DOMAIN-AGNOSTIC INCREMENTAL AUDIO CLASSIFICATION WITH COMPLEMENTARY INCREMENTAL LEARNING SYSTEMS

Technical Report

Kai Pi[1]*, Yunqi Chen*[1]*, Fan Zhong*[1]*, Jiahui Yin*[1]*, Yike Zhang*[1]*, Shihong Tan*[1]*, Xudong Zhao*[2*]*, and Gongping Huang*[1†]

[1] School of Electronic Information, Wuhan University, Wuhan 430072, China, {pikaipk, chenyunqi, fanzhong, jiahuiyin, 2023302121122, shihongtan, gongpinghuang}@whu.edu.cn
[2] Department of Engineering, King's College London, London WC2R 2LS, U.K. xudong.zhao@kcl.ac.uk

ABSTRACT

This report presents our submissions to Task 7 of the Detection and Classification of Acoustic Scenes and Events (DCASE) 2026 Challenge, which focuses on domain-agnostic incremental learning for audio classification. Our approaches provide complementary strategies to improve classification accuracy by reducing domain-routing errors and enhancing the classification models. Specifically, System 1 uses the challenge-provided D1 checkpoint as a frozen CNN14 feature extractor and performs analytic adaptation via hyperspherical random projection and streaming ridge regression. Systems 2 and 4 are both built on the official CNN-BN baseline and improve domain-agnostic inference through a two-stage prototype router and a triple-resolver mechanism, respectively. System 3 follows the same routing principle as System 2, but adopts a separate CRNN-LoRA architecture for the D2 and D3 domains. Among the four submitted systems, System 4 obtains the best local validation performance, achieving 70.2% average macro accuracy on the local D2/D3 development-test sanity split after selecting the final static resolver variant on that split.

Index Terms— domain-incremental learning, audio classification, domain-agnostic inference, prototype-based routing, continual learning

1. INTRODUCTION

Real-world audio classification systems are often deployed in changing acoustic conditions. A model trained in one recording environment may later encounter new devices, rooms, background noise patterns, or source distributions. Re-training a complete model whenever a new domain appears is usually impractical, while directly fine-tuning on the new domain may lead to catastrophic forgetting of previously learned domains. Domain-incremental learning addresses this problem by requiring a system to learn new domains sequentially while maintaining performance on earlier ones [1].

DCASE 2026 Task 7, Domain-Agnostic Incremental Learning for Audio Classification, formalizes this setting for sound event classification [2, 3]. In this task, acoustic domains are introduced in sequence, and each incremental stage must learn from the currently available domain without accessing the original training data of previous domains. During evaluation, the system must predict the sound event label without assuming that the domain identity is given. Therefore, domain inference becomes a key challenge: even a strong classifier may fail if a test sample is routed to an inappropriate domain branch.

Our submission focuses on reducing this domain-routing bottleneck while preserving the constraints of incremental learning. Since the task allows up to four system outputs per participant and subtask, we submit four complementary systems. The first system, DS-RanPAC, analytically adapts frozen CNN14 embeddings [4] through hyperspherical random projection and streaming ridge regression. The second system, ProtoRoute-DIL, combines CNN14-based incremental heads with log-mel statistical prototypes and Mahalanobis-based routing. The third system, MR-CLIAC, uses a CRNN classifier with domain-specific LoRA adapters and Mahalanobis routing for incremental D2/D3 classification. The fourth system is a prototype-routed triple-resolver subsystem built on the official CNN-BN baseline, combining residual MLP domain heads, embedding prototype fusion, soft D2/D3 fusion, and a fixed prediction-only resolver over three member outputs.

Although the four systems differ in model architecture and adaptation strategy, they share the same motivation: separating domain inference from class prediction as much as possible. Log-mel statistical prototypes, Mahalanobis-style distances, embedding prototypes, and soft fusion provide domain and class evidence beyond simple entropy-based branch selection. This is intended to reduce the "low-entropy but wrong" failure mode of classifier-confidence routing. All submitted systems are developed using only the task-provided data and do not use evaluation labels, evaluation-set class counts, or cross-sample decision statistics.

2. METHODS

2.1. System 1: DS-RanPAC

The Deep Streaming Random Projections and Artificial Covariance (DS-RanPAC) [5] subsystem aims to mitigate catastrophic forgetting in domain-agnostic incremental audio classification without relying on replay buffers. We adopt the challenge-provided D1 checkpoint as a completely frozen CNN14 feature extractor. Building upon single-pass embedding adaptation [6], we introduce a dual L_2

*The work of Xudong Zhao was supported by Engineering and Physical Sciences Research Council (EPSRC) under Grant EP/X032981/1.

†Corresponding author.

hyperspherical projection strategy to eliminate audio energy biases and loudness variations. Specifically, the raw 2048-dimensional features are first L_2-normalized, non-linearly mapped into a 16384-dimensional manifold via a frozen, orthogonally initialized random projection layer with GELU activation, and then L_2-normalized again to construct robust domain-invariant representations.

Instead of relying on iterative gradient updates, System 1 performs instant analytic adaptation via streaming ridge regression. The subsystem sequentially accumulates the feature autocorrelation and label cross-correlation matrices in the high-dimensional space. To dynamically balance the retention of old domain knowledge against the plasticity required for new domains, a memory decay coefficient α is continuously applied to these covariance matrices during the streaming updates. The optimal value of $\alpha = 0.97$ was empirically determined via grid search on the local D2/D3 development-test split to maximize the trade-off between stability and plasticity. At inference time, long audio files are processed using a 4-second sliding window with a 1-second hop size (shorter files are zero-padded). Segment-level representations are evaluated sequentially, and the final audio-level predictions are obtained by majority voting across all overlapping segments, with the trailing zero-padded segment discarded in the event of a tie. This closed-form solver provides a stable and robust analytic baseline among the submitted systems.

2.2. System 2: ProtoRoute-DIL

This subsystem tackles the domain-routing bottleneck in domain-agnostic incremental learning [1] by decoupling domain identification from the classifier output. Built on a D1-pretrained CNN14 backbone [4] with per-domain BatchNorm layers, it uses domain-specific classification heads—a single linear layer ($2048 \rightarrow 10$) for D1 inherited from the D1 checkpoint, and non-linear projection heads ($\mathrm{Lin}_{2048,512} \circ \mathrm{ReLU} \circ \mathrm{Drop}_{0.1} \circ \mathrm{Lin}_{512,10}$) for D2/D3. This design provides approximately 10^6 additional parameters per domain compared with a single linear layer, enabling richer domain-specific feature transformations.

The model is trained incrementally under strict incremental constraints. D2 training uses only D2 data with cross-entropy and entropy regularization under a linearly increasing weight $\lambda(t) = 0.2 \cdot t/T_{\max}$. D3 training employs Learning without Forgetting (LwF) [7]: the D2 model is frozen as a teacher, and the D3 loss combines cross-entropy, entropy regularization (half-weight), and KL-divergence between student and teacher D2 logits ($T = 3.0$, $\beta = 0.5$). The final convolutional block (conv_block6) is fine-tuned at a reduced learning rate (10^{-5}) during both D2 and D3 training to provide additional feature-space adaptation capacity. Crucially, BatchNorm layers belonging to previous domains are forced into evaluation mode during training of a new domain, preventing the running statistics from being contaminated by out-of-domain data. The D3 classification head uses a $20\times$ higher learning rate (2×10^{-4}) than D2 to accelerate convergence from random initialization on the more difficult domain.

The core innovation is a two-stage prototype-guided domain router that operates independently of model-internal classification signals. For D2/D3 discrimination, 128-dimensional log-mel statistical prototypes are constructed from each domain's training split: for each audio clip, the per-band temporal mean and standard deviation of the log-mel spectrogram form a feature vector $\mathbf{z} \in R^{128}$. Domain prototypes $\mathcal{P}_d = (\boldsymbol{\mu}_d, \boldsymbol{\Sigma}_d^{-1})$ are estimated via regularized covariance shrinkage ($\alpha = 0.15$, $\epsilon = 10^{-3}$). At inference, domain assignment follows minimum Mahalanobis distance: $d^* = \arg\min_{d \in \{D2,D3\}} (\mathbf{z} - \boldsymbol{\mu}_d)^\top \boldsymbol{\Sigma}_d^{-1} (\mathbf{z} - \boldsymbol{\mu}_d)$. For D1 detection, a Conv feature prototype is derived from the L2-normalized rows of the D1 linear classifier weights. D1 is selected when both (i) the D1 branch yields the lowest softmax entropy and (ii) the cosine distance to the D1 Conv prototype falls below 0.6. This routing mechanism relies solely on raw acoustic statistics for D2/D3 and on frozen D1 weights for D1 fallback, thereby eliminating the "low-entropy but wrong" failure mode of classifier-confidence-based routers. At inference, audio clips are preprocessed by truncation or zero-padding to a fixed 4 s duration, followed by a single model forward pass.

2.3. System 3: Mahalanobis-Routed CRNN-and-LoRA-based Incremental Audio Classifier(MR-CLIAC)

For the D1 baseline, we directly adopt the pre-trained CNN14 model provided by the challenge organizers without further training or modification. For domains D2 and D3, we employ a CRNN backbone consisting of four convolutional blocks (64→128→256→384 channels) with batch normalization, max-pooling, and dropout, followed by a 2-layer bidirectional GRU with 384 hidden units per direction. The GRU output is mean-pooled over time, projected to a 384-dimensional L2-normalized feature space, and classified by domain-specific linear heads. Low-Rank Adaptation (LoRA) [8] is applied to all convolutional layers and the feature projection layer, with two separate adapters using asymmetric ranks: $r = 8\,(\alpha = 16)$ for D2 and $r = 64$ ($\alpha = 128$) for D3, providing greater capacity for the more difficult domain.

The CRNN backbone is first trained on the provided D2 data for 1,200 epochs using AdamW (lr $= 1 \times 10^{-4}$) with cosine annealing warm restarts, Mixup [9] ($\alpha = 0.8$), SpecAugment [10], and label smoothing ($\epsilon = 0.1$). Incremental learning proceeds in two phases: the first LoRA adapter (task_0, $r = 8$) is trained on D2 for 600 epochs, then a second adapter (task_1, $r = 64$) is trained on D3 for 600 epochs with a $3\times$ higher learning rate. Only the current domain's LoRA parameters and classification head are updated; the base model, previous LoRA adapters, and batch normalization statistics remain frozen to mitigate catastrophic forgetting.

At inference time, domain identity is inferred via a two-stage routing mechanism: D1 samples are detected through cosine distance between CNN14 features and L2-normalized class prototypes, while D2 and D3 are discriminated using Mahalanobis distance on log-mel feature statistics (per-frame mean and standard deviation), with a regularized covariance matrix. Long audio files are segmented into 4-second windows (shorter segments zero-padded), each independently routed and classified. Final predictions are obtained by majority voting across segments, with the trailing zero-padded segment discarded in case of ties.

2.4. System 4: Prototype-Routed Triple-Resolver Subsystem

This subsystem is built on the official CNN-BN incremental baseline. Its main goal is to reduce domain-routing errors in domain-agnostic incremental audio classification. We keep the official CNN14-style backbone and train residual MLP heads for D2 and D3 using only the corresponding provided development data.

For domain classification, we use a prototype-based router instead of relying only on classifier entropy. For each domain, log-mel statistical prototypes are computed from the provided development data. Given a test sample, we extract its log-mel statistics and compare them with the domain prototypes using a hybrid Mahalanobis-entropy score. The Mahalanobis distance measures how close the

sample is to each domain distribution, while the entropy term reflects the confidence of the model prediction. The two cues are combined to estimate the most likely domain or to produce soft D2/D3 routing weights.

After domain routing, class-level embedding prototypes are used as auxiliary classification evidence and fused with the residual MLP logits. To avoid the risk of a single wrong hard domain decision, the system also uses soft D2/D3 fusion. We construct two complementary soft-fusion members. One member includes a small D3-specific class-prior correction selected on the development data, while the other removes this correction to avoid over-compensation. A third complementary member is a timemask-based hard-router model, which provides predictions with a different error pattern.

The final prediction is produced by a fixed 27-rule static triple resolver. For each test sample, the three members independently output one label, and the resolver maps this three-label tuple to the final class. Earlier routing, fusion, and member choices are screened on internal validation splits, while the final 27-rule resolver variant is selected using labels from the local D2/D3 development-test sanity split before evaluation. After this selection, the resolver is frozen. During evaluation, the resolver uses only the current sample's predictions and does not use evaluation labels, evaluation-set class counts, or cross-sample statistics. This subsystem achieves 70.208831 official-style macro accuracy on the local D2/D3 development-test sanity split.

3. EXPERIMENTS

3.1. Dataset

We conduct experiments on the task-provided development data of DCASE 2026 Task 7. The task follows a domain-incremental learning protocol in which acoustic domains are introduced sequentially. In each incremental stage, the system is allowed to learn only from the currently available domain data, while the original training data from previously learned domains are not revisited. The evaluation setting is domain-agnostic: the domain identity of a test sample is not assumed to be known during inference.

In our local validation protocol, the development data are used to construct incremental training subsets and a D2/D3 development-test sanity split. The evaluation set is used only to generate final submission predictions. We do not use evaluation labels, evaluation-set class counts, or cross-sample decision statistics for model selection or prediction-time decision making.

3.2. Experimental Setup

We compare the official CNN-BN baseline [1] with our four submitted systems. The baseline is the organizer-provided CNN14-style incremental classifier with domain-specific BatchNorm branches. The four submitted systems are designed as complementary and progressively enhanced solutions to the same domain-routing and forgetting problem. System 1, DS-RanPAC, uses the provided D1 baseline as a frozen CNN14 feature extractor and performs analytic adaptation. System 2, ProtoRoute-DIL, modifies the domain-agnostic routing mechanism of the official CNN-BN baseline using log-mel statistical prototypes and Mahalanobis-based routing. System 3, MR-CLIAC, introduces LoRA-adapted CRNN classifiers for D2 and D3. System 4, the Prototype-Routed Triple-Resolver subsystem, combines residual MLP domain heads, embedding prototype fusion, soft D2/D3 fusion, and a fixed prediction-only resolver.

Table 1: Macro accuracy comparison of the official baseline and the four submitted systems. D2/D2 is the D2 macro accuracy after learning D2, D2/D3 is the retained D2 macro accuracy after learning D3, and D3/D3 is the D3 macro accuracy after learning D3. Overall is the final average of D2/D3 and D3/D3. All values are percentages.

System	D2/D2	D2/D3	D3/D3	Overall
Official CNN-BN baseline	58.6	59.0	46.1	52.5
System 1: DS-RanPAC	69.4	67.5	57.2	62.3
System 2: ProtoRoute-DIL	75.9	70.1	59.4	64.8
System 3: MR-CLIAC	**82.5**	73.2	**65.9**	69.6
System 4: Triple-Resolver	74.5	**75.3**	65.1	**70.2**

All systems are trained or adapted using only the task-provided data. During testing, each audio sample is processed independently, and the final prediction does not depend on the predicted labels or class distribution of other test samples.

3.3. Evaluation Metric

We report domain-wise macro accuracy under the incremental evaluation setting. The notation D2/D2 denotes the macro accuracy on D2 after learning D2. The notation D2/D3 denotes the retained macro accuracy on D2 after subsequently learning D3, and D3/D3 denotes the macro accuracy on D3 after learning D3. For the local D2/D3 development-test sanity split, the overall score is computed after the final D3 stage as the average of D2/D3 and D3/D3. The D2/D2 entry is reported as an intermediate diagnostic result after learning D2 and is not included in the final D2/D3 overall score.

This evaluation reflects both plasticity and stability. A system with high D3/D3 but low D2/D3 may adapt well to the newest domain but suffer from forgetting of the previous domain. Conversely, a system with balanced D2/D3 and D3/D3 indicates a better trade-off between learning the new domain and retaining previously acquired knowledge.

3.4. Results

Table 1 reports the local D2/D3 development-test sanity results of the official CNN-BN baseline and our four submitted systems. All values are reported as percentages and rounded to one decimal place. The D2/D2 column measures the performance on D2 immediately after learning D2. The D2/D3 and D3/D3 columns measure the retained D2 performance and the newly learned D3 performance after the final D3 stage, respectively. The overall score is computed after learning D3 as the average of D2/D3 and D3/D3. Since values are rounded independently, the displayed overall score may not exactly match the average computed from the rounded entries.

Compared with the official CNN-BN baseline, all four submitted systems improve the final overall score. DS-RanPAC improves the baseline from 52.5% to 62.3%, showing that analytic adaptation of frozen CNN14 features provides a stronger incremental-learning anchor than the organizer baseline. ProtoRoute-DIL further improves the final score to 64.8%, indicating that prototype-based domain routing is effective for reducing errors caused by entropy-only branch selection. MR-CLIAC obtains the highest D2/D2 accuracy and the highest D3/D3 accuracy, suggesting that the LoRA-adapted

CRNN classifiers provide strong domain-specific adaptation capacity.

The best final overall score is achieved by System 4, the Prototype-Routed Triple-Resolver subsystem. Although System 3 obtains a slightly higher D3/D3 accuracy, System 4 achieves the best retained D2/D3 accuracy after learning D3. This leads to the highest local D2/D3 sanity overall accuracy of 70.2%, which is 17.7 percentage points higher than the official CNN-BN baseline. These results support the motivation of combining prototype-based routing, residual MLP domain heads, embedding prototype fusion, soft D2/D3 fusion, and per-sample prediction-only resolving to improve the stability-plasticity trade-off in domain-agnostic incremental audio classification.

4. REFERENCES

[1] M. Mulimani and A. Mesaros, "Domain-Incremental Learning for Audio Classification," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2025, pp. 1–5.

[2] DCASE Community, "DCASE 2026 Challenge Task 7: Domain-Agnostic Incremental Learning for Audio Classification," 2026. [Online]. Available: DCASE 2026 Challenge Task 7 webpage.

[3] R. Casciotti, M. Mulimani, M. Harju, J. R. Jensen, and A. Mesaros, "Domain-Agnostic Incremental Learning for Sound Classification. A DCASE 2026 Challenge task," *arXiv preprint arXiv:2606.02173*, 2026.

[4] Q. Kong, Y. Cao, T. Iqbal, Y. Wang, W. Wang, and M. D. Plumbley, "PANNs: Large-scale pretrained audio neural networks for audio pattern recognition," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 28, pp. 2880–2894, 2020.

[5] M. D. McDonnell, D. Gong, A. Parvaneh, E. Abbasnejad, and A. van den Hengel, "RanPAC: Random Projections and Pretrained Models for Continual Learning," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

[6] M. Mulimani and A. Mesaros, "Online incremental learning for audio classification using a pretrained audio model," in *Proc. IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, 2025, pp. 1–5.

[7] Z. Li and D. Hoiem, "Learning without forgetting," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 12, pp. 2935–2947, 2018.

[8] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, and W. Chen, "LoRA: Low-rank adaptation of large language models," in *Proc. International Conference on Learning Representations (ICLR)*, 2022.

[9] H. Zhang, M. Cissé, Y. N. Dauphin, and D. Lopez-Paz, "mixup: Beyond empirical risk minimization," in *Proc. International Conference on Learning Representations (ICLR)*, 2018.

[10] D. S. Park, W. Chan, Y. Zhang, C.-C. Chiu, B. Zoph, E. D. Cubuk, and Q. V. Le, "SpecAugment: A simple data augmentation method for automatic speech recognition," in *Proc. Interspeech*, 2019, pp. 2613–2617.