

# Adaptive Data-Free Quantization

Biao Qian, Yang Wang\*, Richang Hong, Meng Wang

Key Laboratory of Knowledge Engineering with Big Data, Ministry of Education,  
School of Computer Science and Information Engineering,  
Hefei University of Technology, China

yangwang@hfut.edu.cn, {hfutqian, hongrc.hfut, eric.mengwang}@gmail.com

## Abstract

*Data-free quantization (DFQ) recovers the performance of quantized network ( $Q$ ) without the original data, but generates the fake sample via a generator ( $G$ ) by learning from full-precision network ( $P$ ), which, however, is totally independent of  $Q$ , overlooking the adaptability of the knowledge from generated samples, i.e., informative or not to the learning process of  $Q$ , resulting into the overflow of generalization error. Building on this, several critical questions — how to measure the sample adaptability to  $Q$  under varied bit-width scenarios? whether the largest adaptability is the best? how to generate the samples with adaptive adaptability to improve  $Q$ 's generalization? To answer the above questions, in this paper, we propose an Adaptive Data-Free Quantization (AdaDFQ) method, which revisits DFQ from a zero-sum game perspective upon the sample adaptability between two players — a generator and a quantized network. Following this viewpoint, we further define the disagreement and agreement samples to form two boundaries, where the margin between two boundaries is optimized to adaptively regulate the adaptability of generated samples to  $Q$ , so as to address the over-and-under fitting issues. Our AdaDFQ reveals: 1) the largest adaptability is NOT the best for sample generation to benefit  $Q$ 's generalization; 2) the knowledge of the generated sample should not be informative to  $Q$  only, but also related to the category and distribution information of the training data for  $P$ . The theoretical and empirical analysis validate the advantages of AdaDFQ over the state-of-the-arts. Our code is available at <https://github.com/hfutqian/AdaDFQ>.*

## 1. Introduction

Deep Neural Networks (DNNs) have encountered great challenges when applied to the resource-constrained devices, owing to the increasing demands for computing and storage resources. Network quantization [6, 10], a promising approach to improve the efficiency of DNNs, reduces

\*Yang Wang is the corresponding author.

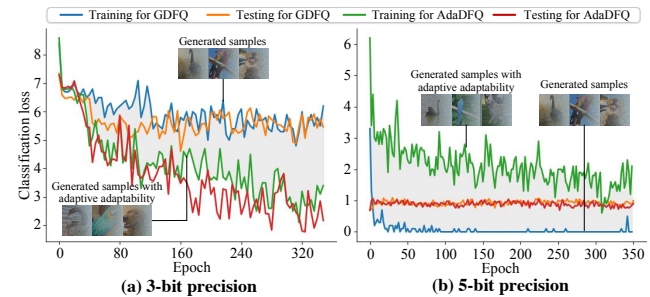


Figure 1. Existing work, e.g., GDFQ [22] (the blue), generally suffers from (a) underfitting issue (both training and testing loss are large) under 3-bit precision and (b) overfitting issue (training loss is small while testing loss is large) under 5-bit precision<sup>1</sup>. Our AdaDFQ (the green) generates the sample with adaptive adaptability to  $Q$ , yielding better generalization of  $Q$  with varied bit widths. The observations are from MobileNetV2 on ImageNet.

the model size by mapping the floating-point weights and activations to low-bit ones. Quantization methods generally recover the performance loss from the quantization errors, such as fine-tuning or calibration operations with the original training data.

However, the original data may not be accessible due to privacy and security issues. To this end, data-free quantization (DFQ) has come up to quantize models without the original data by synthesizing meaningful fake samples, where the quantized network ( $Q$ ) is improved by distilling the knowledge from the pre-trained full-precision model ( $P$ ) [5, 18]. Among the existing arts [1, 23], the increasing attention has recently transformed to the generative models [3, 22, 25], which generally captures the distribution of the original data from  $P$  by utilizing the generative model as a generator ( $G$ ), where  $P$  serves as the discriminator to guide the generation process [22]. To narrow the gap between the synthetic and real data, [3] proposes to restore the decision boundaries via boundary supporting samples; while [25] further optimizes the generator architecture for

<sup>1</sup>3-bit and 5-bit precision are representative for low-bit and high-bit cases, respectively, particularly: 3-bit quantization actually leads to a huge performance loss, which is a major challenge for the existing DFQ methods; while 5-bit or higher-bit quantization usually causes a small performance loss, which is selected to validate the generalization ability.

Table 1. Comparison with the existing DFQ methods. Our AdaDFQ aims to generate the sample with adaptive adaptability to Q with varied bit widths, especially low-bit situation.

Method	Generated sample type	Dependence on Q	Access to low-bit situation
GDFQ [22]	Reconstructed	No	No
Qimera [3]	Reconstructed	No	No
ZAQ [11]	Adversarial	Yes	No
AdaDFQ	Adaptive adaptability	Yes	Yes

better fake samples via a neural architecture search method. Nevertheless, it still suffers from a non-ignorable performance loss under various bit-width settings, stemming from the followings:

- (1) Bearing the limited capacity of the generator, it is impossible for the generated sample with incomplete distribution to fully recover the original dataset, hence a natural question raises up as: whether the knowledge by the generated sample is *informative* or not to benefit the learning process to Q? However, the generated sample by the prior arts customized for P, fail to benefit Q with varied bit-width settings (e.g., 3-bit or 5-bit precision), where only limited information from P can be exploited to recover Q.
- (2) In low-bit precision (e.g., 3 bit), Q suffers from a sharp accuracy drop upon P due to large quantization error, resulting into its poor learning performance. Following that, the generated sample by G may incur a large *disagreement* between the predictions of P and Q, resulting the optimization loss into too large to converge, yielding an *underfitting* issue; see Fig.1(a).
- (3) In high-bit precision (e.g., 5 bit), Q possesses comparable recognition ability with P due to a small accuracy drop. Therefore, most of the generated samples by G, for which Q and P give similar predictions, may not benefit Q. However, trapped by the optimization loss, Q receives no improvement, resulting in an *overfitting* issue; see Fig.1(b).

The above conveys to us that the existing arts overlook the *sample adaptability*, i.e., informative or not to Q, to Q with varied bit widths during the generation process from G, where Q is independent of the generation process; see Table 1. Intuitively, (2) may crave the sample with large agreement between P and Q to avoid the underfitting issue; while for (3), it may be one with large disagreement to avoid the overfitting issue. Such fact promotes us to delve into the following questions: *how to measure the sample adaptability to Q under varied bit-width scenarios? whether the largest adaptability is the best? how to generate the samples with adaptive adaptability to benefit Q’s generalization?*

To answer the above questions, we attempt to generate the samples with large adaptability to Q by taking Q into account during the generation process; see Table 1. Following that, [17] first reformulates the DFQ as a zero-sum

game over sample adaptability between two players — a generator and a quantized network, where one player’s reward is the other’s loss while their sum is zero. Specifically, G generates the sample with large adaptability by enlarging the disagreement between P and Q, to benefit Q; while Q is calibrated to be improved by exploiting the sample with large adaptability. Such process of benefiting Q essentially decreases the sample adaptability, which is adversarial to increasing the sample adaptability for G. However, [17] fails to reveal the underlying *underfitting* (*overfitting*) issues incurred by the sample with the *largest* (*lowest*) adaptability upon the zero-sum game process.

To address the above issues, we define two types of samples: *disagreement* (i.e., P can predict correctly but Q not) and *agreement* (i.e., P and Q have the same prediction) samples, to form the lower and upper boundaries to be balanced. The margin between two boundaries is optimized to adaptively regulate the sample adaptability under the constraint of over-and-under fitting issues, so as to generate the samples with *adaptive* adaptability within these two boundaries to Q, with an Adaptive Data-Free Quantization (AdaDFQ) method. We further conduct the theoretical analysis on the generalization of Q, which reveals: the generated sample with the largest adaptability is NOT the best for Q’s generalization; the knowledge carried by the generated sample should not only be informative to Q, but also related to the category and distribution information of the training data for P. We remark that AdaDFQ optimizes the margin to generate the desirable samples upon the zero-sum game, aiming to address the *over-and-under fitting* issues; while AdaSG [17] focuses primarily on the zero-sum game framework for DFQ, which serves as a *special* case of AdaDFQ in spirit. The theoretical analysis and empirical studies validate the superiority of AdaDFQ to the state-of-the-arts.

## 2. Adaptive Data-Free Quantization

Conventional generative data-free quantization (DFQ) approaches [3, 22, 25] reconstruct the training data via a generator (G) by extracting the knowledge (i.e., the class distribution information about the original data) from a pre-trained full-precision network (P), to recover the performance of quantized network (Q) by the calibration operation. However, we observe that Q is independent of the generation process by the existing arts and whether the knowledge carried by generated samples is informative or not to Q, namely *sample adaptability*, is crucial to Q’s generalization. Before shedding light on our method, we first focus on the sample adaptability to Q, followed by how to reformulate the DFQ as a dynamic zero-sum game process over sample adaptability, as discussed in the next.

### 2.1. How to Measure the Sample Adaptability to Quantized Network?

To measure the sample adaptability to Q, we focus primarily on: 1) the dependence on Q; 2) the advantage of P

over Q; and 3) the disagreement between P and Q. Following [17], we first define the disagreement sample below:

**Definition 1** (Disagreement Sample). *Given a random noise vector  $z \sim N(0, 1)$  and an arbitrary one-hot label  $y$ ,  $G$  generates a sample  $x = G(z|y)$ . Then the logit outputs of  $P$  and  $Q$  are given as  $z_p = P(x)$  and  $z_q = Q(x)$ , respectively. Suppose that  $x$  can be correctly predicted by  $P$ , i.e.,  $\operatorname{argmax}(z_p) = \operatorname{argmax}(y)$ , where  $\operatorname{argmax}(\cdot)$  returns the class index corresponding to maximum value. We determine  $x$  to be the disagreement sample provided  $\operatorname{argmax}(z_p) \neq \operatorname{argmax}(z_q)$ .*

Thus the disagreement between P and Q is encoded by the following probability vector:

$$p_{ds} = \operatorname{softmax}(z_p - z_q) \in \mathbb{R}^C, \quad (1)$$

where  $C$  represents the number of class;  $p_{ds}(c) = \frac{\exp(z_p(c) - z_q(c))}{\sum_{j=1}^C \exp(z_p(j) - z_q(j))}$  ( $c \in \{1, 2, \dots, C\}$ ) denotes the  $c$ -th entry of  $p_{ds}$  as the probability that  $x$  is identified as the disagreement sample of the  $c$ -th class. Based on  $p_{ds}$ , the disagreement is calculated via the information entropy function  $\mathcal{H}_{info}(\cdot)$ , formulated as

$$\mathcal{H}_{info}(p_{ds}) = \sum_{c=1}^C p_{ds}(c) \log \frac{1}{p_{ds}(c)}. \quad (2)$$

In view of the varied  $C$  from different datasets,  $\mathcal{H}_{info}(p_{ds})$  is further normalized as

$$\mathcal{H}_{nor} = 1 - \mathcal{H}'_{info}(p_{ds}) \in [0, 1), \quad (3)$$

which can be exploited to measure the sample adaptability, where  $\mathcal{H}'_{info}(p_{ds}) = \frac{\mathcal{H}_{info}(p_{ds}) - \min(\mathcal{H}_{info}(p_{ds}))}{\max(\mathcal{H}_{info}(p_{ds})) - \min(\mathcal{H}_{info}(p_{ds}))}$ ; the constant  $\max(\mathcal{H}_{info}(p_{ds})) = -\sum_{c=1}^C \frac{1}{C} \log \frac{1}{C}$  represents the maximum value of  $\mathcal{H}_{info}(p_{ds})$ , where each element in  $p_{ds}$  is  $\frac{1}{C}$  (i.e., the same class probability), implying that Q perfectly aligns with P (i.e.,  $z_p = z_q$ ), while  $\min(\cdot)$  returns the minimum value of  $\mathcal{H}_{info}(p_{ds})$  within a batch.

As inspired, we revisit the DFQ as: G generates the sample with large adaptability to Q, to be equivalent to maximizing  $\mathcal{H}_{nor}$ ; upon the generated samples, Q is updated to recover the performance with decreasing the sample adaptability, which is equivalent to minimizing  $\mathcal{H}_{nor}$ , adversarial to maximizing  $\mathcal{H}_{nor}$ . Such fact is in line with the principle of zero-sum game<sup>2</sup> [20], as discussed in the next section.

## 2.2. Zero-Sum Game over Sample Adaptability

On top of [17], we revisit the DFQ from a zero-sum game perspective over sample adaptability between two players — a generator and a quantized network, as follows:

$$\min_{\theta_g \in \Theta_g} \max_{\theta_q \in \Theta_q} \mathcal{H}(\theta_g, \theta_q) = \min_{\theta_g \in \Theta_g} \max_{\theta_q \in \Theta_q} \mathbb{E}_{z,y} [1 - \mathcal{H}'_{info}(p_{ds})], \quad (4)$$

<sup>2</sup> For AdaDFQ, G aims to enlarge the disagreement, i.e., adaptability to Q, between P and Q, while Q decreases it by learning from P, where they cancel each other out, making the overall changing (summation) of the disagreement ( $\mathcal{H}_{info}(p_{ds})$ ) close to 0 (see Fig.3 for such intuition), which is consistent to the intuition of zero-sum game.

where  $\theta_g \in \Theta_g$  and  $\theta_q \in \Theta_q$  are the weight parameters of G and Q, respectively. Particularly, Eq.(4) is alternatively optimized via gradient descent during each iteration:  $\theta_q$  is fixed while  $\theta_g$  is updated to generate the sample with large adaptability by maximizing Eq.(4); alternatively,  $\theta_g$  is fixed while  $\theta_q$  is updated to calibrate Q over the generated sample by minimizing Eq.(4). The optimization process will reach a Nash equilibrium [2] ( $\theta_g^*, \theta_q^*$ ) when the following inequality

$$\mathcal{H}(\theta_g, \theta_q^*) \leq \mathcal{H}(\theta_g^*, \theta_q^*) \leq \mathcal{H}(\theta_g^*, \theta_q) \quad (5)$$

holds for all  $\theta_g \in \Theta_g$  and  $\theta_q \in \Theta_q$ , where  $\theta_g^*$  and  $\theta_q^*$  are the parameters of G and Q under an equilibrium state. Maximizing Eq.(4) is equivalent to maximizing  $\mathcal{H}(\cdot, \cdot)$  throughout learning G to generate the sample with the largest adaptability (i.e., smallest  $\mathcal{H}'_{info}(p_{ds})$ ).

However, such fact may incur:

**Underfitting issue:** the knowledge carried by the generated samples from G (e.g.,  $\bigcirc$  in Fig.2), exhibits excessive information with large adaptability, implying a large disagreement between P and Q, while Q (especially for Q with low bit width) has no sufficient ability to learn informative knowledge from P. Evidently, the sample with the largest adaptability (i.e., smallest  $\mathcal{H}'_{info}(p_{ds})$ ) is not the best. For such case, Q is calibrated by minimizing Eq.(4) over such samples to incur the underfitting, which, in turn, encourages G to generate the samples with lower adaptability (i.e., larger  $\mathcal{H}'_{info}(p_{ds})$ ) by alternatively maximizing Eq.(4). However, encouraging the sample with the lowest adaptability (i.e., largest  $\mathcal{H}'_{info}(p_{ds})$ ) may lead to:

**Overfitting issue:** the knowledge carried by the generated samples (e.g.,  $\square$  in Fig.2) delivers limited information with small adaptability to yield a large agreement between P and Q. For such case, it may not be informative to calibrate Q (especially for Q with high bit width) by minimizing Eq.(4), which alternatively encourages G to generate the samples with larger adaptability by maximizing Eq.(4).

## 2.3. Refining the Maximization of Eq.(4): Generating the Sample with Adaptive Adaptability

The above facts indicate that the sample with either the largest or the lowest adaptability generated by maximizing Eq.(4) is not necessarily the best, incurring over-and-under fitting issues, which fail to be resolved by the above disagreement sample (Definition 1) since it focuses on the larger adaptability (i.e., smaller  $\mathcal{H}'_{info}(p_{ds})$ ). To address the issues, we refine the maximization of Eq.(4) during the zero-sum game by proposing to balance disagreement sample with agreement sample, as discussed in the next.

### 2.3.1 Balancing Disagreement Sample with Agreement Sample

As per Definition 1, the category information from P is crucial to establish the dependence of generated sample on Q. Thereby, to generate the disagreement sample with adaptive

adaptability, we exploit the category information to guide G’s optimization. Given the class label  $y$ , it is expected that the generated sample is identified as disagreement sample with the same label  $y$ . Accordingly, we present to match  $p_{ds}$  and  $y$  via the Cross-Entropy loss  $\mathcal{H}_{CE}(\cdot, \cdot)$  below:

$$\mathcal{L}_{ds} = \mathbb{E}_{z,y}[\mathcal{H}_{CE}(p_{ds}, y)]. \quad (6)$$

Eq.(6) encourages G to generate the disagreement sample that P can predict correctly but Q fails. However, the disagreement sample tends to yield smaller  $\mathcal{H}'_{info}(p_{ds})$ , which mitigates the overfitting while the underfitting issue is still available. To remedy such issue, we further define the agreement sample to weaken the effect of disagreement sample on reducing  $\mathcal{H}'_{info}(p_{ds})$ .

**Definition 2 (Agreement Sample).** Based on Definition 1, we determine the generated sample  $x$  to be the agreement sample if  $\text{argmax}(z_p) = \text{argmax}(z_q)$ .

Thus, similar to  $p_{ds}$  in Eq.(1), the agreement between P and Q is encoded via the probability vector

$$p_{as} = \text{softmax}(z_p + z_q) \in \mathbb{R}^C, \quad (7)$$

where  $p_{as}(c) = \frac{\exp(z_p(c) + z_q(c))}{\sum_{j=1}^C \exp(z_p(j) + z_q(j))}$  is the  $c$ -th entry of  $p_{as}$  as the probability that  $x$  is the agreement sample of the  $c$ -th class. Following Eq.(6),  $p_{as}$  and  $y$  are matched via the following loss function:

$$\mathcal{L}_{as} = \mathbb{E}_{z,y}[\mathcal{H}_{CE}(p_{as}, y)]. \quad (8)$$

Eq.(8) encourages G to generate the agreement sample that both P and Q can correctly predict, yielding larger  $\mathcal{H}'_{info}(p_{ds})$ . Intuitively, the agreement sample is capable of balancing disagreement sample to avoid too small or large  $\mathcal{H}'_{info}(p_{ds})$  for adaptive adaptability, formulated as

$$\mathcal{L}_{bal} = \alpha_{ds}\mathcal{L}_{ds} + \alpha_{as}\mathcal{L}_{as}, \quad (9)$$

where  $\alpha_{ds}$  and  $\alpha_{as}$  are used to facilitate the balance between  $\mathcal{L}_{ds}$  and  $\mathcal{L}_{as}$  (see the parameter study). Eq.(9) denotes the loss between  $p_{ds}$ ,  $p_{as}$  and  $y$  to be minimized during the training phase. From this perspective,  $\mathcal{L}_{ds}$  attempts to generate the disagreement samples to enlarge the gap between P and Q, so as to reduce  $\mathcal{H}'_{info}(p_{ds})$  for agreement samples (see  $\leftarrow$  in Fig.2); to be analogous,  $\mathcal{L}_{as}$  enlarges  $\mathcal{H}'_{info}(p_{ds})$  for disagreement samples (see  $\rightarrow$  in Fig.2).

Intuitively,  $\mathcal{L}_{bal}$  endows the generated sample with adaptive adaptability, *i.e.*, neither too large nor small  $\mathcal{H}'_{info}(p_{ds})$  for the generated samples throughout the balance between disagreement and agreement samples. In other words, we need to study how to control  $\mathcal{H}'_{info}(p_{ds})$  within a desirable range via the balance process; cored on that, it establishes the implicit lower and upper boundary corresponding to the disagreement and agreement samples (see Fig.2), hence the above curse is equivalent to confirming the margin between such lower-and-upper boundary. To this end, we propose to optimize the margin between these two boundaries, as discussed in the next.

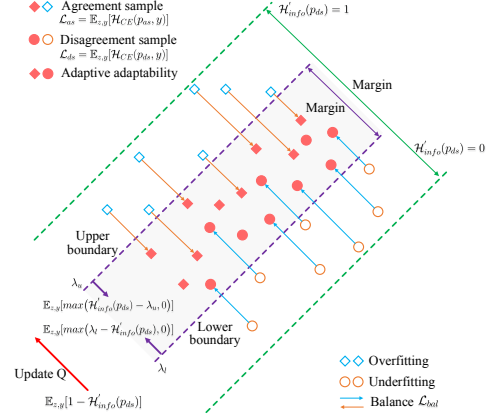


Figure 2. Illustration of generating the sample with adaptive adaptability to address the over-and-under fitting issues.

### 2.3.2 Optimizing the Margin Between two Boundaries

Formally, along with  $\mathcal{L}_{bal}$ , the maximization objective of Eq.(4) is imposed with the desirable bound constraints for  $\mathcal{H}'_{info}(p_{ds})$  (see Fig.2), and formulated as

$$\begin{aligned} \max_{\theta_g \in \Theta_g} \quad & \mathcal{H}(\theta_g, \theta_q) - \beta \mathcal{L}_{bal}, \\ \text{subject to} \quad & \lambda_l < \mathcal{H}'_{info}(p_{ds}) < \lambda_u, \end{aligned} \quad (10)$$

where  $\beta$  is utilized to balance the agreement and disagreement sample.  $\lambda_l$  and  $\lambda_u$  denote the lower and upper bound of  $\mathcal{H}'_{info}(p_{ds})$ , such that  $0 \leq \lambda_l < \lambda_u \leq 1$ . where the goal is to ensure the balance process to generate the sample with adaptive adaptability.  $\lambda_l$  serves to prevent G from generating the sample with too large adaptability (*i.e.*, too small  $\mathcal{H}'_{info}(p_{ds})$ ) via maximizing Eq.(4), where Q is calibrated by minimizing Eq.(4) to incur *underfitting*. By contrast,  $\lambda_u$  aims to avoid the sample with too small adaptability (*i.e.*, too large  $\mathcal{H}'_{info}(p_{ds})$ ), which is not informative to calibrate Q by minimizing Eq.(4), causing *overfitting* issue.

**Discussion on  $\lambda_l$  and  $\lambda_u$ .** The crucial issue is how to specify  $\lambda_l$  and  $\lambda_u$ ? First, we investigate the ideal condition where Q can be well calibrated over varied generated samples from G, it is ideally expected that each generated sample possesses an optimal pairwise of  $\lambda_l$  and  $\lambda_u$ , to help calibrate Q by minimizing Eq.(4), which, unfortunately, is infeasible for individual sample since a large number of generated samples within a batch are generated from G to calibrate Q, only single sample makes no sense; for each batch, the adaptability of each single sample is merely affected by its preceding generated sample, and independent of all other generated samples from G; worse still, the dependencies between varied batches are quite uncertain, since Q varies after being calibrated for each batch. As inspired, instead of trapping into the adaptive rule to each individual sample, we propose to acquire the range of  $\lambda_l$  and  $\lambda_u$  for all generated samples. Particularly, we uniformly select the values of  $\lambda_l$  and  $\lambda_u$  for all samples from  $[0, 1]$  (see the extensive parameter study in Sec.3.4.2), to answer “how much” to the balancing process (Eq.(9)), such that the following demands

meet. That is, avoiding too small  $\lambda_l$  or large  $\lambda_u$ , to address the over-and-under fitting issues, so as to ensure G to well generate the samples with adaptive adaptability.

The above fact discussed the adaptability of single sample to Q, while the calibration process for Q involves the generated samples from a batch. As inspired, we exploit the batch normalization statistics (BNS) information [3, 22] about the training data for P, which is achieved by

$$\mathcal{L}_{BNS} = \sum_{m=1}^M (\|\mu_m^g - \mu_m\|_2^2 + \|\sigma_m^g - \sigma_m\|_2^2), \quad (11)$$

where  $\mu_m^g/\sigma_m^g$  and  $\mu_m/\sigma_m$  denote the mean/variance of the generated samples' distribution within a batch and the corresponding mean/variance parameters for P at the  $m$ -th BN layer of the total  $M$  layers. Eq.(11) encodes the loss between them to be minimized during the training. To this end, the maximization objective of Eq.(4) is refined as

$$\begin{aligned} & \max_{\theta_g \in \Theta_g} \mathbb{E}_{z,y} [-\max(\lambda_l - \mathcal{H}'_{info}(p_{ds}), 0)] + \\ & \mathbb{E}_{z,y} [-\max(\mathcal{H}'_{info}(p_{ds}) - \lambda_u, 0)] - \beta \mathcal{L}_{bal} - \gamma \mathcal{L}_{BNS}, \end{aligned} \quad (12)$$

where the first two terms aim to optimize the margin between  $\lambda_l$  and  $\lambda_u$  via the hinge loss [9].  $\max(\cdot, \cdot)$  returns the maximum.  $\beta$  and  $\gamma$  are the balance parameters, where, the minus “-” for  $\mathcal{L}_{bal}$  and  $\mathcal{L}_{BNS}$  is noted to be minimized during the optimization as aforementioned. With Eq.(12), the minimization objective of Eq.(4) holds to calibrate Q for performance recovery, yielding:

$$\min_{\theta_q \in \Theta_q} \mathbb{E}_{z,y} [1 - \mathcal{H}'_{info}(p_{ds})]. \quad (13)$$

By alternatively optimizing Eq.(12) and (13) during a zero-sum game, the samples with adaptive adaptability can be generated by G to maximally recover the performance of Q until reaching a Nash equilibrium.

#### 2.4. Theoretical Analysis: Why do the Boundaries Improve Q's Generalization?

One may wonder why AdaDFQ can generate the sample with adaptive adaptability. We answer the question by studying the generalization of Q trained on the generated samples from a statistic view. According to the VC theory [12, 13, 21], the classification error of a quantized network (Q) learning from the ground truth label ( $y$ ) on the generated samples by G can be decomposed as

$$R(f_q) - R(f_r) \leq O\left(\frac{|\mathcal{F}_q|_C}{n^{\alpha_{qp}}}\right) + \varepsilon_{qr}, \quad (14)$$

where  $R(\cdot)$  is the error of a specific function.  $f_q \in \mathcal{F}_q$  is the quantized network (Q) function and  $f_r \in \mathcal{F}_r$  is the ground truth label ( $y$ ) function.  $\varepsilon_{qr}$  denotes the approximation error of the quantized network function class  $\mathcal{F}_q$  (considering all possible acquired  $f_q$  by optimizing Q) w.r.t.  $f_r \in \mathcal{F}_r$  (a fixed learning target) on the generated samples during the training phase.  $\alpha_{qr}$  denotes the learning rate for the given generated samples, particularly: when  $\alpha_{qr}$  approaches  $\frac{1}{2}$  (a

slow rate) for the generated samples with the excessive information to Q; while approaching 1 (a fast rate) for the generated samples with the limited information to Q.  $O(\cdot)$  represents the estimation error of a network function over the real data during the testing phase.  $|\cdot|_C$  measures the capacity of a function class and  $n$  is the number of the generated samples from G. Similarly, let  $f_p \in \mathcal{F}_p$  be the full-precision network (P) function, then

$$R(f_p) - R(f_r) \leq O\left(\frac{|\mathcal{F}_p|_C}{n^{\alpha_{pr}}}\right) + \varepsilon_{pr}, \quad (15)$$

where  $\alpha_{pr}$  is related to the learning rate of P upon the ground truth label ( $y$ );  $\varepsilon_{pr}$  denotes the approximation error of the full-precision network (P) function class  $\mathcal{F}_p$  w.r.t.  $f_r \in \mathcal{F}_r$ . For data-free setting, Q is required to learn from P with the generated samples, which yields the following:

$$R(f_q) - R(f_p) \leq O\left(\frac{|\mathcal{F}_q|_C}{n^{\alpha_{qp}}}\right) + \varepsilon_{qp}, \quad (16)$$

where  $\alpha_{qp}$  is related to the learning rate of Q upon P, while  $\varepsilon_{qp}$  is the approximation error of the quantized network (Q) function class  $\mathcal{F}_q$  w.r.t.  $f_p \in \mathcal{F}_p$ . For such case, to study the classification error of Q learning from the ground truth label  $y$  on the generated samples, we combine Eq.(15) and (16) [12, 13], leading to

$$\begin{aligned} R(f_q) - R(f_r) &= R(f_q) - R(f_p) + R(f_p) - R(f_r) \\ &\leq O\left(\frac{|\mathcal{F}_q|_C}{n^{\alpha_{qp}}}\right) + \varepsilon_{qp} + O\left(\frac{|\mathcal{F}_p|_C}{n^{\alpha_{pr}}}\right) + \varepsilon_{pr} \\ &= \underbrace{O\left(\frac{|\mathcal{F}_q|_C}{n^{\alpha_{qp}}}\right) + O\left(\frac{|\mathcal{F}_p|_C}{n^{\alpha_{pr}}}\right)}_{\text{Estimation error}} + \underbrace{\varepsilon_{qp} + \varepsilon_{pr}}_{\text{Approximation error}} \end{aligned} \quad (17)$$

Evidently, Eq.(17) shows that, to benefit the generalization of Q, both estimation and approximation error should be reduced, so that a tighter upper bound can be captured, where we claim that to be consistent with the insights on the optimization of Eq.(12): *first*, as aforementioned, the reasons for the overflow of generalization error stem from: 1) the generated sample with too large  $\mathcal{H}'_{info}(p_{ds})$  is not informative to calibrate Q, *i.e.*, the small  $\varepsilon_{qp}$  during the training stage and large  $O\left(\frac{|\mathcal{F}_q|_C}{n^{\alpha_{qp}}}\right)$  during the testing stage, leading to *overfitting* issue for Q; 2) for the generated sample with too small  $\mathcal{H}'_{info}(p_{ds})$ , Q has no sufficient ability to learn informative knowledge from P, *i.e.*, both  $\varepsilon_{qp}$  during the training stage and  $O\left(\frac{|\mathcal{F}_q|_C}{n^{\alpha_{qp}}}\right)$  during the testing stage are large, leading to *underfitting* issue for Q. Based on that,  $\lambda_l$  and  $\lambda_u$  in Eq.(10) along with the balance process ( $\mathcal{L}_{bal}$ ) in Eq.(9) aim to avoid too large or small  $\varepsilon_{qp}$  by optimizing the margin, thus the estimation error  $O\left(\frac{|\mathcal{F}_q|_C}{n^{\alpha_{qp}}}\right)$  can be reduced, *i.e.*, overcoming the *over-and-under fitting* issues. *Second*, the above fact discussed the adaptability of single sample to Q, as aforementioned, BNS distribution information (Eq.(11)) of the training data extracted from P further facilitates calibrating Q for better generalization via the generated samples within a batch, which ensures the generated samples to be informative to P, *i.e.*, decreasing  $O\left(\frac{|\mathcal{F}_p|_C}{n^{\alpha_{pr}}}\right) + \varepsilon_{pr}$  in

Eq.(15) or (17), along with the category information from P regarding the disagreement and agreement samples.

Based on the above, the sample with adaptive adaptability to Q can be generated by optimizing Eq.(12), reducing both estimation and approximation error (obtaining a tighter upper bound) in Eq.(17), so as to generate the samples by G with adaptive adaptability, which is beneficial to calibrating Q for better generalization by optimizing Eq.(13), where they are alternatively optimized during a zero-sum game until reaching a Nash equilibrium.

### 3. Experiment

#### 3.1. Experimental Settings and Details

We validate AdaDFQ over three typical image classification datasets, including: **CIFAR-10** and **CIFAR-100** [8] contain 10 and 100 classes of images, which are split into 50K training images and 10K testing images; **ImageNet (ILSVRC2012)** [19] consists of 1.2M samples for training and 50k samples for validation with 1000 categories. For data-free setting, only validation sets are adopted to evaluate the performance of the quantized models (Q). We quantize pre-trained full-precision networks (P) including ResNet-20 for CIFAR, and ResNet-18, ResNet-50 and MobileNetV2 for ImageNet, via the following quantizer to yield Q: **Quantizer.** Following [3, 22], we quantize both full-precision (float32) weights and activations into  $n$ -bit precision by a symmetric linear quantization method as [6]:

$$\theta_q = \text{round}\left((2^n - 1) * \frac{\theta - \theta_{min}}{\theta_{max} - \theta_{min}} - 2^{n-1}\right), \quad (18)$$

where  $\theta$  and  $\theta_q$  are the full-precision and quantized value.  $\text{round}(\cdot)$  returns the nearest integer value to the input.  $\theta_{min}$  and  $\theta_{max}$  are the minimum and maximum of  $\theta$ .

For *generation* process, we construct the architecture of the generator G following ACGAN [15], which is trained via Eq.(12) using Adam [7] as an optimizer with a momentum of 0.9 and a learning rate of 1e-3. For *calibration* process, Q is optimized by minimizing Eq.(13), where SGD with Nesterov [14] is adopted as an optimizer with a momentum of 0.9 and weight decay of 1e-4. For CIFAR, the learning rate is initialized to 1e-4 and decayed by 0.1 for every 100 epochs; while it is 1e-5 (1e-4 for ResNet-50) and divided by 10 at epoch 350 on ImageNet. G and Q are alternatively trained for 400 epochs. The batch size is set to 16. For hyper-parameters,  $\alpha_{ds}$  and  $\alpha_{as}$  in Eq.(9);  $\lambda_l$ ,  $\lambda_u$ ,  $\beta$  and  $\gamma$  in Eq.(12) are empirically set to 0.2, 0.1, 0.1, 0.8, 1 and 1 (see our supplementary material for more parameter studies). All experiments are implemented with pytorch [16] via the code of GDFQ [22] and run on an NVIDIA GeForce GTX 1080 Ti GPU and an Intel(R) Core(TM) i7-6950X CPU @ 3.00GHz.

To evaluate AdaDFQ, we offer practical insights into “why” AdaDFQ works, including the comparisons with the state-of-the-arts, ablation studies apart from visual analysis.

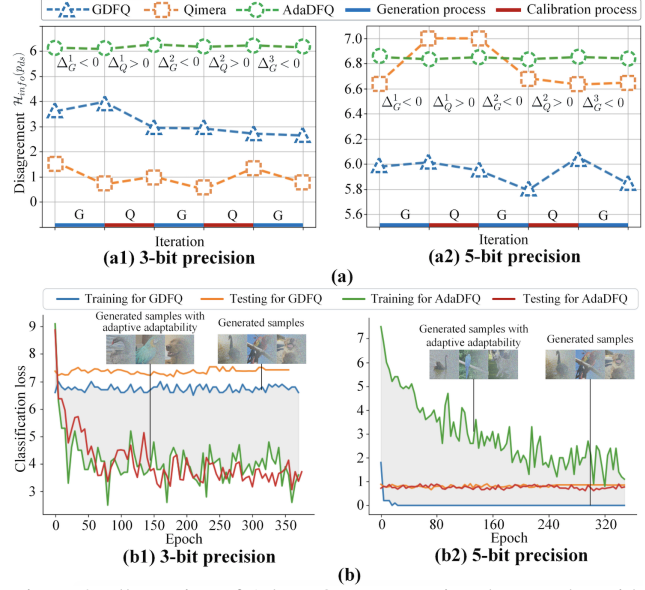


Figure 3. Illustration of AdaDFQ on generating the samples with adaptive adaptability to Q under 3-bit and 5-bit precision. (a) Disagreement between P and Q during the generation (—) and calibration (—) process.  $\Delta_G^i > 0$  and  $\Delta_Q^i < 0$  denote the positive and negative gain of the disagreement, *i.e.*,  $\mathcal{H}_{info}(p_{ds})$ , at  $i$ -th iteration of zero-sum game for the sample generation from G and calibration to Q. (b) Classification loss for Q during the training and testing phases.

#### 3.2. Why does AdaDFQ Work?

We verify the core idea of AdaDFQ — optimizing the margin to generate the sample with adaptive adaptability for better Q’s generalization under varied bit widths. We perform the experiments with ResNet-18 (Fig.3(a)) and ResNet-50 (Fig.3(b)) serving as both P and Q on ImageNet. Fig.3(a) illustrates that, compared to GDFQ [22] and Qimera [3], the disagreement (computed by Eq.(2)) between P and Q for AdaDFQ performs stably within a small range, *i.e.*, the overall changing summation ( $\Delta_G^i + \Delta_Q^i$ ) of the disagreement is *close* to 0, following the principle of zero-sum game (Sec.2.2), which confirms that the generated sample with adaptive adaptability by AdaDFQ is fully exploited to benefit Q, and the lower and upper bound constraints ( $\lambda_l$  and  $\lambda_u$  in Eq.(12)) avoid the generated sample with too large or small adaptability, which results in ridiculously large disagreement or agreement. Fig.3(b) reveals that AdaDFQ achieves better Q’s generalization with 3-bit and 5-bit precision unlike GDFQ, where the generated sample with adaptive adaptability succeeds in overcoming the *underfitting* (both training and testing loss are large) and *overfitting* (small training loss but large testing loss), confirming the analysis in Sec.2.4.

#### 3.3. Comparison with State-of-the-arts

To verify the superiority of AdaDFQ, we compare it with typical DFQ methods, *i.e.*, GDFQ [22], ARC [25]

Table 2. Accuracy (%) comparison with the state-of-the-arts on CIFAR-10, CIFAR-100 and ImageNet. †: the results reimplemented by author-provided code. -: no results are reported. *nwna* indicates the weights and activations are quantized to *n*-bit. The best results are reported with **boldface**.

Dataset	Model (Full precision)	Bit width	ZAQ [11] (CVPR 2021)	IntraQ [24] (CVPR 2022)	ARC+AIT [4] (CVPR 2022)	GDFQ [22] (ECCV 2020)	ARC [25] (IJCAI 2021)	Qimera [3] (NeurIPS 2021)	AdaSG [17] (AAAI 2023)	AdaDFQ (Ours)
CIFAR-10	ResNet-20 (93.89)	3w3a	-	77.07	-	75.11 <sup>†</sup>	-	74.43 <sup>†</sup>	84.14	<b>84.89</b>
		4w4a	92.13	91.49	90.49	90.11	88.55	91.26	92.10	<b>92.31</b>
		5w5a	93.36	-	92.98	93.38	92.88	93.46	93.76	<b>93.81</b>
CIFAR-100	ResNet-20 (70.33)	3w3a	-	48.25	41.34 <sup>†</sup>	47.61 <sup>†</sup>	40.15 <sup>†</sup>	46.13 <sup>†</sup>	<b>52.76</b>	52.74
		4w4a	60.42	64.98	61.05	63.75	62.76	65.10	66.42	<b>66.81</b>
		5w5a	68.70	-	68.40	67.52	68.40	69.02	69.42	<b>69.93</b>
ImageNet	ResNet-18 (71.47)	3w3a	-	-	-	20.23 <sup>†</sup>	23.37	1.17 <sup>†</sup>	37.04	<b>38.10</b>
		4w4a	52.64	66.47	65.73	60.60	61.32	63.84	66.50	<b>66.53</b>
		5w5a	64.54	69.94	70.28	68.49	68.88	69.29	<b>70.29</b>	<b>70.29</b>
	MobileNetV2 (73.03)	3w3a	-	-	-	1.46 <sup>†</sup>	14.30	-	26.90	<b>28.99</b>
		4w4a	0.10	65.10	<b>66.47</b>	59.43	60.13	61.62	65.15	65.41
		5w5a	62.35	71.28	<b>71.96</b>	68.11	68.40	70.45	71.61	71.61
	ResNet-50 (77.73)	3w3a	-	-	-	0.31 <sup>†</sup>	1.63	-	16.98	<b>17.63</b>
		4w4a	53.02	-	68.27	54.16	64.37	66.25	<b>68.58</b>	68.38
		5w5a	73.38	-	76.00	71.63	74.13	75.32	<b>76.03</b>	<b>76.03</b>

Table 3. Ablation study about varied components of AdaDFQ on ImageNet. *nwna* indicates the weights and activations are quantized to *n*-bit. The best results are reported with **boldface**.

Model (Full precision)	$\mathcal{L}_{ds}$	$\mathcal{L}_{as}$	$\lambda_l, \lambda_u$	$\mathcal{L}_{BNS}$	3w3a	5w5a
ResNet-18 (71.47)		✓	✓	✓	19.40	70.03
	✓		✓	✓	31.14	69.77
	✓	✓		✓	18.53	66.27
	✓		✓	✓	32.13	70.06
	✓	✓	✓		20.99	67.80
	✓	✓	✓	✓	<b>38.10</b>	<b>70.29</b>

and Qimera [3]: reconstructing the original data from P; ZAQ [11] focuses primarily on the adversarial sample generation rather than adversarial game process for AdaDFQ; IntraQ [24] optimizes the noise to obtain fake sample without a generator; AIT [4] improves the loss function and gradients for ARC to generate better sample, denoted as ARC+AIT; AdaSG [17] focuses on the zero-sum game framework, serving as a special case of AdaDFQ.

Table 2 summarizes our following findings: 1) AdaDFQ obtains a significant and consistent accuracy gain over the state-of-the-arts, in line with our purpose of optimizing the margin to generate the sample with adaptive adaptability to Q (Sec.2.3). Impressively, AdaDFQ achieves at most 10.46%, 12.59% and 36.93% accuracy gains on CIFAR-10, CIFAR-100 and ImageNet. Notably, compared with GDFQ, ARC and Qimera where Q is independent of the generation process, AdaDFQ obtains accuracy improvement with a large margin, e.g., at least 0.35% gain (ResNet-20 with 5w5a on CIFAR-10), confirming the necessity of AdaDFQ over the sample adaptability to Q in Sec.1. Specifically, without regard for the sample adaptability, ZAQ suffers from a large performance loss caused by many unexpected generated samples, which are harmful to the calibration process of Q. AdaDFQ upgrades beyond AIT despite of the combination with ARC. As expected, AdaDFQ exhibits the obvious advantages over AdaSG, implying the benefits of optimizing the margin upon the zero-sum game. 2) AdaDFQ delivers the substantial gains for Q under varied bit-widths, confirming the importance of adaptive adapt-

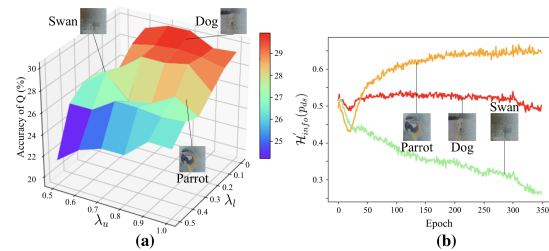


Figure 4. Ablation study about  $\lambda_l$  and  $\lambda_u$ . (a) Accuracy (%) comparison of Q with varied ( $\lambda_l, \lambda_u$ ). (b)  $\mathcal{H}'_{inf}(p_{ds})$  for the generated samples corresponding to different areas in (a).

ability to varied Q (Sec.2.3). Especially for 3-bit situation, most of the competitors suffer from the poor accuracy or convergence, while AdaDFQ obtains at most 36.93% (ResNet-18 with 3w3a) accuracy gains.

### 3.4. Ablation Study

#### 3.4.1 Validating adaptability with disagreement and agreement samples

As aforementioned, the disagreement and agreement samples play a critical role in addressing the over-and-under fitting issues for the adaptive adaptability. We perform the ablation study on  $\mathcal{L}_{ds}$  (Eq.(6)) and  $\mathcal{L}_{as}$  (Eq.(8)) over ImageNet. Table 3 suggests the noticeable superiority (38.10%, 70.29%) of AdaDFQ (including the both) over other cases. Note that, abandoning either or both of  $\mathcal{L}_{ds}$  and  $\mathcal{L}_{as}$  receives a large accuracy loss (at most 19.57% and 4.02%), implying the intuition of balancing the disagreement sample with agreement sample (Sec.2.3.1). Interestingly, the case without  $\lambda_l$  and  $\lambda_u$  (Eq.(12)) obtains the minimal accuracy loss (5.97% and 0.23%), confirming the importance of the bound constraints on the basis of  $\mathcal{L}_{bal}$  (Eq.(9)).

#### 3.4.2 Why can $\lambda_l$ and $\lambda_u$ benefit Q?

The parameters  $\lambda_l$  and  $\lambda_u$  in Eq.(12) serve as a lower and upper bound of the adaptive adaptability (neither too

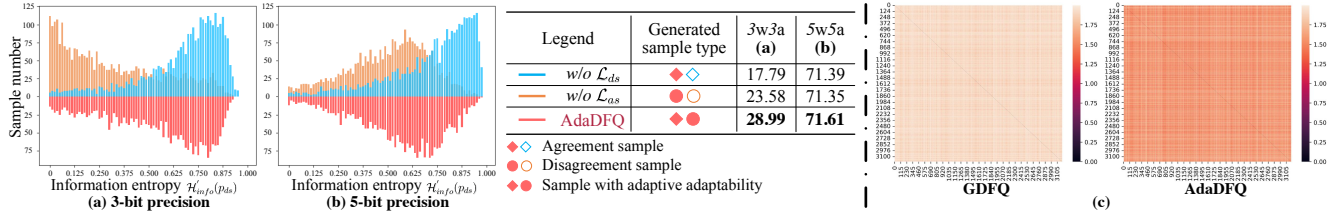


Figure 5. (a)(b) Illustration of how to balance  $\mathcal{L}_{ds}$  and  $\mathcal{L}_{as}$  to generate the sample with adaptive adaptability under 3-bit and 5-bit precision. (c) Visual analysis: the similarity comparison between the generated samples.

small or large  $\mathcal{H}'_{info}(p_{ds})$  for the sample generation, which is critical to address the over-and-under fitting issues. We aim to verify the effectiveness of varied parameter configurations ( $\lambda_l \in \{0, 0.1, 0.2, 0.3, 0.4, 0.5\}$ ,  $\lambda_u \in \{0.5, 0.6, 0.7, 0.8, 0.9, 1.0\}$ ) via the grid search and perform the experiments under 3-bit precision with MobileNetV2 serving as P and Q on ImageNet. Fig.4(a) illustrates that AdaDFQ achieves the significant performance within an optimal range (the red area in Fig.4(a)), *i.e.*,  $\lambda_l \in \{0, 0.1, 0.2\}$  and  $\lambda_u \in \{0.7, 0.8, 0.9\}$ , indicating a wide range between two bounds, where the performance of Q is insensitive to  $\lambda_l$  and  $\lambda_u$ , which offers a guideline for the selection of their values ( $\lambda_l$  and  $\lambda_u$  is set to 0.1 and 0.8 in the main experiments), verifying the feasibility to uniformly select the values of  $\lambda_l$  and  $\lambda_u$  for all samples (Sec.2.3.2). Besides, Fig.4(b) provides evidence that  $\lambda_l$  and  $\lambda_u$  within the optimal range contributes to yielding the adaptive adaptability, where  $\mathcal{H}'_{info}(p_{ds})$  (the red in Fig.4(b)) is neither too small (the green in Fig.4(b)) nor large (the orange in Fig.4(b)).

### 3.4.3 How to balance disagreement sample with agreement sample?

We further study the effectiveness of  $\mathcal{L}_{bal}$  in Eq.(9), and how to balance disagreement sample with agreement sample via two cases: **A**: only generating disagreement sample, denoted as *w/o*  $\mathcal{L}_{as}$ ; and **B**: only generating agreement sample, denoted as *w/o*  $\mathcal{L}_{ds}$ . We perform the experiments and generate 3200 samples under 3-bit and 5-bit precision with MobileNetV2 serving as P and Q on ImageNet. Fig.5(a)(b) illustrates that most of the generated samples (— in Fig.5(a)(b)) from case A yield smaller  $\mathcal{H}'_{info}(p_{ds})$  than those (— in Fig.5(a)(b)) from case B, which provides a basis for balancing  $\mathcal{L}_{ds}$  with  $\mathcal{L}_{as}$ . It is observed that  $\mathcal{H}'_{info}(p_{ds})$  of the generated sample (— in Fig.5(a)(b)) from AdaDFQ is neither too small nor large compared to case **A** and **B**, which is evidence that  $\mathcal{L}_{bal}$  forces the disagreement and agreement samples to move towards each other between two boundaries, in line with the analysis in Sec.2.3.1.

### 3.5. Visual Analysis on Generated Samples

To further show the intuition of the generated samples with adaptive adaptability to Q, we perform the visual analysis over MobileNetV2 serving as both P and Q on ImageNet.

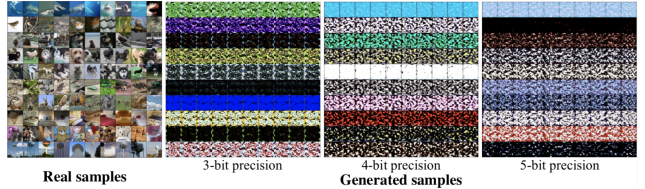


Figure 6. Visualization of real and generated samples, where each row denotes one of 10 randomly chosen classes from ImageNet.

geNet by the similarity matrix (calculated as the  $\ell_1$  norm between  $p_{ds}$  of generated samples), along with the examples of generated samples (10 images per category) from 10 categories. Fig.5(c) illustrates that the generated samples by AdaDFQ exhibit a much larger similarity (the darker, the larger) than those by GDFQ [22], implying that the substantial samples with undesirable adaptability by GDFQ exist against AdaDFQ. Fig.6 shows that the generated samples for different bit widths (*i.e.*, 3 bit, 4 bit and 5bit) vary greatly, confirming the intuition of AdaDFQ — generating the sample with adaptive adaptability to varied Q (Sec.2.3); while the samples from varied categories differ greatly from each other, confirming that the category information is fully exploited (Sec.2.3.1); *due to page limitation, see supplementary material for higher resolution.*

## 4. Conclusion

In this paper, we propose an Adaptive Data-Free Quantization (AdaDFQ) method, which revisits the DFQ from a zero-sum game perspective between two players. Following this viewpoint, the disagreement and agreement samples are further defined to form the lower and upper boundaries. The margin between two boundaries is optimized to address the over-and-under fitting issues, so as to generate the samples with adaptive adaptability between these two boundaries to calibrate Q. The theoretical analysis and empirical studies validate the advantages of AdaDFQ to the existing arts.

## 5. Acknowledgments

This work is supported by National Natural Science Foundation of China under the grant no U21A20470, 62172136, 72188101, U1936217. Key Research and Technology Development Projects of Anhui Province (no.202004a5020043).



## References

- [1] Yaohui Cai, Zhewei Yao, Zhen Dong, Amir Gholami, Michael W Mahoney, and Kurt Keutzer. Zeroq: A novel zero shot quantization framework. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13169–13178, 2020. [1](#)
- [2] Adrian Rivera Cardoso, Jacob Abernethy, He Wang, and Huan Xu. Competing against nash equilibria in adversarially changing zero-sum games. In *International Conference on Machine Learning*, pages 921–930. PMLR, 2019. [3](#)
- [3] Kanghyun Choi, Deokki Hong, Noseong Park, Youngsok Kim, and Jinho Lee. Qimera: Data-free quantization with synthetic boundary supporting samples. *Advances in Neural Information Processing Systems*, 34, 2021. [1](#), [2](#), [5](#), [6](#), [7](#)
- [4] Kanghyun Choi, Hye Yoon Lee, Deokki Hong, Joonsang Yu, Noseong Park, Youngsok Kim, and Jinho Lee. It’s all in the teacher: Zero-shot quantization brought closer to the teacher. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8311–8321, 2022. [7](#)
- [5] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. In *NIPS*, 2015. [1](#)
- [6] Benoit Jacob, Skirmantas Kligys, Bo Chen, Menglong Zhu, Matthew Tang, Andrew Howard, Hartwig Adam, and Dmitry Kalenichenko. Quantization and training of neural networks for efficient integer-arithmetic-only inference. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2704–2713, 2018. [1](#), [6](#)
- [7] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. [6](#)
- [8] A Krizhevsky. Learning multiple layers of features from tiny images. *Master’s thesis, University of Tront*, 2009. [6](#)
- [9] Jae Hyun Lim and Jong Chul Ye. Geometric gan. *arXiv preprint arXiv:1705.02894*, 2017. [5](#)
- [10] Darryl Lin, Sachin Talathi, and Sreekanth Annapureddy. Fixed point quantization of deep convolutional networks. In *International conference on machine learning*, pages 2849–2858. PMLR, 2016. [1](#)
- [11] Yuang Liu, Wei Zhang, and Jun Wang. Zero-shot adversarial quantization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1512–1521, 2021. [2](#), [7](#)
- [12] David Lopez-Paz, Léon Bottou, Bernhard Schölkopf, and Vladimir Vapnik. Unifying distillation and privileged information. *arXiv preprint arXiv:1511.03643*, 2015. [5](#)
- [13] Seyed Iman Mirzadeh, Mehrdad Farajtabar, Ang Li, Nir Levine, Akihiro Matsukawa, and Hassan Ghasemzadeh. Improved knowledge distillation via teacher assistant. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 5191–5198, 2020. [5](#)
- [14] Yuri E Nesterov. A method for solving the convex programming problem with convergence rate  $o(1/k^2)$ . In *Dokl. akad. nauk Sssr*, volume 269, pages 543–547, 1983. [6](#)
- [15] Augustus Odena, Christopher Olah, and Jonathon Shlens. Conditional image synthesis with auxiliary classifier gans. In *International conference on machine learning*, pages 2642–2651. PMLR, 2017. [6](#)
- [16] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019. [6](#)
- [17] Biao Qian, Yang Wang, Richang Hong, and Meng Wang. Rethinking data-free quantization as a zero-sum game. In *Proceedings of the AAAI conference on artificial intelligence*, 2023. [2](#), [3](#), [7](#)
- [18] Biao Qian, Yang Wang, Hongzhi Yin, Richang Hong, and Meng Wang. Switchable online knowledge distillation. In *Computer Vision—ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XI*, pages 449–466, 2022. [1](#)
- [19] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115(3):211–252, 2015. [6](#)
- [20] J v. Neumann. Zur theorie der gesellschaftsspiele. *Mathematische annalen*, 100(1):295–320, 1928. [3](#)
- [21] Vladimir Vapnik. *The nature of statistical learning theory*. Springer science & business media, 1999. [5](#)
- [22] Shoukai Xu, Haokun Li, Bohan Zhuang, Jing Liu, Jiezhong Cao, Chuangrun Liang, and Minghui Tan. Generative low-bitwidth data free quantization. In *European Conference on Computer Vision*, pages 1–17. Springer, 2020. [1](#), [2](#), [5](#), [6](#), [7](#), [8](#)
- [23] Xiangguo Zhang, Haotong Qin, Yifu Ding, Ruihao Gong, Qinghua Yan, Renshuai Tao, Yuhang Li, Fengwei Yu, and Xianglong Liu. Diversifying sample generation for accurate data-free quantization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15658–15667, 2021. [1](#)
- [24] Yunshan Zhong, Mingbao Lin, Gongrui Nan, Jianzhuang Liu, Baochang Zhang, Yonghong Tian, and Rongrong Ji. Intraq: Learning synthetic images with intra-class heterogeneity for zero-shot network quantization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12339–12348, 2022. [7](#)
- [25] Baozhou Zhu, Peter Hofstee, Johan Peltenburg, Jinho Lee, and Zaid Alars. Autorecon: Neural architecture search-based reconstruction for data-free. In *International Joint Conference on Artificial Intelligence*, 2021. [1](#), [2](#), [6](#), [7](#)