Dataset Quantization

Daquan Zhou [1*] Kai Wang[2*] Jianyang Gu[2*] Xiangyu Peng[2] Dongze Lian[2]
Yifan Zhang[2] Yang You[2†] Jiashi Feng[1†]
[1]Bytedance Inc. [2]National University of Singapore
daquan.zhou@u.nus.edu kai.wang@comp.nus.edu.sg gu_jianyang@zju.edu.cn
youy@comp.nus.edu.sg jshfeng@bytedance.com
Code: https://github.com/magic-research/Dataset_Quantization

1. Appendix

We present more explanations of the proposed dataset quantization, experiment results and visualizations in this section.

1.1. Proof of Sec. 3.1

Given the whole dataset $\mathbf{D}$, $|\mathbf{D}| = M \gg 1$. $\forall p \in \mathbf{D}$, $f(p) \in \mathbb{R}^{m\times 1}$. To make a simple proof, we assume $\frac{1}{M}\sum_{p\in\mathbf{D}} f(p) = 0$.

For $n = 0, 1, 2, \cdots, n, \cdots, N$, define set $\mathbf{S}_n \in \mathbf{D}$. And, we define $C_1(x)$ and $C_2(x)$ as follows,

$$C_1(x) = \sum_{p\in\mathbf{S}_1^n} ||f(p) - f(x)||_2^2; \qquad C_2(x) = \sum_{p\in\mathbf{D}\setminus\mathbf{S}_1^i} ||f(p) - f(x)||_2^2; \tag{1}$$

By the policy of GraphCut [4], it aims to maximize $C_1(x)$ and minimize $C_2(x)$ to select $\mathbf{S}_1$. We write it into a united target function to choose x_{k+1} as,

$$x_{k+1} \leftarrow \arg\max_{x\in\mathbf{D}\setminus\mathbf{S}_1^k}(C_1(x) - C_2(x)). \tag{2}$$

We initialize $\mathbf{S}_1^1$ using $\emptyset$ and $\mathbf{S}_1^{k+1} = \mathbf{S}_1^k \cup x_{k+1}$, where $k = 1, 2, \cdots, k, \cdots, K$, and $|\mathbf{S}_1^k| = k$.

Claim: (a). $\mathbf{S}_1^1 = \emptyset$, $x_1 = \arg\min_{x\in\mathbf{D}} ||x||_2^2$, *i.e,* the closest point to $\mathbf{0}$ in $\mathbf{D}$

(b). x_{k+1} is very close to set $\mathbf{S}_1^k$.

Proof: $\mathbf{S}_1^1 = \emptyset$, so $C_1(x) = 0$.

$$\begin{aligned} C_2(x) &= \sum_{p\in\mathbf{D}} ||f(p) - f(x)||_2^2 &(3)\\ &= M||f(x)||_2^2 - 2(\sum_{p\in\mathbf{D}} f(p))^\top f(x) + \sum_{p\in\mathbf{D}} ||f(p)||_2^2 &(4)\\ &= M||f(x)||_2^2 + \sum_{p\in\mathbf{D}} ||f(p)||_2^2, &(5)\end{aligned}$$

where $\sum_{p\in\mathbf{D}} f(p) = 0$.

*Equal first author.
†Corresponding author.

Then, we have

$$x_1 = \arg\max_{x\in\mathbf{D}} -C_2(x) \tag{6}$$
$$= \arg\min_{x\in\mathbf{D}} C_2(x) \tag{7}$$
$$= \arg\min_{x\in\mathbf{D}} M||f(x)||_2^2, \tag{8}$$

(b) Let $C_1(x_k) - C_2(x_k) = 2C_1(x_k) - (C_2(x_k) + C_1(x_k))$. We have:

$$(C_2(x_k) + C_1(x_k)) = \sum_{p\in\mathbf{D}\setminus\mathbf{S}_1^k} ||f(p) - f(x_k)||_2^2 + \sum_{p\in\mathbf{S}_1^k} ||f(p) - f(x_k)||_2^2 \tag{9}$$
$$= \sum_{p\in\mathbf{D}} ||f(p) - f(x)||_2^2 \tag{10}$$
$$= M||f(x)||_2^2 + \sum_{p\in\mathbf{D}} ||f(p)||_2^2 \tag{11}$$
$$= M||f(x)||_2^2 + Const., \tag{12}$$

where 'Const.' denotes constant number.

For $C_1(x_k)$, we have

$$C_1(x_k) = \sum_{p\in\mathbf{S}_1^k} ||f(p) - f(x_k)||_2^2 = k||f(x_k)||_2^2 - 2(\sum_{p\in\mathbf{S}_1^k} f(p)^\top f(x_k) + \sum_{p\in\mathbf{S}_1^k} ||f(p)||_2^2 \tag{13}$$

Define $Q_k = \frac{1}{k}\sum_{p\in\mathbf{S}_1^k} f(p)$ as the weighted center of $\mathbf{S}_1^k$. Then, we can write the submodular gains function as follows,

$$P(x_k) = 2C_1(x_k) - (C_2(x_k) + C_1(x_k)) \tag{14}$$
$$= 2k||f(x_k)||_2^2 - 4kQ_k^\top f(x_k) - M||f(x_k)||_2^2 + Const. \tag{15}$$
$$= (2k - M)||f(x_k) - \frac{2kQ_k}{2k - M}||_2^2 + Const. \tag{16}$$

x_{k+1} is selected as follows,

$$x_{k+1} = \arg\max_{x\in\mathbf{D}\setminus\mathbf{S}_1^k} P(x_k) = \arg\max_{x\in\mathbf{D}\setminus\mathbf{S}_1^k} ||f(x_k) - \frac{2kC_k}{2k - M}||_2^2. \tag{17}$$

Let $\delta_k = \frac{2kC_k}{2k-M}$. We define radius R_1^k of set $\mathbf{S_1^k}$ as,

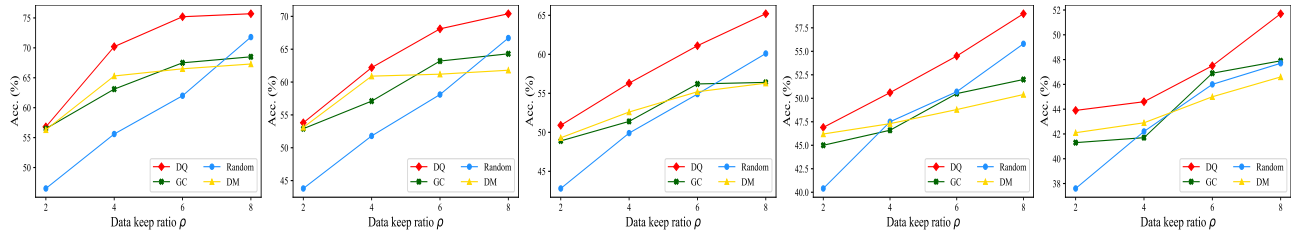
$$R_1^k = max_{p\in\mathbf{S}_1^k} ||f(p)||_2. \tag{18}$$

Therefore, $\forall p \in \mathbf{S}_1^k$, $||f(p)||_2^2 \leq (R_1^k)^2$, which means $\mathbf{S}_1^k$ is included in a ball $\mathbf{B}_k = \{p|||f(p)||_2^2 \leq (R_1^k)^2\}$. Note that,

$$||\delta_k||_2^2 = (2k/2k - M)^2||Q_k||_2^2 \tag{19}$$
$$= (\frac{2k}{2k - M})^2||\frac{1}{k}\sum_{p\in\mathbf{S}_1^k} f(p)||_2^2 \tag{20}$$
$$\leq (\frac{2k}{2k - M})^2\frac{1}{k}\sum_{p\in\mathbf{S}_1^k} ||f(p)||_2^2 \tag{21}$$
$$\leq (\frac{2k}{2k - M})^2(R_1^k)^2. \tag{22}$$

$M \gg k$, so $||\delta_k||_2^2 \leq (R_1^k)^2$ and $\delta_k \in \mathbf{B}_k$ According to Eq. 17, $x_{k+1} = \arg\min_{x\in\mathbf{D}\setminus\mathbf{S}_1^k} ||x - \delta_k||_2^2$ is the closest point in $\mathbf{D}\setminus\mathbf{S}_1^k$ to δ_k, which is in the ball $\mathbf{B}_k$. As $M \gg 1$, $f(x_{k+1})$ is very close to $\mathbf{B}_k$, and thus to $\mathbf{S}_1^k$.

By the proof, GraphCut cannot guarantee the samples diversity under small data keep ratio. Our DQ recursively select samples from $\mathbf{D}$, as the total number of $\mathbf{D}$ reduces, the radius of the ball $\mathbf{B_k}$ will be extended. Therefore the sample diversity is higher than GraphCut method.



(a) Level 1 (b) Level 2 (c) Level 3 (d) Level 4 (e) Level 5

Figure 1: Comparisons of the robustness of trained models via DQ, GC and Random selection on CIFAR10-C dataset.

1.2. Details of Patch Dropping and Reconstruction

As pointed out in Masked Auto-Encoder (MAE) [2], with a pre-trained decoder, some image patches can be dropped without affecting the reconstruction quality of the image. Motivated by it, we propose to reduce the number of pixels utilized for describing each image. Specifically, as shown in pipeline, given an image x, we first feed it into a pretrained feature extractor (ResNet-18 [3]) to obtain the last feature map $\mathcal{M}$ and a prediction score y^c of the image class c. A group of attention scores is then calculated with the gradient values of each pixel in the last feature map following GradCAM++ [1]:

$$a^c = \sum_{i,j} \left[\frac{\frac{\partial^2 y^c}{(\partial \mathcal{M}_{ij})^2}}{2\frac{\partial^2 y^c}{(\partial \mathcal{M}_{ij})^2} + \sum_{m,n} \mathcal{M}_{mn} \{ \frac{\partial^3 y^c}{(\partial \mathcal{M}_{ij})^3} \}} \right] \text{ReLU}\left(\frac{\partial y^c}{\partial \mathcal{M}_{ij}} \right), \tag{23}$$

where a^c is the attention scores for each pixel w.r.t. class c, ReLU is the Rectified Linear Unit activation function, and (i, j) and (m, n) are iterators over the feature map A. The pixel-wise attention score a^c is upsampled to fully cover the original input image. In order to integrate the attention information into image patches, we unify the attention scores of the corresponding pixels of a patch by their average value to generate the patch-wise importance scores p^c as follows,

$$p_k^c = \frac{1}{hw} \sum_{i=h_k}^{h_k+h} \sum_{j=w_k}^{w_k+w} a^c(i,j), \tag{24}$$

where h_k and w_k are the coordinates of the upper left corner of the patch k, and h and w are the height and width of image patches. According to the patch-wise attention scores, we drop a percentage of θ non-informative patches with smallest attention scores to further save the storage cost. At the training stage, we employ a strong pre-trained MAE decoder to reconstruct the dropped patches and the original images.

1.3. Robustness Evaluation

We show the overall robustness evaluation in our paper. Here, we report the detailed results at different corruption levels in Fig. 1. Our proposed DQ achieves state-of-the-art results in all cases.

1.4. Differences between coreset selection and dataset quantization

Coreset VS DQ We here give more detailed explanations on the difference between the coreset selection methods and our proposed dataset quantization. As shown in Fig. 2, the coreset selection only select one subset from the full data distribution. This practice will suffer from a selection bias, resulting in selection results with limited diversity. Besides, when the the size of the selected subset is small, it will suffer a large selection variance. Differently, dataset quantization first divides the full distribution into non-overlapping bins and then sampling from each bin uniformaly. As a result, the sampled data could maximally preserve the original data distribution. To verify this, we use GraphCut [4] as a representation of the coreset based method and 10% and 20% data from ImageNet dataset and compare the results with the data distribution sampled with dataset quantization. We use a pre-trained ResNet-18 model to extract the features of the data and then visualize the extracted data via t-SNE. The results are shown in Fig. 3. It is clearly observed that the data sampled via dataset quantization do capture a more diverse distribution.

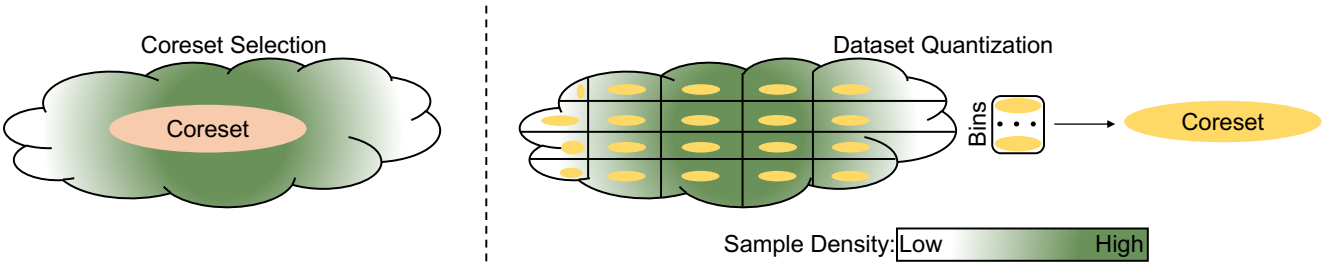


Figure 2: Differences between coreset selection methods and our dataset quantization.

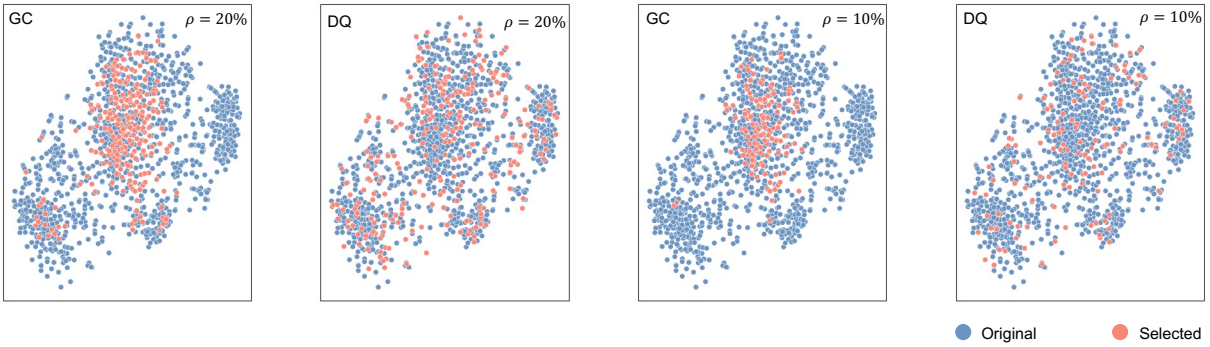


Figure 3: Visualization of the feature distributions among data selected by GraphCut and SQ.

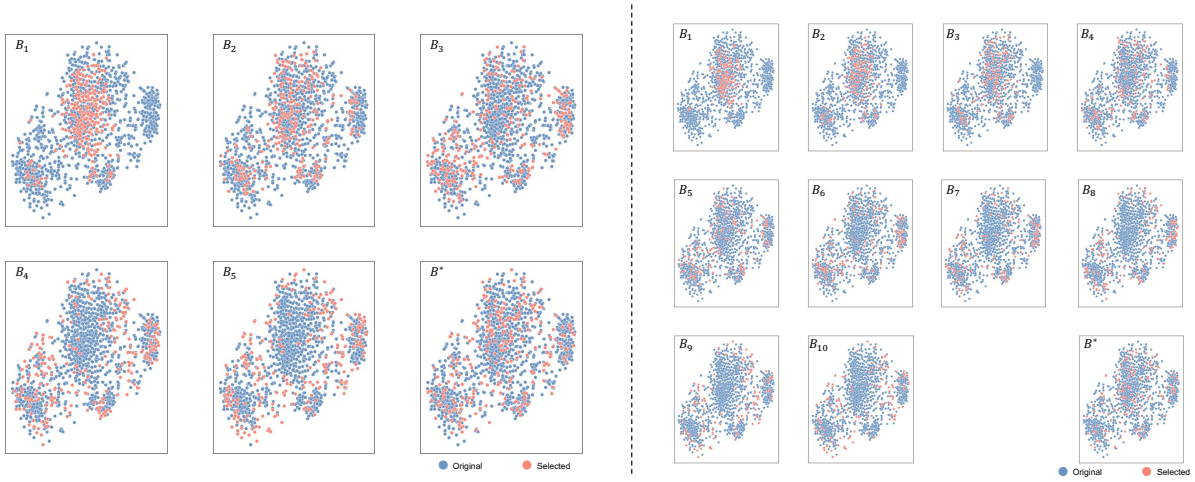


Figure 4: Visualization of the feature distributions among data selected in each bin and the final output of SQ on ImageNet dataset tench class. The bin number N and the data keep ratio ρ are set as (5, 20), (10, 10), respectively for the left and right column.

Bin diversity of DQ To dig deeper for the reason why DQ can better preserved the data distribution. We use the same visualization method as aforementioned for the data contained within each bin. The results are shown in Fig. 4. Each bin contains 20% of the total data in the left column and 10% data in the right column. As shown, different bins are capturing different distributions. As a results, after sampling uniformly from each bin, the combined dataset enjoys a large diversity as well as representativeness over the whole data distribution.

Cross-architecture generalization of DQ We further present more feature distribution visualizations with different network architectures on ImageNet-1K in Fig. 5. The samples are originally selected by ResNet-18 and reconstructed with MAE. Each set contains 10% of the total data. As shown, across all architectures, the generated compact set can effectively cover the whole data distribution, presenting significant cross-architecture generalization capability.

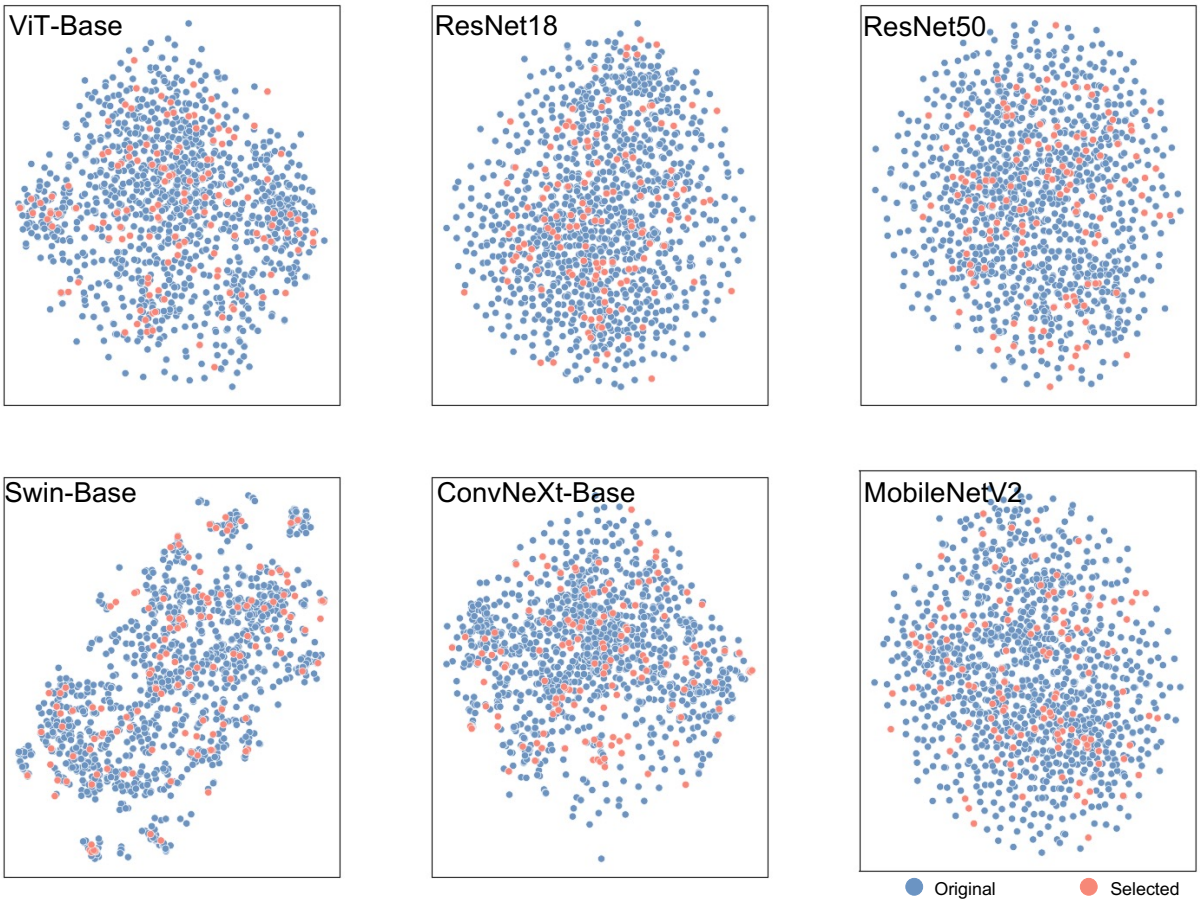


Figure 5: Cross-architecture visualiztaion of the feature distributions among the dataset generated by DQ on ViT-Base on ImageNet dataset tench class.

References

[1] C Aditya, S Anirban, D Abhishek, and H Prantik. Grad-cam++: Improved visual explanations for deep convolutional networks. *arXiv preprint arXiv:1710.11063*, 2017. 3

[2] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16000–16009, 2022. 3

[3] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 3

[4] Rishabh Iyer, Ninad Khargoankar, Jeff Bilmes, and Himanshu Asanani. Submodular combinatorial information measures with applications in machine learning. In *Algorithmic Learning Theory*, pages 722–754. PMLR, 2021. 1, 3