Github repository
ViTyaQua

Numerical Linear Algebra 2023

Final project

ViTyaQua. Model Compression.

Nikita Ligostaev, Nikolay Kalmykov,
Nikita Vasilev, Matvey Skripkin

December 19, 2023

Motivation and Problem Statement

Efficient deployment of Visual Transformer models on edge devices for real-time applications is hindered by high computational costs and memory requirements

We propose optimizing ViT through quantization and Singular Value Decomposition to address these challenges

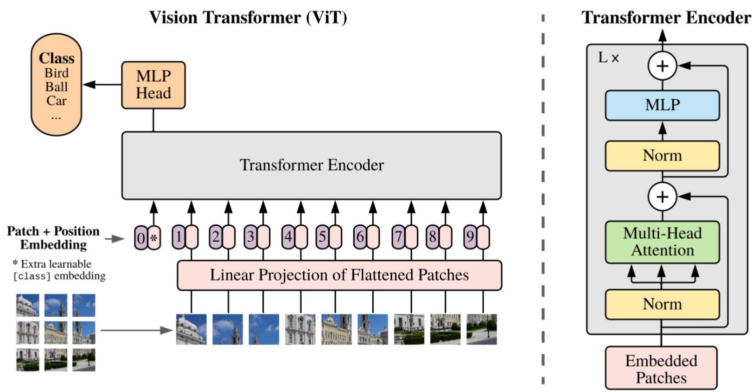


Fig 1. Visual Transformer architecture overview

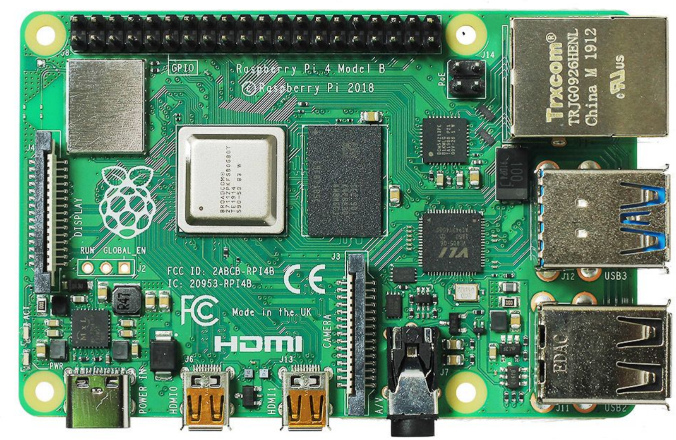


Fig 2. Raspberry PI 4, Broadcom BCM2711,
Quad core Cortex-A72 (ARM v8) 64-bit SoC @ 1.8GHz,
2GB LPDDR4-3200 SDRAM

Current methods of optimization (post-training)

- **Quantization (dynamic)** - represent weights with fewer bits, reducing the precision (convert 32-bit floating-point weights to 8-bit integers)

- **Matrix Decomposition** - decompose high-dimensional tensor into a combination of smaller matrices (SVD.)

- **Pruning** - remove connections (weights) that contribute less to the model's performance.

- etc.

---

1 https://arxiv.org/abs/2101.09671
2 https://link.springer.com/article/10.1007/s10618-019-00619-1

Model Compression

Dynamic Quantization

1) Utilizes Torch framework
2) Minimal loss of model performance
3) Dynamic adjustment of precision allows for a balance between size reduction and accuracy retention

Singular Value Decomposition (SVD)

1) Addresses redundancy in model parameters.
2) Particularly effective for reducing the size of fully connected layers.

Implement SVD in Linear Projection Layer and in MLP in Encoder Layer architecture of ViT.

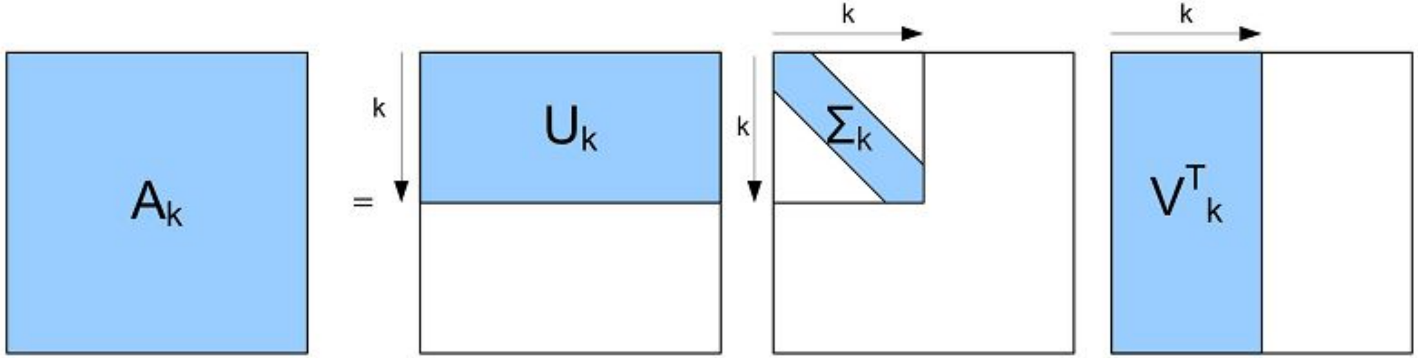
1 https://doi.org/10.48550/arXiv.2201.12220

Proposed method. SVD

Suppose we perform the matrix multiplication $Y = XW^T + b$, where $X \in \mathbb{R}^{m \times p}$, $W \in \mathbb{R}^{n \times p}$, and $b \in \mathbb{R}^n$ resulting in $Y \in \mathbb{R}^{m \times n}$.
We can decompose $W = U\Sigma V^T$, where $U \in \mathbb{R}^{n \times n}$ and $V \in \mathbb{R}^{p \times p}$ are orthogonal matrices.
Σ is a matrix of singular values. We consider only $r < min(p, n)$ singular values to reconstruct matrix $W =>$ $U_r = \mathbb{R}^{n \times r}$, $V_r \in \mathbb{R}^{p \times r}$ and $\Sigma \in \mathbb{R}_+^{r \times r}$. We transpose matrix of weights. Thus, $W^T \in \mathbb{R}^{p \times n}$.

In this case we can rewrite Y as:

$$Y = XV_r \Sigma U_r^T + b$$



Proposed method. Linear quantization

Quantization maps a floating-point value $x \in [\alpha, \beta]$ to a b-bit integer $x_q \in [\alpha_q, \beta_q]$.

The de-quantization process is defined as $x = s(x_q + z)$, and the quantization process is defined as $x_q = \mathrm{round}\left(\frac{1}{s}x - z\right)$, where c and d are variables.

In practice, the quantization process may produce x outside the range $[\alpha, \beta]$, that is why clipping is introduced:

$$x_q = \mathrm{clip}\left(\mathrm{round}\left(\frac{1}{s}x + z\right), \alpha_q, \beta_q\right)$$

where $\mathrm{clip}(x, l, u)$ is defined as:

$$\mathrm{clip}(x, l, u) = \begin{cases} l & \text{if } x < l \\ x & \text{if } l \leq x \leq u \\ u & \text{if } x > u \end{cases}$$

Conducted experiments

CIFAR10 dataset

Task: image classification

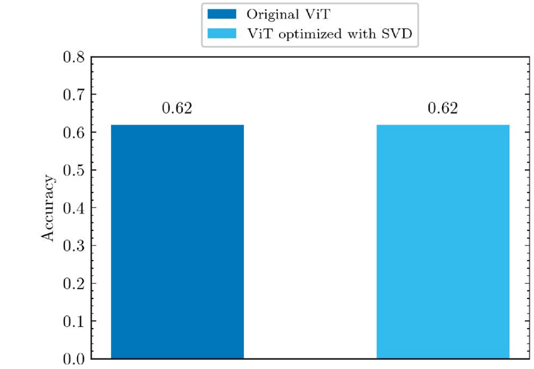
Image size: 32x32 color images

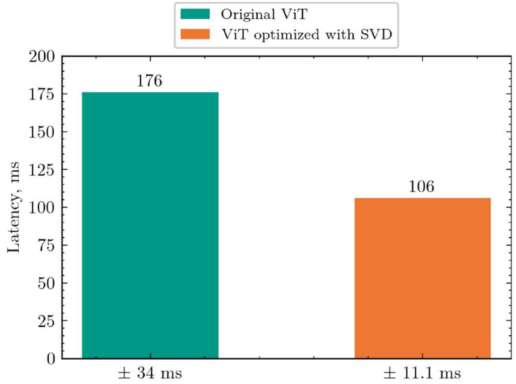
Number of classes: 10

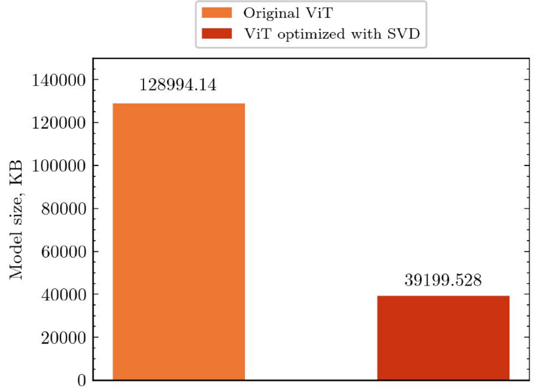
Number objects: Train: 50000, Test: 10000

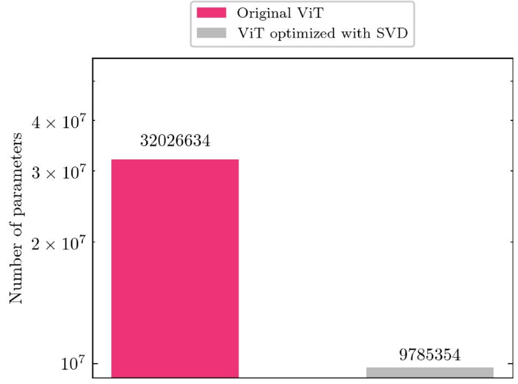
Fig 4. Samples from CIFAR10 dataset

Obtained Results. SVD



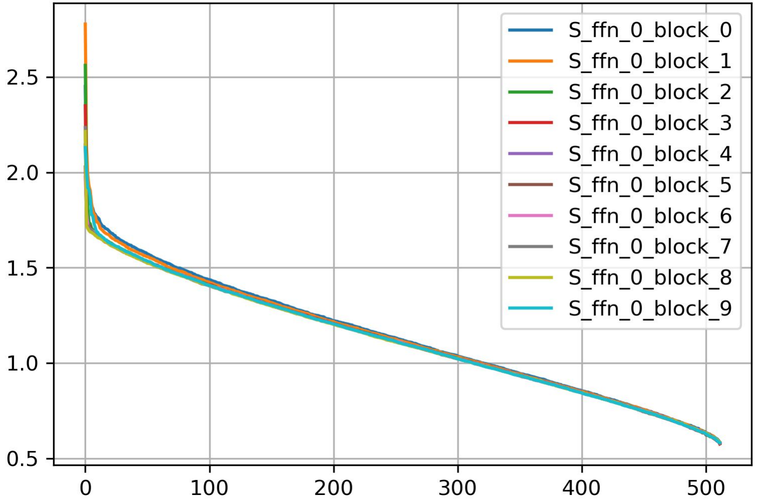


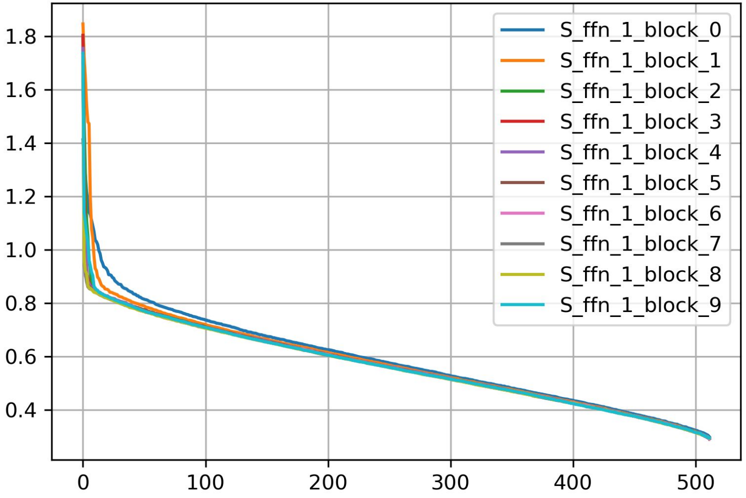




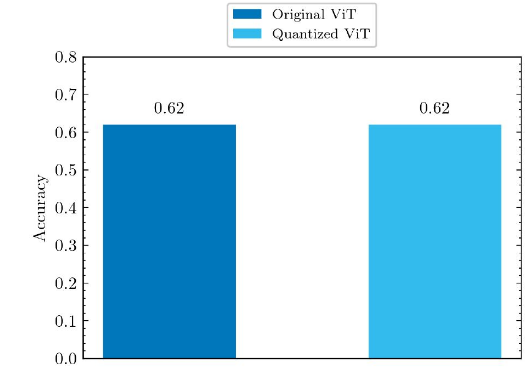
Parameter	Original ViT	ViT optimized with SVD
Accuracy	0.62	0.62
Latency	176 s ± 34 ms	106 ms ± 11.1 ms
Model size, KB	128 994.146	39 199.528
Number of parameters	32 026 634	9 785 354

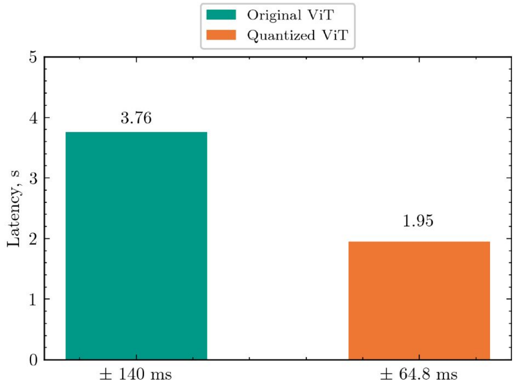
Obtained Results. Singular values of linear layers. SVD

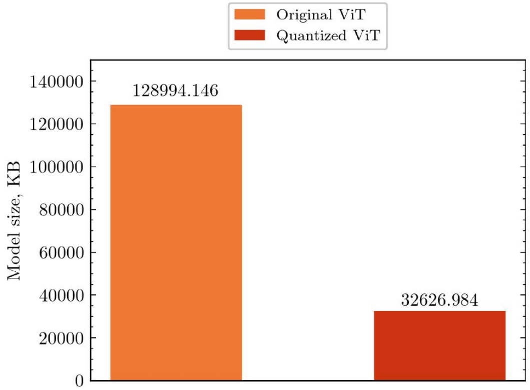


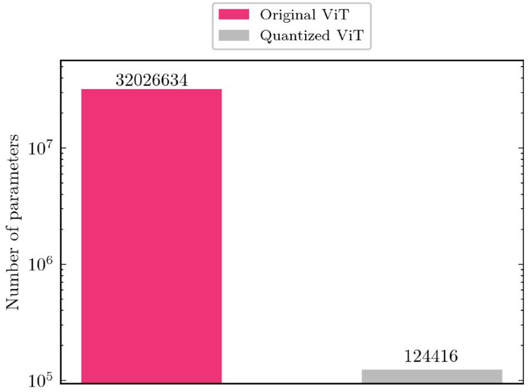


Obtained Results. Dynamic quantization









Parameter	Original ViT	Quantized ViT
Accuracy	0.62	0.62
Latency	3.76 s ± 140 ms	1.95 s ± 64.8 ms
Model size, KB	128 994.146	32 626.984
Number of parameters	32 026 634	124 416

Conclusion

In this study, we explored advanced compression techniques tailored for Visual Transformer models (ViT), demonstrating significant model size reduction without compromising performance:

Baseline ViT Performance:

- - Trained ViT on the CIFAR-10 dataset for image classification, achieved an accuracy of 0.62 on inference.

Singular Value Decomposition (SVD):

- - Applied singular value decomposition (SVD) to linear layers of ViT, specifically targeting projection layers and linear layers in the feed-forward neural network (FFNN) encoder layer.
- - Attained comparable accuracy on the quantized model, while reducing the model size to approximately 3.3 times smaller than the original.

Dynamic Quantization:

- - Maintained high performance with the quantized model achieving the same accuracy as the original, while reducing the model size by a factor of 4.

Contributions

- **Nikita Vasilev**
  - Prepared the presentation
  - Prepared GitHub repository

- **Nikita Ligostaev**
  - Conducted experiments with dynamic, static quantization PyTorch
  - Implemented static quantization to linear layers of ViT on PyTorch
  - Prepared the presentation
  - Prepared GitHub repository

Nikolaus Kalmykov

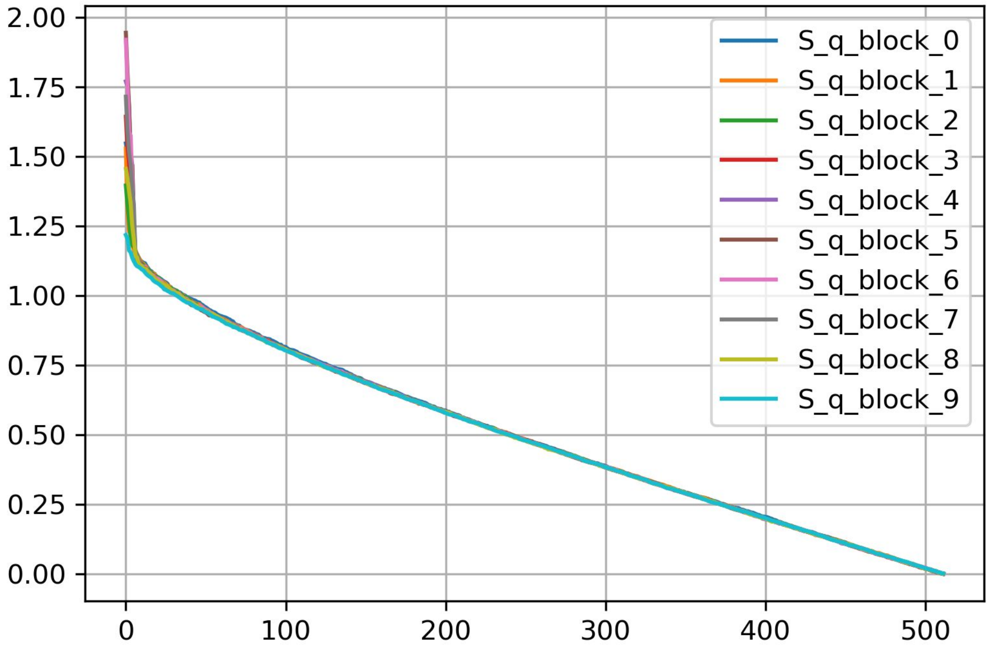
- Conducted experiments with SVD optimization on ViT
- Prepared the presentation
- Prepared GitHub repository

Matvey Skripkin

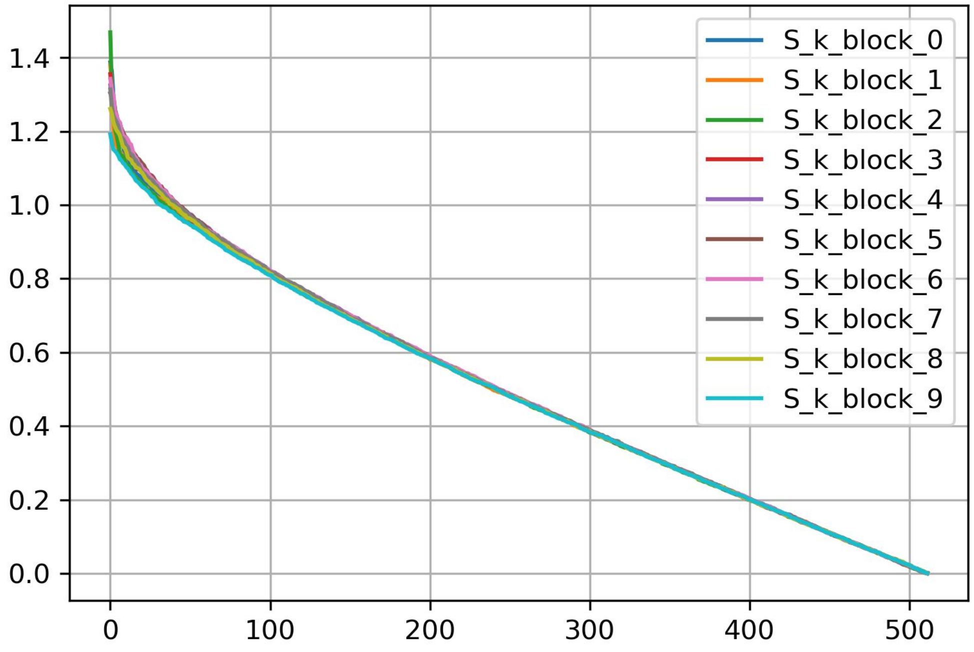
- Implemented ViT on PyTorch
- Implemented SVD optimization for ViT on PyTorch
- Prepared the presentation
- Prepared GitHub repository

Backslides

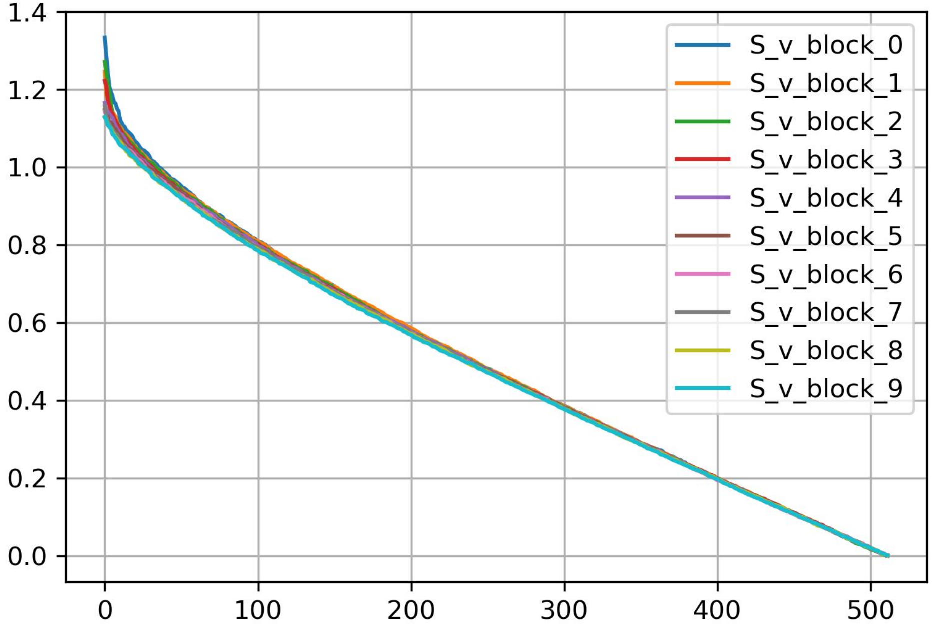
Obtained Results. Singular values of linear layers. SVD



Obtained Results. Singular values of linear layers. SVD



Obtained Results. Singular values of linear layers. SVD



Obtained Results. Singular values of linear layers. SVD