Balancing Accuracy and Efficiency: A Comparative Study of Knowledge Distillation and Post-Training Quantization Sequences

Daniel Gitelman dgitelman@ucsd.edu
Jiangqi Wu jiw118@ucsd.edu
Vishwak Pabba vpabba@ucsd.edu
Zhiqing Wang zhw055@ucsd.edu
Alex Cloninger acloninger@ucsd.edu
Rayan Saab rsaab@ucsd.edu

1 Introduction

Deep neural networks (DNNs) have become the cornerstone of modern machine learning, achieving unprecedented performance in critical tasks such as medical image diagnosis, autonomous driving, and real-time language translation. For instance, architectures like ResNet50 achieve over 95% top-1 accuracy on benchmark datasets like CIFAR-10, rivaling human-level performance in specific domains. However, these models often demand extensive computational resources: ResNet50, for example, comprises 25.6 million parameters and requires approximately 100 MB of storage, making deployment on resource-constrained platforms—such as mobile devices, IoT sensors, or edge computing systems—prohibitively challenging. To address this gap, model compression techniques have emerged as a vital area of research, aiming to reduce computational and storage overhead while preserving accuracy. Existing methods, such as quantization (reducing numerical precision of weights) and knowledge distillation (transferring knowledge from a large "teacher" model to a compact "student" model), have shown promise individually. However, the interplay between these techniques—particularly their combined impact on accuracy, storage efficiency, and inference speed—remains underexplored. In this project, we propose a hybrid compression strategy that systematically integrates knowledge distillation and post-training quantization to optimize the trade-off between model efficiency and performance. Specifically, we focus on compressing ResNet50. Our methodology involves two stages, first involved in identifying an optimal distillation strategy to maximize accuracy retention when transferring knowledge from ResNet50 to ResNet18, as well as post training quantization to further reduce its memory storage. By evaluating our approach on the CIFAR-10 and CIFAR-100 dataset, we demonstrate that this sequential integration of techniques achieves superior compression rates compared to single methods while maintaining competitive accuracy. Our work provides actionable insights for deploying high-accuracy DNNs in environments with strict storage and computational constraints, contributing to the broader goal of sustainable and accessible AI.

1.1 Technical Background

Model compression is a vital area of research focused on reducing the resource requirements of DNNs without substantially compromising their performance. Two prominent techniques in this domain are quantization and knowledge distillation.

Quantization

Quantization involves reducing the number of bits used to represent each model parameter. By converting 32-bit floating-point representations to lower bit-width formats, quantization reduces both the model size and the computational load during inference.

Knowledge Distillation

Knowledge distillation (KD) is a technique where a smaller, compact student model learns to replicate the behavior of a larger teacher model. The student model is trained using the softened output probabilities of the teacher model, capturing the dark knowledge that

encompasses the teacher's generalization capabilities. KD helps in transferring knowledge effectively, enabling the student model to achieve performance comparable to the teacher model despite having fewer parameters.

1.2 Prior Work

1.2.1 Quantization

Quantization, as a technique for model compression, was first introduced in the 1990s to reduce the computational and memory requirements of neural networks. The core idea behind quantization is to represent model parameters with fewer bits, typically reducing the bit-width used to store weight values from 32-bit floating point precision to lower bit-widths such as 8-bit or even binary values. By doing so, quantization significantly reduces the storage footprint of the model and accelerates inference, without sacrificing too much in terms of model accuracy. Traditional quantization methods generally apply fixed bit-widths uniformly across the entire network, simplifying the process but often resulting in a loss of accuracy due to the indiscriminate application of lower bit-widths, especially in more sensitive layers of the network.

The method of quantization that we focus on in this study is the **Greedy Path Following Quantization (GPFQ)** algorithm, a novel approach introduced by Rayan et al. (Zhang, Zhou and Saab 2023). GPFQ refines traditional quantization techniques by minimizing quantization error iteratively, all while preserving high model fidelity. Unlike conventional methods that apply uniform quantization across all layers, GPFQ adapts the quantization strategy for each layer of the network, balancing precision and performance based on the specific characteristics of each layer. This adaptive, layer-wise strategy is driven by a greedy approach that selects the most appropriate bit-width for each layer, aiming to achieve the best trade-off between compression and accuracy. Through this greedy process, GPFQ is able to maintain or even improve the model's accuracy at lower bit-widths compared to other state-of-the-art quantization methods.

Empirical evaluations have shown that GPFQ consistently outperforms existing quantization techniques, particularly in scenarios where high compression ratios are required. For example, on challenging benchmarks such as ImageNet, GPFQ achieves substantial reductions in model size while maintaining competitive or even superior accuracy. This makes GPFQ particularly valuable in resource-constrained environments, where efficient deployment of deep learning models is critical. By iterating on the quantization process and fine-tuning layer-specific precision, GPFQ offers a more flexible and effective solution for compressing neural networks without sacrificing performance, highlighting its potential for real-world applications in mobile and edge computing devices.

1.2.2 Knowledge Distillation

Knowledge distillation, first introduced by Bucilă et al. in 2006, focuses on training a smaller, more manageable student model to approximate the complex behavior of a larger teacher model through the use of pseudo data generated by ensemble models (Bucilă, Caruana and Niculescu-Mizil 2006). This foundational concept was further refined by Hinton et al. in 2015, who advanced the use of soft targets (or class probabilities) derived from the teacher's output to enhance the student's learning process, revealing a richer spectrum of information than traditional hard targets (Hinton, Vinyals and Dean 2015).

Building on this framework, various methods of knowledge distillation have been developed, which utilize three principal types of source knowledge: Response-based, Feature-based, and Relation-based. The Response-based method, utilized by Hinton et al., extracts information from the final output layer probabilities of the teacher (Hinton, Vinyals and Dean 2015). In contrast, FitNet, introduced by Romero et al., leverages the intermediate feature maps of deep neural networks, marking a significant shift towards Feature-based knowledge extraction (Romero et al. 2015). Furthermore, Park et al. introduced the concept of Relation-based Knowledge Distillation (RKD), which captures structural relationships between data examples using novel losses, thereby significantly enhancing model performance across various tasks (Park et al. 2019).

The application of knowledge distillation in model quantization has been explored in previous research, notably by Elthakeb et al., who demonstrated the benefits of applying knowledge distillation after quantization to enhance the performance of low-precision student models under the guidance of high-precision teacher networks (Elthakeb et al. 2020). In contrast, our study proposes to invert this sequence by initiating the process with knowledge distillation followed by quantization. This methodological shift is intended to assess whether such an approach can yield enhanced model efficiency. Additionally, our research adjusts the number and bit-width of the parameters to optimize performance and extends the investigation to various distillation techniques (with different types of knowledge) to evaluate their impact on the subsequent quantization process. This novel approach seeks to fill a significant gap in the existing literature, offering new insights into the synergies between knowledge distillation and model quantization for improving the deployment efficiency and performance of neural networks.

To identify optimal strategies for integrating knowledge distillation with post-training quantization, we evaluate the following approaches:

- **Simple Knowledge Distillation**: Trains a student model using a hybrid loss combining cross-entropy (ground-truth labels) and KL-divergence (teacher's soft targets), where class probabilities are smoothed via temperature scaling (Hinton, Vinyals and Dean 2015).
- **Deep Mutual Learning (DML)**: Co-trains two peer student models through collaborative optimization, where both minimize a supervised loss and a mutual imitation loss that aligns their predicted class distributions (Zhang et al. 2017).
- **Neural Tangent Kernel (NTK) Approach**: Compresses models via kernel-based parameter space dimensionality reduction, preserving functionality through NTK trans-

formations that approximate infinite-width network dynamics (Gu et al. 2024).

- **Feature Transfer Methods**: Distills knowledge by aligning intermediate layer activations (e.g., FitNets (Romero et al. 2015)) or attention maps between teacher and student, transferring spatial feature importance.

1.3 Current Methods Combining Quantization and Distillation

Quantization-Aware Knowledge Distillation (QKD) is a prominent method that combines the strengths of Quantization-Aware Training (QAT) and Knowledge Distillation (KD) to create efficient and compact deep learning models. Quantization-Aware Training simulates the behavior of quantized weights and activations during the training process. This allows the model to adapt to reduced precision, mitigating the performance degradation that often occurs when deploying quantized models. Knowledge Distillation, on the other hand, transfers knowledge from a high-accuracy teacher model to a smaller student model by minimizing a loss function that aligns the student's outputs with those of the teacher.

In QKD, the student model is not only compact but also explicitly trained to operate under quantization constraints. This integrated approach ensures that the student model benefits from both the adaptability of QAT and the accuracy retention capabilities of KD. Recent advancements have further refined this technique through methods like Adaptive Quantization with Distillation. Here, quantization parameters, such as bit-width, are dynamically adjusted during the distillation process based on the complexity of different model layers. This dynamic adjustment enables more efficient compression while minimizing performance trade-offs. Models trained using these integrated techniques have achieved significant reductions in size and computational requirements while maintaining competitive accuracy, making them well-suited for deployment on resource-constrained devices such as mobile phones and edge devices.

2 Experiment Methods

2.1 Experiment Methods Overview

Our methodology combines knowledge distillation (KD) and post-training quantization to compress ResNet50 into a resource-efficient ResNet18-based model. First, we systematically evaluate KD strategies, including hard-label training (using ground-truth labels) and soft-label distillation (leveraging the teacher's probabilistic outputs), to identify the optimal student model initialization, as well as attempting different knowledge distillation methods. Second, we apply a greedy path-following quantization algorithm to compress the distilled student model to 8-bit precision, iteratively selecting layers for quantization based on accuracy impact. By testing combinations of distillation strategies and quantization sequences, we determine the configuration that maximizes accuracy under strict storage constraints. Since the experiment invovled in combining knowledge distillation as well as quantization, the following section will be divided into knowledge distillation part and quantization part.

2.2 Knowledge Distillation

We experiment with various knowledge distillation techniques to identify the one that optimally integrates with the quantization method. This process involves evaluating different strategies to ascertain the most effective approach for combining knowledge distillation with quantization to enhance model performance.

2.2.1 Mixup Method for Data Generation

Mixup is a data augmentation method we implemented in order to improve the overall performance of the knowledge distillation algorithm which was discussed by Beyer et al. (Beyer et al. 2022). It works by generating new training samples through the linear interpolation of pairs of images and their corresponding labels. Specifically, given two randomly selected samples (x_i, y_i) and (x_j, y_j), Mixup creates a new sample $(\tilde{x}, \tilde{y})$ using the following formulas:

$$\tilde{x} = \lambda x_i + (1-\lambda)x_j$$

$$\tilde{y} = \lambda y_i + (1-\lambda)y_j$$

where λ is drawn from a Beta distribution Beta(α, α). By blending both input data and their labels, Mixup encourages the model to learn smoother decision boundaries, making it more robust to adversarial examples and improving generalization. In our implementation, we applied Mixup to augment the training set before distillation, allowing the student model to learn from a more diverse and interpolated feature space, which ultimately enhanced its performance.

2.2.2 Teacher Model

For this experiment, our teacher model was the pretrained ResNet50, tailored by Eduardo Dadalto for the CIFAR-10 dataset and available on Hugging Face, achieving an accuracy of 92.25% on our testing suite.

2.2.3 Without Knowledge Distillation

Initially, we trained a ResNet18 model using the CIFAR-10 dataset, utilizing cross-entropy loss to quantify the divergence between the predicted and actual labels, independent of any assistance from a teacher model. This training served as a benchmark to assess the effectiveness of Knowledge Distillation. The model attained an accuracy of 88.91%.

2.2.4 Vanilla Knowledge Distillation

We explored traditional knowledge distillation, focusing on compressing the teacher model. During hyperparameter fine-tuning, we discovered that cross-entropy proved more effective

than KL-divergence for our training objectives. Consequently, we predominantly employed cross-entropy, ultimately attaining an accuracy of 91.52%.

2.2.5 Deep Mutual Learning (DML)

We utilized two student models, each based on the ResNet18 architecture and devoid of knowledge distillation techniques. One model was initially pretrained using the CIFAR10 dataset, which logically showed superior performance in the early epochs due to its prior training on the same dataset. The other model was pretrained with the ImageNet dataset. As the training advanced, we calculated loss using both the traditional supervised learning approach, based on label accuracy, and a method involving the comparison of each student's class posterior probabilities using KL divergence. This approach facilitated mutual learning between the models, leading to a convergence in their performance and an overall enhancement in their capabilities. Following this phase, we selected the student model that achieved the highest test accuracy, recorded as 0.8902, for further quantization experiments at various bit sizes.

2.2.6 Decoupled Knowledge Distillation (DKD)

We implemented the Decoupled Knowledge Distillation (DKD) method, introduced by Zhao et al. in CVPR 2022 (Zhao et al. 2022). DKD aims to address the imbalance between the logit-based distillation losses commonly used in knowledge distillation. Traditional methods often combine Kullback-Leibler (KL) divergence between the student and teacher logits with the standard cross-entropy loss, leading to suboptimal learning when the student's training objective becomes overly dependent on the teacher's output distribution. To mitigate this issue, DKD explicitly decouples the distillation loss into two components: target class knowledge distillation (TCKD) and non-target class knowledge distillation (NCKD). TCKD focuses on aligning the student's predicted probability of the correct class with the teacher's, ensuring the student learns the correct classification decision. NCKD, on the other hand, encourages the student to replicate the relative probability distribution of incorrect classes as predicted by the teacher. By adjusting the balance between these two components, DKD provides a more flexible and effective distillation process. This decoupling allows DKD to outperform conventional distillation methods, particularly in cases where the teacher provides overly confident predictions that may misguide the student. The method introduces two hyperparameters, alpha and beta, to control the contributions of TCKD and NCKD, enabling more fine-tuned knowledge transfer through adjusting the weight of these two components.

By experimenting with various baseline values for the α and β parameters, we were able to replicate some of the experiments from the original DKD paper. However, our results differed from the findings reported in the paper. Specifically, we tested α values of 0.5, 1, and 2, alongside β values of 4, 8, and 10. Our observations indicated a positive correlation between higher $\alpha - \beta$ combinations and improved test accuracy. The lowest accuracy, 86.68%, was observed with $\alpha = 0.5$ and $\beta = 10$, whereas the highest accuracy, 90.64%,

was achieved with $\alpha = 2$ and $\beta = 10$ as shown in Figure 2. According to the paper, the best-performing configurations involved α values close to 1 and β values near 8. However, we were unable to replicate these results in our experiments. This discrepancy may stem from differences in our training setup. Notably, we utilized a pretrained ResNet-50 model sourced from Hugging Face, which may have introduced variations in initialization and learning dynamics. Additionally, differences in other hyperparameters and training conditions could have contributed to the observed deviations from the paper's reported performance.

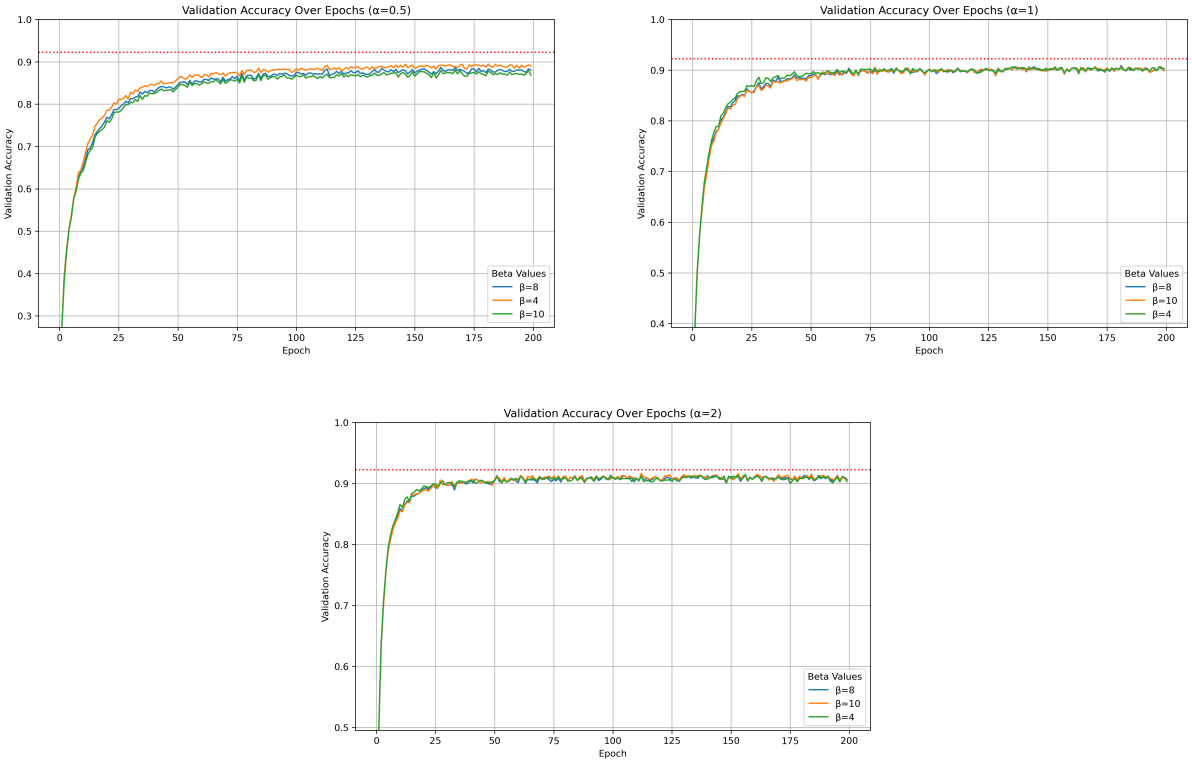

Figure 1: Validation accuracy for knowledge distillation across β values at each α

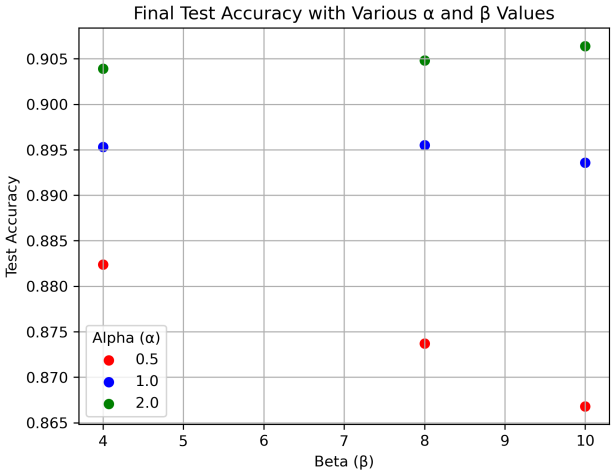

Figure 2: Final test accuracy of each model trained with different combination of α and β.

2.3 Post-Training Quantization

After distilling knowledge into the ResNet18 student model, we apply post-training quantization to further compress its storage footprint and accelerate inference. Our quantization strategy employs a greedy path-following algorithm to iteratively quantize layers to 8 bit. Once all layers are quantized, we evaluate the fully compressed model against the original student as well as the teacher model to compare accuracy, storage reduction systematically.

3 Results

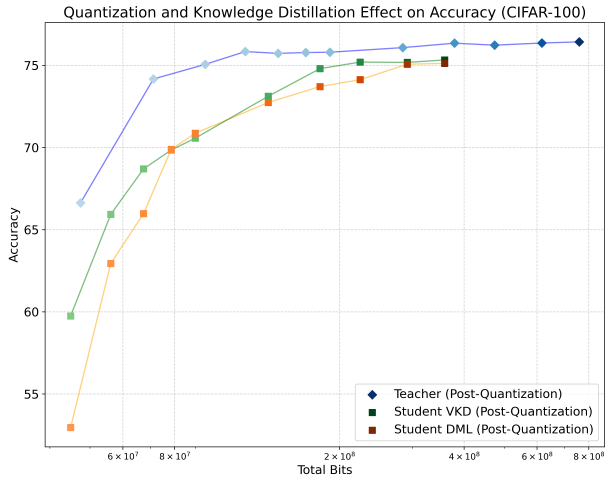

Figure 3: Final test accuracy of each model trained with different combination of α and β.

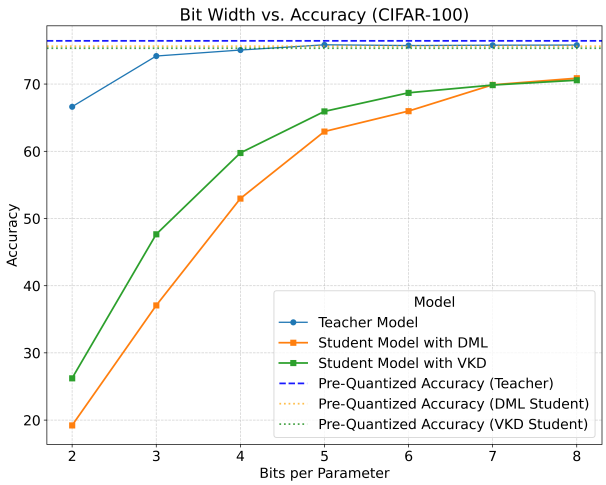

Figure 4: Final test accuracy of each model trained with different combination of α and β.

4 Discussion

5 Conclusion

This study investigated the effects of quantization and knowledge distillation on teacher and student models using CIFAR-100 and CIFAR-10 datasets. For CIFAR-100, the student model exhibited a more pronounced accuracy drop at quantization levels between 2 and 6 bits compared to the teacher. While knowledge distillation aids in addressing the complexities inherent in this dataset, it concurrently reduces the compressibility of the student model for further quantization. For CIFAR-10, the student model also experienced a sharper accuracy drop at 2 to 4 bits, yet both models maintained robustness at higher bit widths, with distillation enabling the student to nearly match its pre-quantization performance. These findings suggest that on less complex datasets, quantizing a student model is more efficient, whereas on more complex datasets, the quantized teacher outperforms the quantized student.

References

Beyer, Lucas, Xiaohua Zhai, Amélie Royer, Larisa Markeeva, Rohan Anil, and Alexander Kolesnikov. 2022. "Knowledge distillation: A good teacher is patient and consistent." [Link]

Bucilă, Cristian, Rich Caruana, and Alexandru Niculescu-Mizil. 2006. "Model compression." In *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. New York, NY, USA Association for Computing Machinery. [Link]

Elthakeb, Ahmed T., Prannoy Pilligundla, Alex Cloninger, and Hadi Esmaeilzadeh.

2020. “Divide and Conquer: Leveraging Intermediate Feature Representations for Quantized Training of Neural Networks.” [Link]

Gu, Lingyu, Yongqi Du, Yuan Zhang, Di Xie, Shiliang Pu, Robert C. Qiu, and Zhenyu Liao. 2024. ““Lossless” Compression of Deep Neural Networks: A High-dimensional Neural Tangent Kernel Approach.” [Link]

Hinton, Geoffrey E., Oriol Vinyals, and Jeffrey Dean. 2015. “Distilling the Knowledge in a Neural Network.” *CoRR* abs/1503.02531. [Link]

Park, Wonpyo, Dongju Kim, Yan Lu, and Minsu Cho. 2019. “Relational Knowledge Distillation.” [Link]

Romero, Adriana, Nicolas Ballas, Samira Ebrahimi Kahou, Antoine Chassang, Carlo Gatta, and Yoshua Bengio. 2015. “FitNets: Hints for Thin Deep Nets.” In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*. [Link]

Zhang, Jinjie, Yixuan Zhou, and Rayan Saab. 2023. “Post-training Quantization for Neural Networks with Provable Guarantees.” [Link]

Zhang, Ying, Tao Xiang, Timothy M. Hospedales, and Huchuan Lu. 2017. “Deep Mutual Learning.” [Link]

Zhao, Borui, Quan Cui, Renjie Song, Yiyu Qiu, and Jiajun Liang. 2022. “Decoupled Knowledge Distillation.” [Link]