

# Poster: Self-Supervised Quantization-Aware Knowledge Distillation

1<sup>st</sup> Kaiqi Zhao

School of Computing and Augmented Intelligence  
Arizona State University  
kzhao27@asu.edu

2<sup>nd</sup> Ming Zhao

School of Computing and Augmented Intelligence  
Arizona State University  
mingzhao@asu.edu

**Abstract**—Quantization-aware training (QAT) achieves competitive performance and is widely used for image classification tasks in model compression. Existing QAT works start with a pre-trained full-precision model and perform quantization during retraining. However, these works require supervision from the ground-truth labels whereas sufficient labeled data are infeasible in real-world environments. Also, they suffer from accuracy loss due to reduced precision, and no algorithm consistently achieves the best or the worst performance on every model architecture. To address the aforementioned limitations, this paper proposes a novel Self-Supervised Quantization-Aware Knowledge Distillation framework (SQAKD). SQAKD unifies the forward and backward dynamics of various quantization functions, making it flexible for incorporating the various QAT works. With the full-precision model as the teacher and the low-bit model as the student, SQAKD reframes QAT as a co-optimization problem that simultaneously minimizes the KL-Loss (i.e., the Kullback-Leibler divergence loss between the teacher’s and student’s penultimate outputs) and the discretization error (i.e., the difference between the full-precision weights/activations and their quantized counterparts). This optimization is achieved in a self-supervised manner without labeled data. The evaluation shows that SQAKD significantly improves the performance of various state-of-the-art QAT works (e.g., PACT, LSQ, DoReFa, and EWGS). SQAKD establishes stronger baselines and does not require extensive labeled training data, potentially making state-of-the-art QAT research more accessible.

## I. INTRODUCTION

Quantization is one of the model compression approaches [1]–[5] to address the mismatch issue between resource-hungry DNNs and resource-constrained edge devices [6]–[8]. Various quantization techniques [9]–[12] have achieved great results in creating low-bit models through Quantization-Aware Training (QAT), which starts with a pre-trained model and performs quantization during retraining. However, most of them result in some degree of accuracy loss due to reduced precision [13] and no algorithm consistently achieves the best or the worst performance on every model architecture (e.g., VGG, ResNet, MobileNet, etc.) [14]. Also, the various QAT works are motivated by different intuitions and lack a commonly agreed theory, which makes it challenging to generalize. Moreover, all the QAT works assume labeled training data are always available while labeling the data can be time-consuming and sometimes even infeasible, particularly in specialized domains or for specific tasks.

To address the aforementioned limitations and improve the performance of the state-of-the-art (SOTA) QAT for model compression, we propose a simple yet effective framework — Self-Supervised Quantization-Aware Knowledge Distillation (SQAKD). SQAKD first unifies the forward and backward dynamics of various quantization functions and shapes

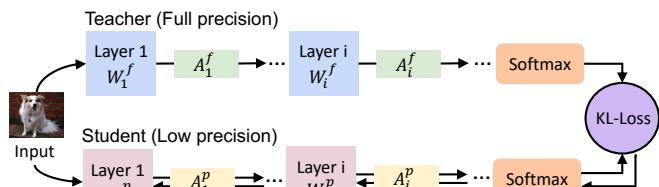


Fig. 1: Workflow of SQAKD.

QAT as minimizing the discretization error between original weights/activations and their quantized counterparts. Then, SQAKD lets the full-precision and low-bit models as the teacher and student, respectively, and excludes CE-Loss (i.e., cross-entropy loss with labels) and keeps only the KL-Loss (i.e., the Kullback-Leibler divergence loss between the teacher’s and student’s penultimate outputs), reframing QAT as a co-optimization problem that simultaneously minimizes the KL-Loss and the discretization error without supervision from labels.

Compared to existing QAT methods, SQAKD has several advantages. First, SQAKD is flexible for incorporating the various QAT works since it unifies the optimization of their forward and backward dynamics. Second, SQAKD operates in a self-supervised manner without labeled data, supporting a more extensive scope of applications in practical scenarios. Finally, SQAKD improves the various SOTA QAT works significantly in both convergence speed and accuracy.

## II. METHODOLOGY

Figure 1 illustrates the workflow of the proposed SQAKD.

**Forward Propagation.** Let  $Quant(\cdot)$  denote a uniform quantizer that converts a full-precision input  $x$  to a quantized output  $x_q = Quant(x)$ .  $x$  can be the activations or weights of the network. First, the quantizer  $Quant(\cdot)$  applies a clipping function  $Clip(\cdot)$  to normalize and clip the input  $x$  to smaller range, producing a full-precision latent presentation:  $x_c = Clip(x, \{p_i\}_{i=1}^{i=K_c}, v, m)$ , where  $v$  and  $m$  are the lower and upper bound of the range, respectively,  $\{p_i\}_{i=1}^{i=K_c}$  denotes the set of trainable parameters needed for quantization, and  $K_c$  denotes the number of parameters. Then, the quantizer  $Quant(\cdot)$  converts the clipped value  $x_c$  to a discrete quantization point  $x_q$  using the function  $R(\cdot)$  that contains a round function:  $x_q = R(x_c, b, \{q_i\}_{i=1}^{i=K_r})$ , where  $b$  is the bit width and  $\{q_i\}_{i=1}^{i=K_r}$  denotes the set of trainable parameters. Thus,  $Quant(\cdot)$  can be described as:  $x_q = Quant(x, \alpha, b, v, m)$ , here we use  $\alpha$  as a shorthand for the set of all the parameters in the functions  $R(\cdot)$  and  $Clip(\cdot)$ :  $\alpha = \{\{p_i\}_{i=1}^{i=K_c}, \{q_i\}_{i=1}^{i=K_r}\}$ .

TABLE I: Top-1 test accuracy (%) on CIFAR-10 and CIFAR-100.

	Model	VGG-8 (FP: 91.27)			ResNet-20 (FP: 92.58)		
		W1A1	W2A2	W4A4	W1A1	W2A2	W4A4
CIFAR-10	Bit-width						
	EWGS [12]	87.77	90.84	90.95	86.42	91.41	92.40
	<b>SQAKD (EWGS)</b>	<b>89.05</b>	<b>91.55</b>	<b>91.31</b>	<b>86.47</b>	<b>91.80</b>	<b>92.59</b>
		<b>(+1.28)</b>	<b>(+0.71)</b>	<b>(+0.36)</b>	<b>(+0.05)</b>	<b>(+0.39)</b>	<b>(+0.19)</b>
	Model	VGG-13 (FP: 76.36)			ResNet-32 (FP: 71.33)		
	Bit-width	W1A1	W2A2	W4A4	W1A1	W2A2	W4A4
CIFAR-100	EWGS [12]	65.55	73.31	73.41	59.25	69.37	70.50
	<b>SQAKD (EWGS)</b>	<b>68.56</b>	<b>74.65</b>	<b>74.67</b>	<b>59.41</b>	<b>69.99</b>	<b>71.65</b>
		<b>(+3.01)</b>	<b>(+1.34)</b>	<b>(+1.26)</b>	<b>(+0.16)</b>	<b>(+0.62)</b>	<b>(+1.15)</b>

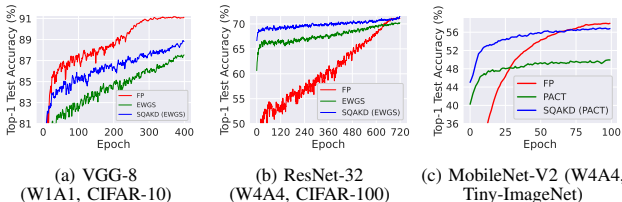


Fig. 2: Top-1 test accuracy evolution of full-precision models (FP), and quantized models using EWES and EWGS+SQAKD, in each epoch during training on CIFAR-10 and CIFAR-100. “W\*A×” denotes that the weights and activations are quantized into \*bit and ×bit, respectively.

**Backward Propagation.** Different from prevalent quantization works using Straight-Through Estimator (STE) [15] in backpropagation, we propose a novel formulation of backpropagation by integrating the discretization error ( $x_c - x_q$ ):

$$\frac{\partial L}{\partial x_c} = \frac{\partial L}{\partial x_q} + \mu \cdot (x_c - x_q), \quad (1)$$

where  $\mu$  is a non-negative value. STE is represented by setting  $\mu$  to zero, and  $\mu$  can also be updated by other schemes, like Curriculum Learning driven strategy [1], [16].

**Optimization Objective.** To apply Knowledge Distillation (KD) [17] into quantization, we let a pre-trained full-precision network act as the teacher, guiding a low-bit student with the same architecture. KD [17] defines the training loss as a linear combination of the cross-entropy loss (term “CE-Loss”) with labels and the KL divergence (termed “KL-Loss”) between the teacher’s and student’s soft logits, controlled by a hyperparameter  $\lambda$ :  $L = (1 - \lambda)L_{CE} + \lambda L_{KL}$ .

However, we find out that CE-Loss does not cooperate effectively with KL-Loss, and their combination may potentially degrade the network performance. Solely minimizing KL-Loss is sufficient for achieving optimal gradient updates in the quantized network. So we drop the CE-Loss and keep only the KL-Loss in SQAKD. The optimization objection is defined as:

$$\begin{aligned} \min_{W_f^S, \alpha_W, \alpha_A} & KL(S(h^T/\rho) || S(h^S/\rho)) \\ \text{s.t.} & W_q^S = \text{Quant}_W(W_f^S, \alpha_W, b_W, v_W, m_W) \\ & A_q^S = \text{Quant}_A(A_f^S, \alpha_A, b_A, v_A, m_A), \end{aligned} \quad (2)$$

where  $\rho$  is the temperature, which makes distribution softer for using the dark knowledge,  $Y$  is the ground-truth labels,  $h^T$  and  $h^S$  are the penultimate layer outputs of the teacher and student, respectively.  $\text{Quant}_W(\cdot)$  and  $\text{Quant}_A(\cdot)$  are the quantization function for the student’s weights and activations, and  $W_f^S/A_f^S$  and  $W_q^S/A_q^S$  are the student’s full-precision weights/activations and quantized weights/activations.

TABLE II: Top-1 test accuracy (%) of ResNet and VGG on Tiny-ImageNet.

Model	ResNet-18 (FP: 65.59)			VGG-11 (FP: 59.47)		
	W3A3	W4A4	W8A8	W3A3	W4A4	W8A8
Bit-width						
PACT [10]	58.09	61.06	64.91	52.94	57.10	58.08
<b>SQAKD (PACT)</b>	<b>61.34</b>	<b>61.47</b>	<b>65.78</b>	<b>57.25</b>	<b>59.05</b>	<b>59.44</b>
	<b>(+3.25)</b>	<b>(+0.41)</b>	<b>(+0.87)</b>	<b>(+4.31)</b>	<b>(+1.95)</b>	<b>(+1.36)</b>
LSQ [11]	61.99	64.10	65.08	58.39	59.14	59.25
<b>SQAKD (LSQ)</b>	<b>65.21</b>	<b>65.34</b>	<b>65.96</b>	<b>58.43</b>	<b>59.19</b>	<b>59.42</b>
	<b>(+3.22)</b>	<b>(+1.24)</b>	<b>(+0.88)</b>	<b>(+0.04)</b>	<b>(+0.05)</b>	<b>(+0.17)</b>
DoReFa [9]	61.94	62.72	63.23	56.72	57.28	57.54
<b>SQAKD (DoReFa)</b>	<b>64.10</b>	<b>64.56</b>	<b>64.88</b>	<b>57.02</b>	<b>58.93</b>	<b>58.91</b>
	<b>(+2.16)</b>	<b>(+1.84)</b>	<b>(+1.65)</b>	<b>(+0.3)</b>	<b>(+1.65)</b>	<b>(+1.37)</b>

TABLE III: Top-1 and top-5 test accuracy (%) of MobileNet-V2, ShuffleNet-V2, and SqueezeNet on Tiny-ImageNet.

Model	Bit-width	Method	Top-1 Acc.	Top-5 Acc.
MobileNet-V2	FP	-	58.07	80.97
	W3A3	PACT [10]	47.77	73.44
		<b>SQAKD (PACT)</b>	<b>52.73</b>	<b>77.68</b>
		PACT [10]	50.33	75.08
		<b>SQAKD (PACT)</b>	<b>57.14</b>	<b>80.61</b>
ShuffleNet-V2	W4A4	DoReFa [9]	45.96	71.93
		<b>SQAKD (DoReFa)</b>	<b>47.33</b>	<b>73.85</b>
	W8A8	DoReFa [9]	56.26	79.64
	<b>SQAKD (DoReFa)</b>	<b>58.13</b>	<b>81.3</b>	
SqueezeNet1_0	FP	-	49.91	76.05
	W4A4	PACT [10]	27.09	52.54
		<b>SQAKD (PACT)</b>	<b>41.11</b>	<b>68.4</b>
		DoReFa [9]	45.96	71.93
		<b>SQAKD (DoReFa)</b>	<b>47.33</b>	<b>73.85</b>
SqueezeNet1_0	W4A4	LSQ [11]	35.37	62.75
		<b>SQAKD (LSQ)</b>	<b>47.40</b>	<b>73.18</b>
		DoReFa [9]	42.66	69.25
		<b>SQAKD (DoReFa)</b>	<b>46.62</b>	<b>73.02</b>
W8A8	DoReFa [9]	42.66	69.25	
	<b>SQAKD (DoReFa)</b>	<b>46.62</b>	<b>73.02</b>	

### III. EVALUATION

**Experiment Setup.** We conduct an extensive evaluation on diverse models (ResNet [18], VGG [19], MobileNet [20], ShuffleNet [21], SqueezeNet [22], and AlexNet [23]) and various datasets, including CIFAR-10 [24], CIFAR-100 [24], and Tiny-ImageNet [25]. We incorporate state-of-the-art (SOTA) quantization methods (PACT [10], LSQ [11], DoReFa [9], and EWGS [12]) into SQAKD, and compare to their original results to show the improvement made by SQAKD.

**Accuracy on CIFAR-10 and CIFAR-100.** Table I shows that SQAKD improves the accuracy of EWGS by a large margin for 1, 2, and 4-bit quantization on CIFAR-10 and CIFAR-100. Specifically, on CIFAR-10, SQAKD improves EWGS by 0.36% to 1.28% on VGG-8 and 0.05% to 0.39% on ResNet-20; and on CIFAR-100, SQAKD improves EWGS by 1.26% to 3.01% on VGG-13 and 0.16% to 1.15% on ResNet-32.

**Accuracy on Tiny-ImageNet.** Tables II and III show that SQAKD consistently improves the accuracy of various quantization methods, including PACT [10], LSQ [11], and DoReFa [9], by a large margin in all cases. Specifically, SQAKD improves 1) PACT by 0.41% to 15.86%, 2) LSQ by 0.04% to 12.03%, and 3) DoReFa 0.3% to 3.96% for 3, 4, and 8-bit quantization.

**Convergence Speed.** Figure 2 illustrate the top-1 test accuracy evolution for 1-bit VGG-18 (CIFAR-10), 4-bit ResNet-32 (CIFAR-100), and 4-bit MobileNet-V2 (Tiny-ImageNet), respectively, in each epoch during training. SQAKD improves the convergence speed of the existing quantization methods on all model architectures. Furthermore, on MobileNet-V2, SQAKD enables the quantized student to converge much faster than the full-precision teacher whereas the quantization method alone cannot achieve that.

## REFERENCES

- [1] K. Zhao, H. D. Nguyen, A. Jain, N. Susanj, A. Mouchtaris, L. Gupta, and M. Zhao, “Knowledge distillation via module replacing for automatic speech recognition with recurrent neural network transducer,” 2022.
- [2] K. Zhao, Y. Chen, and M. Zhao, “A contrastive knowledge transfer framework for model compression and transfer learning,” in *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1–5, IEEE, 2023.
- [3] K. Zhao, A. Jain, and M. Zhao, “Automatic attention pruning: Improving and automating model pruning using attentions,” in *International Conference on Artificial Intelligence and Statistics*, pp. 10470–10486, PMLR, 2023.
- [4] M. Á. Carreira-Perpiñán and Y. Idelbayev, “Model compression as constrained optimization, with application to neural nets. part v: combining compressions,” *arXiv preprint arXiv:2107.04380*, 2021.
- [5] Y. Idelbayev and M. Á. Carreira-Perpiñán, “Lc: A flexible, extensible open-source toolkit for model compression,” in *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*, pp. 4504–4514, 2021.
- [6] Y. Chen, K. Zhao, B. Li, and M. Zhao, “Exploring the use of synthetic gradients for distributed deep learning across cloud and edge resources,” in *2nd USENIX Workshop on Hot Topics in Edge Computing (HotEdge 19)*, 2019.
- [7] M. Zhao, “Knowledgenet: Disaggregated and distributed training and serving of deep neural networks,”
- [8] Y. Chen, S. Biookaghazadeh, and M. Zhao, “Exploring the capabilities of mobile devices in supporting deep learning,” in *Proceedings of the 4th ACM/IEEE Symposium on Edge Computing*, pp. 127–138, 2019.
- [9] S. Zhou, Y. Wu, Z. Ni, X. Zhou, H. Wen, and Y. Zou, “Dorefa-net: Training low bitwidth convolutional neural networks with low bitwidth gradients,” *arXiv preprint arXiv:1606.06160*, 2016.
- [10] J. Choi, Z. Wang, S. Venkataramani, P. I.-J. Chuang, V. Srinivasan, and K. Gopalakrishnan, “Pact: Parameterized clipping activation for quantized neural networks,” *arXiv preprint arXiv:1805.06085*, 2018.
- [11] S. K. Esser, J. L. McKinstry, D. Bablani, R. Appuswamy, and D. S. Modha, “Learned step size quantization,” *arXiv preprint arXiv:1902.08153*, 2019.
- [12] J. Lee, D. Kim, and B. Ham, “Network quantization with element-wise gradient scaling,” in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 6448–6457, 2021.
- [13] Y. Boo, S. Shin, J. Choi, and W. Sung, “Stochastic precision ensemble: self-knowledge distillation for quantized deep neural networks,” in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, pp. 6794–6802, 2021.
- [14] Y. Li, M. Shen, J. Ma, Y. Ren, M. Zhao, Q. Zhang, R. Gong, F. Yu, and J. Yan, “Mqbench: Towards reproducible and deployable model quantization benchmark,” *arXiv preprint arXiv:2111.03759*, 2021.
- [15] Y. Bengio, N. Léonard, and A. Courville, “Estimating or propagating gradients through stochastic neurons for conditional computation,” *arXiv preprint arXiv:1308.3432*, 2013.
- [16] Y. Bengio, J. Louradour, R. Collobert, and J. Weston, “Curriculum learning,” in *Proceedings of the 26th annual international conference on machine learning*, pp. 41–48, 2009.
- [17] G. Hinton, O. Vinyals, J. Dean, *et al.*, “Distilling the knowledge in a neural network,” *arXiv preprint arXiv:1503.02531*, vol. 2, no. 7, 2015.
- [18] K. He, X. Zhang, S. Ren, and J. Sun, “Identity mappings in deep residual networks,” in *Computer Vision—ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part IV 14*, pp. 630–645, Springer, 2016.
- [19] D. Eigen and R. Fergus, “Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture,” in *Proceedings of the IEEE international conference on computer vision*, pp. 2650–2658, 2015.
- [20] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, “Mobilenetv2: Inverted residuals and linear bottlenecks,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4510–4520, 2018.
- [21] X. Zhang, X. Zhou, M. Lin, and J. Sun, “Shufflenet: An extremely efficient convolutional neural network for mobile devices,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 6848–6856, 2018.
- [22] F. N. Iandola, S. Han, M. W. Moskewicz, K. Ashraf, W. J. Dally, and K. Keutzer, “Squeezenet: Alexnet-level accuracy with 50x fewer parameters and 0.5 mb model size,” *arXiv preprint arXiv:1602.07360*, 2016.
- [23] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “Imagenet classification with deep convolutional neural networks,” *Advances in neural information processing systems*, vol. 25, 2012.
- [24] A. Krizhevsky, G. Hinton, *et al.*, “Learning multiple layers of features from tiny images,” tech. rep., Citeseer, 2009.
- [25] Y. Le and X. Yang, “Tiny imagenet visual recognition challenge,” *CS 231N*, vol. 7, no. 7, p. 3, 2015.