GQKVA: Efficient Pre-training of Transformers by Grouping Queries, Keys, and Values

Farnoosh Javadi, Walid Ahmed, Habib Hajimolahoseini, Foozhan Ataiefard, Mohammad Hassanpour, Saina Asani, Austin Wen , Omar Mohamed Awad, Kangling Liu, Yang Liu
*Ascend Team, Toronto Research Center, Huawei Technologies
farnoosh.javadi@huawei.com

Abstract

Massive transformer-based models face several challenges, including slow and computationally intensive pre-training and over-parametrization. This paper addresses these challenges by proposing a versatile method called GQKVA, which generalizes query, key, and value grouping techniques. GQKVA is designed to speed up transformer pre-training while reducing the model size. Our experiments with various GQKVA variants highlight a clear trade-off between performance and model size, allowing for customized choices based on resource and time limitations. Our findings also indicate that the conventional multi-head attention approach is not always the best choice, as there are lighter and faster alternatives available. We tested our method on ViT, which achieved an approximate 0.3% increase in accuracy while reducing the model size by about 4% in the task of image classification. Additionally, our most aggressive model reduction experiment resulted in a reduction of approximately 15% in model size, with only around a 1% drop in accuracy.

1 Introduction

Transformers (Vaswani et al., 2023) have dominated RNNs in natural language processing tasks. While CNNs are generally considered the mainstay for various computer vision tasks (Wang et al., 2023; Javadi Fishani, 2020; Ahmed et al., 2023) transformers have demonstrated their competitive capabilities in many instances (Liu et al., 2021; Touvron et al., 2021; Dosovitskiy et al., 2020). Their scalability and power have prompted a trend in the literature which is scaling up the transformer-based models by increasing their parameters and layers to achieve superior performance. However, the expansion of these models has introduced several challenges, such as heavy computation demands and slow pre-training, fine-tuning, and inference process. Furthermore, (Hoffmann et al., 2022) have found out that many of large language models (LLMs) such as GPT3 175B are over-parameterized and inadequately trained, meaning that abundance of parameters doesn't truly translate into enhanced performance. Accordingly, there is a need for techniques that introduce more modestly parameterized transformers to address the issue of over-parametrization.
Studies offer various techniques for efficiently fine-tuning massive transformers such as LORA (Hu et al., 2021; Hajimolahoseini et al., 2022) and prompt-tuning (Lester et al., 2021). The literature is also rich in the direction of speeding up inference of transformers (Ainslie et al., 2023; Shazeer, 2019; Leviathan et al., 2023; Pope et al., 2022). However, accelerating pre-training of transformer-based models is not well studied. One avenue of research in this direction seeks to lower the time complexity of multi-head attention from being quadratic to linear (Choromanski et al., 2022; Han et al., 2023; Wang et al., 2020). This is achieved by introducing a kernel metric that allows for a change in the order of attention computation. However, this direction can lead to accuracy degradation. Another line of research on pre-training acceleration techniques focuses on reducing the number of tokens fed into the transformers (Hou et al., 2022; Yao et al., 2022). It is important to note that both of these directions

37th Conference on Neural Information Processing Systems (NeurIPS 2023).

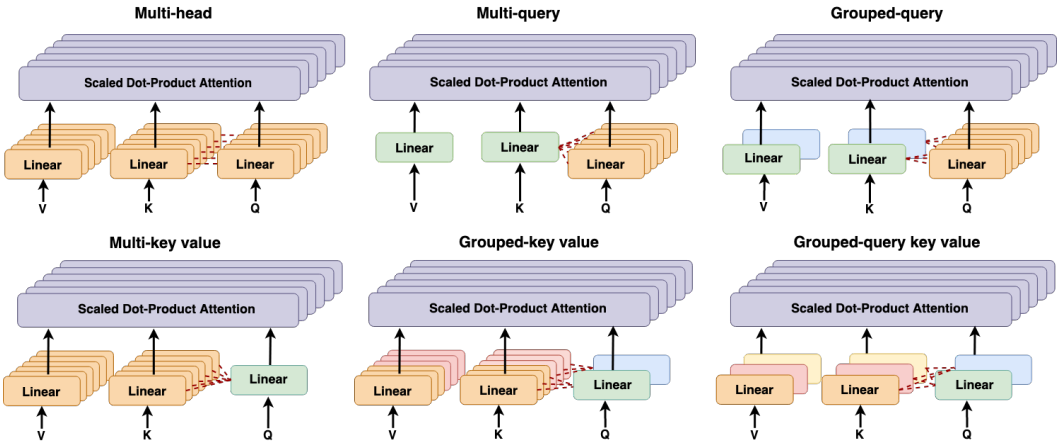


Figure 1: Illustration of various strategies for grouping queries, keys, and values within the attention mechanism, including Vanilla MHA, MQA, GQA, MKVA, GKVA, and GQKVA.

maintain the same model size and do not address the issue of over-parameterization. This paper aims to contribute to the relatively limited literature that proposes methods for both expediting the pre-training of transformers and reducing the model size simultaneously. MQA (Shazeer, 2019) and GQA (Ainslie et al., 2023) have recently shown a promising speedup for speeding up the decoder inference. We propose a more general approach called GQKVA a pre-training acceleration technique. GQKVA partitions queries, keys, and values within the self-attention mechanism to reduce the time needed for attention computation. In our evaluation across vision transformer architecture (Dosovitskiy et al., 2020), we demonstrate that the practice of grouping queries, keys, and values results in expedited training and a more compact model size. Moreover, we analyzed the trade-offs and implications of grouping Q, K, and Vs in-depth, shedding light on its effects on model convergence, and the number of parameters. Our contributions include:

1. Proposing an efficient general attention computation mechanism called GQKVA, which involves grouping queries, keys, and values.
2. Thoroughly exploring various ways of grouping Q, K, V matrices during pre-training, including MQA, GQA, MKVA, GKVA, etc.
3. Obtaining a clear trade-off between performance versus model-size and TPS allows for tailored choices based on resource and time constraints.

2 Method

2.1 Preliminaries

Multi-Head Attention (MHA) employs a set of h distinct attention heads, each ideally specializing in learning unique aspects of the input. For each head, separate query Q, key K, and value V matrices are created by passing the input x through a linear layer of dimensions $d \times 3d$, referred to as the **qkv layer** (d represents the embedding size). The dot-product attention is then computed for each head using Equation 1, resulting in unique outputs for each attention head. These output vectors capture diverse aspects of the input and are concatenated before being fed through a second linear layer, which applies a transformation to the combined output.

$$\text{Attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{softmax}\left(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d}}\right)\mathbf{V} \tag{1}$$

Multi-Query Attention (MQA) initially introduced to enhance inference speed (Shazeer, 2019) is a variation of MHA. In this variant, we still maintain h distinct heads, each equipped with a separate Q matrix. However, there's a key difference: we employ a single shared K and V matrix across all

heads. Consequently, the number of parameters in the qkv layer is reduced to $d \times (d + 2 \times \text{head-dim})$, head-dim $= \frac{d}{\text{number of heads}}$.

Grouped-Query Attention (GQA) introduced for faster inference (Ainslie et al., 2023), partitions the queries into *g* distinct groups, where each group shares a single *K* and *V*. Consequently, in this scheme, we have *h* Q matrices, along with *g* shared K and V matrices. Notably, when *g* equals 1, GQA is equivalent to MQA, while if *g* equals *h*, GQA aligns with the traditional MHA. The parameter count for the qkv layer is determined by: $d \times (d + 2 \times g \times \text{head-dim})$

2.2 Proposed Methods

Multi-Key Value Attention (MKVA) and Grouped Key Value (GKVA). We propose MKVA and GKVA as novel variations, akin to MQA and GQA. However, MKVA and GKVA differ in that they group keys and values into *g* distinct groups instead of queries; while queries are shared within each group. It's essential to emphasize that there exists a one-on-one mapping between keys and values. Therefore, however K matrices are grouped, V matrices must undergo the same grouping to maintain this correspondence. To clarify, when a single Q matrix is shared among all *h* Ks and Vs, we refer to the method as MKVA and when *g* Qs are shared, it's called GKVA.

Grouped-Query Key Value Attention (GQKVA) To further optimize the parameter count and computational efficiency, we propose a comprehensive approach named GQKVA. In GQKVA, we partition the Q matrices into g_q groups and the K, V matrices into g_{kv} groups, where $h = g_q \times g_{kv}$. Subsequently, dot-product attention is computed for each combination of $(Q_i, KV_j)_{i \in [1,g_q], j \in [1,g_{kv}]}$, resulting in *h* distinct outputs similar to the behavior of MHA. It's crucial to note that the usage of Qs and K,Vs must ensure there is no repetition of (Q, KV) pairs to preserve *h* effective heads. Otherwise, we would end up with identical outputs from the dot-product attention, diminishing the model's capacity. GQKVA serves as a unifying generalization of all the methods discussed earlier. For instance, with a single Q matrix shared among all *h* K,Vs, the model corresponds to MKVA. If there are *g* K,Vs shared among all queries, we have GQA-g. This approach offers versatility while optimizing parameterization and computational efficiency.

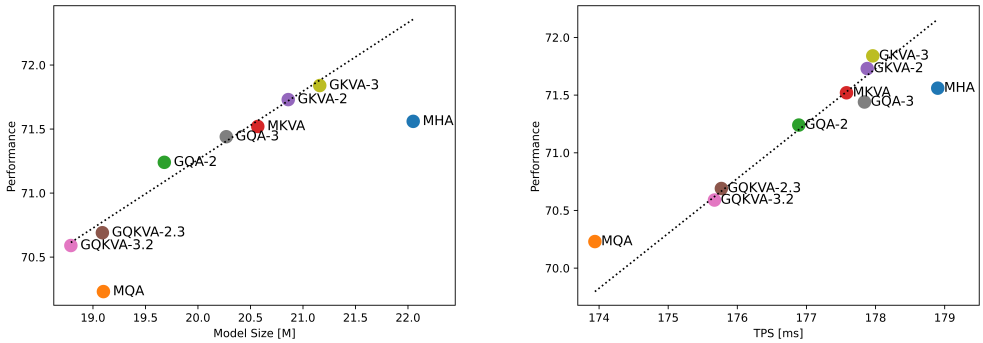
3 Experiments

We assessed the performance of the mentioned methods on ViT (Dosovitskiy et al., 2020) for the task of image classification. We trained ViT-small with 6 heads and 22 million parameters from scratch. The training process was carried out on six 32GB V100 GPU cores using data parallelism for 300 epochs, and a batch size of 288. We used AdamW (Loshchilov and Hutter, 2017) optimizer and an initial learning rate of 0.001. Through an extensive series of experiments, we compared the following methods: MHA, MQA, GQA-2 (two sets of K,V matrices shared among queries), GQA-3, MKVA, GKVA-2 (two groups of Qs shared among all keys and values), GKVA-3, GQKVA-2.3 (two Q matrices and three K,V matrices), and GQKVA-3.2. The summarized results are presented in Table 1.

Our findings indicate that GKVA-2 and GKVA-3 achieved the highest accuracy, reaching 71.73 and 71.84, respectively, while reducing model sizes 5.4% and 4.04% compared to multi-head attention. Moreover, it's seen that GQKVA-2.3 and GQKVA-3.2 have the same or less number of parameters than MQA while outperforming its accuracy respectively by 0.46% and 0.36%. Figure 2a and 2b portray the trade-off between model size and performance, as well as the relationship between time per sample (TPS) and performance, respectively. TPS is measured based on the time required to train a batch. As seen in the figures, both TPS and model size exhibit a linear correlation with performance. Notably, MHA falls below the trend line in both figures, indicating that there are faster and lighter alternatives to MHA and it doesn't necessarily benefit from its larger parameter count. In general, for methods other than MHA and MQA, an increase in model size leads to improved accuracy, while larger models tend to have higher TPS. Therefore, the choice of the most suitable method can be based on resource and time constraints, allowing for a balanced consideration of model size, training speed, and performance.

Table 1: Comparative evaluation of different grouping strategies within the attention mechanism applied on ViT-small as explained in Section 2. TPS represents the time that training a batch takes.

Model	TPS (Time per sample - ms)	Acc-top1 (%)	#Parameters (M)	Model size (mb)
ViT-MHA	178.90	71.56	22.05	84.11
ViT-GKVA-3	177.96 (0.53%)	**71.84**	21.16 (-4.04%)	80.73
ViT-GKVA-2	177.88 (0.57%)	71.73	20.86 (-5.40%)	79.60
ViT-MKVA	177.58 (0.74%)	71.52	20.57 (-6.71%)	78.47
ViT-GQA-3	177.84 (0.59%)	71.44	20.27 (-8.07%)	77.34
ViT-GQA-2	176.89 (1.12%)	71.24	19.68 (-10.75%)	75.09
ViT-MQA	**173.94** (2.77%)	70.23	19.09 (-13.42%)	72.83
ViT-GQKVA-2.3	175.77 (1.75%)	70.69	19.09 (-13.42%)	72.83
ViT-GQKVA-3.2	175.67 (1.82%)	70.59	**18.79 (-14.78%)**	**71.70**



(a) Performance versus Model Size

(b) Performance versus TPS

Figure 2: Both figures highlight the presence of faster and lighter attention mechanisms compared to MHA. They also show performance correlates linearly with model size and TPS.

4 Conclusion

Transformer models usually suffer from huge model sizes and computationally intensive pre-training. In this paper, we introduce a versatile solution named GQKVA, designed to expedite transformer pre-training while simultaneously reducing model sizes. GQKVA serves as a generalization of various Q, K, V grouping techniques encompassing methods like MQA and GQA. Our experiments involving different GQKVA variants unveil a clear trade-off between model size and performance. This trade-off allows practitioners to tailor their choices based on resource constraints and training time limitations. Our results demonstrate that the conventional MHA is not always the optimal choice, as there exist lighter and faster alternatives. It's worth noting that the proposed GQKVA method is general and can be applied to any transformer architecture. However, due to our current time and resource constraints, we limited our exploration to the ViT-small model. Future research should extend these techniques to larger transformers, where the potential for greater speed-up and memory savings awaits discovery.

References

Walid Ahmed, Habib Hajimolahoseini, Austin Wen, and Yang Liu. 2023. Speeding up resnet architecture with layers targeted low rank decomposition. *arXiv preprint arXiv:2309.12412*.

Joshua Ainslie, James Lee-Thorp, Michiel de Jong, Yury Zemlyanskiy, Federico Lebrón, and Sumit Sanghai. 2023. Gqa: Training generalized multi-query transformer models from multi-head checkpoints.

Krzysztof Choromanski, Valerii Likhosherstov, David Dohan, Xingyou Song, Andreea Gane, Tamas Sarlos, Peter Hawkins, Jared Davis, Afroz Mohiuddin, Lukasz Kaiser, David Belanger, Lucy Colwell, and Adrian Weller. 2022. Rethinking attention with performers.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. 2020. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*.

Habib Hajimolahoseini, Walid Ahmed, Mehdi Rezagholizadeh, Vahid Partovinia, and Yang Liu. 2022. Strategies for applying low rank decomposition to transformer-based models. In *36th Conference on Neural Information Processing Systems (NeurIPS2022)*.

Dongchen Han, Xuran Pan, Yizeng Han, Shiji Song, and Gao Huang. 2023. Flatten transformer: Vision transformer using focused linear attention. *arXiv preprint arXiv:2308.00442*.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, Tom Hennigan, Eric Noland, Katie Millican, George van den Driessche, Bogdan Damoc, Aurelia Guy, Simon Osindero, Karen Simonyan, Erich Elsen, Jack W. Rae, Oriol Vinyals, and Laurent Sifre. 2022. Training compute-optimal large language models.

Le Hou, Richard Yuanzhe Pang, Tianyi Zhou, Yuexin Wu, Xinying Song, Xiaodan Song, and Denny Zhou. 2022. Token dropping for efficient bert pretraining.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models.

Farnoosh Javadi Fishani. 2020. *Hierarchical part-based disentanglement of pose and appearance*. Ph.D. thesis, University of British Columbia.

Brian Lester, Rami Al-Rfou, and Noah Constant. 2021. The power of scale for parameter-efficient prompt tuning.

Yaniv Leviathan, Matan Kalman, and Yossi Matias. 2023. Fast inference from transformers via speculative decoding.

Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. 2021. Swin transformer: Hierarchical vision transformer using shifted windows.

Ilya Loshchilov and Frank Hutter. 2017. Fixing weight decay regularization in adam. *CoRR*, abs/1711.05101.

Reiner Pope, Sholto Douglas, Aakanksha Chowdhery, Jacob Devlin, James Bradbury, Anselm Levskaya, Jonathan Heek, Kefan Xiao, Shivani Agrawal, and Jeff Dean. 2022. Efficiently scaling transformer inference.

Noam Shazeer. 2019. Fast transformer decoding: One write-head is all you need.

Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. 2021. Training data-efficient image transformers & distillation through attention. In *International conference on machine learning*, pages 10347–10357. PMLR.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2023. Attention is all you need.

Sinong Wang, Belinda Z Li, Madian Khabsa, Han Fang, and Hao Ma. 2020. Linformer: Self-attention with linear complexity. *arXiv preprint arXiv:2006.04768*.

Wenhai Wang, Jifeng Dai, Zhe Chen, Zhenhang Huang, Zhiqi Li, Xizhou Zhu, Xiaowei Hu, Tong Lu, Lewei Lu, Hongsheng Li, Xiaogang Wang, and Yu Qiao. 2023. Internimage: Exploring large-scale vision foundation models with deformable convolutions.

Zhewei Yao, Xiaoxia Wu, Conglong Li, Connor Holmes, Minjia Zhang, Cheng Li, and Yuxiong He. 2022. Random-ltd: Random and layerwise token dropping brings efficient training for large-scale transformers.