai-org/grok-1

against their quantized FP16 counterparts. We study this problem and observe that Ternary LLMs still offer better performance for their size than quantized models.

We first build TriLM, a set of Ternary Language Models based on a modified version of BitNet's architecture. These models range from 99M parameters to 3.9B parameters, each trained for 300B tokens. We complement these models by training an FP16 LLM with the same parameter counts for roofline reference and comparison across the two families of models. For each FloatLM, we build a corresponding quantized LLM (QuantLM) using GPTQ (Frantar et al., 2022), one of the most popular and state-of-the-art methods, across bit-widths of 3, 4, 6, and 8. We leverage these models for our analysis and also open source the model weights to further the research. We establish its peculiar training dynamics, characterized by a sudden drop at the halfway point due to its learning schedule. To further research on better optimization techniques for TriLMs and understanding its training dynamics, we also release 500+ intermediate checkpoints across the two families of models.

We rigorously evaluate TriLM, FloatLM and QuantLM models over 10 different benchmarks spanning from commonsense and reasoning to the model's knowledge capacity. TriLMs offer the best performance across reasoning and commonsense tasks for their size, with the largest model in the family, TriLM 3.9B, surpassing even FloatLM 2.4B and remaining competitive with FloatLM 3.9B, despite having fewer bits than FloatLM 830M. Across tasks requiring knowledge capacity, TriLMs still outperform QuantLMs of comparable sizes, with TriLM 3.9B performing similarly to a larger QuantLM 4-Bit 2.4B. However, when TriLMs lag behind in parameter count, TriLM 3.9B offers 10-shot Exact Match performance on TriviaQA only halfway between FloatLM 1.5B and FloatLM 2.4B.

In this paper, we build an open suite of Ternary Language Models - TriLM, across 9 sizes ranging from 99M to 3.9B parameters along with their FloatLM FP16 counterparts and QuantLM quantized models. We make over 500 checkpoints of the model weights publicly available to advance research. These are also the first open family of LLMs optimized from scratch with effective precision of weights below FP16. Additionally, We establish the training schedule and corresponding loss curves for TriLMs, demonstrating that they significantly outperform their same-sized (measured in bits) FloatLM counterparts on validation loss. Furthermore, we evaluate the families of models across commonsense and reasoning benchmarks and establish that at 3B+ parameter scale, TriLMs offers similar performance to QuantLM 4-bit of same parameter count despite being at half size. Finally, we establish that TriLMs still offer better performance for its size on Knowledge based tasks than QuantLM (or FloatLM), although the gains are much lesser as TriLMs do not perform close to QuantLMs of same parameter count.

2. TriLM, FloatLM and QuantLM

In this section, we outline the preparation of three families of LLMs for our analysis. We first discuss the architectural choices and training setup, which are closely based on those of Pythia (Biderman et al., 2023) and BitNet's b1.58 (Ma et al., 2024), to ensure a fair comparison and stable training. This is followed by a detailed explanation of the optimization schedule and training setup for TriLMs. Finally, we discuss quantization of FloatLM to create QuantLM.

Architecture Both TriLM and FloatLM are LLaMa-style (Touvron et al., 2023a) autoregressive transformers (Vaswani et al., 2017) model with RMSNorm (Zhang and Sennrich, 2019) instead of LayerNorm (Ba et al., 2016), SwiGLU Gated MLP (Shazeer, 2020) instead of standard transformer MLP, Rotary Position Embedding (RoPE) (Su et al., 2021), Multi-Headed Attention and no bias terms. We show these two architectures in Figure 2.

In FloatLM, the parameters in weight matrix of linear layers are represented as floating point (FP16/BF16) numbers. Whereas, in TriLM, these parameters are represented in one of three possible ternary states $\{-1, 0, 1\}$ along with a single additional floating point number - 'scale' for the entire matrix. For better performance and stable training in FP16, we follow GPT3's Pre-Normalization (Brown et al., 2020) rather than BitNet's architecture that normalizes before each linear layer. Thus, normalization is done twice in each transformer layer, at the inputs of the two sub-layers - attention and Gated MLP. The embedding, language model head and activations are not quantized.

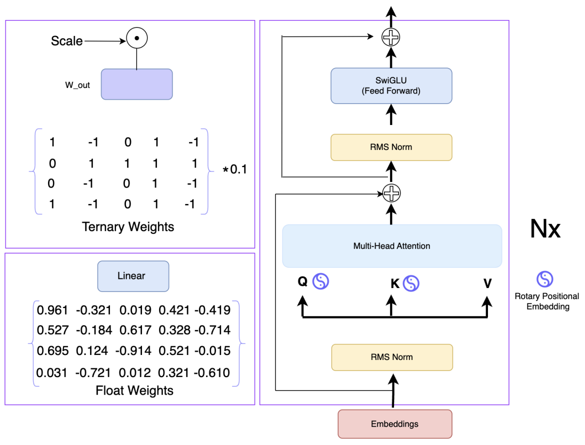

Figure 2. Architecture of TriLM and FloatLM are both based on the standard LLaMa architecture. All the linear layers in FloatLM are represented by an FP16 matrix, whereas in TriLM these are represented via a ternary matrix and a floating point scale.

During training of TriLM, the *latent* weights are maintained in floating point precision to allow accumulation of small updates over multiple iterations. During forward pass, floating point latent weight weights are ternarized on the fly by computing the absolute mean of the latent weights as the scale and rounding off to the nearest ternary state. During the backward pass, a straight through estimator (Bengio et al., 2013) is used to estimate backward pass on floating point latent weights. We expand on formal description of these forward pass, backward pass and inference time equations in the Appendix (§A.1)

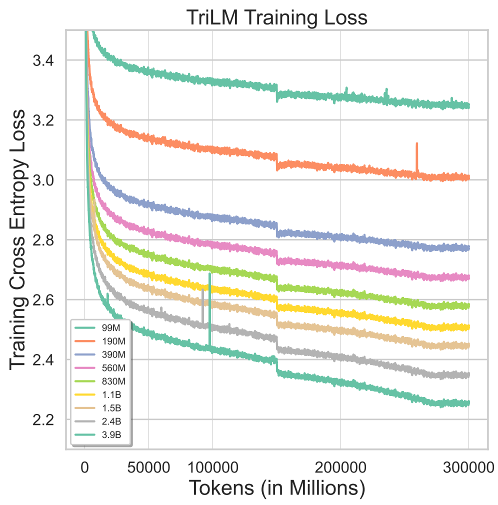

Figure 3. Training Cross Entropy Loss across steps for the TriLM family of models. At halfway point (150B tokens) when we lower the peak learning rate and weight decay, we observe a sudden drop in training loss.

Training Our optimization schedule for TriLM closely follows the recommended settings from BitNet (Ma et al., 2024). It is trained with a two staged linear decay scheduling, where, at the halfway point, the weight decay is reduced from 0.1 to 10^{-5} and the peak learning rate is also dropped. The lower weight decay settings is commonly adopted in quantization aware training (Liu et al., 2023a; Yuan et al., 2024) and fully quantized training (Bethge et al., 2018; Le and Li, 2023) as quantizing to low bit-width already significant regularization. We train FloatLM with cosine decay scheduling and 0.1 weight decay, similar to popular LLMs like Pythia, LLaMa (Touvron et al., 2023a;b) and LLM-360 (Liu et al., 2023b). We list architectural hyperparameters and learning rates across families of models in the Appendix (§A.4).

Figure 3 shows the Training loss curves for all the TriLMs trained. We observe similar scaling as FloatLM models. However, the training loss in TriLMs, has a sudden drop at halfway point when the weight decay and peak learning rate is lowered, beyond which the loss reduces linearly with iterations, until the final few iterations, when the model converges.

Quantization We create QuantLM across 3, 4, 6 and 8 bits by applying the GPTQ (Frantar et al., 2022) Post Training Quantization on FloatLM across all the weights in the transformer layers. We set group size to 128 for the 3 and 4 bits, leading to 3.25 and 4.25 effective bit per parameter for these quantization levels. Similar to TriLM, we do not quantize the embedding, language model head and the activations. We use symmetric quantization for a fairer comparison.

3. Evaluation

We evaluate the families of LLMs on three aspects - commonsense & reasoning tasks, knowledge based tasks and validation cross entropy loss, all of which are crucial measures of their downstream performance.

Commonsense and Reasoning We consider 9 different commonsense and reasoning benchmarks consisting of tasks from logical reasoning and science question to grounded and physical commonsense tasks: Arc Easy and Challenge (Clark et al., 2018), BoolQ (Clark et al., 2019), HellaSWAG (Zellers et al., 2019), WinoGrande (Sakaguchi et al., 2021), PIQA (Bisk et al., 2019), LAMBADA (Paperno et al., 2016), SciQ (Welbl et al., 2017), LogiQA (Liu et al., 2021). Figure 1 shows the average performance of the LLMs on first 6 of these benchmarks (same as those reported in BitNet). When comparing among QuantLMs and FloatLMs, we observe that 4-bit is the optimal size among within QuantLMs echoing prior observation (Dettmers and Zettlemoyer, 2023). However, TriLMs offers much better performance for its size than even 4-bit QuantLM at billion+ parameter scale. In fact, at 3.9B parameter scale the performance gap between TriLM and FloatLM is significantly diminished, despite former having only 1.58 bit per parameter instead of 16 bits. The trends remain consistent across the remaining three benchmarks. Readers may refer to Tables 4 and 5 for detailed benchmarking across all datasets.

Knowledge Several downstream practical uses of LLMs requires these models to have knowledge about common subjects and topics. We evaluate the performance of the LLMs on the TriviaQA benchmark across 0-shot, 1-shot, 3-shot, 5-shot and 10-shot settings. Figure 4 shows the Exact Match performance of the model families in a 10-shot settings. Compared to commonsense & reasoning benchmarks, models represented with less number of bits (like TriLM and QuantLM 4-bit) show lower performance than FP16 models of same parameter count. However, the performance degradation observed in 4-bit quantization are less that the 2x reduction factor from previous works (Allen-Zhu and Li, 2024) in toy settings. From the observed scaling laws, TriLMs still maintains a performance superiority for its size as TriLM 3.9B reaches performance close to a larger QuantLM 4-bit 2.4B. However, when measuring relative

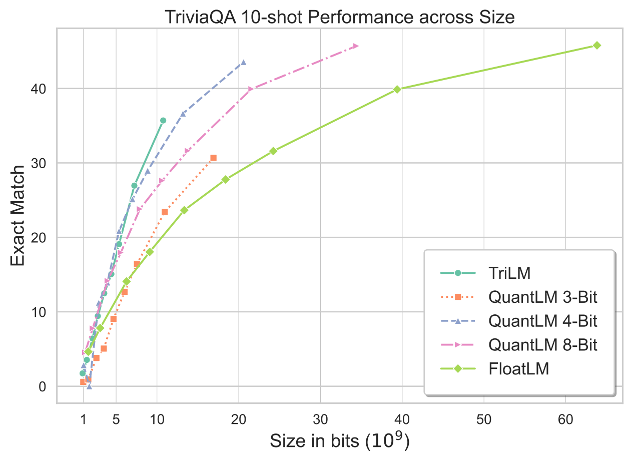

Figure 4. TriviaQA 10-shot performance. From scaling laws, we observe little gains from TriLM over similar sized 4-bit QuantLMs.

performance for its parameter count, it only manages to rank halfway between FloatLM 1.5B and 2.4B. This shows that low-bitwidth LLMs have lesser knowledge capacity.

Our observations remain consistent across few shot settings. At 0-shot settings, TriLMs performs much worse, with TriLM 3.9B being outperformed by a model half its size - QuantLM 4-bit 830M. Table 6 shows these results.

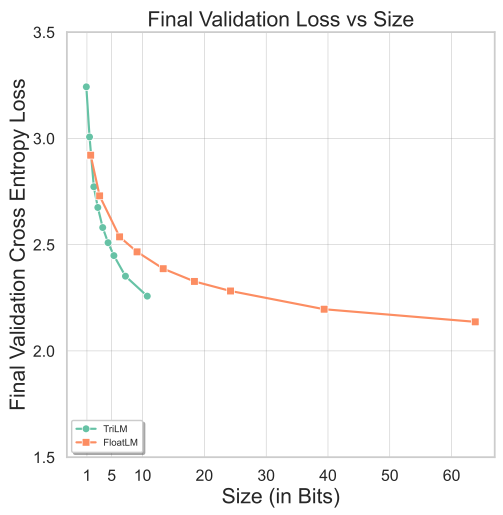

Figure 5. Final validation cross entropy loss for TriLM and FloatLM across size (in bits). TriLM offers significantly lower validation loss than similarly size FloatLMs.

Validation Loss Though the task of next word prediction is noisy, cross entropy loss is an good proxy for intelligence as it measures compression which correlate with intelligence (Huang et al., 2024). Figure 5 shows the final validation cross entropy loss across TriLM and FloatLM family of models. Here too, we observe similar characteristics as knowledge based tasks - where TriLM offers best performance for its size, but lags behind FloatLMs of same parameter count. Specifically TriLM 3.9B has better loss than FloatlM 1.5B with double its size, but slightly falls behind than FloatLM 2.4B.

4. Related Work

The increasing size of Large Language Models (LLMs) poses deployment challenges and raises environmental concerns. One way to tackle this is through post-training quantization. These techniques are discussed in Xiao et al. (2024); Chee et al. (2024); **?**. They offer a solution by creating low-bit models for inference, reducing both memory and computational requirements. This led to the adoption of 4-bit variants over previously favored 16-bit models. However, recent research on 1-bit and 1.5-bit (ternary) model architectures, exemplified by Wang et al. (2023); Ma et al. (2024), presents a promising avenue for cost reduction in LLMs while maintaining effectiveness.

In parallel, initiatives such as Pythia (Biderman et al., 2023) and LLM360 (Liu et al., 2023b) have emerged within the LLM development community, advocating for transparency and collaborative research. Pythia provides a suite of meticulously trained LLMs with a wide range of parameters, offering access to numerous checkpoints and data reconstruction tools for comprehensive exploration of LLM dynamics. On the other hand, LLM360 pushes AI research democratization by openly sharing intermediate checkpoints, training data, metrics, and source code.

5. Conclusion

We introduce TriLM, an open family of ternary LLMs along with corresponding FP16 LLMs - FloatLM and quantized QuantLM models, all pretrained for 300B tokens. We demonstrate that, at 3.9B parameter scale, TriLMs offers competitive performance across commonsense and reasoning tasks to QuantLM and FloatLM of same parameter count, despite have less size (bits) than FloatLM 830M model. We establish that TriLMs maintain superiority in knowledge-capacity, even though the relative gains in performance to same sized QuantLMs reduces. We open source 500+ checkpoints across these models, to enable research on understanding and improving ternary LLMs.

Acknowledgement

We acknowledge the support from the Mozilla Responsible AI Grant, the Canada CIFAR AI Chair Program and the Canada Excellence Research Chairs Program. This research was enabled by the computational resources provided by the Summit supercomputer, awarded through the Frontier DD allocation and INCITE 2023 program for the project "Scalable Foundation Models for Transferable Generalist AI" and SummitPlus allocation in 2024. These resources were supplied by the Oak Ridge Leadership Computing Facility at the Oak Ridge National Laboratory, with support from the Office of Science of the U.S. Department of Energy. We extend special thanks to Jens Glaser for his assistance with the Summit and Frontier supercomputers. We would also like to thank Ishita Hirmath for helping with proofreading.

References

AI@Meta. Llama 3 model card. 2024. URL https://github.com/meta-llama/llama3/blob/main/MODEL_CARD.md.

Z. Allen-Zhu and Y. Li. Physics of language models: Part 3.3, knowledge capacity scaling laws, 2024.

A. Andonian, Q. Anthony, S. Biderman, S. Black, P. Gali, L. Gao, E. Hallahan, J. Levy-Kramer, C. Leahy, L. Nestler, K. Parker, M. Pieler, J. Phang, S. Purohit, H. Schoelkopf, D. Stander, T. Songz, C. Tigges, B. Thérien, P. Wang, and S. Weinbach. GPT-NeoX: Large Scale Autoregressive Language Modeling in PyTorch, 9 2023. URL https://www.github.com/eleutherai/gpt-neox.

J. L. Ba, J. R. Kiros, and G. E. Hinton. Layer normalization, 2016.

Y. Bengio, N. Léonard, and A. Courville. Estimating or propagating gradients through stochastic neurons for conditional computation, 2013.

J. Bethge, M. Bornstein, A. Loy, H. Yang, and C. Meinel. Training competitive binary neural networks from scratch. *ArXiv*, abs/1812.01965, 2018. URL https://api.semanticscholar.org/CorpusID:54458838.

S. Biderman, H. Schoelkopf, Q. Anthony, H. Bradley, K. O'Brien, E. Hallahan, M. A. Khan, S. Purohit, U. S. Prashanth, E. Raff, A. Skowron, L. Sutawika, and O. van der Wal. Pythia: A suite for analyzing large language models across training and scaling, 2023.

L. Biewald. Experiment tracking with weights and biases, 2020. URL https://www.wandb.com/. Software available from wandb.com.

Y. Bisk, R. Zellers, R. L. Bras, J. Gao, and Y. Choi. Piqa: Reasoning about physical commonsense in natural language. In *AAAI Conference on Artificial Intelligence*, 2019. URL https://api.semanticscholar.org/CorpusID:208290939.

S. Black, S. Biderman, E. Hallahan, Q. Anthony, L. Gao, L. Golding, H. He, C. Leahy, K. McDonell, J. Phang, M. Pieler, U. S. Prashanth, S. Purohit, L. Reynolds, J. Tow, B. Wang, and S. Weinbach. Gpt-neox-20b: An open-source autoregressive language model, 2022.

T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. M. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, and D. Amodei. Language models are few-shot learners, 2020.

J. Chee, Y. Cai, V. Kuleshov, and C. D. Sa. Quip: 2-bit quantization of large language models with guarantees, 2024.

C. Clark, K. Lee, M.-W. Chang, T. Kwiatkowski, M. Collins, and K. Toutanova. BoolQ: Exploring the surprising difficulty of natural yes/no questions. In J. Burstein, C. Doran, and T. Solorio, editors, *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2924–2936, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1300. URL https://aclanthology.org/N19-1300.

P. Clark, I. Cowhey, O. Etzioni, T. Khot, A. Sabharwal, C. Schoenick, and O. Tafjord. Think you have solved question answering? try arc, the ai2 reasoning challenge, 2018. URL https://api.semanticscholar.org/CorpusID:3922816.

T. Dettmers and L. Zettlemoyer. The case for 4-bit precision: k-bit inference scaling laws, 2023.

T. Dettmers, M. Lewis, Y. Belkada, and L. Zettlemoyer. Llm.int8(): 8-bit matrix multiplication for transformers at scale, 2022.

V. Egiazarian, A. Panferov, D. Kuznedelev, E. Frantar, A. Babenko, and D. Alistarh. Extreme compression of large language models via additive quantization, 2024.

E. Frantar and D. Alistarh. Marlin: a fast 4-bit inference kernel for medium batchsizes. https://github.com/IST-DASLab/marlin, 2024.

E. Frantar, S. Ashkboos, T. Hoefler, and D. Alistarh. GPTQ: Accurate post-training compression for generative pretrained transformers. *arXiv preprint arXiv:2210.17323*, 2022.

L. Gao, S. Biderman, S. Black, L. Golding, T. Hoppe, C. Foster, J. Phang, H. He, A. Thite, N. Nabeshima, S. Presser, and C. Leahy. The pile: An 800gb dataset of diverse text for language modeling, 2020.

L. Gao, J. Tow, B. Abbasi, S. Biderman, S. Black, A. DiPofi, C. Foster, L. Golding, J. Hsu, A. Le Noac'h, H. Li, K. McDonell, N. Muennighoff, C. Ociepa, J. Phang, L. Reynolds, H. Schoelkopf, A. Skowron, L. Sutawika, E. Tang, A. Thite, B. Wang, K. Wang, and A. Zou. A framework for few-shot language model evaluation, 12 2023. URL https://zenodo.org/records/10256836.

Y. Huang, J. Zhang, Z. Shan, and J. He. Compression represents intelligence linearly, 2024.

M. Joshi, E. Choi, D. Weld, and L. Zettlemoyer. TriviaQA: A large scale distantly supervised challenge dataset for reading comprehension. In R. Barzilay and M.-Y. Kan, editors, *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1601–1611, Vancouver, Canada, July 2017. Association for Computational Linguistics. doi: 10.18653/v1/P17-1147. URL https://aclanthology.org/P17-1147.

S. Kim, C. Hooper, A. Gholami, Z. Dong, X. Li, S. Shen, M. W. Mahoney, and K. Keutzer. Squeezellm: Dense-and-sparse quantization, 2024.

D. P. Kingma and J. Ba. Adam: A method for stochastic optimization, 2017.

P.-H. C. Le and X. Li. Binaryvit: Pushing binary vision transformers towards convolutional models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pages 4664–4673, June 2023.

J. Lin, J. Tang, H. Tang, S. Yang, W.-M. Chen, W.-C. Wang, G. Xiao, X. Dang, C. Gan, and S. Han. Awq: Activation-aware weight quantization for llm compression and acceleration, 2024.

J. Liu, L. Cui, H. Liu, D. Huang, Y. Wang, and Y. Zhang. Logiqa: a challenge dataset for machine reading comprehension with logical reasoning. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence*, IJCAI'20, 2021. ISBN 9780999241165.

Z. Liu, B. Oguz, C. Zhao, E. Chang, P. Stock, Y. Mehdad, Y. Shi, R. Krishnamoorthi, and V. Chandra. Llm-qat: Data-free quantization aware training for large language models. *arXiv preprint arXiv:2305.17888*, 2023a.

Z. Liu, A. Qiao, W. Neiswanger, H. Wang, B. Tan, T. Tao, J. Li, Y. Wang, S. Sun, O. Pangarkar, R. Fan, Y. Gu, V. Miller, Y. Zhuang, G. He, H. Li, F. Koto, L. Tang, N. Ranjan, Z. Shen, X. Ren, R. Iriondo, C. Mu, Z. Hu, M. Schulze, P. Nakov, T. Baldwin, and E. P. Xing. Llm360: Towards fully transparent open-source llms, 2023b.

S. Ma, H. Wang, L. Ma, L. Wang, W. Wang, S. Huang, L. Dong, R. Wang, J. Xue, and F. Wei. The era of 1-bit llms: All large language models are in 1.58 bits, 2024.

D. Paperno, G. Kruszewski, A. Lazaridou, N. Q. Pham, R. Bernardi, S. Pezzelle, M. Baroni, G. Boleda, and R. Fernández. The LAMBADA dataset: Word prediction requiring a broad discourse context. In K. Erk and N. A. Smith, editors, *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1525–1534, Berlin, Germany, Aug. 2016. Association for Computational Linguistics. doi: 10.18653/v1/P16-1144. URL https://aclanthology.org/P16-1144.

S. Rajbhandari, J. Rasley, O. Ruwase, and Y. He. Zero: memory optimizations toward training trillion parameter models. In *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*, SC '20. IEEE Press, 2020. ISBN 9781728199986.

J. Rasley, S. Rajbhandari, O. Ruwase, and Y. He. Deepspeed: System optimizations enable training deep learning models with over 100 billion parameters. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, KDD '20, page 3505–3506, New York, NY, USA, 2020. Association for Computing Machinery. ISBN 9781450379984. doi: 10.1145/3394486.3406703. URL https://doi.org/10.1145/3394486.3406703.

K. Sakaguchi, R. L. Bras, C. Bhagavatula, and Y. Choi. Winogrande: an adversarial winograd schema challenge at scale. *Commun. ACM*, 64(9):99–106, aug 2021. ISSN 0001-0782. doi: 10.1145/3474381. URL https://doi.org/10.1145/3474381.

N. Shazeer. Glu variants improve transformer, 2020.

M. Shoeybi, M. Patwary, R. Puri, P. LeGresley, J. Casper, and B. Catanzaro. Megatron-lm: Training multi-billion parameter language models using model parallelism, 2019.

J. Su, Y. Lu, S. Pan, A. Murtadha, B. Wen, and Y. Liu. Roformer: Enhanced transformer with rotary position embedding, 2021.

H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar, A. Rodriguez, A. Joulin, E. Grave, and G. Lample. Llama: Open and efficient foundation language models, 2023a.

H. Touvron, L. Martin, K. Stone, P. Albert, A. Almahairi, Y. Babaei, N. Bashlykov, S. Batra, P. Bhargava, S. Bhosale, D. Bikel, L. Blecher, C. C. Ferrer, M. Chen, G. Cucurull, D. Esiobu, J. Fernandes, J. Fu, W. Fu, B. Fuller, C. Gao, V. Goswami, N. Goyal, A. Hartshorn, S. Hosseini, R. Hou, H. Inan, M. Kardas, V. Kerkez, M. Khabsa, I. Kloumann, A. Korenev, P. S. Koura, M.-A. Lachaux, T. Lavril, J. Lee, D. Liskovich, Y. Lu, Y. Mao, X. Martinet, T. Mihaylov, P. Mishra, I. Molybog, Y. Nie, A. Poulton, J. Reizenstein, R. Rungta, K. Saladi, A. Schelten, R. Silva, E. M. Smith, R. Subramanian, X. E. Tan, B. Tang, R. Taylor, A. Williams, J. X. Kuan, P. Xu, Z. Yan, I. Zarov, Y. Zhang, A. Fan, M. Kambadur, S. Narang, A. Rodriguez, R. Stojnic, S. Edunov, and T. Scialom. Llama 2: Open foundation and fine-tuned chat models, 2023b.

A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. u. Kaiser, and I. Polosukhin. Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf.

H. Wang, S. Ma, L. Dong, S. Huang, H. Wang, L. Ma, F. Yang, R. Wang, Y. Wu, and F. Wei. Bitnet: Scaling 1-bit transformers for large language models, 2023.

J. Welbl, N. F. Liu, and M. Gardner. Crowdsourcing multiple choice science questions. *ArXiv*, abs/1707.06209, 2017. URL https://api.semanticscholar.org/CorpusID:1553193.

T. Wolf, L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi, P. Cistac, T. Rault, R. Louf, M. Funtowicz, J. Davison, S. Shleifer, P. von Platen, C. Ma, Y. Jernite, J. Plu, C. Xu, T. L. Scao, S. Gugger, M. Drame, Q. Lhoest, and A. M. Rush. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online, Oct. 2020. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/2020.emnlp-demos.6.

G. Xiao, J. Lin, M. Seznec, H. Wu, J. Demouth, and S. Han. Smoothquant: Accurate and efficient post-training quantization for large language models, 2024.

Z. Yuan, Y. Shang, and Z. Dong. PB-LLM: Partially binarized large language models. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=BifeBRhikU.

R. Zellers, A. Holtzman, Y. Bisk, A. Farhadi, and Y. Choi. HellaSwag: Can a machine really finish your sentence? In A. Korhonen, D. Traum, and L. Màrquez, editors, *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4791–4800, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1472. URL https://aclanthology.org/P19-1472.

B. Zhang and R. Sennrich. *Root mean square layer normalization*. Curran Associates Inc., Red Hook, NY, USA, 2019.

A. Architecture and Training Details

A.1. Forward Pass, Backward Pass and Inference Equations

Table 1 show the equations across TriLM vs FloatLM for forward pass, backward pass and inference.

Let input be $X \in R_{b\times n}$ for a linear layer with FP16 weight matrix $W \in R_{m\times n}$ and $Y \in R_{b\times m}$ be the output. The same matrix W is also used to denote latent weights in TriLMs during training.

For ternarized layers in TriLMs, we also have a scalar scale $\gamma \in R$, matrix with ternarized states $\widehat{W} \in \{-1, 0, 1\}_{n\times m}$ and ternarized matrix $\widetilde{W} \in R_{n\times m}$. We set $\epsilon = 1e-5$.

A.2. Data and Tokenizer

Due to lack of availability of Pile 300B (Gao et al., 2020) used in Pythia or BitNet's data subset, we opted to use a 300B token sample of deduplicated Slim Pajama dataset[2]. We sample from each subset with the probability proportional to it's size. We use the GPT-NeoX 20B (Black et al., 2022) tokenizer following Pythia. For speeding up training, we round embedding rounding of to nearest multiple of 128 times the model parallel size.

For creating QuantLM, we use a subset of the Slimpajama-627B dataset, comprising 512 samples of sequence length of 2048.

A.3. PreTraining Setup

We scale using 2D-parallelism with Megatron-style sharding (Shoeybi et al., 2019) and use ZeRO stage 2 Deep-

[2]We also make this subset public.

Type	Forward Pass	Backward Pass	Inference
FloatLM	$Y = XW^T$	$\frac{\partial L}{\partial X} = \frac{\partial L}{\partial Y} W$ $\frac{\partial L}{\partial W} = \frac{\partial L}{\partial Y}^T X$	$Y = XW^T$
TriLM	$\gamma = \epsilon + \frac{1}{nm} \sum_{i=1}^{n} \sum_{j=1}^{m} \lvert W_{ij} \rvert$ $\widehat{W_{ij}} = round(min(\frac{max(W_{ij}}{\gamma}, -1), 1))$ $\widetilde{W_{ij}} = \gamma \widehat{W_{ij}}$ $Y = X\widetilde{W}^T$	$\frac{\partial L}{\partial X} = \frac{\partial L}{\partial Y} \widetilde{W}$ $\frac{\partial L}{\partial W} = \frac{\partial L}{\partial Y}^T X$	Compute once and cache $\widehat{W}$ and γ $\widetilde{W_{ij}} = \gamma \widehat{W_{ij}}$ $Y = X\widetilde{W}^T$

Table 1. Equations of Linear Layer in TriLMs and FloatLMs.

Dataset	**Size (Tokens)**
Arxiv	13B
Book	13B
C4	80B
Common Crawl	156B
GitHub	16B
Stack Exchange	10B
Wikipedia	12B
Total	300B

Table 2. 300B Subset of Slim Pajama

speed (Rasley et al., 2020) for ZeRO (Rajbhandari et al., 2020). Our implementation was based on GPT NeoX Codebase (Andonian et al., 2023). We use Adam (Kingma and Ba, 2017) for optimization. We train on nodes with with IBM Power9 PC CPUs and 6x16GB V100. Due to lack of BFloat16 support in V100, we train both TriLM and FloatLM in FP16 using Mixed Precision Training and Dynamic Loss Scaling. We extensively use Huggingface (Wolf et al., 2020) and Wandb (Biewald, 2020) for handling the checkpoints and experiment tracking.

A.4. Hyperparameters

Table 3 shows the hyperparameters for TriLM and FloatLM's transformer architecture and their learning rate. We set Adam β are set to (0.9, 0.95) for both families of models and all the reported runs are trained to 2048 sequence length. FloatLM and TriLM are respectively trained with batch sizes of $2M$ and $1M$ tokens respectively.

Known Implementation Artifacts

- Because we train in FP16, we may expect some artifacts from training. However, we do not expect a reasonable performance difference from mixed precision training with BF16 or even FP32 because the lowest values of loss scales observed during any of the runs were much higher than 1. Moreover, in BitNet b1.58 (Section 3), they compared models to their reproduced FP16 LLaMA LLM. Thus, our setting closely resemble theirs.
- Similar to BitNet (Wang et al., 2023), our models have artifacts from model parallelism. Specifically, computing the scale γ across all the entire weight matrix - which has been sharded across multiple devices requires a costly communication overhead from all-reduce. In our implementation, we compute these scales over the portion of weight matrix local to each device. Thus, for inference over TriLM models, scales should be independently computed over each model parallel group. It should be noted that this negligible change on effective on bits/parameter of $< 10^{-5}$, even at highest model parallelism of 6 for our largest model.

B. Benchmark Details

We consider standard benchmarks for testing the two families of models across Knowledge, CommonSense and Reasoning. We leverage the LM Evaluation Harness (Gao et al., 2023) to run our experiments for TriLMs, FloatLMs and different levels of quantization of FloatLMs. We also add Pythia suite of models (70M to 2.8B params) and BitNet b.158 performance scores from their paper for comparison . For each task of the commonsense and reasoning benchmark, we report the zero-shot accuracy (both length normalized and unnormalized) or perplexity score wherever applicable. The Pythia suite of models are based on the standard GPT2 architecture with rotary positional embedding and GPT-NeoX 20B tokenizer. These were trained on 300B tokens of non-deduplicated Pile dataset with a batch size of 2M. Pythia models. We use 70M, 160M, 410M, 1B, 1.4B, and 2.8B Pythia models in our experiments in FP16 precision. We did not use Pythia v0 due to non-uniformity in batch size and learning rate schedule across the models. For a fair comparison with TriLM and FloatLM, we compare the models with approximately same number of bits.

B.1. Commonsense and Reasoning

We report commonsense and reasoning benchmark scores across 6 benchmarks previously considered by BitNet b.158 in Table 4 and rest in Table 5. Each is considered in a zero shot setting. Following are the details of each of the benchmark considered:

Params	Hidden	GLU	Heads	Layers	FloatLM LR	TriLM LR
99.74M (99M)	512	1280	8	16	$4.0 * 10^{-4}$	$2.4 * 10^{-3} \rightarrow 1.5 * 10^{-3}$
190.0M (190M)	768	2048	12	16	$4.0 * 10^{-4}$	$2.4 * 10^{-3} \rightarrow 1.5 * 10^{-3}$
392.4M (390M)	1024	2560	16	24	$3.0 * 10^{-4}$	$1.8 * 10^{-3} \rightarrow 1.2 * 10^{-3}$
569.2M (560M)	1280	3072	20	24	$2.8 * 10^{-4}$	$1.6 * 10^{-3} \rightarrow 1.1 * 10^{-3}$
834.0M (830M)	1536	4096	24	24	$2.5 * 10^{-4}$	$1.5 * 10^{-3} \rightarrow 1.0 * 10^{-3}$
1.149B (1.1B)	1792	5120	28	24	$2.2 * 10^{-4}$	$1.3 * 10^{-3} \rightarrow 9.0 * 10^{-4}$
1.515B (1.5B)	2048	6144	32	24	$2.0 * 10^{-4}$	$1.2 * 10^{-3} \rightarrow 8.0 * 10^{-4}$
2.461B (2.4B)	2304	7680	36	30	$2.0 * 10^{-4}$	$1.2 * 10^{-3} \rightarrow 8.0 * 10^{-4}$
3.989B (3.9B)	3072	9216	24	30	$1.5 * 10^{-4}$	$1.2 * 10^{-3} \rightarrow 8.0 * 10^{-4}$

Table 3. Hyperparameters across model sizes for TriLM and FloatLM.

- **ARC Challenge and Easy**: (Clark et al., 2018) ARC dataset comprises 7787 multiple-choice science questions divided into two sets: Challenge and Easy. We calculate accuracy and normalised accuracy across both of these sets.

- **BoolQ**: (Clark et al., 2019) BoolQ is a reading comprehension dataset consisting of naturally occurring yes/no questions. We calculate the accuracy on this tasks.

- **HellaSwag**: (Zellers et al., 2019) HellaSWAG is a dataset multiple choice questions for testing grounded commonsense. The incorrect options are generated through Adversarial Filtering (AF) to fool machines but not humans. We calculate the accuracy and normalised accuracy on this task.

- **WinoGrande**: (Sakaguchi et al., 2021) WinoGrande is a collection of 44k problems for testing commonsense reasoning formulated as a fill-in-a-blank task with binary options. We report the accuracy on this task.

- **PIQA**: (Bisk et al., 2019) Physical Interaction: Question Answering (PIQA) a physical commonsense reasoning benchmark dataset to test the physical knowledge of language models. We calculate the accuracy and normalised accuracy on this task.

- **LAMBADA OpenAI**: (Paperno et al., 2016) LAMBADA is a dataset to evaluate text understanding by next word prediction. It is a collection of narrative passages BooksCorpus To succeed on LAMBADA, models must integrate broader discourse information, not solely rely on local context. We calculate the perplexity and the accuracy of the model on this task.

- **SciQ**: (Welbl et al., 2017) The SciQ dataset contains multiple-choice questions with 4 answer options from crowd-sourced science exams. The questions range from Physics, Chemistry and Biology and several other fields. We calculate the accuracy and length normalised accuracy on this task.

- **LogiQA**: (Liu et al., 2021) LogiQA is a dataset for testing human logical reasoning. It contains questions spanning multiple types of deductive reasoning. We calculate the accuracy and normalised accuracy on this task.

B.2. Knowledge

We report performance on TriviaQA across zero-shot, 1-shot, 3-shot, 5-shot and 10-shot settings in report commonsense and reasoning benchmark scores across 6 benchmarks previously considered by BitNet b.158 in Table 4 and rest in Table 5. Each is considered in a zero shot setting. Following are the details of each of the benchmark considered:

For report commonsense and reasoning benchmark scores across 6 benchmarks previously considered by BitNet b.158 in Table 4 and rest in Table 5. Following are the details of each of the benchmark considered:

The knowledge-based evaluation included the following tasks:

- **TriviaQA**: (Joshi et al., 2017) TriviaQA is a reading comprehension dataset containing question-answer-evidence triples. We calculate the exact match accuracy on this task.

Models	Bits	Bits/	Arc Easy		Arc Challenge		HellaSwag		BoolQ	PIQA		WinoGrande
	$(*10^9)$	Param	Acc Norm.	Acc	Acc Norm.	Acc	Acc Norm.	Acc	Acc	Acc Norm.	Acc	Acc
TriLM 99M	0.9	1.58	36.4± 1.0	40.3± 1.0	21.8± 1.2	18.2± 1.1	28.5± 0.5	27.6± 0.4	61.6± 0.9	60.6± 1.1	59.4± 1.1	51.5± 1.4
FloatLM 99M	0.98	3.25	34.8± 1.0	36.1± 1.0	23.2± 1.2	19.5± 1.2	29.2± 0.5	27.7± 0.4	48.4± 0.9	57.2± 1.2	58.2± 1.2	49.2± 1.4
FloatLM 99M	1.03	4.25	37.1± 1.0	41.7± 1.0	22.6± 1.2	18.0± 1.1	31.0± 0.5	28.9± 0.5	52.2± 0.9	62.2± 1.1	60.9± 1.1	50.4± 1.4
FloatLM 99M	1.11	6	38.8± 1.0	44.8± 1.0	23.2± 1.2	19.7± 1.2	31.7± 0.5	29.2± 0.5	58.9± 0.9	62.8± 1.1	63.1± 1.1	50.2± 1.4
Pythia 70M	1.13	16	24.8± 0.9	24.8± 0.9	22.0± 1.2	22.1± 1.2	25.1± 0.4	25.1± 0.4	38.5± 0.9	49.8± 1.2	49.9± 1.2	49.1± 1.4
FloatLM 99M	1.21	8	39.4± 1.0	45.3± 1.0	23.8± 1.2	19.6± 1.2	31.7± 0.5	29.0± 0.5	58.5± 0.9	62.6± 1.1	63.0± 1.1	50.0± 1.4
TriLM 190M	1.42	1.58	38.7± 1.0	43.0± 1.0	23.8± 1.2	19.7± 1.2	32.0± 0.5	28.8± 0.5	61.8± 0.8	61.8± 1.1	62.2± 1.1	52.0± 1.4
FloatLM 99M	1.6	16	39.1± 1.0	45.1± 1.0	23.8± 1.2	19.9± 1.2	31.6± 0.5	29.1± 0.5	58.2± 0.9	62.8± 1.1	63.2± 1.1	50.2± 1.4
FloatLM 190M	1.6	3.25	37.1± 1.0	39.7± 1.0	22.5± 1.2	19.4± 1.2	32.0± 0.5	28.8± 0.5	56.5± 0.9	58.1± 1.2	58.7± 1.1	50.1± 1.4
FloatLM 190M	1.72	4.25	26.5± 0.9	26.8± 0.9	25.2± 1.3	19.9± 1.2	26.0± 0.4	25.7± 0.4	40.9± 0.9	49.3± 1.2	51.7± 1.2	51.0± 1.4
FloatLM 190M	1.92	6	42.0± 1.0	48.0± 1.0	23.8± 1.2	20.0± 1.2	36.3± 0.5	31.5± 0.5	59.1± 0.9	65.6± 1.1	64.3± 1.1	51.9± 1.4
TriLM 390M	2.11	1.58	42.6± 1.0	47.6± 1.0	24.2± 1.3	22.3± 1.2	37.3± 0.5	32.1± 0.5	57.1± 0.9	65.3± 1.1	64.9± 1.1	53.4± 1.4
FloatLM 190M	2.14	8	43.0± 1.0	48.5± 1.0	24.4± 1.3	20.3± 1.2	36.5± 0.5	31.4± 0.5	59.3± 0.9	65.6± 1.1	64.8± 1.1	51.7± 1.4
FloatLM 390M	2.59	3.25	41.6± 1.0	43.6± 1.0	24.9± 1.3	21.5± 1.2	39.5± 0.5	32.9± 0.5	56.3± 0.9	63.8± 1.1	63.2± 1.1	53.0± 1.4
Pythia 160M	2.6	16	26.7± 0.9	26.6± 0.9	23.8± 1.2	23.1± 1.2	25.1± 0.4	25.0± 0.4	38.3± 0.9	53.1± 1.2	53.1± 1.2	47.3± 1.4
TriLM 560M	2.76	1.58	46.5± 1.0	51.9± 1.0	25.5± 1.3	22.5± 1.2	41.2± 0.5	33.5± 0.5	61.7± 0.9	67.5± 1.1	66.3± 1.1	53.1± 1.4
FloatLM 390M	2.88	4.25	45.2± 1.0	49.6± 1.0	25.1± 1.3	21.3± 1.2	43.4± 0.5	35.1± 0.5	50.8± 0.9	68.1± 1.1	68.3± 1.1	53.7± 1.4
FloatLM 190M	3.05	16	43.0± 1.0	48.4± 1.0	24.1± 1.3	20.5± 1.2	36.6± 0.5	31.4± 0.5	59.1± 0.9	65.6± 1.1	64.8± 1.1	51.9± 1.4
FloatLM 390M	3.38	6	46.8± 1.0	51.8± 1.0	24.8± 1.3	21.5± 1.2	44.2± 0.5	35.6± 0.5	55.3± 0.9	69.0± 1.1	68.4± 1.1	53.0± 1.4
FloatLM 560M	3.49	3.25	42.3± 1.0	45.8± 1.0	24.0± 1.2	21.2± 1.2	41.7± 0.5	33.4± 0.5	59.0± 0.9	63.5± 1.1	63.2± 1.1	49.7± 1.4
TriLM 830M	3.55	1.58	48.0± 1.0	53.6± 1.0	25.3± 1.3	21.4± 1.2	45.1± 0.5	36.3± 0.5	61.6± 0.9	68.8± 1.1	68.2± 1.1	53.8± 1.4
FloatLM 560M	3.93	4.25	46.3± 1.0	52.4± 1.0	25.9± 1.3	23.0± 1.2	46.7± 0.5	37.0± 0.5	58.8± 0.9	67.8± 1.1	67.1± 1.1	53.1± 1.4
FloatLM 390M	3.96	8	46.6± 1.0	51.0± 1.0	24.6± 1.3	21.2± 1.2	44.5± 0.5	35.7± 0.5	54.6± 0.9	68.8± 1.1	68.6± 1.1	52.6± 1.4
TriLM 1.1B	4.42	1.58	49.1± 1.0	55.5± 1.0	27.6± 1.3	25.7± 1.3	48.5± 0.5	38.2± 0.5	60.3± 0.9	70.0± 1.1	70.3± 1.1	56.9± 1.4
FloatLM 830M	4.68	3.25	46.8± 1.0	50.5± 1.0	27.1± 1.3	22.7± 1.2	45.5± 0.5	35.9± 0.5	56.3± 0.9	66.1± 1.1	66.6± 1.1	53.5± 1.4
FloatLM 560M	4.7	6	47.6± 1.0	54.2± 1.0	26.0± 1.3	23.5± 1.2	47.6± 0.5	37.7± 0.5	57.3± 0.9	68.7± 1.1	68.8± 1.1	53.5± 1.4
Pythia 410M	4.87	16	45.7± 1.0	51.6± 1.0	24.7± 1.3	21.2± 1.2	40.3± 0.5	33.8± 0.5	60.0± 0.9	67.2± 1.1	66.3± 1.1	53.5± 1.4
TriLM 1.5B	5.36	1.58	50.8± 1.0	57.3± 1.0	27.7± 1.3	24.9± 1.3	51.7± 0.5	40.1± 0.5	62.6± 0.8	70.5± 1.1	70.7± 1.1	56.0± 1.4
FloatLM 830M	5.36	4.25	50.5± 1.0	56.2± 1.0	27.6± 1.3	23.3± 1.2	50.2± 0.5	39.2± 0.5	58.1± 0.9	70.6± 1.1	71.1± 1.1	56.0± 1.4
FloatLM 560M	5.58	8	48.3± 1.0	54.1± 1.0	26.5± 1.3	23.6± 1.2	47.6± 0.5	37.7± 0.5	57.6± 0.9	68.9± 1.1	68.9± 1.1	53.8± 1.4
FloatLM 1.1B	6.03	3.25	48.9± 1.0	55.0± 1.0	29.2± 1.3	27.0± 1.3	51.3± 0.5	39.4± 0.5	62.1± 0.8	69.4± 1.1	68.4± 1.1	54.8± 1.4
FloatLM 390M	6.28	16	46.5± 1.0	51.0± 1.0	24.7± 1.3	21.3± 1.2	44.4± 0.5	35.7± 0.5	54.7± 0.9	68.7± 1.1	68.4± 1.1	51.8± 1.4
FloatLM 830M	6.55	6	51.6± 1.0	57.7± 1.0	27.6± 1.3	24.7± 1.3	51.5± 0.5	40.2± 0.5	61.3± 0.9	71.3± 1.1	71.8± 1.0	56.2± 1.4
FloatLM 1.1B	7.0	4.25	53.6± 1.0	59.0± 1.0	30.3± 1.3	26.0± 1.3	54.9± 0.5	42.0± 0.5	61.3± 0.9	71.6± 1.1	70.4± 1.1	54.8± 1.4
TriLM 2.4B	7.23	1.58	55.4± 1.0	62.5± 1.0	30.1± 1.3	27.4± 1.3	57.8± 0.5	44.0± 0.5	63.1± 0.8	72.1± 1.0	72.4± 1.0	58.5± 1.4
FloatLM 1.5B	7.55	3.25	49.7± 1.0	54.8± 1.0	27.8± 1.3	25.2± 1.3	53.7± 0.5	41.0± 0.5	53.7± 0.9	70.0± 1.1	69.4± 1.1	55.0± 1.4
FloatLM 830M	7.91	8	51.7± 1.0	57.3± 1.0	28.2± 1.3	25.1± 1.3	51.4± 0.5	40.1± 0.5	60.9± 0.9	71.2± 1.1	71.7± 1.1	55.9± 1.4
FloatLM 1.1B	8.7	6	54.3± 1.0	60.2± 1.0	29.8± 1.3	25.5± 1.3	54.9± 0.5	42.6± 0.5	62.9± 0.8	71.9± 1.0	71.2± 1.1	56.1± 1.4
FloatLM 1.5B	8.86	4.25	55.2± 1.0	60.4± 1.0	29.4± 1.3	26.9± 1.3	56.9± 0.5	43.2± 0.5	62.5± 0.8	72.7± 1.0	72.4± 1.0	57.1± 1.4
FloatLM 560M	9.11	16	48.4± 1.0	54.4± 1.0	26.5± 1.3	23.9± 1.2	47.6± 0.5	37.7± 0.5	57.9± 0.9	68.8± 1.1	69.0± 1.1	53.7± 1.4
FloatLM 1.1B	10.64	8	54.1± 1.0	60.2± 1.0	28.9± 1.3	26.1± 1.3	55.2± 0.5	42.6± 0.5	62.6± 0.8	72.1± 1.0	71.2± 1.1	56.2± 1.4
TriLM 3.9B	10.76	1.58	60.4± 1.0	66.1± 1.0	34.4± 1.4	30.6± 1.3	62.8± 0.5	46.8± 0.5	64.6± 0.8	74.3± 1.0	74.3± 1.0	63.6± 1.4
FloatLM 2.4B	10.95	3.25	54.2± 1.0	58.4± 1.0	29.7± 1.3	28.4± 1.3	58.6± 0.5	43.5± 0.5	55.7± 0.9	72.7± 1.0	70.8± 1.1	57.2± 1.4
FloatLM 1.5B	11.15	6	56.8± 1.0	62.2± 1.0	30.1± 1.3	26.0± 1.3	57.5± 0.5	44.2± 0.5	63.4± 0.8	74.0± 1.0	73.0± 1.0	59.7± 1.4
FloatLM 2.4B	13.18	4.25	59.6± 1.0	64.1± 1.0	33.3± 1.4	30.8± 1.3	62.2± 0.5	46.5± 0.5	59.0± 0.9	75.4± 1.0	74.5± 1.0	61.7± 1.4
FloatLM 830M	13.34	16	51.6± 1.0	57.3± 1.0	28.0± 1.3	24.5± 1.3	51.3± 0.5	40.1± 0.5	61.0± 0.9	71.4± 1.1	71.7± 1.1	56.4± 1.4
FloatLM 1.5B	13.77	8	56.6± 1.0	62.4± 1.0	29.8± 1.3	26.0± 1.3	57.8± 0.5	44.3± 0.5	63.3± 0.8	73.7± 1.0	73.1± 1.0	59.4± 1.4
Pythia 1B	16.18	16	49.0± 1.0	57.0± 1.0	27.0± 1.3	24.4± 1.3	47.2± 0.5	37.7± 0.5	60.8± 0.9	69.3± 1.1	70.8± 1.1	53.2± 1.4
FloatLM 3.9B	16.91	3.25	55.5± 1.0	62.1± 1.0	32.1± 1.4	29.3± 1.3	61.2± 0.5	45.9± 0.5	60.0± 0.9	72.6± 1.0	72.3± 1.0	59.3± 1.4
FloatLM 2.4B	17.09	6	60.4± 1.0	65.4± 1.0	32.7± 1.4	30.6± 1.3	62.9± 0.5	47.0± 0.5	62.0± 0.8	75.7± 1.0	74.7± 1.0	61.1± 1.4
FloatLM 1.1B	18.39	16	54.0± 1.0	60.4± 1.0	29.1± 1.3	26.1± 1.3	55.2± 0.5	42.6± 0.5	62.9± 0.8	72.2± 1.0	71.3± 1.1	56.3± 1.4
FloatLM 3.9B	20.59	4.25	61.2± 1.0	68.3± 1.0	34.7± 1.4	32.9± 1.4	65.0± 0.5	49.0± 0.5	65.4± 0.8	75.5± 1.0	75.6± 1.0	62.7± 1.4
FloatLM 2.4B	21.55	8	60.3± 1.0	65.7± 1.0	32.6± 1.4	30.0± 1.3	62.7± 0.5	47.1± 0.5	62.1± 0.8	75.4± 1.0	74.9± 1.0	61.4± 1.4
Pythia 1.4B	22.62	16	54.0± 1.0	60.4± 1.0	28.7± 1.3	26.0± 1.3	52.0± 0.5	40.4± 0.5	63.2± 0.8	70.8± 1.1	70.6± 1.1	57.1± 1.4
FloatLM 1.5B	24.23	16	56.4± 1.0	62.6± 1.0	29.7± 1.3	26.2± 1.3	57.8± 0.5	44.3± 0.5	63.2± 0.8	73.9± 1.0	73.1± 1.0	59.4± 1.4
FloatLM 3.9B	27.03	6	63.3± 1.0	68.0± 1.0	35.1± 1.4	32.1± 1.4	65.9± 0.5	49.7± 0.5	65.6± 0.8	75.5± 1.0	75.6± 1.0	62.2± 1.4
FloatLM 3.9B	34.39	8	63.0± 1.0	68.1± 1.0	34.6± 1.4	31.9± 1.4	66.0± 0.5	49.7± 0.5	65.4± 0.8	75.9± 1.0	75.5± 1.0	62.9± 1.4
FloatLM 2.4B	39.38	16	60.5± 1.0	65.5± 1.0	32.7± 1.4	30.1± 1.3	62.7± 0.5	47.1± 0.5	62.1± 0.8	75.2± 1.0	74.9± 1.0	61.8± 1.4
Pythia 2.8B	44.39	16	58.8± 1.0	64.3± 1.0	33.2± 1.4	29.2± 1.3	59.2± 0.5	45.3± 0.5	63.7± 0.8	73.8± 1.0	73.9± 1.0	58.9± 1.4
FloatLM 3.9B	63.83	16	63.0± 1.0	68.3± 1.0	34.6± 1.4	32.1± 1.4	66.1± 0.5	49.7± 0.5	65.9± 0.8	75.8± 1.0	75.4± 1.0	62.8± 1.4
BitNet 700M	-	1.58	51.8		21.4		35.1		58.2	68.1		55.2
BitNet 1.3B	-	1.58	54.9		24.2		37.7		56.7	68.8		55.8
BitNet 3B	-	1.58	61.4		28.3		42.9		61.5	71.5		59.3
BitNet 3.9B	-	1.58	64.2		28.7		44.2		63.5	73.2		60.5

Table 4. Common Sense Task Performance: Arc Easy, Arc Challenge, HellaSwag, BoolQ, PIQA, WinoGrande. BitNet b1.58's scores are taken from Ma et al. (2024)

Models	Bits ($*10^9$)	Bits/Param	LAMBADA		SciQ		LogiQA	
			Perp.	Acc	Acc Norm.	Acc	Acc Norm.	Acc
TriLM 99M	0.9	1.58	160.4± 8.2	20.8± 0.6	59.5± 1.6	68.0± 1.5	24.9± 1.7	20.0± 1.6
FloatLM 99M	0.98	3.25	4765.4± 413.0	4.5± 0.3	51.9± 1.6	57.0± 1.6	25.3± 1.7	19.8± 1.6
FloatLM 99M	1.03	4.25	211.6± 17.3	16.7± 0.5	61.2± 1.5	70.7± 1.4	24.9± 1.7	20.7± 1.6
FloatLM 99M	1.11	6	89.9± 7.4	26.1± 0.6	61.8± 1.5	73.9± 1.4	28.1± 1.8	20.3± 1.6
Pythia 70M	1.13	16	NaN	0.9± 0.1	0.9± 0.3	1.1± 0.3	20.3± 1.6	20.3± 1.6
FloatLM 99M	1.21	8	85.8± 7.0	26.6± 0.6	62.8± 1.5	73.7± 1.4	27.8± 1.8	21.0± 1.6
TriLM 190M	1.42	1.58	118.1± 6.2	26.3± 0.6	61.9± 1.5	73.5± 1.4	25.0± 1.7	20.3± 1.6
FloatLM 99M	1.6	16	85.0± 6.9	26.5± 0.6	62.9± 1.5	73.6± 1.4	27.6± 1.8	21.2± 1.6
FloatLM 190M	1.6	3.25	664.5± 41.1	12.4± 0.5	58.5± 1.6	66.4± 1.5	26.3± 1.7	21.0± 1.6
FloatLM 190M	1.72	4.25	72479077.3	0	25.6± 1.4	22.9± 1.3	23.3± 1.7	20.7± 1.6
FloatLM 190M	1.92	6	55.3± 3.0	30.0± 0.6	64.2± 1.5	77.0± 1.3	26.1± 1.7	22.4± 1.6
TriLM 390M	2.11	1.58	74.7± 3.5	28.1± 0.6	65.6± 1.5	76.9± 1.3	25.5± 1.7	20.7± 1.6
FloatLM 190M	2.14	8	48.7± 2.6	31.5± 0.6	65.5± 1.5	77.1± 1.3	27.0± 1.7	22.3± 1.6
FloatLM 390M	2.59	3.25	115.0± 5.6	23.0± 0.6	67.4± 1.5	76.7± 1.3	25.7± 1.7	21.8± 1.6
Pythia 160M	2.6	16	NaN	9.7± 0.4	1.5± 0.4	1.6± 0.4	20.3± 1.6	20.3± 1.6
TriLM 560M	2.76	1.58	47.1± 2.2	34.2± 0.7	73.8± 1.4	82.2± 1.2	26.0± 1.7	21.0± 1.6
FloatLM 390M	2.88	4.25	30.2± 1.3	39.1± 0.7	77.1± 1.3	84.1± 1.2	25.8± 1.7	23.3± 1.7
FloatLM 190M	3.05	16	50.3± 2.7	31.1± 0.6	65.1± 1.5	77.3± 1.3	27.2± 1.7	22.1± 1.6
FloatLM 390M	3.38	6	24.3± 1.0	40.6± 0.7	75.5± 1.4	83.7± 1.2	27.6± 1.8	23.2± 1.7
FloatLM 560M	3.49	3.25	146.3± 7.1	20.1± 0.6	71.1± 1.4	75.9± 1.4	25.0± 1.7	21.8± 1.6
TriLM 830M	3.55	1.58	25.9± 1.1	41.5± 0.7	75.6± 1.4	83.1± 1.2	25.5± 1.7	20.3± 1.6
FloatLM 560M	3.93	4.25	24.9± 1.1	40.8± 0.7	73.6± 1.4	82.0± 1.2	27.0± 1.7	21.7± 1.6
FloatLM 390M	3.96	8	21.7± 0.9	42.3± 0.7	75.7± 1.4	84.1± 1.2	28.3± 1.8	24.1± 1.7
TriLM 1.1B	4.42	1.58	21.5± 0.9	44.3± 0.7	72.5± 1.4	83.2± 1.2	26.7± 1.7	18.9± 1.5
FloatLM 830M	4.68	3.25	47.7± 2.0	30.5± 0.6	74.1± 1.4	80.1± 1.3	28.1± 1.8	21.2± 1.6
FloatLM 560M	4.7	6	21.7± 0.9	42.8± 0.7	74.4± 1.4	83.6± 1.2	25.8± 1.7	20.9± 1.6
Pythia 410M	4.87	16	11.9± 0.4	49.9± 0.7	70.8± 1.4	80.9± 1.2	28.7± 1.8	21.8± 1.6
TriLM 1.5B	5.36	1.58	17.3± 0.7	45.8± 0.7	79.6± 1.3	86.8± 1.1	28.3± 1.8	20.9± 1.6
FloatLM 830M	5.36	4.25	15.4± 0.6	47.3± 0.7	78.8± 1.3	85.1± 1.1	25.5± 1.7	21.2± 1.6
FloatLM 560M	5.58	8	20.9± 0.9	44.2± 0.7	74.7± 1.4	83.6± 1.2	27.3± 1.7	20.7± 1.6
FloatLM 1.1B	6.03	3.25	26.9± 1.1	39.1± 0.7	78.7± 1.3	85.0± 1.1	25.8± 1.7	20.7± 1.6
FloatLM 390M	6.28	16	21.9± 0.9	42.2± 0.7	75.6± 1.4	84.2± 1.2	28.1± 1.8	23.8± 1.7
FloatLM 830M	6.55	6	13.3± 0.5	49.1± 0.7	77.8± 1.3	85.4± 1.1	26.3± 1.7	20.1± 1.6
FloatLM 1.1B	7.0	4.25	13.9± 0.5	49.3± 0.7	81.2± 1.2	87.6± 1.0	28.4± 1.8	20.3± 1.6
TriLM 2.4B	7.23	1.58	8.9± 0.3	55.0± 0.7	79.7± 1.3	86.6± 1.1	27.8± 1.8	20.0± 1.6
FloatLM 1.5B	7.55	3.25	17.8± 0.7	45.3± 0.7	75.5± 1.4	82.1± 1.2	28.4± 1.8	22.7± 1.6
FloatLM 830M	7.91	8	13.5± 0.5	49.4± 0.7	78.5± 1.3	86.1± 1.1	26.6± 1.7	20.0± 1.6
FloatLM 1.1B	8.7	6	11.7± 0.4	51.0± 0.7	82.3± 1.2	88.1± 1.0	27.5± 1.8	21.5± 1.6
FloatLM 1.5B	8.86	4.25	10.4± 0.4	53.0± 0.7	81.1± 1.2	86.9± 1.1	25.7± 1.7	20.3± 1.6
FloatLM 560M	9.11	16	20.8± 0.9	44.1± 0.7	74.7± 1.4	83.5± 1.2	27.0± 1.7	20.7± 1.6
FloatLM 1.1B	10.64	8	11.7± 0.4	51.2± 0.7	82.1± 1.2	88.1± 1.0	27.8± 1.8	21.2± 1.6
TriLM 3.9B	10.76	1.58	6.6± 0.2	61.1± 0.7	85.8± 1.1	89.4± 1.0	27.6± 1.8	19.4± 1.5
FloatLM 2.4B	10.95	3.25	15.6± 0.6	45.0± 0.7	79.9± 1.3	86.7± 1.1	28.6± 1.8	21.4± 1.6
FloatLM 1.5B	11.15	6	9.5± 0.3	55.4± 0.7	81.4± 1.2	87.6± 1.0	25.7± 1.7	20.3± 1.6
FloatLM 2.4B	13.18	4.25	8.9± 0.3	56.1± 0.7	84.8± 1.1	89.7± 1.0	29.6± 1.8	20.9± 1.6
FloatLM 830M	13.34	16	13.3± 0.5	49.6± 0.7	78.4± 1.3	85.9± 1.1	26.3± 1.7	20.1± 1.6
FloatLM 1.5B	13.77	8	9.5± 0.3	55.5± 0.7	81.3± 1.2	87.5± 1.0	25.7± 1.7	20.6± 1.6
Pythia 1B	16.18	16	7.9± 0.2	56.1± 0.7	76.0± 1.4	83.8± 1.2	29.8± 1.8	22.1± 1.6
FloatLM 3.9B	16.91	3.25	14.0± 0.5	47.1± 0.7	83.1± 1.2	88.6± 1.0	27.0± 1.7	21.5± 1.6
FloatLM 2.4B	17.09	6	7.9± 0.3	58.9± 0.7	87.3± 1.1	90.9± 0.9	29.6± 1.8	20.9± 1.6
FloatLM 1.1B	18.39	16	11.7± 0.4	51.2± 0.7	82.2± 1.2	88.1± 1.0	27.3± 1.7	20.9± 1.6
FloatLM 3.9B	20.59	4.25	7.4± 0.2	58.5± 0.7	86.1± 1.1	90.8± 0.9	28.6± 1.8	20.1± 1.6
FloatLM 2.4B	21.55	8	7.7± 0.3	59.2± 0.7	87.1± 1.1	91.0± 0.9	29.5± 1.8	21.5± 1.6
Pythia 1.4B	22.62	16	6.1± 0.2	61.5± 0.7	79.1± 1.3	86.7± 1.1	27.8± 1.8	20.9± 1.6
FloatLM 1.5B	24.23	16	9.4± 0.3	55.5± 0.7	80.9± 1.2	87.4± 1.0	26.1± 1.7	20.9± 1.6
FloatLM 3.9B	27.03	6	6.8± 0.2	60.8± 0.7	86.6± 1.1	91.3± 0.9	25.8± 1.7	20.4± 1.6
FloatLM 3.9B	34.39	8	6.7± 0.2	61.1± 0.7	86.2± 1.1	91.0± 0.9	26.6± 1.7	20.6± 1.6
FloatLM 2.4B	39.38	16	7.7± 0.3	59.3± 0.7	87.2± 1.1	91.0± 0.9	29.5± 1.8	21.5± 1.6
Pythia 2.8B	44.39	16	5.1± 0.1	64.6± 0.7	83.6± 1.2	88.5± 1.0	28.3± 1.8	22.0± 1.6
FloatLM 3.9B	63.83	16	6.7± 0.2	61.1± 0.7	86.5± 1.1	90.9± 0.9	26.9± 1.7	20.9± 1.6

Table 5. Common Sense Task Performance (Contd.): LAMBADA OpenAI, SciQ, LogiQA

Models	Bits ($*10^9$)	Bits/Param	TriviaQA Exact Match Score				
			0-shot	1-shot	3-shot	5-shot	10-shot
TriLM 99M	0.9	1.58	0.4± 0.0	0.8± 0.1	1.4± 0.1	1.5± 0.1	1.7± 0.1
FloatLM 99M	0.98	3.25	0.1± 0.0	0.1± 0.0	0.3± 0.0	0.4± 0.0	0.6± 0.1
FloatLM 99M	1.03	4.25	0.3± 0.0	1.6± 0.1	2.6± 0.1	2.7± 0.1	2.8± 0.1
FloatLM 99M	1.11	6	0.6± 0.1	2.8± 0.1	3.7± 0.1	3.9± 0.1	4.3± 0.2
Pythia 70M	1.13	16	0.1± 0.0	0.1± 0.0	0.2± 0.0	0.2± 0.0	0.2± 0.0
FloatLM 99M	1.21	8	0.6± 0.1	2.8± 0.1	4.0± 0.1	4.2± 0.2	4.6± 0.2
TriLM 190M	1.42	1.58	0.2± 0.0	2.5± 0.1	3.7± 0.1	3.6± 0.1	3.5± 0.1
FloatLM 99M	1.6	16	0.6± 0.1	2.9± 0.1	4.0± 0.1	4.3± 0.2	4.6± 0.2
FloatLM 190M	1.6	3.25	0.1± 0.0	0.5± 0.1	1.0± 0.1	0.9± 0.1	0.9± 0.1
FloatLM 190M	1.72	4.25	0.0± 0.0	0.0± 0.0	0.0± 0.0	0.0± 0.0	0.0± 0.0
FloatLM 190M	1.92	6	0.7± 0.1	4.8± 0.2	6.6± 0.2	7.1± 0.2	7.3± 0.2
TriLM 390M	2.11	1.58	0.9± 0.1	4.5± 0.2	5.9± 0.2	6.0± 0.2	6.4± 0.2
FloatLM 190M	2.14	8	0.7± 0.1	5.1± 0.2	7.2± 0.2	7.7± 0.2	7.8± 0.2
FloatLM 390M	2.59	3.25	0.8± 0.1	1.6± 0.1	3.3± 0.1	3.6± 0.1	3.8± 0.1
Pythia 160M	2.6	16	0.5± 0.1	1.1± 0.1	1.4± 0.1	1.7± 0.1	1.7± 0.1
TriLM 560M	2.76	1.58	2.4± 0.1	6.3± 0.2	8.6± 0.2	9.0± 0.2	9.4± 0.2
FloatLM 390M	2.88	4.25	1.3± 0.1	8.0± 0.2	10.5± 0.2	11.2± 0.2	11.2± 0.2
FloatLM 190M	3.05	16	0.6± 0.1	5.2± 0.2	7.1± 0.2	7.6± 0.2	7.8± 0.2
FloatLM 390M	3.38	6	2.4± 0.1	9.5± 0.2	12.7± 0.2	13.4± 0.3	13.8± 0.3
FloatLM 560M	3.49	3.25	1.5± 0.1	2.5± 0.1	3.7± 0.1	4.6± 0.2	5.1± 0.2
TriLM 830M	3.55	1.58	2.7± 0.1	10.2± 0.2	11.6± 0.2	12.0± 0.2	12.5± 0.2
FloatLM 560M	3.93	4.25	2.1± 0.1	10.4± 0.2	12.8± 0.2	13.5± 0.3	13.9± 0.3
FloatLM 390M	3.96	8	2.9± 0.1	9.7± 0.2	13.1± 0.3	13.7± 0.3	14.2± 0.3
TriLM 1.1B	4.42	1.58	1.0± 0.1	10.1± 0.2	13.3± 0.3	14.2± 0.3	15.1± 0.3
FloatLM 830M	4.68	3.25	3.1± 0.1	6.3± 0.2	8.4± 0.2	8.9± 0.2	9.0± 0.2
FloatLM 560M	4.7	6	3.5± 0.1	12.5± 0.2	15.5± 0.3	16.4± 0.3	17.5± 0.3
Pythia 410M	4.87	16	1.7± 0.1	6.8± 0.2	8.7± 0.2	9.1± 0.2	9.5± 0.2
TriLM 1.5B	5.36	1.58	2.1± 0.1	15.8± 0.3	18.2± 0.3	18.7± 0.3	19.1± 0.3
FloatLM 830M	5.36	4.25	10.6± 0.2	17.7± 0.3	19.7± 0.3	20.6± 0.3	20.8± 0.3
FloatLM 560M	5.58	8	4.7± 0.2	13.0± 0.3	16.0± 0.3	16.8± 0.3	18.0± 0.3
FloatLM 1.1B	6.03	3.25	6.8± 0.2	10.5± 0.2	11.7± 0.2	12.2± 0.2	12.7± 0.2
FloatLM 390M	6.28	16	2.8± 0.1	9.6± 0.2	13.0± 0.3	13.7± 0.3	14.1± 0.3
FloatLM 830M	6.55	6	8.5± 0.2	19.3± 0.3	21.8± 0.3	22.6± 0.3	23.3± 0.3
FloatLM 1.1B	7.0	4.25	9.3± 0.2	21.0± 0.3	23.3± 0.3	24.7± 0.3	25.1± 0.3
TriLM 2.4B	7.23	1.58	3.1± 0.1	23.5± 0.3	26.0± 0.3	26.8± 0.3	27.0± 0.3
FloatLM 1.5B	7.55	3.25	4.2± 0.1	12.0± 0.2	15.0± 0.3	15.5± 0.3	16.4± 0.3
FloatLM 830M	7.91	8	8.5± 0.2	19.7± 0.3	22.2± 0.3	23.0± 0.3	23.8± 0.3
FloatLM 1.1B	8.7	6	12.4± 0.2	24.0± 0.3	26.1± 0.3	26.9± 0.3	27.5± 0.3
FloatLM 1.5B	8.86	4.25	9.0± 0.2	24.3± 0.3	27.0± 0.3	27.9± 0.3	29.0± 0.3
FloatLM 560M	9.11	16	4.6± 0.2	13.1± 0.3	16.0± 0.3	16.8± 0.3	18.1± 0.3
FloatLM 1.1B	10.64	8	12.7± 0.2	24.4± 0.3	26.0± 0.3	27.2± 0.3	27.6± 0.3
TriLM 3.9B	10.76	1.58	9.4± 0.2	29.6± 0.3	33.7± 0.4	34.6± 0.4	35.7± 0.4
FloatLM 2.4B	10.95	3.25	10.9± 0.2	18.2± 0.3	21.8± 0.3	23.0± 0.3	23.4± 0.3
FloatLM 1.5B	11.15	6	11.3± 0.2	26.8± 0.3	29.6± 0.3	30.5± 0.3	31.5± 0.3
FloatLM 2.4B	13.18	4.25	21.1± 0.3	32.7± 0.4	35.3± 0.4	36.4± 0.4	36.6± 0.4
FloatLM 830M	13.34	16	8.5± 0.2	19.6± 0.3	22.1± 0.3	23.0± 0.3	23.6± 0.3
FloatLM 1.5B	13.77	8	12.5± 0.2	27.3± 0.3	30.0± 0.3	30.9± 0.3	31.6± 0.3
Pythia 1B	16.18	16	4.2± 0.1	10.5± 0.2	12.8± 0.2	13.5± 0.3	14.0± 0.3
FloatLM 3.9B	16.91	3.25	8.2± 0.2	24.5± 0.3	28.7± 0.3	29.8± 0.3	30.7± 0.3
FloatLM 2.4B	17.09	6	20.4± 0.3	35.7± 0.4	38.4± 0.4	39.3± 0.4	39.7± 0.4
FloatLM 1.1B	18.39	16	12.9± 0.3	24.3± 0.3	26.1± 0.3	27.3± 0.3	27.8± 0.3
FloatLM 3.9B	20.59	4.25	17.9± 0.3	37.5± 0.4	41.9± 0.4	42.8± 0.4	43.6± 0.4
FloatLM 2.4B	21.55	8	20.7± 0.3	35.8± 0.4	38.6± 0.4	39.7± 0.4	39.9± 0.4
Pythia 1.4B	22.62	16	5.6± 0.2	13.7± 0.3	16.9± 0.3	18.1± 0.3	19.2± 0.3
FloatLM 1.5B	24.23	16	12.2± 0.2	27.3± 0.3	30.1± 0.3	31.0± 0.3	31.6± 0.3
FloatLM 3.9B	27.03	6	21.0± 0.3	39.7± 0.4	44.1± 0.4	44.9± 0.4	45.6± 0.4
FloatLM 3.9B	34.39	8	21.7± 0.3	39.8± 0.4	44.5± 0.4	45.0± 0.4	45.7± 0.4
FloatLM 2.4B	39.38	16	20.7± 0.3	36.0± 0.4	38.6± 0.4	39.6± 0.4	39.9± 0.4
Pythia 2.8B	44.39	16	9.9± 0.2	21.8± 0.3	25.7± 0.3	27.2± 0.3	28.3± 0.3
FloatLM 3.9B	63.83	16	21.5± 0.3	39.9± 0.4	44.6± 0.4	45.2± 0.4	45.8± 0.4

Table 6. Knowledge Based Benchmark (TriviaQA) Performance