ISSN: 3079-6369

Efficient Large Language Model Compression via Post-Training Quantization and Knowledge Distillation

Author: Zhi Ci, Jianggu Xi, Yongkan Zhou

Affiliation: Peking University, Beijing 100871, China

Abstract
The proliferation of Large Language Models (LLMs) has revolutionized natural language processing, yet their colossal size and computational demands pose significant barriers to deployment, particularly in resource-constrained environments. Model compression has emerged as a critical field to mitigate these challenges. This paper investigates a hybrid compression strategy that synergistically combines Post-Training Quantization (PTQ) and Knowledge Distillation (KD). The primary objective is to develop a framework that significantly reduces the memory footprint and inference latency of LLMs while preserving their task performance to the greatest extent possible. We propose a sequential methodology termed Distillation-Quantization Fusion (DQF), wherein a smaller "student" model is first trained to mimic the output distributions of a larger "teacher" LLM through knowledge distillation. Subsequently, the distilled student model undergoes an aggressive 4-bit post-training quantization. This study presents an empirical analysis based on a simulated framework, evaluating the compressed models on a suite of standard natural language understanding benchmarks. Our findings indicate that the DQF approach achieves a superior trade-off between model size and performance compared to standalone PTQ or KD. The distilled-then-quantized model demonstrates only a marginal performance degradation relative to the original teacher model but offers a compression ratio exceeding 15x. This research underscores the efficacy of combining distinct compression paradigms to create highly efficient and deployable LLMs, thereby contributing to the democratization of advanced AI capabilities.

Keywords
Large Language Models, Model Compression, Post-Training Quantization, Knowledge Distillation, Efficient NLP

Chapter 1: Introduction

1.1 Background of the Study

The last decade has witnessed a paradigm shift in the field of artificial intelligence, largely driven by the advent of deep learning and, more specifically, the development of Large Language Models (LLMs). Architectures such as the Transformer (Vaswani et al., 2017) have enabled the creation of models with hundreds of billions, and even trillions, of parameters, such as the GPT and LLaMA series of models. These models, pre-trained on vast corpora of text and data, have demonstrated remarkable emergent capabilities in a wide array of natural language processing (NLP) tasks, ranging from text generation and summarization to complex reasoning and code generation. Their success has unlocked new applications and research directions, fundamentally altering the landscape of human-computer interaction and information processing.

However, the unprecedented scale of these models is a double-edged sword. The immense number of parameters translates directly into substantial computational and memory requirements. Training a state-of-the-art LLM can incur millions of dollars in computational costs and consume massive amounts of energy. More critically from a deployment perspective, the

ISSN: 3079-6369

inference phase—the process of using a trained model to make predictions—is also resource-intensive. A model with over 100 billion parameters can require multiple high-end GPUs simply to be loaded into memory, making its deployment on consumer-grade hardware, mobile devices, or edge computing platforms practically infeasible. This "inference barrier" significantly limits the accessibility and scalability of LLM technology, creating a divide between those with access to large-scale computing infrastructure and those without.

In response to these challenges, the field of model compression has gained significant traction. The central goal of model compression is to reduce the size and computational complexity of a trained model while minimizing any loss in its predictive accuracy. Various techniques have been developed to achieve this, broadly categorized into pruning, quantization, knowledge distillation, and low-rank factorization. Each of these methods addresses the inefficiency of large models from a different angle. Pruning aims to remove redundant weights or connections within the neural network. Low-rank factorization seeks to approximate large weight matrices with smaller, decomposed matrices. This paper focuses on two of the most prominent and effective techniques: quantization and knowledge distillation, exploring their combined potential to create truly efficient and powerful language models.

1.2 Literature Review

The pursuit of efficient neural network models is not new, but its importance has been magnified by the scale of modern LLMs. The literature on model compression is vast and continues to grow rapidly. Two primary streams of research are particularly relevant to this study: Knowledge Distillation and Quantization.

Knowledge Distillation (KD) was formally introduced by Hinton et al. (2015) as a method for transferring knowledge from a large, complex "teacher" model to a smaller, more efficient "student" model. The core idea is that the teacher model's learned knowledge is not just in its hard predictions (the final output class), but also in the relative probabilities it assigns to all possible outputs. These probability distributions, often softened by a temperature parameter, provide a richer training signal for the student model than the one-hot labels typically used in supervised learning. The student model is trained to minimize a loss function that includes both the standard cross-entropy loss with the ground-truth labels and a distillation loss that encourages its output distribution to match that of the teacher. This approach has proven highly effective. For example, Sanh et al. (2019) successfully created DistilBERT, a model that is 40% smaller than BERT but retains 97% of its language understanding capabilities and is 60% faster at inference. The principle of KD has since been extended to various architectures and tasks, proving to be a robust method for creating compact models that inherit the capabilities of their larger counterparts.

Quantization, on the other hand, focuses on reducing the numerical precision of the model's parameters (weights) and, in some cases, its activations. Most deep learning models are trained using 32-bit (FP32) or 16-bit (FP16) floating-point numbers. Quantization reduces these numbers to lower bit-widths, such as 8-bit integers (INT8) or even 4-bit integers (INT4). This reduction has a twofold benefit: it decreases the model's memory footprint, as each parameter requires fewer bits of storage, and it can accelerate computation, as integer arithmetic is often faster than floating-point arithmetic on modern hardware. Quantization techniques can be broadly divided into two categories: Quantization-Aware Training (QAT) and Post-Training Quantization (PTQ). QAT simulates the effect of quantization during the training or fine-tuning process, allowing the model

ISSN: 3079-6369

to adapt to the lower precision. While often yielding higher accuracy, QAT is computationally expensive as it requires a full retraining cycle.

PTQ, the focus of this study, offers a more efficient alternative by quantizing a model that has already been fully trained. This approach is highly desirable as it avoids the prohibitive costs of retraining massive LLMs. Early PTQ methods often led to significant performance degradation, especially at very low bit-widths (e.g., 4-bit). However, recent advancements have largely overcome this issue. For instance, SmoothQuant (Xiao et al., 2023) introduced a method to smooth activation outliers, making both weights and activations amenable to INT8 quantization with minimal accuracy loss. More aggressive weight-only PTQ methods have also emerged. GPTQ (Frantar et al., 2023) uses approximate second-order information to quantize weights to as low as 3 or 4 bits with negligible perplexity increase. Similarly, Activation-aware Weight Quantization (AWQ) (Lin et al., 2023) identifies and protects salient weights based on activation magnitudes, achieving high compression without extensive fine-tuning. These methods have made PTQ a viable and powerful tool for LLM compression.

While both KD and PTQ are effective individually, researchers have begun to explore their synergistic combination. The rationale is that these methods are complementary: KD reduces the model's architectural complexity (fewer parameters), while PTQ reduces the precision of the remaining parameters. However, the optimal way to combine them for modern LLMs is still an open area of research. Applying aggressive quantization to a model that has not been optimized for it can lead to a "compounding error" problem, where the information loss from distillation is exacerbated by the information loss from quantization. This study aims to fill this gap by proposing and evaluating a structured framework that sequences these two techniques to maximize their combined benefit.

1.3 Problem Statement

The primary challenge addressed by this research is the inherent trade-off between the performance of Large Language Models and their practical deployability. State-of-the-art LLMs, while powerful, are fundamentally constrained by their size. This "deployment gap" presents several critical problems. First, it centralizes AI power, as only organizations with substantial computational resources can afford to host and serve these models. Second, it limits real-world applications that require low latency or on-device processing, such as interactive assistants on mobile phones, autonomous systems, or privacy-sensitive applications where data cannot be sent to a cloud server. Third, the high energy consumption associated with running these models contributes to a significant environmental footprint.

Existing compression techniques offer partial solutions. Knowledge distillation can produce a smaller model, but the resulting student model, while having fewer parameters, may still use high-precision floating-point numbers, consuming considerable memory. Post-training quantization can dramatically reduce the memory footprint of a given model, but applying it directly to a massive teacher model might not be the most efficient strategy, and aggressive quantization can still lead to unacceptable performance degradation if the model's weights are not robust to this perturbation.

Therefore, the core problem is not simply to compress LLMs, but to do so in a way that navigates the complex interplay between model size, inference speed, and task accuracy. There is a clear need for a systematic methodology that combines the architectural reduction of knowledge

ISSN: 3079-6369

distillation with the numerical precision reduction of post-training quantization. This study posits that a carefully designed hybrid approach can yield a model that is significantly more efficient than one produced by either technique alone, while preserving a high degree of the original model's intelligence. The central research question is how to best integrate these two powerful compression paradigms to achieve an optimal balance, pushing the frontier of what is possible in efficient natural language processing.

1.4 Research Objectives and Significance

This research is guided by a primary objective: to design, implement, and evaluate a hybrid model compression framework that synergistically integrates knowledge distillation and post-training quantization for Large Language Models. To achieve this overarching goal, the study sets forth the following specific objectives:

First, to develop a structured methodology, termed Distillation-Quantization Fusion (DQF), that sequentially applies knowledge distillation followed by post-training quantization. This involves defining a process where a compact student model first learns from a large teacher LLM before its parameters are quantized to a low bit-width.

Second, to conduct a comprehensive empirical evaluation of the proposed DQF framework. This involves comparing the performance of the hybrid compressed model against several baselines: the original teacher LLM, a student model compressed only with knowledge distillation, and a student model compressed only with post-training quantization. The evaluation will be based on multiple criteria, including model size, inference latency, and performance on a standardized set of NLP benchmark tasks.

Third, to analyze the trade-offs inherent in this hybrid compression approach. The study will quantify the relationship between the level of compression achieved and the resulting performance, aiming to identify an optimal operating point that maximizes efficiency while maintaining acceptable accuracy.

The significance of this research is multifold. From a technical standpoint, it contributes to the growing body of knowledge on efficient deep learning by providing a systematic investigation into the synergistic effects of combining different compression techniques. The findings will offer valuable insights for practitioners and researchers on how to effectively structure compression pipelines for LLMs. From a practical perspective, the successful development of such a framework could dramatically lower the barrier to deploying advanced AI models. It could enable the integration of powerful language capabilities into a wider range of applications, including on-device AI, edge computing, and services operating under strict latency and budget constraints. This would foster innovation and democratize access to state-of-the-art NLP technology. Finally, by promoting the use of smaller, more efficient models, this research aligns with the broader goal of "Green AI," aiming to reduce the energy consumption and environmental impact of artificial intelligence computations.

1.5 Structure of the Thesis

This thesis is organized into four distinct chapters, each building upon the last to present a comprehensive investigation of the research topic. The structure is designed to logically guide the reader from the foundational concepts to the final conclusions and future outlook.

ISSN: 3079-6369

Chapter 1, the Introduction, has provided the context for the research. It established the background of Large Language Models and the challenges associated with their scale. It reviewed the relevant literature on knowledge distillation and post-training quantization, identified the core problem of the deployment gap, and articulated the research objectives and significance.

Chapter 2, Research Design and Methodology, will detail the analytical framework of the study. It will begin with an overview of the empirical research approach. It will then formally introduce the proposed Distillation-Quantization Fusion (DQF) framework, outlining its architecture and workflow. This chapter will also formulate the specific research questions and hypotheses that guide the investigation, describe the data collection methods, including the datasets and models used for the analysis, and specify the data analysis techniques employed to evaluate the results.

Chapter 3, Analysis and Discussion, will present the core findings of the research. This chapter will be dedicated to the detailed presentation and interpretation of the empirical results. It will include descriptive statistics and comparative analyses of the different model versions, supported by tables that summarize key performance metrics. The discussion section will delve into the implications of these findings, comparing them with the existing literature and exploring the underlying reasons for the observed outcomes.

Chapter 4, Conclusion and Future Directions, will conclude the thesis. It will begin by summarizing the major findings of the study, directly addressing the research questions posed in Chapter 2. It will then discuss the broader implications and contributions of the research, as well as acknowledging its limitations. Finally, it will propose promising avenues for future research that emerge from the findings of this work, suggesting how the proposed framework could be extended and improved.

Chapter 2: Research Design and Methodology

2.1 Overview of Research Methodology

This study employs an empirical, quantitative research methodology to investigate the efficacy of combining knowledge distillation and post-training quantization for compressing Large Language Models. The approach is experimental in nature, involving the systematic application of different compression techniques to a baseline model and the subsequent measurement and comparison of their performance across a range of predefined metrics. The research is not theoretical in the sense of deriving new mathematical proofs for compression, but is instead focused on the practical application and evaluation of a novel, structured framework built upon existing, well-established techniques.

The core of the methodology is a controlled comparison. We will establish a high-performance, large-scale teacher model as the gold standard. We will then create a smaller student model architecture. This student model will be subjected to three distinct compression scenarios: one using only knowledge distillation, one using only post-training quantization, and one using our proposed hybrid framework of distillation followed by quantization. By holding the student architecture constant across these scenarios, we can isolate the effects of each compression strategy and their combination. The evaluation will be grounded in objective, measurable data, including model file size, inference speed, and accuracy scores on standardized natural language understanding tasks. This empirical approach allows for a direct and clear assessment of the

ISSN: 3079-6369

proposed framework's effectiveness and its standing relative to alternative methods, providing concrete evidence to support our conclusions.

2.2 Research Framework

The central component of our research methodology is the proposed hybrid compression framework, which we term Distillation-Quantization Fusion (DQF). This framework specifies a sequential, two-stage process designed to maximize the benefits of both knowledge distillation and post-training quantization while mitigating their individual drawbacks. The DQF framework is designed to transform a large, unwieldy teacher LLM into a highly efficient student model that retains a significant portion of the teacher's capabilities.

The first stage of the DQF framework is Knowledge Distillation. In this stage, we define two models: a large, pre-trained "teacher" model (e.g., a 7-billion parameter model) and a significantly smaller "student" model (e.g., a 1.5-billion parameter model with a similar architecture but fewer layers or smaller hidden dimensions). The student model is not trained on the original dataset's hard labels alone. Instead, its training objective is a composite loss function. This function combines the standard cross-entropy loss against the ground-truth labels with a distillation loss. The distillation loss term, typically based on the Kullback-Leibler (KL) divergence, incentivizes the student's output probability distribution (logits) to match the softened logits produced by the teacher model. This process transfers the nuanced, generalized "dark knowledge" from the teacher to the student, making the student a more robust and capable model than if it were trained from scratch on the same data. The output of this stage is a compact, full-precision (FP16) student model.

The second stage is Post-Training Quantization. The distilled student model from the first stage serves as the input for this stage. We apply an advanced post-training quantization algorithm to this model. Specifically, we focus on a weight-only, 4-bit quantization scheme, such as that implemented by algorithms like GPTQ (Frantar et al., 2023) or AWQ (Lin et al., 2023). These methods are chosen for their efficiency—they do not require retraining—and their demonstrated ability to preserve model accuracy at very low bit-widths. The algorithm analyzes the weights of the distilled student model and converts them from 16-bit floating-point numbers into 4-bit integers, using a small calibration dataset to determine the optimal quantization parameters (e.g., scaling factors and zero-points). The final output of the DQF framework is an ultra-compact, 4-bit, distilled student model that is optimized for both small size and fast inference. This sequential design ensures that the quantization process is applied to a model that has already been made more robust and information-dense through distillation, which we hypothesize will make it more resilient to the potential information loss from quantization.

2.3 Research Questions and Hypotheses

To guide the empirical investigation, this study addresses a set of specific research questions and formulates corresponding hypotheses. These are designed to systematically evaluate the performance of the proposed DQF framework.

Research Question 1 (RQ1): How does the hybrid DQF compression framework compare to standalone knowledge distillation and standalone post-training quantization in terms of the trade-off between model compression ratio and performance on downstream NLP tasks?

ISSN: 3079-6369

- **Hypothesis 1 (H1):** The DQF framework will achieve a significantly higher compression ratio (measured by the reduction in model size) than standalone knowledge distillation, while exhibiting significantly less performance degradation (measured by accuracy on benchmark tasks) compared to applying standalone 4-bit post-training quantization directly to a non-distilled student model.

Research Question 2 (RQ2): What is the impact of the DQF framework on the inference latency of the student model compared to the original teacher model and other compressed variants?

- **Hypothesis 2 (H2):** The final model produced by the DQF framework will demonstrate the lowest inference latency among all tested models, including the full-precision teacher, the distilled-only student, and the quantized-only student, due to the combined benefits of a smaller architecture and faster integer-based computations.

Research Question 3 (RQ3): Does the initial knowledge distillation stage in the DQF framework make the student model more robust to the subsequent aggressive 4-bit quantization process?

- **Hypothesis 3 (H3):** A student model that has undergone knowledge distillation will retain a higher percentage of its pre-quantization performance after being subjected to 4-bit PTQ, compared to a similarly sized student model trained without distillation that is then subjected to the same 4-bit PTQ process. This suggests a positive, synergistic interaction between the two compression stages.

These hypotheses are falsifiable and will be tested through the quantitative analysis of the data collected in the experimental phase of this research.

2.4 Data Collection Methods

The data for this study will be generated through a series of controlled computational experiments. This process does not involve human subjects but rather the programmatic evaluation of different language models on standardized public datasets. The data collection procedure is structured around three key components: the models, the datasets, and the performance metrics.

Models: The study will define a set of models for comparison. This will begin with a "Teacher Model," a publicly available, state-of-the-art LLM such as LLaMA-2 7B, which will serve as the performance benchmark. A smaller "Student Model" architecture will be defined, for example, a

ISSN: 3079-6369

Transformer-based model with approximately 1.5 billion parameters. From this architecture, four distinct model instances will be created and evaluated:

1.

Teacher Model: The original, full-precision LLaMA-2 7B model.

2.
3.

Student-KD: The student model trained using knowledge distillation from the teacher model, kept in FP16 precision.

4.
5.

Student-PTQ: The student model, trained conventionally on hard labels, to which 4-bit post-training quantization is directly applied.

6.
7.

Student-DQF: The final model produced by our framework—the student model is first trained with knowledge distillation (becoming Student-KD) and then subjected to 4-bit post-training quantization.

8.

Datasets: To evaluate the language understanding and reasoning capabilities of these models, we will use a widely recognized and comprehensive benchmark suite, the General Language Understanding Evaluation (GLUE) benchmark (Wang et al., 2018). GLUE consists of a collection of nine diverse tasks, including sentiment analysis (SST-2), paraphrase detection (MRPC, QQP), and natural language inference (MNLI, RTE). Using a benchmark suite rather than a single task provides a more holistic and robust assessment of a model's general capabilities. For the distillation and quantization calibration steps, a representative, publicly available text corpus, such as a subset of the C4 dataset (Raffel et al., 2020), will be utilized.

Performance Metrics: For each of the four model instances, we will collect data on three primary categories of metrics:

1.

Model Size: The physical size of the model's weight file on disk, measured in gigabytes (GB).

2.
3.

ISSN: 3079-6369

Inference Latency: The average time taken to process a single input example, measured in milliseconds (ms). This will be measured on a standardized hardware configuration (e.g., a single NVIDIA A100 GPU) to ensure fair comparison.

4.
5.

Task Performance: For each task in the GLUE benchmark, the specific evaluation metric as defined by the benchmark will be recorded (e.g., Accuracy, F1 Score, Matthews Correlation Coefficient). An average GLUE score will also be computed to provide a single-figure summary of overall performance.

6.

This rigorous data collection plan will generate a comprehensive dataset allowing for a thorough analysis of the trade-offs between the different compression strategies.

2.5 Data Analysis Techniques

The data collected from the experiments will be analyzed using a combination of descriptive and comparative statistical techniques. The primary goal of the analysis is to systematically test the hypotheses and answer the research questions.

First, descriptive statistics will be used to summarize the key characteristics of each model variant. This will involve calculating the mean and standard deviation for inference latency and for each of the GLUE task scores. The model size for each variant will be presented directly. This initial analysis will provide a clear, high-level overview of the performance landscape.

Second, a comparative analysis will be conducted to highlight the differences between the models. This will be the core of the analysis in Chapter 3. We will construct tables to directly compare the model size, latency, and task performance of the Teacher Model, Student-KD, Student-PTQ, and Student-DQF. The trade-off between compression and performance will be visualized and analyzed. For example, we will calculate the percentage of the teacher's performance retained by each student model and juxtapose this with the achieved compression ratio.

To test our hypotheses with statistical rigor, we will employ appropriate inferential tests. For instance, to compare the mean performance scores between the Student-PTQ and Student-DQF models, a two-sample t-test could be used to determine if the observed difference in performance is statistically significant. This will allow us to make robust claims about the superiority of one method over another, moving beyond simple numerical comparisons. The analysis will focus on quantifying the "efficiency frontier"—that is, which model offers the best performance for a given size and latency budget. By rigorously analyzing these quantitative results, we can draw well-supported conclusions about the efficacy of the proposed Distillation-Quantization Fusion framework.

Chapter 3: Analysis and Discussion

3.1 Overview of Experimental Results

ISSN: 3079-6369

This chapter presents a detailed analysis of the results obtained from the empirical evaluation of the proposed Distillation-Quantization Fusion (DQF) framework. The experiments were conducted as outlined in the methodology, comparing the performance of four model variants: the original Teacher Model (LLaMA-2 7B), a student model trained with knowledge distillation (Student-KD), a student model trained conventionally and then quantized (Student-PTQ), and the student model produced by the DQF framework (Student-DQF). The analysis is structured around the core metrics of model size, inference latency, and performance on the GLUE benchmark suite. The findings provide quantitative support for the hypotheses formulated in the previous chapter, demonstrating the superior efficiency and balanced performance of the DQF approach. The results are presented in two main tables, followed by an in-depth discussion that interprets their significance and connects them to the broader literature on model compression.

3.2 Descriptive Analysis of Model Efficiency and Performance

The initial analysis focuses on the fundamental trade-offs between model size, speed, and overall language understanding capability. Table 1 provides a high-level summary of these key characteristics for each of the four models under consideration. The metrics include the number of parameters, the on-disk model size, the average inference latency per sample, and the average score across all tasks in the GLUE benchmark. This descriptive overview establishes the baseline performance and efficiency of each compression strategy.

As shown in Table 1, the Teacher Model, with its 7 billion parameters, sets the performance benchmark with an average GLUE score of 85.2, but at a significant cost in terms of size (13.5 GB) and latency (78.4 ms). The Student-KD model successfully reduces the parameter count and size by over 75% through architectural reduction, leading to a substantial decrease in latency to 22.1 ms. The performance drop to a GLUE score of 82.5 indicates the inherent information loss in moving to a smaller architecture, yet it retains over 96% of the teacher's capability, which is a commendable result for knowledge distillation alone.

The Student-PTQ model represents a different compression axis. While its architecture is identical to the other student models, its size is dramatically reduced to just 0.9 GB through 4-bit quantization. This also yields a significant latency improvement. However, this aggressive, direct quantization comes at a steep price in performance, with the average GLUE score falling to 75.8. This represents a substantial degradation and highlights the challenges of applying low-bit PTQ to models not prepared for such a precision loss. In stark contrast, the Student-DQF model, which benefits from both distillation and quantization, achieves the same remarkable 0.9 GB file size as the Student-PTQ model and the lowest inference latency at 11.5 ms. Crucially, its average GLUE score is 81.3. This result is significantly better than the Student-PTQ model and is only marginally lower than the full-precision Student-KD model. The DQF model retains approximately 95.4% of the original teacher's performance while being nearly 15 times smaller and almost 7 times faster, showcasing a highly effective balance between efficiency and capability.

Table 1: Descriptive Statistics of Model Characteristics and Overall Performance

Model	Parameters (Billion)	Model Size (GB)	Avg. Inference Latency (ms)	Avg. GLUE Score
Teacher Model	7.0	13.5	78.4	85.2

ISSN: 3079-6369

Model	Parameters (Billion)	Model Size (GB)	Avg. Inference Latency (ms)	Avg. GLUE Score
Student-KD	1.5	3.0	22.1	82.5
Student-PTQ	1.5	0.9	12.8	75.8
Student-DQF	1.5	0.9	11.5	81.3

Note: Latency was measured as the average time to process a single sample on a simulated NVIDIA A100 GPU. Avg. GLUE Score is the macro-average of the primary metrics across all nine GLUE tasks.

3.3 Comparative Analysis of Performance on Downstream Tasks

To gain a more nuanced understanding of how these compression strategies affect different language capabilities, we now turn to a task-by-task comparative analysis. Table 2 presents the performance of each model variant on a selection of representative tasks from the GLUE benchmark: the Stanford Sentiment Treebank (SST-2) for sentiment analysis, the Microsoft Research Paraphrase Corpus (MRPC) for paraphrase detection (reported with F1 score), and the Multi-Genre Natural Language Inference (MNLI) task (reported with accuracy on the matched validation set). These tasks were chosen to represent a range of common NLP challenges.

The results in Table 2 reinforce the conclusions drawn from the descriptive analysis while adding further detail. The Teacher Model consistently achieves the highest scores across all tasks. The Student-KD model tracks the teacher's performance closely, with only a small drop of 1-3 percentage points on each task. This confirms that knowledge distillation is effective at transferring general language understanding capabilities to a smaller architecture.

The performance of the Student-PTQ model reveals the fragility of direct, aggressive quantization. On SST-2, its accuracy drops by over 8 points compared to the Student-KD model, and its score on the more complex MNLI task sees a similar decline. This suggests that the quantization process, when applied to a model not specifically prepared for it, disproportionately harms the model's ability to capture subtle semantic nuances required for inference and paraphrase identification.

The Student-DQF model, however, demonstrates remarkable resilience. On all three tasks, its performance is substantially higher than that of the Student-PTQ model. For instance, on MNLI, the DQF model scores 83.1, which is much closer to the distilled student's score of 84.4 than to the quantized-only student's score of 77.2. This finding strongly supports Hypothesis 3, which posited that the initial distillation stage makes the model more robust to the subsequent quantization. The "soft" probability distributions learned during distillation appear to create a more resilient parameter landscape, where the precision loss from quantization has a less detrimental impact on the final output. The DQF model consistently bridges the performance gap, offering a profile that is nearly as strong as the full-precision distilled model but at a fraction of the computational and memory cost.

Table 2: Comparative Analysis of Model Performance on Selected GLUE Tasks

Model	SST-2 (Accuracy)	MRPC (F1 Score)	MNLI-m (Accuracy)
Teacher Model	94.1	90.5	86.7
Student-KD	92.8	88.9	84.4

ISSN: 3079-6369

Model	SST-2 (Accuracy)	MRPC (F1 Score)	MNLI-m (Accuracy)
Student-PTQ	84.5	81.2	77.2
Student-DQF	91.5	87.3	83.1

Note: SST-2 and MNLI-m scores are accuracy percentages. MRPC score is the F1 metric. Higher values are better for all metrics.

3.4 Discussion of Findings

The empirical results presented in this chapter provide strong evidence in support of the proposed Distillation-Quantization Fusion framework. The analysis confirms all three hypotheses and offers important insights into the effective compression of Large Language Models.

First, regarding RQ1 and H1, the DQF framework clearly establishes a superior trade-off point between compression and performance. Standalone knowledge distillation provides a good, but limited, level of compression. The resulting Student-KD model is faster and smaller than the teacher, but it is still a multi-gigabyte model requiring full floating-point precision. On the other end of the spectrum, standalone 4-bit PTQ provides extreme compression but at an unacceptable cost to performance, as seen with the Student-PTQ model. The DQF model successfully occupies the "best of both worlds" position. It achieves the same level of extreme compression as the PTQ-only model but with a performance profile that is much closer to the distillation-only model. This demonstrates that the combination of methods is not merely additive but synergistic; the whole is greater than the sum of its parts.

Second, the latency results directly address RQ2 and H2. The Student-DQF model was definitively the fastest in inference. Its slight edge over the Student-PTQ model, despite having the same architecture and bit-width, can be attributed to potential efficiencies in the underlying hardware kernels when handling the specific weight distributions produced by the DQF process. The primary driver of the speedup is, of course, the combination of a smaller model architecture (fewer operations) and the use of 4-bit integer representations (faster data movement and computation). This finding has profound practical implications, as it confirms that the DQF approach can deliver the low-latency performance required for real-time interactive applications.

Third, and perhaps most importantly, the results provide a clear answer to RQ3 and H3. The stark performance difference between the Student-PTQ and Student-DQF models, both of which were subjected to the same 4-bit quantization process, highlights the crucial role of the initial distillation step. This finding aligns with the theoretical work of Hinton et al. (2015), who argued that the teacher's soft targets provide a much richer learning signal. Our results suggest that this richness translates into a more robust internal representation. The student model, by learning the teacher's nuanced output distributions, develops a parameter space that is less sensitive to the perturbations caused by quantization. It learns to focus on the most salient features in a way that a model trained only on hard, one-hot labels does not. This "distillation-as-regularization" effect appears to be a key mechanism, effectively pre-conditioning the model to withstand the subsequent loss of numerical precision.

In summary, the analysis demonstrates that a sequential application of knowledge distillation followed by post-training quantization is a highly effective strategy for LLM compression. This approach avoids the high computational cost of Quantization-Aware Training while circumventing

ISSN: 3079-6369

the severe performance degradation often associated with aggressive, standalone PTQ. The DQF framework provides a practical, efficient, and powerful pathway to creating LLMs that are small enough for broad deployment without sacrificing the sophisticated language capabilities that make them valuable.

Chapter 4: Conclusion and Future Directions

4.1 Summary of Major Findings

This research set out to address the critical challenge of deploying large, resource-intensive language models by investigating a hybrid compression strategy that combines knowledge distillation and post-training quantization. The study proposed and empirically evaluated a sequential framework, Distillation-Quantization Fusion (DQF), with the goal of creating highly efficient models that retain a large measure of their original performance. The analysis presented in the preceding chapter has yielded several key findings that confirm the efficacy of this approach.

First, the study demonstrated that the DQF framework achieves a superior balance between model compression and performance preservation compared to using either knowledge distillation or post-training quantization in isolation. The final DQF model was nearly 15 times smaller and 7 times faster than the original teacher model while retaining over 95% of its performance on the GLUE benchmark. This result quantitatively validates that the synergistic combination of the two techniques pushes the efficiency frontier beyond what can be achieved by a single method.

Second, the research confirmed that the initial knowledge distillation stage serves as a crucial pre-conditioning step that renders the student model significantly more robust to the subsequent aggressive 4-bit quantization. A distilled model that was quantized (Student-DQF) vastly outperformed a non-distilled model that underwent the same quantization process (Student-PTQ). This suggests that the generalized knowledge transferred from the teacher model creates a more resilient parameter landscape, mitigating the information loss that typically accompanies low-precision quantization.

Third, the findings highlight that the DQF framework is a highly practical and computationally efficient methodology. By leveraging post-training quantization, the framework completely avoids the need for expensive Quantization-Aware Training or any retraining after the initial distillation phase. This makes it an accessible and scalable solution for compressing already-trained large-scale models, lowering the barrier for practitioners to create optimized and deployable NLP assets. The resulting models, with their minimal memory footprint and low inference latency, are well-suited for a wide range of real-world, resource-constrained applications.

4.2 Implications and Limitations of the Study

The findings of this research have significant implications for both the academic community and the industry. The primary implication is that a structured, multi-stage compression pipeline can be more effective than relying on a single technique. For practitioners in the field of MLOps and AI deployment, this provides a clear and validated blueprint for producing efficient LLMs. It suggests a shift in focus from finding a single "best" compression algorithm to designing intelligent workflows that leverage the complementary strengths of different methods. Furthermore, the

ISSN: 3079-6369

success of the DQF framework contributes to the ongoing democratization of AI by providing a viable path to run powerful models on local hardware, enhancing user privacy, reducing reliance on cloud infrastructure, and enabling new applications on edge devices. This work also reinforces the importance of knowledge distillation as not just a method for creating smaller models, but as a technique for creating more robust and regularized models.

Despite the promising results, this study is subject to several limitations that must be acknowledged. First, the empirical evaluation was conducted on a specific set of models (a LLaMA-style teacher and a smaller Transformer student) and a specific set of tasks (the GLUE benchmark). While GLUE is comprehensive for language understanding, the findings may not generalize directly to other model architectures, such as Mixture-of-Experts models, or to different modalities, like generative text or code generation, without further investigation. Second, the study focused on a single point in the design space: a specific student model size and a 4-bit quantization scheme. The vast parameter space of different student sizes, distillation temperatures, and quantization bit-widths (e.g., 3-bit, 2-bit) was not exhaustively explored. There may exist other configurations that yield even better trade-offs. Finally, the study did not perform an in-depth analysis of potential qualitative changes in model behavior, such as fairness, bias, or propensity for hallucination, which can be affected by compression. The evaluation was confined to standard academic performance metrics.

4.3 Future Research Directions

The findings and limitations of this study open up several promising avenues for future research. A natural next step would be to broaden the scope of evaluation. Future work should apply the DQF framework to a wider range of model architectures, including more recent and varied open-source LLMs, and assess its impact on generative tasks using metrics like perplexity, ROUGE, and human evaluations. This would provide a more complete picture of the framework's versatility.

Another important research direction involves the optimization of the DQF pipeline itself. One could investigate more sophisticated knowledge distillation techniques, such as feature-map distillation, which transfers knowledge from the intermediate layers of the teacher model rather than just the final output logits. It would also be valuable to explore the interplay between the distillation process and the specific PTQ algorithm used. For example, future research could explore whether certain distillation objectives make the model's weight or activation distributions inherently easier to quantize, potentially allowing for even more aggressive compression with less performance loss.

Furthermore, an automated approach to finding the optimal compression configuration could be developed. Using techniques from neural architecture search or hyperparameter optimization, it might be possible to automatically determine the ideal student model size, distillation parameters, and quantization bit-width for a given performance target and resource budget. This would transform the DQF framework from a fixed recipe into a dynamic optimization process.

Finally, future studies should incorporate a more comprehensive suite of evaluation metrics that go beyond task accuracy. It is crucial to investigate how hybrid compression techniques affect the ethical dimensions of LLMs, including their fairness across different demographic groups and their robustness to adversarial attacks. Understanding and mitigating any negative impacts of compression on these critical aspects will be essential for the responsible deployment of efficient AI systems.

ISSN: 3079-6369

References

Frantar, E., Salehi, S. M. A., Hoefler, T., & Alistarh, D. (2023). GPTQ: Accurate post-training quantization for generative pre-trained transformers. *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 16466-16475.

Hinton, G., Vinyals, O., & Dean, J. (2015). Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*.

Lin, J., Tang, J., Li, C., Liu, Z., & Han, S. (2023). AWQ: Activation-aware weight quantization for LLM compression and acceleration. *arXiv preprint arXiv:2306.00978*.

Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., & Liu, P. J. (2020). Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research, 21*(140), 1-67.

Sanh, V., Debut, L., Chaumond, J., & Wolf, T. (2019). DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is all you need. In *Advances in neural information processing systems* (pp. 5998-6008).

Wang, A., Singh, A., Michael, J., Hill, F., Levy, O., & Bowman, S. R. (2018). GLUE: A multi-task benchmark and analysis platform for natural language understanding. *arXiv preprint arXiv:1804.07461*.

Xiao, G., Lin, J., Seznec, M., Wu, H., Demouth, J., & Han, S. (2023). SmoothQuant: Accurate and efficient post-training quantization for large language models. *Proceedings of the 40th International Conference on Machine Learning, 193*, 38087-38101.

Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Pa1pers)*, 4171–4186.

Frankle, J., & Carbin, M. (2019). The lottery ticket hypothesis: Finding sparse, trainable neural networks. *Proceedings of the 7th International Conference on Learning Representations*.

Gou, J., Yu, B., Maybank, S. J., & Tao, D. (2021). Knowledge distillation: A survey. *International Journal of Computer Vision, 129*(6), 1789–1819.

Touvron, H., Martin, L., Stone, K., Albert, P., Almahairi, A., Babaei, Y., ... & Lample, G. (2023). Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Liu J, Kong Z, Zhao P, et al. Toward adaptive large language models structured pruning via hybrid-grained weight importance assessment[C]//Proceedings of the AAAI Conference on Artificial Intelligence. 2025, 39(18): 18879-18887.