A Collaborative Multi-Compression Acceleration Mechanism for Neural Networks in Keyword Spotting

Junbang Jiang , Rui Pu , Jin Li * , Man Zhu *

Posted Date: 15 May 2026

doi: 10.20944/preprints202605.1063.v1

Keywords: keyword spotting; neural network; model compression; mixed-precision quantization; structured pruning; knowledge distillation

 doi:10.20944/preprints202605.1063.v1

Article

A Collaborative Multi-Compression Acceleration Mechanism for Neural Networks in Keyword Spotting

Junbang Jiang [1], Rui Pu [1], Jin Li [1,*] and Man Zhu [2,3,*]

[1] National"111 Research Center"Microelectronics and Integrated Circuits, School of Science, Hubei University of Technology, Wuhan 430068, China

[2] Intelligent Transportation Systems Research Center, Wuhan University of Technology, Wuhan 430063, China

[3] State Key Laboratory of Maritime Technology and Safety, Wuhan University of Technology, Wuhan430063, China

* Correspondence: lijin@hbut.edu.cn (J.L.); man.zhu.393@whut.edu.cn (M.Z.)

Abstract

To address the large model size, high computational cost, and limited deployment resources of keyword spotting models on edge platforms, this study proposes a collaborative multi-compression acceleration framework for lightweight deployment. Built on an end-to-end convolutional neural network for keyword spotting, the framework integrates adaptive structured pruning, hardware-friendly mixed-precision dynamic quantization, and quantization-aware multi-stage knowledge distillation into a unified compression pipeline. To eliminate the influence of inconsistent training budgets and data partitions across different compression branches, the results of quantization, pruning, distillation, and joint compression are reorganized under a unified evaluation protocol with multi-seed mean ± std reporting. Under this protocol, the retrained baseline reaches 97.13% ± 0.85. Experimental results show that, in the quantization branch, MPDQ achieves 95.78% ± 1.69 with a compression ratio of 9.56×, demonstrating the most favorable balance between accuracy and storage efficiency; in the pruning branch, AIASP reaches 95.63% ± 0.67 at 30% sparsity with a compression ratio of 1.43×, indicating a balanced compromise between accuracy retention and stability; in the distillation branch, PMKD, Multi-Teacher KD, and Fixed-T KD achieve 96.81% ± 0.69, 95.99% ± 1.18, and 96.70% ± 0.74, respectively, showing that the student model can maintain strong recognition performance under approximately 4× structural compression; and the final joint compression scheme reaches 96.16% ± 0.53 with a trade-off score of 4.26 at a compression ratio of 9.89×. These results indicate that the main advantage of collaborative multi-compression lies in achieving a more balanced optimization among accuracy, model size, and compression efficiency under stringent deployment constraints.

Keywords: keyword spotting; neural network; model compression; mixed-precision quantization; structured pruning; knowledge distillation

1. Introduction

In recent years, with the rapid development of intelligent speech interaction technologies, keyword spotting (KWS) has become a key front-end component in smart homes, in-vehicle terminals, wearable devices, and edge speech nodes [1–3]. Compared with large-vocabulary continuous speech recognition tasks, KWS is typically oriented toward limited-vocabulary and always-listening scenarios and therefore places stronger emphasis on low latency, low power consumption, and high reliability. Consequently, efficient deployment on resource-constrained platforms has become an important research topic [1–3]. The introduction of the Speech Commands

 doi:10.20944/preprints202605.1063.v1

dataset provided a unified public benchmark for limited-vocabulary speech recognition and keyword spotting, thereby promoting the development of compact speech models and embedded KWS systems [1].

Early research on keyword spotting model design mainly focused on lightweight convolutional neural networks. Representative models such as MatchboxNet reduce parameter count and computational complexity while maintaining recognition performance through depthwise separable convolutions, compact topologies, and edge-oriented design strategies [2]. However, the emergence of self-supervised speech representation models such as wav2vec 2.0 and HuBERT has substantially improved feature representation capability, while also increasing model size, storage demand, and inference cost. This trend poses greater challenges for direct deployment on resource-constrained platforms such as FPGAs, MCUs, and mobile devices [4–6]. Consequently, how to reduce model size and computational cost while preserving KWS accuracy and maintaining hardware-friendly deployment characteristics has become a central issue in current research.

To address these challenges, recent studies have mainly explored low-bit quantization, structured pruning, and knowledge distillation. In quantization, binarization and ultra-low-bit modeling have achieved promising results for compact KWS networks, and related work has shown that low-bit representations can further reduce storage overhead and multiply-accumulate complexity [7–10]. In distillation, teacher-student learning frameworks have been widely adopted for performance compensation in compact models and have shown strong potential in device-constrained scenarios, self-supervised representation transfer, and in-memory computing settings [9–13]. In hardware-aware co-design, several studies have jointly considered KWS network structures, quantization precision, and accelerator architectures to improve throughput and energy efficiency on FPGA platforms [10,11,16–18]. These studies provide an important basis for lightweight deployment of KWS models; however, most of them focus on a single compression strategy, and the interaction among multiple compression mechanisms remains insufficiently studied.

In broader speech-model compression research, joint optimization of pruning, distillation, and quantization has gradually become an important trend. Existing studies indicate that the combination of distillation and pruning can effectively compress self-supervised speech models, and structured pruning has also been extended to end-to-end automatic speech recognition models [13–15,18]. In addition, integrated compression, one-pass multi-model compression, and ultra-low-bit mixed-precision quantization suggest that model compression is evolving from conventional staged optimization toward unified modeling and joint search [19–23]. Overall, quantization, pruning, and distillation have each been shown to improve deployment efficiency to some extent, yet their effective collaboration under hardware-oriented deployment constraints remains insufficiently explored.

Current studies still exhibit several limitations. First, many methods focus on a single compression strategy, such as quantization alone or pruning alone, and therefore lack systematic analysis of the interaction among multiple compression mechanisms [7–15]. Second, although some methods achieve high compression ratios, they pay insufficient attention to practical deployment constraints such as on-chip storage, DSP utilization, and memory bandwidth, which limits the translation of algorithmic gains into real acceleration benefits [10,11,16–18]. Third, the combination of low-bit quantization and structured pruning often introduces non-negligible performance degradation, while existing recovery mechanisms still leave room for improvement in training stability, task-specific knowledge transfer, and deployment consistency [9–15]. It is therefore of both theoretical and practical interest to develop a collaborative multi-compression method that jointly balances model accuracy, compression efficiency, and deployment friendliness.

Motivated by these issues, this study proposes a collaborative multi-compression acceleration method for efficient deployment of KWS models on resource-constrained platforms. The proposed framework integrates hardware-friendly mixed-precision dynamic quantization, adaptive structured pruning, and quantization-aware multi-stage knowledge distillation into a unified optimization pipeline. Through coordinated design across parameter representation, network structure, and knowledge transfer, the framework jointly balances model-size reduction, computational-cost

 doi:10.20944/preprints202605.1063.v1

reduction, and accuracy preservation. The main contributions are as follows. First, a hardware-friendly mixed-precision dynamic quantization method is proposed for KWS deployment, where Fisher sensitivity, activation outliers, and hardware cost are jointly used for bit-width allocation. Second, an adaptive structured pruning method is introduced, where channel gating, importance evaluation, and local structure preservation are combined to remove redundant channels in a regularized manner. Third, a quantization-aware multi-stage knowledge distillation method is proposed to improve accuracy recovery of compressed models. Fourth, these components are integrated into a unified collaborative optimization pipeline and validated experimentally on the KWS task. The results show that the proposed method effectively reduces model complexity while maintaining high recognition accuracy, providing a practical reference for lightweight deployment of KWS models.

2. Keyword Spotting Model Optimization

The system is built on a lightweight convolutional neural network tailored to keyword spotting. The network consists of a front-end temporal feature extraction module and a back-end discriminative classification module. The front end directly processes raw monaural speech waveforms and uses large receptive-field convolution kernels to model initial band-response patterns. Subsequent layers employ multiple stacked small convolution kernels to progressively extract local temporal patterns and channel-wise features. A fully convolutional structure is maintained throughout the model, thereby avoiding the large parameter overhead of fully connected layers while preserving representation capability and deployment efficiency.

As shown in Figure 1, the system input is a 1-s monaural speech segment after cropping and normalization, with a sampling rate of 16 kHz, consistent with the standard setting of Speech Commands Dataset v2. The first layer adopts relatively large convolution kernels to enhance the capture of local time-domain band patterns, followed by multi-stage convolution and downsampling modules that progressively reduce the temporal dimension and aggregate discriminative information. Batch normalization is applied in each layer to improve training stability. The final classification layer directly outputs the posterior distribution over keyword categories, enabling the model to complete end-to-end keyword recognition with relatively low computational overhead and providing a unified compression target for subsequent pruning, quantization, and distillation.

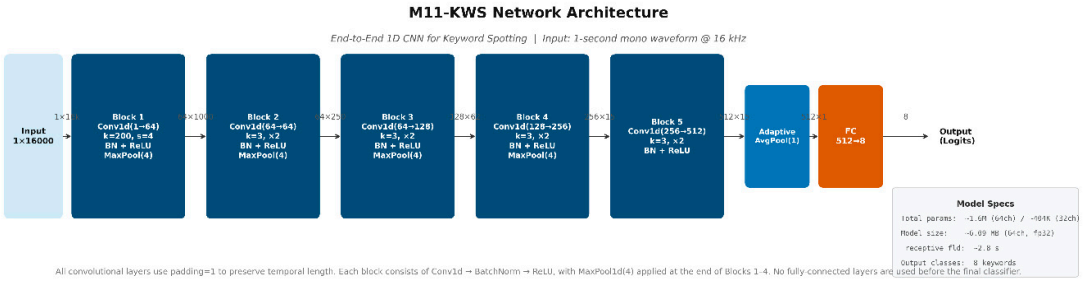


Figure 1. Convolutional neural network architecture for keyword spotting

3. Model Optimization with a Multi-Compression Strategy

To address limited storage resources, constrained computational capability, and imbalanced hardware resource allocation in edge deployment, this study proposes a collaborative multi-compression optimization method for lightweight keyword spotting models. The method is built on three technical components: mixed-precision dynamic quantization (MPDQ), adaptive importance-aware structured pruning (AIASP), and progressive multi-stage knowledge distillation (PMKD). In

 doi:10.20944/preprints202605.1063.v1

addition, hardware-cost constraints, quantization-noise modeling, and joint distillation mechanisms are introduced to perform systematic compression optimization for an end-to-end CNN-based KWS model. Figure 2 illustrates the proposed multi-compression framework.

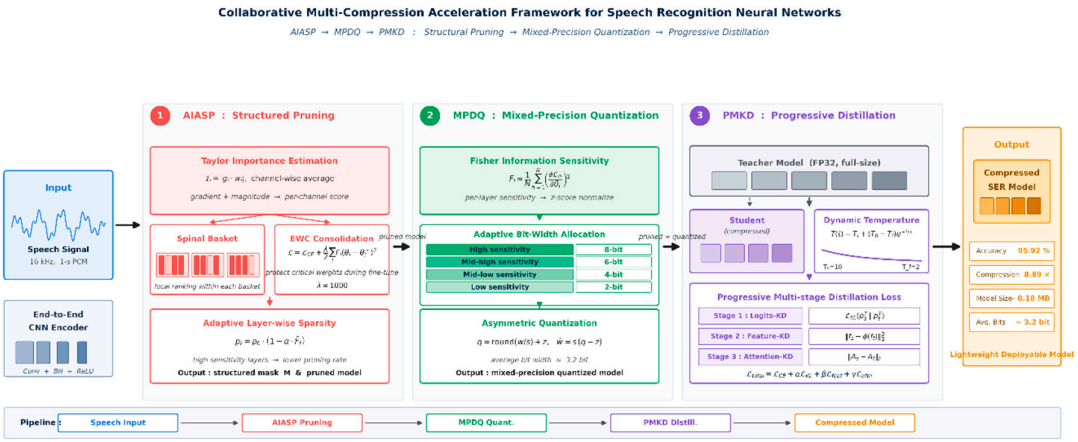


Figure 2. Framework of the multi-compression model optimization strategy

The overall optimization procedure can be divided into three stages, although these stages are not isolated serial modules. First, AIASP performs regularized structured pruning through learnable channel gating, Taylor-based importance modeling, and local structure preservation, while using a latency proxy as feedback to constrain pruning intensity. Second, MPDQ applies hardware-friendly mixed-precision bit-width allocation to the compact network after pruning by jointly considering Fisher information, activation outliers, and a hardware-cost proxy. Third, PMKD introduces quantization-aware distillation and dynamic scheduling under low-bit and sparse structural conditions to recover the accuracy of the compressed model. This process forms a closed-loop compression pipeline involving structural pruning, low-bit adaptation, distillation-based compensation, and hardware feedback, enabling compression ratio, recognition accuracy, and deployment friendliness to be optimized within a unified framework.

3.1. Mixed-Precision Dynamic Quantization

Quantization maps high-precision floating-point parameters into low-bit integer representations and is a key technique for reducing model storage and computational overhead. As KWS networks are increasingly oriented toward edge deployment and lightweight implementation, the role of quantization extends beyond parameter compression and must also account for operator throughput, on-chip storage consumption, and robustness to low-bit noise. Conventional uniform-precision quantization applies the same bit width to all layers and therefore cannot simultaneously protect critical layers and aggressively compress redundant ones. In addition, bit-width allocation methods based only on weight distributions tend to ignore activation outliers and differences in hardware mapping. To address these issues, this study proposes a hardware-friendly mixed-precision dynamic quantization method that jointly considers Fisher sensitivity, activation outlier strength, and deployment cost to obtain bit-width allocation better aligned with practical acceleration requirements.

3.1.1. Sensitivity Modeling

The Fisher Information Matrix (FIM) characterizes the impact of parameter perturbations on the model loss, serving as an effective tool for measuring quantization sensitivity. For a network parameterized by θ, paper adopt the diagonal approximation of the FIM to estimate the importance of individual parameters. The diagonal element of the FIM for the i-th parameter is defined as:

 doi:10.20944/preprints202605.1063.v1

$$F_i = \frac{1}{N}\sum_{n=1}^{N}\left(\frac{\partial L_n}{\partial \theta_i}\right)^2 \tag{1}$$

where N denotes the number of calibration samples and L_n represents the loss function corresponding to the n-th sample. A larger F_i indicates higher sensitivity of the parameter to the model output, thus necessitating the retention of higher precision during low-bit quantization. However, relying solely on the diagonal elements of the FIM for bit-width allocation tends to underestimate the amplification effect of local peak activations in the shallow layers of keyword spotting (KWS) convolutional networks on quantization errors. To address this, paper further introduce an activation outlier statistic, O_l, to characterize the dynamic range fluctuations within the l-th layer, and jointly model it with the intra-layer average Fisher sensitivity. The joint score for the l-th layer is defined as:

$$S_l = \alpha \cdot \mathrm{Norm}\left(\frac{1}{|\Theta_l|}\sum_{i\in\Theta_l} F_i\right) + \beta \cdot \mathrm{Norm}(O_l) \tag{2}$$

where Θ_l denotes the parameter set of the l-th layer, $\mathrm{Norm}(\cdot)$ represents the normalization operation, and α and β are balancing coefficients. This score concurrently reflects the impact of the parameters on the loss function and the amplification effect of low bit-widths on peak responses. At the implementation level, performing forward and backward propagation batch-by-batch on the calibration set to record the squared parameter gradients, layer output peaks, and activation dynamic ranges. Quantile truncation and exponential moving average (EMA) are then employed for robust estimation. Compared to methods that rely solely on weight magnitudes or single-pass statistics for quantization, the proposed strategy provides a more stable identification of critical and vulnerable layers, thereby yielding more reliable priors for subsequent bit-width allocation.

3.1.2. Bit-Width Allocation and Quantization Implementation

Upon obtaining the layer-wise joint sensitivity, it is necessary to map the continuous scores to a discrete set of bit-widths. Unlike conventional partitioning methods based on fixed thresholds, paper formulate the bit-width allocation as a combinatorial optimization problem subject to hardware budget constraints. Subject to the limitations of on-chip memory, DSP utilization, and memory access bandwidth, the optimal configuration for each layer is selected from the candidate bit-width set $\{2,4,6,8\}$ to achieve an optimal trade-off between quantization error and hardware overhead. The objective can be formulated as:

$$\min_{\{b_l\}} \sum_{l=1}^{L}\left(\lambda_1 S_l E_q(b_l) + \lambda_2 C_{hw}(b_l)\right) \tag{3}$$

where b_l denotes the quantization bit-width of the l-th layer, $E_q(b_l)$ represents the quantization error term under the corresponding bit-width, $C_{hw}(b_l)$ denotes the hardware cost function, and λ_1 and λ_2 are balancing coefficients.

Specifically, for layers with high sensitivity that significantly impact the final keyword spotting (KWS) results, 8-bit or 6-bit representations are preferentially retained. For layers with moderate sensitivity that still need to balance compression gains, 4-bit quantization is favored. For layers with low sensitivity that constitute a large proportion of the hardware overhead, the representation can be further reduced to 2-bit. This strategy circumvents two common pitfalls: "over-compression of critical layers" and "excessive resource consumption by low-sensitivity layers," thereby aligning the bit-width allocation more closely with the practical requirements of lightweight deployment. Furthermore, construct a hardware cost proxy based on BRAM utilization, multiply-accumulate (MAC) operations, and parallel unrolling granularity. Consequently, the bit-width selection relies not only on statistical characteristics but also reflects actual deployment constraints.(Table 1).

 doi:10.20944/preprints202605.1063.v1

Table 1. Mixed-precision bit-width allocation rules

Fisher Normalization Range	Sensitivity Level	Assigned Bit-width
[0.75, 1.00]	High Sensitivity	8-bit
[0.50, 0.75)	Medium-High Sensitivity	6-bit
[0.25, 0.50)	Medium-Low Sensitivity	4-bit
[0.00, 0.25)	Low Sensitivity	2-bit

For the quantization implementation, paper employ a per-channel asymmetric integer mapping scheme. The quantization process can be expressed as:

$$q = \text{clip}\left(\text{round}\left(\frac{x}{s}\right) + z, q_{\min}, q_{\max}\right) \tag{4}$$

where x is the input tensor, s is the scaling factor, z is the zero-point offset, and q is the quantized integer value. This approach can preserve richer numerical distribution information under lower bit-width conditions, making it particularly suitable for KWS models with unbalanced output ranges across convolutional layers. To mitigate the degradation of feature representation caused by quantization errors, further integrate per-channel scaling with outlier clipping, ensuring that the error evaluation during the training phase more closely approximates the actual inference environment. Compared to uniform-scale quantization, this strategy better adapts to the dynamic range discrepancies across different convolutional layers and feature channels, thereby enhancing the stability of the compressed model under low-bit deployment conditions.

Overall, the proposed mixed-precision dynamic quantization method transcends simplistic bit-width partitioning based solely on statistical thresholds; instead, it constitutes a joint bit-width optimization mechanism that integrates Fisher sensitivity, outlier awareness, and hardware cost feedback. While preserving KWS accuracy, this method further reduces model storage and computational overhead, establishing a more robust low-bit foundation for subsequent structured pruning and distillation compensation.

3.2. Adaptive Structured Pruning

After baseline training and sensitivity analysis, the model still contains substantial structural redundancy. To further reduce parameter count and computational complexity, this study extends the original structured pruning framework by introducing learnable gating and hardware-latency feedback, allowing redundant channels and filters to be pruned in a regularized manner. Unlike traditional pruning approaches that rely solely on static ranking, the proposed method emphasizes coordinated design across stable importance evaluation, local structure preservation, and deployment-cost constraints, making the pruning results better suited to efficient acceleration on resource-constrained platforms.

3.2.1. Channel Importance Evaluation

The crux of pruning lies in evaluating the importance of each channel to the model output. Traditional weight magnitude-based methods solely focus on parameter size, which fails to accurately reflect the true impact on the loss function. To address this, the Taylor expansion is retained as the theoretical foundation, while the object of importance modeling is extended from individual weights to channel-level gating variables. Let $g_{l,c}$ denote the gating variable corresponding to the c-th output channel of the l-th layer; the change in loss induced by the attenuation of this channel can then be approximated as:

$$\Delta L_{l,c} \approx \left|\frac{\partial L}{\partial g_{l,c}} g_{l,c}\right| \tag{5}$$

where L is the loss function for the keyword spotting (KWS) task. This metric directly reflects the instantaneous impact of channel pruning on model performance, providing an approximation that is more aligned with the actual pruning effect than evaluations based solely on weight magnitudes. Considering the significant gradient fluctuations inherent in single-batch samples, channel activation

 doi:10.20944/preprints202605.1063.v1

energy and layer-level sensitivity information are further incorporated to construct a joint importance score:

$$I_{l,c} = \alpha \left| \frac{\partial L}{\partial g_{l,c}} g_{l,c} \right| + \beta \mathrm{Norm}(A_{l,c}) + \gamma \mathrm{Norm}(F_l) \tag{6}$$

where $A_{l,c}$ represents the average activation energy of the c-th channel in the l-th layer, F_l denotes the Fisher sensitivity statistic of that layer, and α, β, and γ are balancing coefficients. This score simultaneously accounts for gradient sensitivity, channel response strength, and layer vulnerability, thereby enabling a more stable identification of high-contribution channels, redundant channels, and fragile channels sensitive to noise. At the implementation level, forward and backward propagation are performed batch-by-batch on the calibration set, the gating sensitivity is averaged across multiple samples, and the results are integrated with activation statistics through normalized fusion. Compared to methods that rank channels based solely on instantaneous gradients or weight norms, this strategy is more robust to temporal fluctuations and class distribution biases in KWS tasks, providing a more reliable structural prior for subsequent local pruning and recovery training.

3.2.2. Local Gated Pruning

Conventional channel pruning typically employs a global uniform ranking scheme, wherein all channels are sorted by their importance scores and the lowest-scoring channels are directly pruned. Although simple to implement, this method is prone to disrupting local functional structures, particularly causing concentrated damage to consecutive convolutional channels responsible for short-term pattern extraction in KWS networks. To mitigate this, the local partitioning philosophy of Spinal Basket is retained, and a gating competition mechanism is introduced within each local unit, allowing channels to undergo a two-stage "soft selection - hard pruning" filtering process inside local functional blocks. Specifically, the output channels of each layer are partitioned into several local Baskets. Within each Basket, channels are ranked based on their joint importance scores, while the retention ratio is dynamically adjusted in conjunction with layer-wise hardware latency. The local retention ratio for the k-th Basket is defined as:

$$r_{l,k} = r_0 - \eta_1 \mathrm{Norm}(T_l) + \eta_2 \mathrm{Norm}(\bar{T}_{l,k}) \tag{7}$$

where r_0 is the baseline retention rate, T_l denotes the latency cost proxy for the l-th layer, $\bar{T}_{l,k}$ represents the average importance of the k-th Basket, and η_1 and η_2 are regulation coefficients. For latency-sensitive layers with low importance, the local pruning ratio is increased; conversely, for critical layers that contribute significantly to the final KWS accuracy, the pruning intensity is appropriately reduced. This approach not only avoids the collapse of functional units caused by global uniform pruning but also aligns the number of retained channels more closely with the parallel array and cache organization schemes in resource-constrained deployment scenarios. Furthermore, a progressive gated annealing strategy is adopted to execute structured pruning. Specifically, prunable channels are first searched via soft gating, and the gating variables are then progressively forced towards a binary form, deleting the corresponding channels. The gating mechanism can be expressed as:

$$\hat{g}_{l,c}^{(t)} = \sigma\left(\frac{u_{l,c}}{\tau_t}\right) \tag{8}$$

where $u_{l,c}$ is the learnable gating parameter, $\sigma(\cdot)$ is the Sigmoid function, and τ_t is a temperature parameter that decays progressively with training epochs. As τ_t decreases, the gating output gradually approaches a 0–1 form, thereby realizing a smooth transition from soft searching to hard pruning. Compared to one-shot hard pruning, this approach significantly mitigates the accuracy regression induced by structural abruptness and preserves a smoother optimization trajectory for subsequent recovery training.

 doi:10.20944/preprints202605.1063.v1

3.2.3. Stable Recovery Training

The model architecture undergoes alterations following pruning; direct fine-tuning can easily lead to the excessive modification of important parameters, thereby triggering performance degradation. To alleviate this issue, the Elastic Weight Consolidation (EWC) mechanism continues to be employed but is extended as a stabilisation constraint tailored for the compression-recovery phase: on the one hand, the Fisher diagonal information is utilized to protect critical parameters; on the other hand, distillation signals are jointly applied to constrain the representation drift of the low-bit sparse model during recovery training. Equation (2-10) defines the total loss function during the recovery phase:

$$L_{\text{total}} = L_{\text{CE}}(p^S, y) + \lambda_{\text{ewc}} \sum_i F_i(\theta_i - \theta_i^*)^2 + \lambda_{\text{kd}} L_{\text{KD}}(p^S, p^T) \tag{9}$$

where the first term, cross-entropy loss, maintains task supervision; the second term, the EWC regularization term, restricts high-importance parameters from deviating from the original solution; and the third term, the distillation auxiliary term L_{KD}, helps the student model inherit the discriminative boundaries of the pre-pruning teacher network. λ_{ewc} and λ_{kd} denote the weight coefficients for the stabilization and distillation constraints, respectively. It is worth noting that EWC acts exclusively on the retained parameters, whereas the channels eliminated during the gating annealing phase are exempt from this constraint. Through this joint strategy of "gated pruning + EWC protection + distillation recovery," the upgraded AIASP can further reduce model parameters and computational complexity with minimal accuracy loss.(Table 2)

Table 2. Complete pruning procedure of AIASP

Step	Operation	Output	Step
Step 1	Taylor importance calculation	Channel importance scores	Step 1
Step 2	Sensitivity normalization	Normalized importance metrics	Step 2
Step 3	Adaptive pruning rate calculation	Pruning rate for each layer	Step 3
Step 4	Spinal Basket pruning	Generation of structural mask	Step 4
Step 5	EWC weight registration	Saving Fisher information and parameter snapshots	Step 5
Step 6	Fine-tuning training	Model accuracy recovery via EWC + distillation fine-tuning	Step 6

3.3. Quantization-Aware Multi-Stage Knowledge Distillation

After quantization and pruning, the model is substantially compressed; however, discretization error, structural sparsification, and weakened representation capacity may jointly reduce KWS performance. To recover the performance of the compressed model, this study extends the conventional knowledge distillation framework by introducing quantization-aware training and stage-wise knowledge transfer, thereby constructing a multi-stage distillation method for low-bit sparse models. The method considers not only the output-distribution discrepancy between teacher and student models but also the effects of intermediate feature structure and quantization perturbation on the distillation process, enabling better consistency between knowledge transfer and the actual deployment environment.

3.3.1. Response Distillation

The core objective of traditional logits distillation is to make the output distribution of the student model approximate that of the teacher model. However, for low-bit student models, performing distillation solely on the full-precision training trajectory often underestimates the

 doi:10.20944/preprints202605.1063.v1

performance shift induced by quantization noise. To address this, quantization-aware response distillation is introduced in the first stage. Specifically, quantization operators or pseudo-quantization processes are explicitly retained during the forward propagation of the student network, allowing the distillation loss to act directly on the low-bit response space rather than an idealized floating-point space. The distillation objective of this stage is defined as:

$$L_{\text{resp}} = \lambda_{\text{ce}} L_{\text{ce}} + \lambda_{\text{kd}} T^2 D_{\text{KL}}\left(p_i^{(T)} \parallel p_s^{(T)}\right) \tag{10}$$

where L_{ce} is the cross-entropy loss for the keyword spotting (KWS) task, $D_{\text{KL}}(\cdot)$ denotes the Kullback-Leibler (KL) divergence, $p_i^{(T)}$ and $p_s^{(T)}$ represent the soft output distributions of the teacher and student models at temperature T, respectively, and λ_{ce} and λ_{kd} are balancing coefficients. This loss enables the low-bit student model to learn the inter-class relational information from the teacher model while maintaining task discrimination capability, thereby mitigating the destruction of decision boundaries caused by quantization errors.

Compared to traditional single-stage output distillation, quantization-aware response distillation places greater emphasis on the "consistency between the distillation process and the deployment state." Instead of first learning in the floating-point space and subsequently adapting to quantization errors, knowledge transfer is accomplished directly within the compressed environment. Consequently, this approach is more suitable for the optimization of KWS models targeted at resource-constrained deployments.

3.3.2. Feature Transfer

Relying solely on output-layer distillation remains insufficient for fully recovering the internal representation capability of the compressed model. To further enhance the student model's performance, intermediate-layer feature transfer and relational distillation are incorporated in the second stage, enabling the student model to simultaneously approximate the teacher model in terms of temporal feature expression and channel correlation. First, projection layers are employed to align the intermediate feature dimensions between the teacher and the student, and a feature reconstruction loss is applied to constrain local representation consistency:

$$L_{\text{feat}} = \sum_{l=1}^{L_f} \left\| P_l(F_s^l) - F_t^l \right\|_2^2 \tag{11}$$

where F_s^l and F_t^l denote the feature representations of the student and teacher models at the l-th distillation layer, respectively, $P_l(\cdot)$ represents the projection mapping, and L_f is the number of layers participating in distillation. Second, to alleviate the disruption of feature dependencies following structured pruning, relational distillation is further introduced. A Gram matrix is constructed for the features of each layer, and the correlation discrepancy between the teacher and the student is minimized:

$$L_{\text{rel}} = \sum_{l=1}^{L_f} \left\| G\left(P_l(F_s^l)\right) - G(F_t^l) \right\|_F^2 \tag{12}$$

where $G(\cdot)$ denotes the Gram matrix mapping, and $\|\cdot\|_F$ represents the Frobenius norm. This term preserves the relative structural relationships among channels in the teacher model, which is more suitable for knowledge transfer in compression scenarios than mere point-wise alignment. By integrating the constraints of the above two parts, the distillation objective of the second stage can be formulated as:

$$L_{\text{stage 2}} = L_{\text{resp}} + \lambda_{\text{feat}} L_{\text{feat}} + \lambda_{\text{rel}} L_{\text{rel}} \tag{13}$$

where λ_{feat} and λ_{rel} denote the weights for feature distillation and relational distillation, respectively. Through the optimization of this stage, the student model not only learns the final output of the teacher model but also inherits its intermediate-layer representation patterns and class structural information, thereby improving the generalization capability of the compressed model on complex keyword samples.

 doi:10.20944/preprints202605.1063.v1

3.3.3. Dynamic Scheduling

Fixed temperatures and fixed distillation weights often struggle to accommodate the requirements of different training phases. In the early stage of training, the student model has not yet converged, necessitating a higher temperature to obtain smoother inter-class relational information. In the middle stage, feature distillation and relational distillation need to be strengthened to help the model establish stable internal representations. In the late stage, the temperature should be lowered and auxiliary constraints weakened, allowing the model to focus on converging towards the ultimate KWS objective. Based on this, a dynamic temperature and distillation intensity collaborative scheduling strategy is designed. The temperature parameter varies dynamically with the training epochs, expressed as:

$$T(t) = T_{\min} + (T_{\max} - T_{\min})\exp(-t/\tau) \tag{14}$$

where t denotes the current training epoch, $T_{\max}$ and $T_{\min}$ represent the initial and final temperatures, respectively, and τ is the decay constant. As training progresses, the temperature gradually decreases, enabling the distillation process to transition smoothly from "high-temperature soft supervision" to "low-temperature discriminative convergence."

Simultaneously, stage-wise collaborative adjustment is adopted for λ_{feat}, λ_{rel}, and λ_{kd}: the early stage focuses on response distillation, the middle stage enhances feature transfer and relation preservation, and the late stage progressively attenuates the weights of the auxiliary terms. This strategy helps to reduce the sensitivity of fixed hyperparameter schemes to experimental environments, cuts down the cost of manual parameter tuning, and improves the stability and reproducibility of the joint compression training.

In summary, the proposed quantization-aware multi-stage knowledge distillation method enhances the compressed model's capability to inherit teacher knowledge through three mechanisms: response distillation, feature transfer, and dynamic scheduling. This forms a tighter synergy with the aforementioned mixed-precision quantization and structured pruning. Compared to traditional cascaded compression frameworks, this approach better aligns with the current developmental demands for the integrated optimization of lightweight and edge-deployed KWS models.

3.4. Joint Method Coordination

Overall, the model initially performs sensitivity analysis and gated pruning, followed by hardware-constrained bit-width allocation, and subsequently executes quantization-aware multi-stage distillation within the low-bit sparse structure. Compared to the traditional cascaded paradigm of "compress first, remediate later," this pipeline places greater emphasis on the information feedback loop between compression decisions and recovery training.

The synergistic effects among the three components are primarily reflected in three aspects: First, gated structured pruning initially removes redundant channels, thereby reducing the subsequent search space for mixed-precision and enhancing the stability of bit-width allocation. Second, hardware-constrained quantization concentrates resources on critical layers, yielding higher resource utilization efficiency for low-bit deployment. Third, quantization-aware distillation recovers decision boundaries and internal representations under real compression conditions, thereby compensating for the performance degradation caused by the superimposition of pruning and quantization. Ultimately, the joint compression pipeline forms a closed-loop mechanism of "structural compression—bit-width optimization—representation recovery—hardware feedback."(Tables 3 and 4)

 doi:10.20944/preprints202605.1063.v1

Table 3. Multi-stage PMKD distillation strategy

Stage	Knowledge Type	Transfer Form	Loss Function
Stage 1	Output distribution knowledge	Teacher/Student soft logits	$L_{CE} + \lambda 1 \cdot T^2 \cdot KL(p_t \mid\mid p_s)$
Stage 2	Intermediate layer feature knowledge	Intermediate features & Gram relation matrix	$\lambda 2 \cdot L_{feat} + \lambda 3 \cdot L_{rel}$
Stage 3	Attention knowledge (optional)	Temporal/Channel attention mapping	$\lambda 4 \cdot L_{att}$ (optional)

Table 4. Parameter configuration of the joint compression pipeline

Stage	Method	Key Parameters	Expected Effect
Stage 1	AIASP Pruning	Base sparsity 10%-60%, Basket=4, progressive annealing	Reduce parameters, generate structural mask
Stage 2	MPDQ Quantization	Candidate bit-widths {2, 4, 6, 8}, asymmetric, per-layer mixed allocation	Average bit-width ~3.2-bit
Stage 3	PMKD Distillation	, , , epochs=40	Recover compressed model accuracy

In general, through quantization-aware response distillation, relational-level feature transfer, and dynamic collaborative scheduling, the joint pipeline enhances the compressed model's capacity to inherit knowledge from the teacher, establishing a tighter synergy with the previously described hardware-friendly quantization and gated structured pruning. Compared to traditional cascaded compression frameworks, this pipeline better aligns with the developmental trend of integrated optimization for lightweight speech models and edge deployment.

4. Experimental Results and Analysis

Experiments are conducted on Speech Commands Dataset v2 released by Google. Under the unified evaluation protocol adopted in the revised manuscript, the baseline model and all compression branches use the same data split, optimizer, and learning-rate setting, and mean ± std values are reported over multiple random seeds. The retrained baseline reaches 97.13% ± 0.85. Accordingly, all comparisons in Sections 3.1-3.4 are centered on this reference. On this basis, the main and ablation results of quantization, pruning, distillation, and joint compression are synchronized to ensure consistency among figures, tables, and textual conclusions under the same statistical conditions.

The distillation branch is further evaluated under unified teacher and student models with a consistent training budget. Specifically, the teacher model uses 64 channels and 20 epochs, whereas the student model uses 32 channels and 10 epochs. Fixed-T KD, Feature-KD, Multi-Teacher KD, and PMKD are all evaluated on the same student-side architecture, such that the performance differences primarily reflect the knowledge-transfer mechanism itself rather than differences in model size or training protocol. It should be noted that the 20/10-epoch configuration in Table 9 is used for a fair comparison among distillation methods, whereas the 40-epoch setting of PMKD in Table 4 corresponds to the recovery-training stage of the joint compression pipeline; the two therefore belong to different experimental scenarios. Because multi-seed statistics are reported in this section, Table 9 presents mean ± std for all distillation methods to more fully characterize both average performance and result variability under the unified teacher-student setting.

 doi:10.20944/preprints202605.1063.v1

4.1. Quantization Results and Ablation Analysis

4.1.1. Main Quantization Results

Figure 3(a) presents the final accuracy of the main quantization results under the unified benchmark, while Figure 3(b) illustrates the relationship between accuracy and the compression ratio. The results indicate that MPDQ achieves the highest accuracy of 95.78% ± 1.69% among the quantization branches, followed by GPTQ-4-bit at 95.58% ± 0.93% and RTN-4-bit at 95.28% ± 1.59%. Although all three methods fall below the unified baseline of 97.13% ± 0.85%, the performance gap has converged to within 2 percentage points. This demonstrates that under the current, more rigorous baseline reference, the primary distinction among quantization methods is no longer whether they surpass the baseline, but rather their ability to achieve higher compression gains with minimal accuracy degradation.

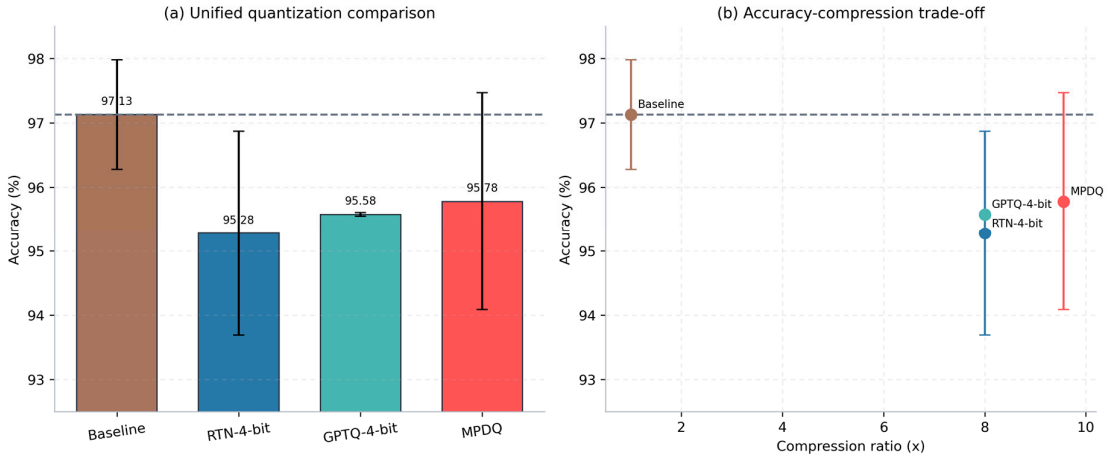


Figure 3. Main results of unified benchmark quantization: Method comparison and accuracy-compression ratio trade-off.

Table 5 further summarizes the accuracy, compression ratio, and model size of each method. In conjunction with Figure 3, it is observed that the average accuracies of GPTQ-4-bit and RTN-4-bit are relatively comparable; however, the standard deviation of GPTQ-4-bit is 0.93%, indicating the presence of certain training fluctuations under the current corrected implementation. In contrast, MPDQ compresses the model to 0.162 MB, corresponding to a 9.56× compression ratio, while maintaining the highest quantization accuracy. As evidenced by the positional distribution in the figure and the model sizes in the table, the mixed-precision bit-width allocation does not merely pursue a higher compression ratio; rather, it achieves a superior comprehensive trade-off between accuracy and storage under the current settings.

Table 5. Unified quantization main results

Method	Configuration	Accuracy (mean±std, %)	Compression Ratio (×)	Model Size (MB)
Baseline	Independently retrained baseline	97.13 ± 0.85	1.00	1.544
RTN-4-bit	4-bit uniform	95.28 ± 1.59	8.00	0.193
GPTQ-4-bit	4-bit corrected GPTQ	95.58 ± 0.93	8.00	0.193
MPDQ	Mixed precision	95.78 ± 1.69	9.56	0.162
RTN-4-bit	4-bit uniform	95.28 ± 1.59	8.00	0.193

 doi:10.20944/preprints202605.1063.v1

4.1.2. Quantization Ablation Analysis

Figure 4(a) demonstrates that the quantization ablation results exhibit a distinct stratification with variations in bit-width. Among them, 8-bit symmetric quantization achieves the highest accuracy of 96.78% ± 1.44%, followed by 8-bit asymmetric quantization at 96.63% ± 0.29%. As the bit-width decreases to 6-bit and 4-bit, the accuracies sequentially drop to approximately 95% and 94%, respectively. This indicates that under the current model and data conditions, bit-width remains the dominant factor affecting quantization accuracy, and the quantization errors introduced by lower bit-widths gradually accumulate in high compression regimes.

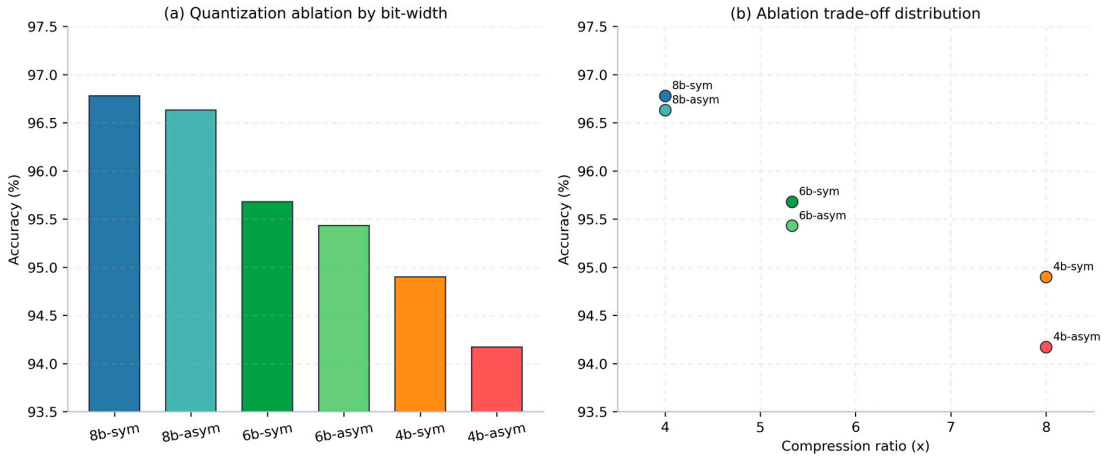


Figure 4. Unified benchmark quantization ablation: Bit-width, quantizer type, and accuracy-compression trade-off.

Table 6 further reveals that the discrepancy between symmetric and asymmetric quantization at the same bit-width is relatively limited. Particularly under the 8-bit configuration, the difference between the two is merely about 0.15 percentage points, suggesting that the zero-point formulation is not the primary source of performance variation in this context. In conjunction with Figure 4, it can be observed that the quantization ablation reflects localized trends under fixed single bit-width configurations, whereas MPDQ in Table 5 achieves the main result of 95.78% ± 1.69% at a 9.56× compression ratio through cross-layer mixed allocation. Therefore, while the quantization ablation uncovers the accuracy trends associated with single-layer bit-width variations, the main results demonstrate the advantages of the mixed-precision strategy in terms of overall deployment efficiency, establishing a complementary relationship between the two.

Table 6. Unified quantization ablation results

Configuration	Accuracy (mean±std, %)
8-bit, symmetric	96.78 ± 1.44
8-bit, asymmetric	96.63 ± 0.29
6-bit, symmetric	95.68 ± 1.25
6-bit, asymmetric	95.43 ± 1.28
4-bit, symmetric	94.90 ± 1.37
4-bit, asymmetric	94.17 ± 1.69

 doi:10.20944/preprints202605.1063.v1

4.2. Pruning Results and Ablation Analysis

4.2.1. Main Pruning Results

Figure 5(a) indicates that, under the unified benchmark and statistics aggregated over five random seeds, AIASP achieves a main result of 95.63% ± 0.67% with a 30% sparsity configuration, outperforming Magnitude, Random, and Gradient, which yield 95.29% ± 0.66%, 94.72% ± 0.35%, and 94.02% ± 1.20%, respectively. Although this result remains lower than the unified baseline of 97.13% ± 0.85%, it maintains the highest mean accuracy among the current pruning comparisons. Figure 5(b) further shows that AIASP corresponds to a compression ratio of 1.43×, indicating that the current main result strikes a balance between moderate structural pruning and relatively stable accuracy performance.

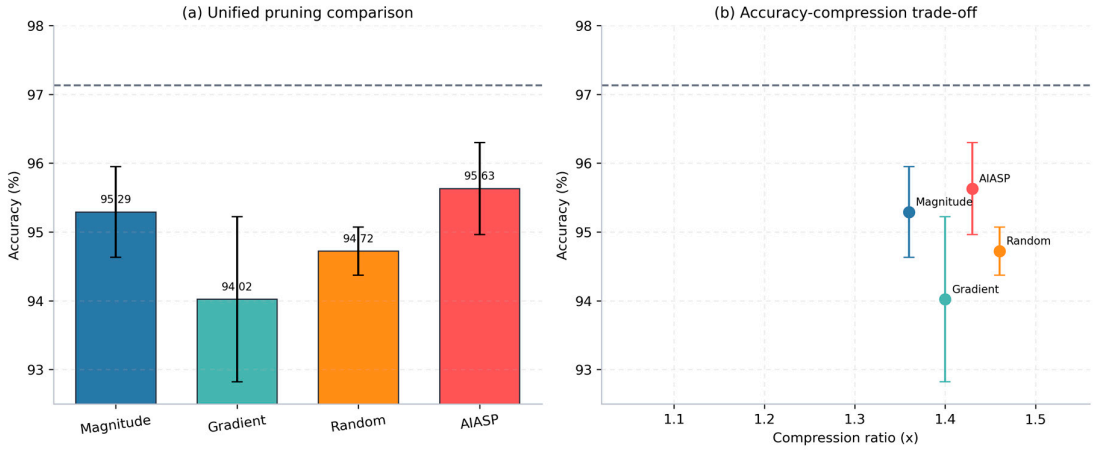


Figure 5. Main results of unified benchmark pruning: Method comparison and accuracy-compression ratio trade-off.

Table 7 summarizes the accuracy, model size, and conclusions for each method. In conjunction with Figure 5, it is evident that AIASP not only preserves the highest mean accuracy within the pruning branch under the 30% configuration but also maintains a standard deviation of 0.67, demonstrating a well-balanced outcome between accuracy and stability. Meanwhile, the model size corresponding to AIASP in the table is 1.081 MB, consistent with the 1.43× compression ratio, which indicates that the current main result does not rely on overly lenient pruning settings. Consequently, it can be concluded that AIASP provides a more representative overall performance at 30% sparsity, establishing a relatively stable structural foundation for subsequent joint compression.

Table 7. Unified pruning main results

Method	Final Accuracy (mean±std, %)	Configuration	Compression Ratio (×)	Model Size (MB)	Conclusion
Magnitude	95.29 ± 0.66	30%	1.43	1.081	Magnitude criterion is effective but with limited retention capability
Gradient	94.02 ± 1.20	30%	1.43	1.081	Gradient criterion exhibits larger fluctuations
Random	94.72 ± 0.35	30%	1.43	1.081	Serves as a lower-bound baseline; overall weaker performance

 doi:10.20944/preprints202605.1063.v1

AIASP	95.63 ± 0.67	30%	1.43	1.081	Demonstrates a more balanced accuracy and stability at the 30% configuration

4.2.2. Pruning Ablation Analysis

Figure 6(a) reveals that AIASP exhibits a generally declining trend as sparsity increases within the base sparsity range of 10% to 60%. Specifically, the 10% configuration achieves the highest accuracy of 96.11% ± 1.61%, followed by the 20% configuration at 95.95% ± 0.52% and the 30% configuration at 95.63% ± 0.67%, while the 60% configuration further decreases to 94.54% ± 1.65%. This indicates that lower sparsity still corresponds to a higher accuracy ceiling; however, from the perspective of main result selection, the 30% configuration exhibits more stable statistical characteristics while maintaining relatively high accuracy. When juxtaposing AIASP-30% with representative pruning methods in Figure 6(b), it can be observed that it still leads the other baseline methods in accuracy, thereby substantiating its representativeness as the current main result.

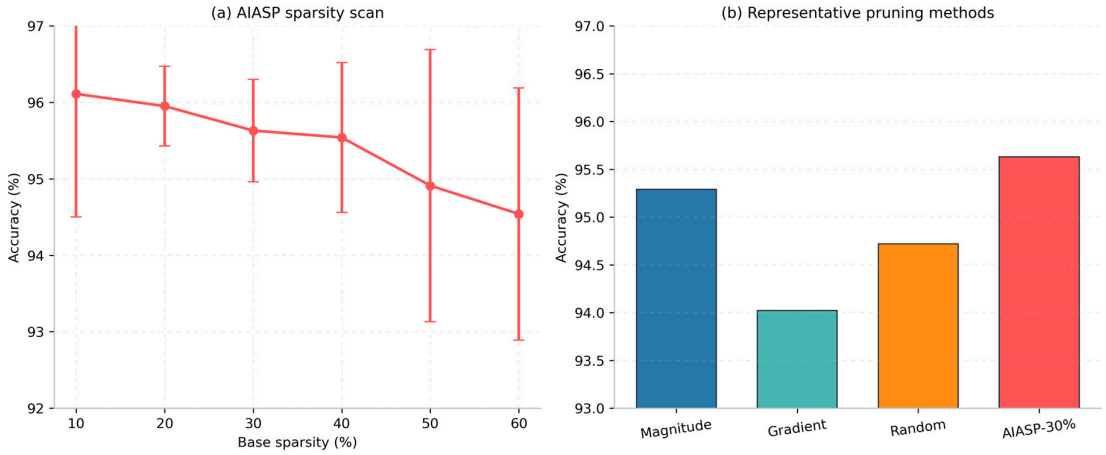


Figure 6. Unified benchmark pruning ablation: AIASP sparsity scanning and comparison with representative methods.

Table 8 further demonstrates that a higher sparsity is not invariably better, as model accuracy generally declines with an increase in pruning intensity. The selection of the main result among different configurations depends not only on the magnitude of the mean accuracy but also requires a consideration of stability and representativeness. Although the 10% and 20% configurations possess higher mean accuracies, the 30% configuration maintains a more balanced level of accuracy and fluctuation at 95.63% ± 0.67%, indicating that the current model can achieve more robust recovery in the moderate sparsity regime. Combining Figure 6 and Table 7, it is evident that the rationale for adopting AIASP-30% as the current main result lies in its superior suitability as a comprehensive representative of the pruning branch, rather than merely serving as a local maximum in the ablation table.

Table 8. Unified pruning ablation results

CONFIGURATION	FINAL ACCURACY (MEAN±STD, %)
AIASP-10%	96.11 ± 1.61
AIASP-20%	95.95 ± 0.52
AIASP-30%	95.63 ± 0.67
AIASP-40%	95.54 ± 0.98
AIASP-50%	94.91 ± 1.78

 doi:10.20944/preprints202605.1063.v1

AIASP-60%	94.54 ± 1.65

4.3. Distillation Results and Ablation Analysis

4.3.1. Main Distillation Results

Table 9. Unified main distillation results (mean±std)

Distillation Method	Accuracy (mean±std, %)
Teacher (64ch, 20ep)	97.13 ± 0.85
Student-Standalone (32ch, 10ep)	95.35 ± 0.91
Fixed-T KD	96.70 ± 0.74
Feature-KD	95.53 ± 0.86
Multi-Teacher KD	95.99 ± 1.18
PMKD	96.81 ± 0.69

Figure 7(a) shows that under the current teacher model and unified student model configurations, the Teacher achieves 97.13% ± 0.85%, while the Student-Standalone reaches 95.35% ± 0.91%. This indicates that merely by reducing the student scale and training epochs, an average performance gap of approximately 1.78 percentage points emerges between the student and teacher models. On this basis, Fixed-T KD and PMKD improve to 96.70% ± 0.74% and 96.81% ± 0.69%, respectively, both surpassing the Student-Standalone. This demonstrates that the distillation mechanism can effectively narrow the performance gap between the teacher and student models. In contrast, Feature-KD yields 95.53% ± 0.86% and Multi-Teacher KD achieves 95.99% ± 1.18%, indicating that while different knowledge transfer approaches bring certain gains, PMKD maintains the highest average result.

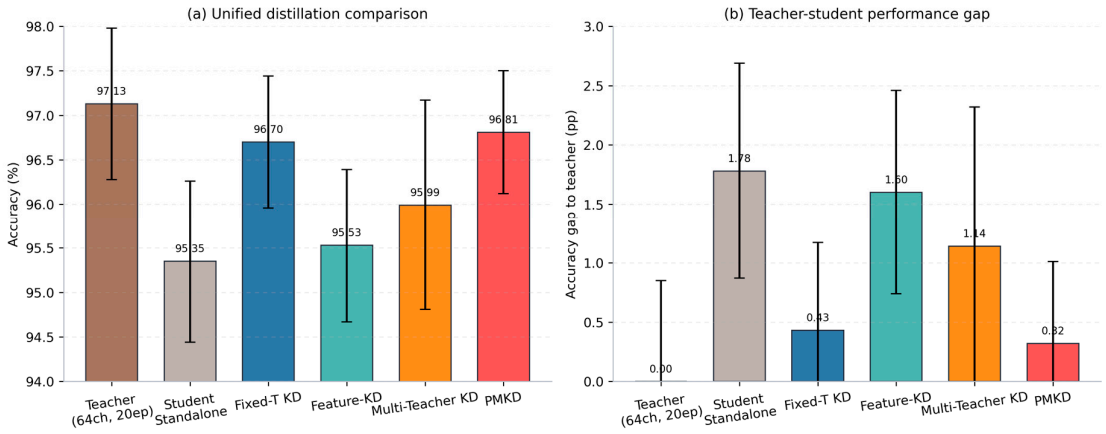


Figure 7. Unified benchmark distillation main results (mean±std): Teacher-student comparison and performance differences among methods.

Table 9 further places the teacher model, the standalone student model, and various distillation strategies under the same benchmark for comparison. In conjunction with Figure 7, it can be observed that the average results of PMKD and Fixed-T KD are highly comparable, differing by only 0.11 percentage points, and their standard deviations are both controlled within 1 percentage point. This indicates that under the current student capacity and training budget, both exhibit relatively stable knowledge transfer effects. Meanwhile, Multi-Teacher KD reaches 95.99% ± 1.18%; although its mean is higher than that of Feature-KD, its fluctuation range is relatively larger, suggesting that multi-teacher information fusion provides certain benefits under the current training protocol but is simultaneously more sensitive to the training process. Overall, under the conditions of a unified

 doi:10.20944/preprints202605.1063.v1

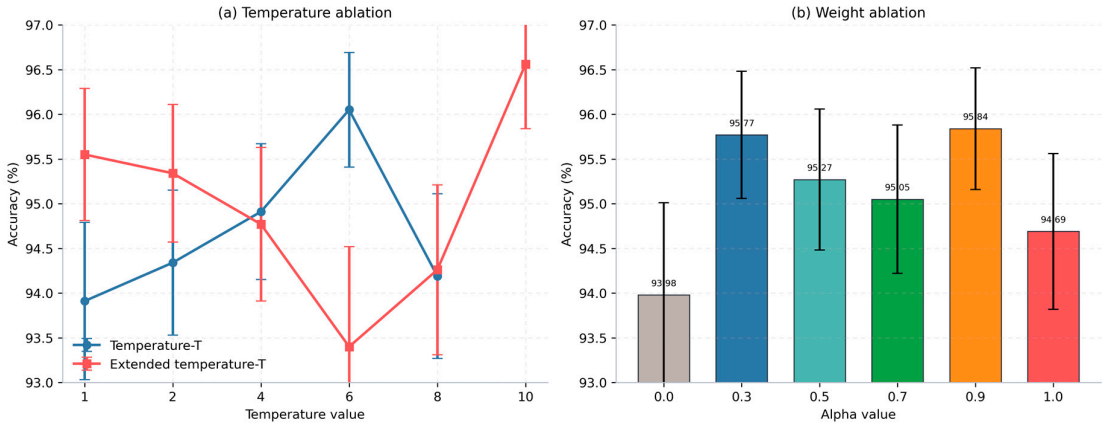
teacher, student, and budget, PMKD, Fixed-T KD, and Multi-Teacher KD all improve the student model's performance, with PMKD maintaining the optimum and exhibiting a superior level of stability.

4.3.2. Distillation Ablation Analysis

Table 10. Unified distillation ablation results (mean±std)

Configuration	Accuracy (mean±std, %)
Temp-T=1.0	93.91 ± 0.88
Temp-T=2.0	94.34 ± 0.81
Temp-T=4.0	94.91 ± 0.76
Temp-T=6.0	96.05 ± 0.64
Temp-T=8.0	94.19 ± 0.92
Weight-=0.0	93.98 ± 1.03
Weight-=0.3	95.77 ± 0.71
Weight-=0.5	95.27 ± 0.79
Weight-=0.7	95.05 ± 0.83
Weight-=0.9	95.84 ± 0.68
Weight-=1.0	94.69 ± 0.87
Extended-Temp-T=1.0	95.55 ± 0.74
Extended-Temp-T=2.0	95.34 ± 0.77
Extended-Temp-T=4.0	94.77 ± 0.86
Extended-Temp-T=6.0	93.40 ± 1.12
Extended-Temp-T=8.0	94.26 ± 0.95
Extended-Temp-T=10.0	96.56 ± 0.72

Figure 8(a) demonstrates that the distillation temperature has a distinct impact on the results, but this effect is not monotonic. In the baseline temperature scan, T=6.0 achieves the highest accuracy of 96.05% ± 0.64%; in the extended temperature scan, T=10.0 further reaches 96.56% ± 0.72%. This indicates that the temperature setting needs to be matched with the current student model capacity, training stage, and scheduling approach, rather than being simply interpreted as "the higher the temperature, the better." It should be noted that the baseline temperature scan and the extended temperature scan in the figure are not repeated experiments under identical configurations; the latter corresponds to extended scheduling settings. Therefore, the discrepancies at the same temperature points between the two scan groups are explainable. Figure 8(b) shows that the optimal point for the distillation weight occurs α occurs at $\alpha = 0.9$, corresponding to $95.84\% \pm 0.68\%$; when $\alpha = 0.0$, the result drops to $93.98\% \pm 1.03\%$, indicating that soft target supervision plays a crucial role in the current distillation task.



 doi:10.20944/preprints202605.1063.v1

Figure 8. Unified benchmark distillation ablation (mean±std): Temperature and distillation weight scanning.

The itemized results in Table 3-6 further reveal that distillation performance is highly sensitive to hyperparameters. In both the baseline and extended temperature scans, the optimal results appear in the mid-to-high temperature regions, suggesting that moderately enhancing the smoothness of soft labels helps improve the student model's knowledge absorption efficiency. The process of increasing α from 0.0 to 0.9 also corresponds to an overall performance uplift, indicating that a reasonable balance must be maintained between the distillation loss and hard label supervision. Combining the mean and standard deviation, it is observable that the regions around T=6.0, T=10.0, and $\alpha = 0.9$ not only exhibit higher mean values but also maintain acceptable fluctuation ranges, demonstrating that these configurations balance performance and stability under the current training budget. In conjunction with Figure 7, it can be inferred that the value of the PMKD method does not lie in relying on a single local hyperparameter peak, but rather in progressively organizing and integrating these effective factors more stably into the complete training pipeline.

4.4. Joint Compression Results Analysis

To uniformly measure the overall performance of different compression branches in balancing accuracy retention and compression gains, the comprehensive score is defined as Score=(Acc/Acc_base)×(1+log2(R)), where denotes the accuracy of the current scheme, represents the unified baseline accuracy, and denotes the compression ratio. Figure 9(a) presents the comprehensive trade-off scores of each scheme under this metric, while Figure 9(b) illustrates their positional relationships in the accuracy-compression ratio space. The results show that the comprehensive score of Final-Joint is 4.26, slightly higher than MPDQ's 4.20, and surpasses the individual scores of standalone AIASP (1.49) and standalone PMKD (2.99). This demonstrates that while maintaining a high compression ratio of 9.89×, joint compression still achieves superior overall deployment benefits compared to any single branch. In the scatter plot, Final-Joint and MPDQ are both located in the high compression ratio region, but the accuracy of Final-Joint improves to 96.16% with the standard deviation converging to 0.53, indicating that joint compression exhibits excellent comprehensive effects in terms of both accuracy and stability.

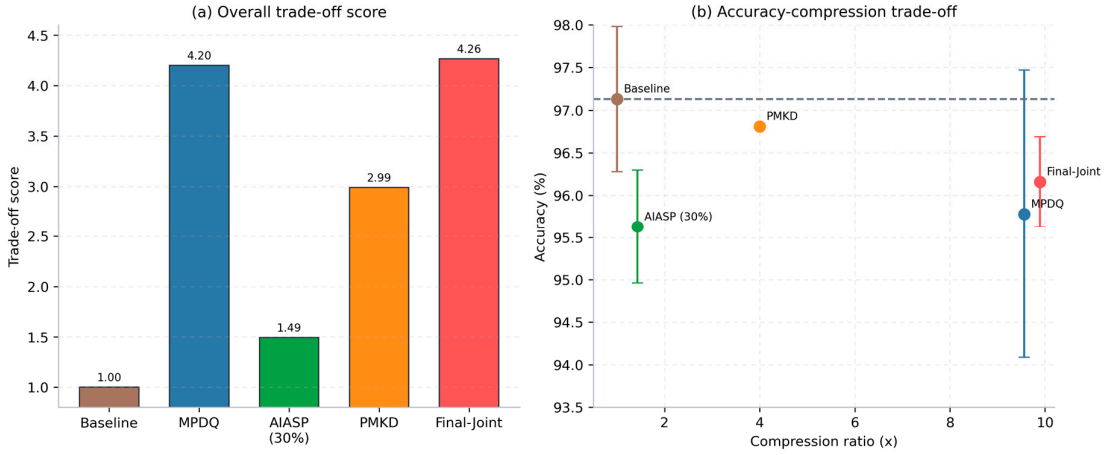


Figure 9. Unified benchmark joint compression results: Comprehensive trade-off score and accura-cy-compression ratio relationship.

Table 11 adopts the individual main results consistent with Sections 3.1-3.3 as references: standalone MPDQ corresponds to 95.78% ± 1.69% and a 9.56× compression ratio; standalone AIASP utilizes the current main result AIASP-30%, corresponding to 95.63% ± 0.67% and a 1.43× compression ratio; standalone PMKD yields 96.81% ± 0.69% and a 4.00× compression ratio; Final-Joint corresponds to 96.16% ± 0.53% and a 9.89× compression ratio. After recalculation using the

 doi:10.20944/preprints202605.1063.v1

aforementioned formula, Final-Joint maintains the highest comprehensive score under the high compression ratio condition, indicating that the objective of joint compression is not to pursue the highest single-item accuracy, but rather to achieve a more balanced accuracy-compression trade-off under a smaller storage overhead. The results in the table also show that MPDQ remains the single scheme with the strongest comprehensive benefit within the quantization branch, whereas Final-Joint further integrates this high compression advantage with the accuracy recovery brought by distillation. Consequently, it demonstrates more complete deployment value under the unified benchmark.

Table 11. Unified joint compression results (accuracy reported as mean±std)

Scheme (Configuration)	Accuracy (mean±std, %)	Compression Ratio (×)	Comprehensive Score
Baseline	97.13 ± 0.85	1.00	1.00
MPDQ	95.78 ± 1.69	9.56	4.20
AIASP (30%)	95.63 ± 0.67	1.43	1.49
PMKD	96.81 ± 0.69	4.00	2.99
Final-Joint	96.16 ± 0.53	9.89	4.26

5. Conclusions

This study investigated a collaborative multi-compression acceleration mechanism for deploying neural network models in keyword spotting tasks on resource-constrained platforms and established a unified optimization framework consisting of mixed-precision dynamic quantization, importance-aware structured pruning, and progressive multi-stage knowledge distillation. To address the difficulty faced by conventional single compression methods in balancing compression ratio, recognition accuracy, and hardware friendliness, coordinated design was carried out from three perspectives: parameter-representation compression, network-structure pruning, and knowledge-transfer compensation. In the final stage, unified data partitions, unified training budgets, and multi-seed statistics were introduced to re-examine the actual performance differences among compression branches.

The unified evaluation results show that the retrained baseline reaches 97.13% ± 0.85. In the quantization branch, MPDQ achieves 95.78% ± 1.69 with a compression ratio of 9.56×, whereas GPTQ-4-bit reaches 95.58% ± 0.93, indicating that its mean difference from RTN-4-bit becomes relatively small after implementation correction. In the pruning branch, AIASP-30% reaches 95.63% ± 0.67 with a compression ratio of 1.43×, indicating that the selected pruning result corresponds to a medium-sparsity configuration emphasizing the balance between accuracy and stability. In the distillation branch, PMKD reaches 96.81% ± 0.69, Multi-Teacher KD reaches 95.99% ± 1.18, and Fixed-T KD reaches 96.70% ± 0.74; under the same student-model protocol, these results indicate that distillation can maintain high recognition accuracy while reducing model scale. In the joint compression branch, Final-Joint reaches 96.16% ± 0.53 and achieves a trade-off score of 4.26 at a compression ratio of 9.89×. Overall, the value of collaborative multi-compression lies not in maximizing any single metric, but in achieving a more balanced trade-off among accuracy, model size, and compression efficiency under stringent high-compression deployment constraints.

Overall, the collaborative multi-compression acceleration mechanism proposed in this work provides a feasible path for the lightweight design and efficient deployment of keyword-spotting neural networks, and may serve as a useful reference for real-time speech processing on edge intelligent hardware.

Author Contributions: Conceptualization, J.J. , R.P. , J.L. and M.Z.; methodology, J.J. , R.P. and J.L.; software, J.J. , R.P. and J.L.; validation, J.J. , R.P. and J.L; formal analysis, J.J. , R.P. and J.L; investigation, J.J. , R.P. and J.L; resources, J.J. , R.P. and J.L; data curation, J.J. , R.P. and J.L; writing—original draft preparation, J.J. , R.P. and J.L; writing—review and editing, M.Z; supervision, M.Z; funding acquisition, M.Z All authors have read and agreed to the published version of the manuscript.

 doi:10.20944/preprints202605.1063.v1

Funding: Guiding project of Scientific Research Plan of Hubei Provincial Department of Education, Design of Application-Specific Integrated Circuit for Voice Command Recognition Based on Compute-in-Memory Architecture (No. B2020046); Cooperation Agreement on the Project of Design and Process Manufacturing Technology of 3D Stacked Code Flash Memory Chip (No. 2021802); PhD Research Foundation Project of Hubei University of Technology, Research on Sun Tracking Control and System State Monitoring of Butterfly Concentrating Photovoltaic Based on Deep Learning Image Analysis (No. 00185).

Data Availability Statement: The Speech Commands Dataset v2 is open to the public and can be obtained from the TensorFlow official repository at https://storage.googleapis.com/download.tensorflow.org/data/speech_commands_v0.02.tar.gz, released by Google's TensorFlow and AIY teams for keyword spotting research.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. WARDEN P. Speech Commands: A Dataset for Limited-Vocabulary Speech Recognition[EB/OL]. arXiv:1804.03209, 2018[2026-05-09]. https://arxiv.org/abs/1804.03209
2. MAJUMDAR S, GINSBURG B. MatchboxNet: 1D Time-Channel Separable Convolutional Neural Network Architecture for Speech Commands Recognition[EB/OL]. arXiv:2004.08531, 2020[2026-05-09]. https://arxiv.org/abs/2004.08531
3. BAEVSKI A, ZHOU H, MOHAMED A, et al. wav2vec 2.0: A Framework for Self-Supervised Learning of Speech Representations[EB/OL]. arXiv:2006.11477, 2020[2026-05-09]. https://arxiv.org/abs/2006.11477
4. HSU W N, BOLTE B, TSAI Y H H, et al. HuBERT: Self-Supervised Speech Representation Learning by Masked Prediction of Hidden Units[EB/OL]. arXiv:2106.07447, 2021[2026-05-09]. https://arxiv.org/abs/2106.07447
5. KIM D H, LEE J H, MO J H, et al. W2V2-Light: A Lightweight Version of Wav2vec 2.0 for Automatic Speech Recognition[C]//Proc. Interspeech 2022. Incheon, Korea: ISCA, 2022: 3038-3042. https://doi.org/10.21437/Interspeech.2022-10339
6. QIN H, MA X, DING Y, et al. BiFSMN: Binary Neural Network for Keyword Spotting[C]//Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence. Vienna: IJCAI, 2022: 4346-4352. https://doi.org/10.24963/ijcai.2022/603
7. WANG X, CHENG S, LI J, et al. Low-complex and Highly-performed Binary Residual Neural Network for Small-footprint Keyword Spotting[C]//Proc. Interspeech 2022. Incheon, Korea: ISCA, 2022: 3233-3237. https://doi.org/10.21437/Interspeech.2022-573
8. HARD A, PARTRIDGE K, CHEN N, et al. Production Federated Keyword Spotting via Distillation, Filtering, and Joint Federated-Centralized Training[C]//Proc. Interspeech 2022. Incheon, Korea: ISCA, 2022: 76-80. https://doi.org/10.21437/Interspeech.2022-11050
9. SONG Z, LIU Q, YANG Q, et al. Knowledge Distillation for In-Memory Keyword Spotting Model[C]//Proc. Interspeech 2022. Incheon, Korea: ISCA, 2022: 4128-4132. https://doi.org/10.21437/Interspeech.2022-633
10. HE K, CHEN D, SU T. A Configurable Accelerator for Keyword Spotting Based on Small-Footprint Temporal Efficient Neural Network[J]. Electronics, 2022, 11(16): 2571. https://doi.org/10.3390/electronics11162571
11. YANG G P, GU Y, TANG Q, et al. On-Device Constrained Self-Supervised Speech Representation Learning for Keyword Spotting via Knowledge Distillation[C]//Proc. Interspeech 2023. Dublin, Ireland: ISCA, 2023. https://www.isca-archive.org/interspeech_2023/yang23y_interspeech.html
12. PENG Y, SUDO Y, MUHAMMAD S, et al. DPHuBERT: Joint Distillation and Pruning of Self-Supervised Speech Models[C]//Proc. Interspeech 2023. Dublin, Ireland: ISCA, 2023: 62-66. https://www.isca-archive.org/interspeech_2023/peng23c_interspeech.html
13. BEKAL D, GOPALAKRISHNAN K, MUNDNICH K, et al. A Metric-Driven Approach to Conformer Layer Pruning for Efficient ASR Inference[C]//Proc. Interspeech 2023. Dublin, Ireland: ISCA, 2023: 4079-4083. https://doi.org/10.21437/Interspeech.2023-2183

 doi:10.20944/preprints202605.1063.v1

14. JIANG H, ZHANG L L, LI Y, et al. Accurate and Structured Pruning for Efficient Automatic Speech Recognition[C]//Proc. Interspeech 2023. Dublin, Ireland: ISCA, 2023. https://www.isca-archive.org/interspeech_2023/jiang23d_interspeech.html
15. WANG H, DU J, ZHOU H, et al. A Multiple-Teacher Pruning Based Self-Distillation (MT-PSD) Approach to Model Compression for Audio-Visual Wake Word Spotting[C]//Proc. Interspeech 2023. Dublin, Ireland: ISCA, 2023: 2678-2682. https://doi.org/10.21437/Interspeech.2023-1717
16. YOON J, KIM N, LEE D, et al. A Resource-Efficient Keyword Spotting System Based on a One-Dimensional Binary Convolutional Neural Network[J]. Electronics, 2023, 12(18): 3964. https://doi.org/10.3390/electronics12183964
17. BAE S, KIM H, LEE S, et al. FPGA Implementation of Keyword Spotting System Using Depthwise Separable Binarized and Ternarized Neural Networks[J]. Sensors, 2023, 23(12): 5701. https://doi.org/10.3390/s23125701
18. BEN LETAIFA L, ROUAS J L. Variable Scale Pruning for Transformer Model Compression in End-to-End Speech Recognition[J]. Algorithms, 2023, 16(9): 398. https://doi.org/10.3390/a16090398
19. PARK E, AHN D, KIM H. RepTor: Re-parameterizable Temporal Convolution for Keyword Spotting via Differentiable Kernel Search[C]//Proc. Interspeech 2024. Kos, Greece: ISCA, 2024: 4518-4522. https://doi.org/10.21437/Interspeech.2024-233
20. GU T, LIU B, SHAO H, et al. SparseWAV: Fast and Accurate One-Shot Unstructured Pruning for Large Speech Foundation Models[C]//Proc. Interspeech 2024. Kos, Greece: ISCA, 2024: 4498-4502. https://doi.org/10.21437/Interspeech.2024-607
21. LI Z, XU H, WANG T, et al. One-pass Multiple Conformer and Foundation Speech Systems Compression and Quantization Using An All-in-one Neural Model[C]//Proc. Interspeech 2024. Kos, Greece: ISCA, 2024: 4503-4507. https://doi.org/10.21437/Interspeech.2024-703
22. GU T, LIU B, WANG H, et al. Ultra-Low Bit Post-Training Quantization of Large Speech Models via K-Means Clustering and Mixed Precision Allocation[C]//Proc. Interspeech 2025. Rotterdam, The Netherlands: ISCA, 2025: 1988-1992. https://doi.org/10.21437/Interspeech.2025-503
23. LI Z, XU H, JIN Z, et al. Towards One-bit ASR: Extremely Low-bit Conformer Quantization Using Co-training and Stochastic Precision[C]//Proc. Interspeech 2025. Rotterdam, The Netherlands: ISCA, 2025: 1973-1977. https://doi.org/10.21437/Interspeech.2025-18