FROM GENERAL EXPERTS TO CHANNEL SPECIALISTS: RECASTING MIXTURE-OF-EXPERTS FOR MULTIVARIATE TIME SERIES FORECASTING

Anonymous authors
Paper under double-blind review

ABSTRACT

Channels in a multivariate time series can favor different parameter updates. An expert's training channels therefore shape the competence available for forecast combination. We introduce cMoE, a plug-in mixture-of-experts mechanism that uses historical channel–expert forecasting errors to guide both training and prediction. Every expert evaluates every channel; an exponential-moving-average Scoreboard tracks channel–expert losses and determines the responsibility matrix. Unlike prediction-only weighting, this matrix both combines forecasts and defines each expert's channel-weighted training objective. Our analysis characterizes how responsibility changes expert updates and bounds future weighted expert-loss advantage over equal-expert assignment under exchangeability. Paired interventions on linear, frequency, and attention backbones show that responsibility-weighted training develops complementary experts and increases the average forecasting benefit of the same historical channel weights. Their value thus depends on the competence developed through training, not only on their use at prediction time. Across 180 settings with five backbones and nine multivariate datasets, cMoE improves every backbone and dataset on average, reducing MSE by 4.92% and MAE by 3.20%. Together, these findings support using outcome history to develop channel specialists as well as combine their forecasts. Our code is available at `https://anonymous.4open.science/r/cMoE`.

1 INTRODUCTION

Multivariate time series contain channels with different temporal regularities, noise levels, physical meanings, and predictive dependencies (Liu et al., 2024; Chen et al., 2024a). Therefore, recent forecasting methods have explored channel-separable processing, variate-centric representations, and selective or grouped cross-channel modeling to better accommodate this heterogeneity (Wu et al., 2023; Zeng et al., 2023; Nie et al., 2023; Liu et al., 2024; Wang et al., 2024a; Han et al., 2024; Wang et al., 2024b; Qiu et al., 2025). These design choices primarily determine how channel information is represented, exchanged, or grouped. Yet this heterogeneity also raises an optimization question: *when multiple channels share an expert, which channels should shape its update, and by how much?* We refer to this allocation as *channel-specific responsibility*. Parameter sharing helps learn common temporal structure, but averaging channel-level losses can compromise the updates preferred by individual channels (Sener & Koltun, 2018; Yu et al., 2020). Adaptive responsibility offers a way to retain shared experts while changing which channels shape each expert's learning.

Mixture-of-experts models provide adaptive responsibility through local experts and their hierarchical extensions (Jacobs et al., 1991; Jordan & Jacobs, 1994). Input- or hidden-state-dependent routers distribute examples, patches, or paths across experts, shaping both forecasts and the updates each expert receives (Shazeer et al., 2017; Lepikhin et al., 2021; Fedus et al., 2022; Chen et al., 2024b; Ni et al., 2024; Shi et al., 2025; Liu et al., 2025a; Zhang et al., 2025; Liu et al., 2026). These routers infer responsibility from input or model representations. Observed forecasting errors provide a complementary signal. Prediction errors have long informed model-level weighting (Littlestone & Warmuth, 1994; Raftery et al., 2010; Montero-Manso et al., 2020), and recent methods derive model-selection targets from validation performance or add expert-specific loss terms to training

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

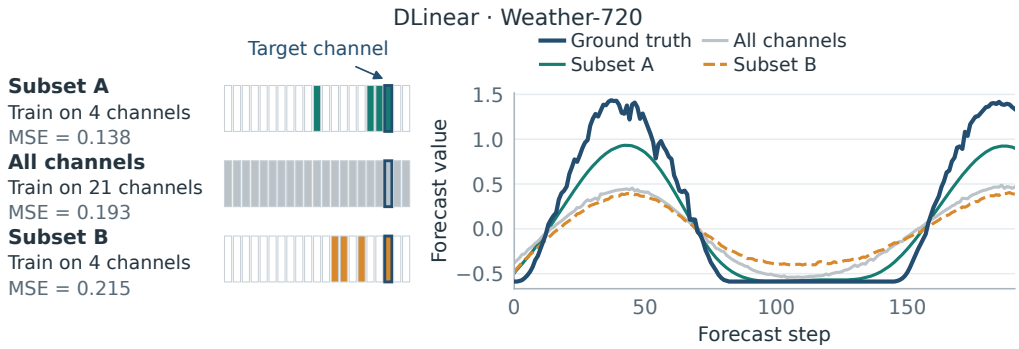


Figure 1: Training composition shapes channel-specific competence on DLinear/Weather-720. Left panel: training channels and target-channel validation MSE. Right panel: the first 192 steps of a 720-step forecast from each trained model.

(Li et al., 2026b; El Mahtout & Ziel, 2026). Our focus is to reuse historical channel–expert outcomes as a common responsibility signal for forecast combination and channel-weighted expert training.

Figure 1 gives the starting observation: changing the channels that train an expert changes its error on the same target channel. This motivates our central hypothesis: historical channel–expert outcomes should determine not only how experts are combined at prediction time, but also which channels train each expert, because the value of channel-specific forecast weights depends on the channel-specific competence developed during training. We therefore propose cMoE to connect outcome-based forecast combination with expert training through a channel–expert Scoreboard. The Scoreboard determines how much each expert contributes to a channel's forecast and how much each channel contributes to an expert's training. We first test whether responsibility-weighted training itself develops channel-relative differences in expert forecasting competence. Paired interventions retain the same historical forecast weights while changing training composition, testing whether training increases their forecasting benefit. A complementary control tests whether fitting combination weights after equal-channel expert training can match cMoE. Figures 3–5 examine the resulting expert competence, learned identities, and channel structure. Our contributions are:

- **Outcome-Written channel responsibility.** We introduce a plug-in mechanism that maintains a channel–expert loss history and directly converts it into responsibility for forecast aggregation and channel-weighted expert training. Experts share the same backbone architecture, but learn their channel roles without predefined groups or a separately trained router.
- **Analysis of responsibility-weighted learning.** We characterize how channel-weighted gradients separate or contract expert parameters through an exact, measurable update identity. A calibrated certificate bounds future expert-loss advantage over equal-expert assignment.
- **Training-dependent value of forecast weights.** In linear, frequency, and attention settings, paired interventions show that responsibility-weighted training increases the mean benefit of the same historical forecast weights. A 180-setting evaluation, ablations, and routing controls examine forecasting performance, outcome memory, and training allocation.

2 Related Work

Modeling channel heterogeneity. Existing approaches to channel heterogeneity are largely concerned with how information and parameters are shared across channels (Qiu et al., 2026). Channel-independent methods share temporal parameters while processing channels separately (Zeng et al., 2023; Nie et al., 2023), whereas channel-dependent architectures explicitly model cross-channel interactions (Liu et al., 2024; Han et al., 2024; Wang et al., 2024b). Adaptive approaches further condition representations or sharing structures on channel characteristics (Liu et al., 2023a; Dong et al., 2024; Jiang et al., 2023; Chen et al., 2024a; Qiu et al., 2025; Li et al., 2026a). Temporal modeling addresses a complementary source of variation within each series. Frequency-domain and

multi-scale representations capture patterns at different frequencies and resolutions (Wu et al., 2023; Wang et al., 2024a; Yi et al., 2023; Wang et al., 2025), including coarse-to-fine trend reconstruction in diffusion forecasting (Shen et al., 2024). Normalization instead addresses changing data distributions (Deng et al., 2021; Ye et al., 2024): slice-level adaptive normalization predicts local statistics (Liu et al., 2023b), while IN-Flow learns invertible transformations (Fan et al., 2025). These architectural choices determine how information is represented and shared; an expert pool additionally requires a rule for allocating predictive responsibility.

Expert responsibility in time series forecasting. Forecast responsibility is commonly inferred from input features or learned representations, as in input-conditioned mixtures (Chen et al., 2024b; Ni et al., 2024; Shi et al., 2025; Liu et al., 2025a; Zhang et al., 2025; Ma et al., 2025) and feature-based model combination such as FFORMA and TimeFuse (Montero-Manso et al., 2020; Liu et al., 2025b). DisenTS further conditions channel-wise routing on input features and moving-average forecaster representations, with a similarity constraint promoting distinct patterns (Liu et al., 2026). LeMoLE combines linear experts at different lookback lengths through text-conditioned feature modulation and learned fusion (Zhang et al., 2024). Recent work structures expert roles around temporal regularities. Frequency information guides spectral-band weighting in FreqMoE and routing over frozen frequency experts and trainable complementary experts in Super-Linear (Liu, 2025; Nochumsohn et al., 2026). Locally, TFPS and Seg-MoE route patches and contiguous segments, respectively (Sun et al., 2025; Ortigossa & Segal, 2026). Structural constraints guide specialization: AME-TS regularizes routing toward temporal-descriptor priors, while FM-LLM separates shared periodic modeling from routed residual decoding (Wang et al., 2026; Gu et al., 2026).

Predictive performance provides another source of allocation evidence. Adaptive combination updates model-level weights from observed performance (Littlestone & Warmuth, 1994; Raftery et al., 2010), while FAME uses validation-derived expert-suitability targets to train a forecastability-based router over frozen heterogeneous experts (Li et al., 2026b). Thus, outcomes may update weights directly or supervise a learned weighting function. For trainable experts, the consequences of this allocation depend on how prediction weights enter the objective.

Loss-based training allocation. Loss-based allocation regulates which observations update each model and how their losses contribute to the objective. Boosting upweights misclassified examples (Freund & Schapire, 1997), whereas self-paced learning progresses from low-loss to higher-loss examples (Kumar et al., 2010). To allocate examples among ensemble members, Multiple Choice Learning minimizes the loss of the best prediction for each example (Guzman-Rivera et al., 2012); its stochastic form assigns each example's gradient to the current lowest-loss model (Lee et al., 2016). At the objective level, Expert Loss Integration augments the global MoE objective with expert-specific losses (El Mahtout & Ziel, 2026). cMoE jointly specifies forecast weighting and soft training allocation at channel–expert resolution. One responsibility matrix, obtained directly from historical relative losses, governs forecast aggregation and channel-weighted expert objectives within the same training step. Our same-backbone controls compare this coupling with a learned static gate, outcome-supervised gating, and expert-loss augmentation.

3 METHODOLOGY

Problem setup. Given a C-channel lookback window X_t, K separately parameterized experts with the same backbone predict future channel values. Initially, they are *general experts*, with no assigned channel roles. Indices c, k, t denote channel, expert, and training step. Let $\hat{y}^k_{t,c}$ and $y_{t,c}$ be forecast and target vectors; the matrix Y_t collects all channel targets. Scoreboard and training losses are $\ell^k_{t,c} = \mathcal{L}(\hat{y}^k_{t,c}, y_{t,c})$ and $\ell^{\text{train},k}_{t,c} = \mathcal{L}_{\text{train}}(\hat{y}^k_{t,c}, y_{t,c})$, respectively. The training criterion incorporates any configured horizon weights and may differ from $\mathcal{L}$.

Design principle. Our design principle is to use historical channel–expert outcomes to jointly determine how experts are combined and which channels shape each expert's subsequent learning. Figure 2 summarizes the procedure.

Accordingly, cMoE first converts historical outcomes into responsibility, then uses the same responsibility for forecasting and expert learning, before writing new outcomes back to the Scoreboard.

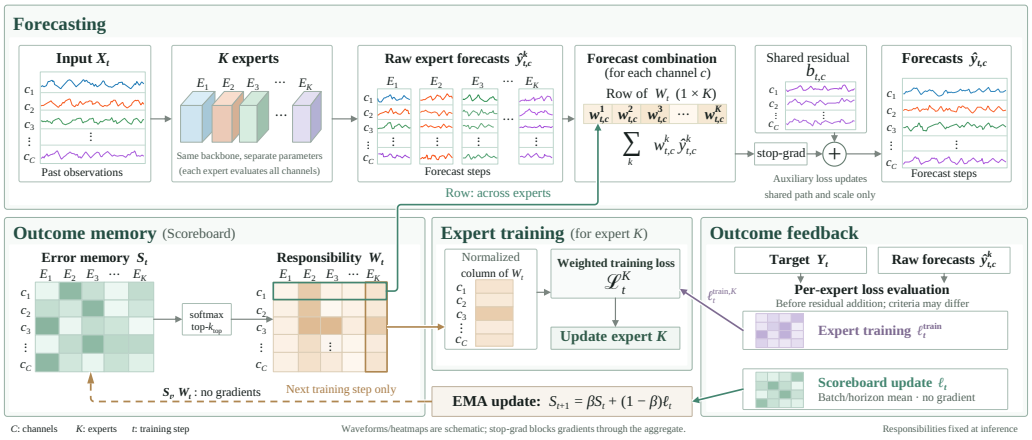


Figure 2: The cMoE responsibility loop. Every expert evaluates every channel. Historical channel–expert losses determine W_t: rows combine forecasts, while normalized columns weight expert training. Superscript train marks the configured training criterion, which may differ from the Scoreboard criterion. The training panel assumes positive total responsibility; the supplement specifies the zero-responsibility case. EMA updates apply at the next training step, and responsibility is fixed at inference. Heatmaps and waveforms are schematic.

3.1 Outcome-Written Responsibility and Forecasting

The Scoreboard $S_t \in \mathbb{R}^{C\times K}$ has entries $s^k_{t,c}$. Channel-wise normalization and softmax at temperature $\tau > 0$ favor lower losses. Soft balancing redistributes total responsibility across experts; top-k_{top} retains each channel's largest weights, kept soft and renormalized to sum to one. $W_t \in \mathbb{R}^{C\times K}$ has entries $w^k_{t,c} \geq 0$. The supplement gives transformation details and schedules. For a fixed channel, a row of W_t distributes responsibility across experts; for a fixed expert, a column distributes responsibility across channels. We first use the row-wise responsibility to combine expert forecasts:

$$\hat{y}_{t,c} = \sum_{k=1}^{K} w^k_{t,c}\hat{y}^k_{t,c} + b_{t,c}, \tag{1}$$

where $b_{t,c}$ is the shared residual prediction, including its learned scale. The shared path learns the residual of the combined forecast and does not change the routed expert objective. The same responsibility matrix is next used to determine which channels shape each expert's learning.

3.2 Responsibility-Weighted Expert Learning

Viewing W_t column-wise, cMoE defines the channel-weighted objective for expert k as

$$\mathcal{L}^k_t = \sum_{c=1}^{C} \frac{w^k_{t,c}}{m^k_t}\ell^{\text{train},k}_{t,c}, \qquad m^k_t = \sum_c w^k_{t,c}. \tag{2}$$

For $m^k_t > 0$, the normalized weights sum to one; the supplement specifies the zero-responsibility case. Thus, Eq. 2 changes the channel composition of an expert's training without changing the total weight of its objective. We use *specialization* to denote channel-relative forecasting competence; parameter dispersion below is only an update-level diagnostic.

Update-level effect. We compare responsibility-weighted and *equal-channel training* (weight $1/C$ per channel) through virtual gradient steps from the same parameter state, assuming $m^k_t > 0$. For each expert, both objectives have unit total weight, isolating channel composition. All quantities in the update test are observable after the current target arrives and before either virtual step.

Let θ_t^k be expert k's parameters, $g_{t,c}^k = \nabla_{\theta_t^k} \ell_{t,c}^{\text{train},k}$ its channel gradient, and $\bar{\theta}_t = K^{-1} \sum_k \theta_t^k$ the expert mean. Define the expert-separation pressure as

$$\mathcal{P}_t = -\sum_{c,k} \left(\frac{w_{t,c}^k}{m_t^k} - \frac{1}{C} \right) (\theta_t^k - \bar{\theta}_t)^\top g_{t,c}^k. \tag{3}$$

The weight difference measures the change in channel composition; its interaction with each channel gradient gives the additional separation pressure relative to equal-channel training.

Expert dispersion is the mean squared parameter distance from the expert mean. Let $\eta > 0$ be their common virtual step size and L_t the dispersion difference between responsibility-weighted and equal-channel updates from the same state. Direct expansion yields

$$L_t = \frac{2\eta}{K} \mathcal{P}_t + R_t, \tag{4}$$

where R_t is the exact difference between the quadratic update components. Thus $L_t > 0$ means greater dispersion after the responsibility-weighted step than after the equal-channel step. If $|2\eta \mathcal{P}_t / K| > |R_t|$, the sign of $\mathcal{P}_t$ determines the sign of L_t. Current parameters, channel gradients, and weights determine $\mathcal{P}_t$ and R_t before either virtual step. The supplement gives the derivation and finite-window decomposition. This update analysis characterizes how responsibility changes expert learning, but parameter separation alone does not imply better forecasting. We therefore connect this training effect to forecast combination under a scoped fixed-feature setting.

For linear heads with shared feature second moments and a frozen responsibility matrix, population-optimal responsibility-weighted training has no larger mixture MSE than equal-channel training. With positive-definite feature second moments, the reduction is strict when the resulting mixture preserves channel-level coefficient differences. Supplementary Section A.3 gives the coefficient formula and proof. This result provides a scoped analytical connection between responsibility-weighted training and the channel-specific forecast combination in Eq. 1; the general training-dependent effect is evaluated in Section 4.

3.3 Full-Information Outcome Feedback

Every expert evaluates every channel, including pairs with small prediction weights. This full-information audit keeps currently low-responsibility channel–expert pairs observable for future allocation. After observing Y_t, cMoE collects their losses in the matrix ℓ_t and updates the Scoreboard as

$$s_{t+1,c}^k = \beta s_{t,c}^k + (1 - \beta) \ell_{t,c}^k, \tag{5}$$

where $\beta \in [0, 1)$ is the Scoreboard momentum. The current training losses update expert parameters, while the Scoreboard losses enter routing at the next step through S_{t+1}. Thus, the current target does not affect the responsibility W_t used for the current prediction.

At inference, expert forecasts change with the input while the learned responsibility remains fixed. Because the Scoreboard carries channel–expert outcomes across steps, we finally examine whether this history remains informative for future allocation.

Supporting result: future value of outcome memory. We compare responsibility-weighted expert loss over the next H batches with *equal-expert assignment*, which gives each expert weight $1/K$ per channel. The Scoreboard estimates this loss advantage. Subtracting a calibration quantile q_t gives

$$\mathcal{B}_t = \frac{1}{C} \sum_c \left(\frac{1}{K} \sum_k s_{t,c}^k - \sum_k w_{t,c}^k s_{t,c}^k \right) - q_t. \tag{6}$$

For error probability $\alpha \in (0, 1)$, q_t is the finite-sample-corrected upper quantile of past estimation residuals: estimated minus observed H-batch advantage. Calibration uses only completed, non-overlapping windows within a fixed routing phase, Scoreboard rule, expert pool, and responsibility transformation. Assume that these residuals and the next residual are exchangeable, which means their joint distribution is unchanged by permutation. Then $\mathcal{B}_t$ is a marginal $1 - \alpha$ lower bound on the weighted expert-loss advantage of the next window. A positive bound supports the continued value of the historical allocation. Supplementary Section A.5 gives the full probability statement, quantile construction, and proof; Section 4.5 reports empirical coverage on training trajectories.

Table 1: Forecasting results with and without cMoE, averaged over three runs.

Dataset	H	DLinear				FITS				OLinear				iTransformer				MixLinear			
		Base		+cMoE		Base		+cMoE		Base		+cMoE		Base		+cMoE		Base		+cMoE	
		MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
ETTh1	96	0.385	0.398	**0.381**	**0.393**	0.392	0.400	**0.385**	**0.393**	0.361	0.382	**0.361**	**0.382**	0.395	0.410	**0.389**	**0.405**	0.386	0.391	**0.385**	**0.391**
	192	0.442	0.437	**0.437**	**0.433**	0.440	0.426	**0.433**	**0.420**	0.416	0.414	**0.415**	**0.413**	0.449	0.441	**0.442**	**0.437**	0.454	0.438	**0.440**	**0.422**
	336	0.480	0.457	**0.478**	**0.454**	0.479	0.448	**0.475**	**0.442**	0.458	0.438	**0.454**	**0.432**	0.492	0.465	**0.485**	**0.462**	0.481	0.446	**0.476**	**0.439**
	720	0.523	0.517	**0.508**	**0.504**	0.476	0.475	**0.468**	**0.462**	0.463	0.462	0.477	**0.458**	0.522	0.504	**0.502**	**0.490**	0.465	0.459	**0.462**	**0.454**
ETTh2	96	0.329	0.386	**0.300**	**0.349**	0.295	0.343	**0.289**	**0.337**	0.284	0.329	**0.283**	**0.327**	0.300	0.350	**0.295**	**0.346**	0.309	0.348	**0.301**	**0.344**
	192	0.482	0.479	**0.447**	**0.445**	0.380	0.393	**0.377**	**0.390**	0.360	0.379	**0.356**	**0.377**	0.382	0.400	**0.375**	**0.394**	0.388	0.396	**0.385**	**0.394**
	336	0.593	0.542	**0.481**	**0.481**	0.418	0.427	**0.414**	**0.424**	0.409	0.416	**0.405**	**0.414**	0.424	0.432	**0.419**	**0.429**	0.427	0.429	**0.421**	**0.427**
	720	0.834	0.658	**0.649**	**0.574**	0.421	0.439	**0.416**	**0.436**	0.421	0.437	**0.413**	**0.432**	0.426	0.445	**0.423**	**0.443**	0.423	0.438	**0.420**	**0.437**
ETTm1	96	0.346	0.374	**0.331**	**0.354**	0.357	0.377	**0.351**	**0.373**	0.302	0.334	**0.302**	**0.332**	0.341	0.377	**0.334**	**0.372**	0.350	0.370	**0.341**	**0.369**
	192	0.383	0.392	**0.375**	**0.386**	0.394	0.395	**0.393**	**0.394**	0.357	0.363	**0.356**	**0.361**	0.380	0.395	**0.375**	**0.390**	0.399	0.398	**0.387**	**0.392**
	336	0.414	0.415	**0.406**	**0.413**	0.426	0.415	**0.416**	**0.411**	0.387	0.385	**0.385**	**0.383**	0.420	0.419	**0.410**	**0.412**	0.434	0.419	**0.421**	**0.415**
	720	0.473	0.451	**0.467**	**0.449**	0.487	0.449	**0.480**	**0.444**	0.452	0.426	**0.452**	**0.423**	0.486	0.456	**0.479**	**0.452**	0.495	0.455	**0.487**	**0.450**
ETTm2	96	0.193	0.293	**0.183**	**0.274**	0.184	0.268	**0.177**	**0.258**	0.169	0.249	**0.168**	**0.246**	0.183	0.267	**0.179**	**0.261**	0.190	0.272	**0.184**	**0.263**
	192	0.285	0.361	**0.270**	**0.348**	0.248	0.306	**0.243**	**0.300**	0.232	0.290	**0.231**	**0.288**	0.254	0.313	**0.251**	**0.310**	0.253	0.310	**0.249**	**0.305**
	336	0.371	0.421	**0.314**	**0.362**	0.308	0.343	**0.303**	**0.338**	0.291	0.328	**0.291**	**0.327**	0.312	0.350	**0.310**	**0.349**	0.316	0.351	**0.309**	**0.343**
	720	0.556	0.523	**0.433**	**0.438**	0.408	0.398	**0.404**	**0.394**	0.389	0.387	**0.388**	**0.386**	0.412	0.407	**0.410**	**0.404**	0.416	0.406	**0.410**	**0.399**
Electricity	96	0.196	0.280	**0.189**	**0.273**	0.213	0.298	**0.192**	**0.270**	0.131	0.221	**0.128**	**0.217**	0.149	0.240	**0.142**	**0.235**	0.210	0.280	**0.202**	**0.276**
	192	0.194	0.282	**0.188**	**0.275**	0.213	0.301	**0.206**	**0.295**	0.152	0.240	**0.149**	**0.236**	0.164	0.256	**0.163**	**0.254**	0.205	0.281	**0.199**	**0.276**
	336	0.207	0.296	**0.202**	**0.292**	0.227	0.314	**0.219**	**0.308**	0.164	0.253	**0.162**	**0.250**	0.179	0.272	**0.170**	**0.266**	0.219	0.297	**0.213**	**0.291**
	720	0.242	0.329	**0.241**	**0.328**	0.267	0.344	**0.261**	**0.339**	0.191	0.279	**0.184**	**0.272**	0.228	0.312	**0.198**	**0.293**	0.260	0.330	**0.255**	**0.324**
Exchange	96	0.113	0.246	**0.087**	**0.206**	0.087	0.209	**0.086**	**0.207**	0.088	0.207	**0.086**	**0.206**	0.100	0.222	**0.095**	**0.218**	0.094	0.217	**0.094**	**0.217**
	192	0.163	0.296	**0.160**	**0.292**	0.180	0.303	**0.180**	**0.303**	0.183	0.303	**0.182**	**0.302**	0.204	0.324	**0.182**	**0.306**	0.183	0.306	**0.177**	**0.300**
	336	0.265	0.384	**0.265**	**0.384**	0.326	0.414	**0.323**	**0.412**	0.354	0.430	**0.328**	**0.413**	0.343	0.428	**0.337**	**0.420**	0.320	0.411	**0.317**	**0.408**
	720	0.635	0.624	**0.601**	**0.598**	0.879	0.713	**0.865**	**0.705**	0.849	0.692	**0.841**	**0.691**	0.842	0.697	**0.832**	**0.689**	0.848	0.696	**0.842**	**0.690**
Solar	96	0.283	0.372	0.287	0.377	0.330	0.364	**0.327**	**0.361**	0.207	0.208	**0.178**	**0.205**	0.222	0.249	**0.203**	**0.243**	0.511	0.447	**0.414**	**0.398**
	192	0.320	0.393	**0.317**	0.395	0.374	0.394	**0.373**	**0.394**	0.244	0.230	**0.208**	**0.226**	0.247	0.272	**0.239**	**0.270**	0.543	0.458	**0.450**	**0.417**
	336	0.347	0.409	0.349	0.412	0.416	0.409	**0.410**	**0.405**	0.260	0.250	**0.236**	**0.240**	0.266	0.281	**0.251**	**0.272**	0.555	0.456	**0.475**	**0.419**
	720	0.356	0.413	**0.353**	**0.413**	0.416	0.399	**0.413**	**0.397**	0.252	0.254	**0.248**	**0.244**	0.262	0.281	**0.251**	**0.274**	0.522	0.432	**0.461**	**0.404**
Weather	96	0.196	0.256	**0.161**	**0.227**	0.196	0.236	**0.181**	**0.229**	0.152	0.190	**0.147**	**0.184**	0.175	0.216	**0.166**	**0.207**	0.197	0.240	**0.183**	**0.231**
	192	0.238	0.295	**0.210**	**0.279**	0.241	0.272	**0.230**	**0.267**	0.202	0.238	**0.200**	**0.235**	0.225	0.258	**0.215**	**0.253**	0.246	0.277	**0.231**	**0.270**
	336	0.283	0.335	**0.257**	**0.320**	0.293	0.308	**0.284**	**0.305**	0.259	0.281	**0.254**	**0.276**	0.280	0.298	**0.273**	**0.295**	0.296	0.314	**0.287**	**0.309**
	720	0.348	0.384	**0.339**	**0.380**	0.366	0.355	**0.357**	**0.351**	0.340	0.334	**0.333**	**0.328**	0.361	0.351	**0.355**	**0.348**	0.371	0.361	**0.363**	**0.358**
ILI	24	4.847	1.672	**2.729**	**1.108**	7.029	1.986	**4.849**	**1.611**	2.305	0.945	**2.021**	**0.849**	2.437	0.971	**2.330**	**0.918**	5.518	1.787	**5.064**	**1.703**
	36	4.393	1.551	**2.721**	**1.101**	5.955	1.771	**4.864**	**1.605**	2.011	0.912	**1.943**	**0.891**	2.202	0.958	**2.152**	**0.946**	5.013	1.661	**4.509**	**1.564**
	48	4.069	1.469	**2.480**	**1.067**	5.095	1.637	**3.915**	**1.426**	2.190	0.936	**2.144**	**0.918**	2.403	1.000	**2.166**	**0.942**	4.773	1.598	**4.492**	**1.552**
	60	4.178	1.475	**3.002**	**1.174**	5.082	1.632	**4.254**	**1.488**	2.114	0.938	**2.044**	**0.921**	2.341	1.005	**2.136**	**0.951**	4.721	1.584	**4.406**	**1.529**

4 EXPERIMENTS

We assess cMoE's forecasting gains and test whether responsibility-weighted training increases the value of the same channel-specific forecast weights. Analyses of expert competence and learned assignments characterize the resulting channel roles.

4.1 EXPERIMENTAL SETUP

Datasets. We evaluate on nine multivariate benchmarks: ETTh1, ETTh2, ETTm1, and ETTm2 (Zhou et al., 2021); Electricity, Exchange, and Solar (Lai et al., 2018); and Weather and ILI (Wu et al., 2021). Input length is 96; prediction horizons are $\{96, 192, 336, 720\}$ except for ILI ($\{24, 36, 48, 60\}$). We report mean squared error (MSE) and mean absolute error (MAE).

Baselines. We evaluate cMoE with five forecasting backbones: DLinear (Zeng et al., 2023), FITS (Xu et al., 2024), OLinear (Yue et al., 2025), iTransformer (Liu et al., 2024), and MixLinear (Ma et al., 2026). For each backbone, we compare cMoE with its original model under matched settings.

Implementations. Experiments use PyTorch 2.4.1 (Paszke et al., 2019) with CUDA 12.1 on four NVIDIA RTX A6000 GPUs using Adam optimization (Kingma & Ba, 2015). Supplementary Section B provides preprocessing details, configuration profiles, training settings, and checkpoint selection. Code is available through the anonymous repository linked in the abstract.

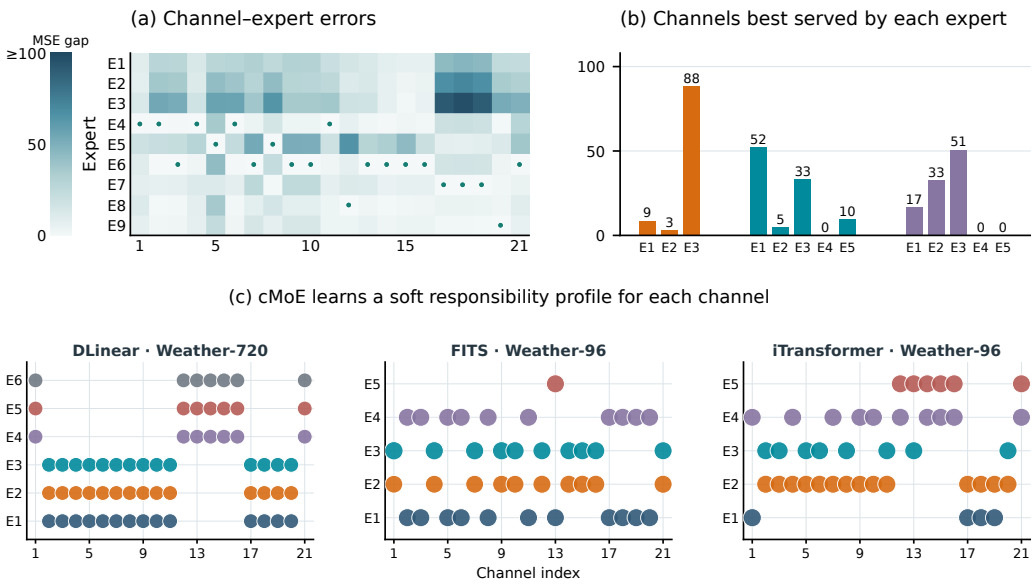


Figure 3: Channel-wise competence and responsibility. (a) Validation MSE excess (%) over each channel's best expert (dots), DLinear/Weather-720. (b) Percentage of channels best predicted by each expert (validation MSE): MixLinear/Solar-96 (orange), FITS/Weather-96 (teal), and iTransformer/Electricity-96 (purple). (c) Separate Weather snapshots; larger circles indicate larger soft weights, summing to one per channel. Protocols: Supplementary Section F.

4.2 Overall Forecasting Performance

Table 1 reports three-run means for all 180 settings. cMoE matches or improves the corresponding baseline in 177 MSE cells, 177 MAE cells, and 176 cells on both metrics. The average relative improvements are 4.92% in MSE and 3.20% in MAE.

The average MSE gains range from 2.60% for OLinear to 9.70% for DLinear. Dataset-level gains range from 1.11% on ETTh1 to 15.75% on ILI. FITS, MixLinear, and iTransformer show the most consistent results across their 36 settings. Solar improves by 6.32% on average.

Direct routing controls test the proposed use of outcome history against alternative responsibility mechanisms (Supplementary Table S7). On nine 96-step settings, cMoE has lower reported mean MSE than static and outcome-supervised gates in all nine, and than the control combining forecast and expert-specific losses (ELI-style) in seven. The ELI-style control performs better on DLinear/Electricity and FITS/Electricity. The supplement reports configurations and run-level variation.

4.3 Ablation Study

Table 2 compares four variants trained from initialization on three settings under matched configurations, including expert counts, shared residuals, and training budgets. Uniform responsibility uses equal weights, controlling for capacity. Forecast-only responsibility retains historical forecast weighting but trains experts equally across channels, testing whether prediction weighting alone recovers the gains. Instant outcome retains adaptive prediction and training but uses only the latest observed losses at the next step ($\beta = 0$), testing the contribution of accumulated history. Forecast-only responsibility remains close to uniform, while instant outcomes recover part of the improvement. Full cMoE achieves the lowest mean MSE and MAE, reducing MSE by 3.83%, 3.29%, and 3.52% relative to uniform responsibility. Supplementary Section F reports run-level variation.

Table 2: Component ablation: averaged over three runs. Bold marks the best result in each column.

Variant	DLinear / Weather-720		FITS / Electricity-96		iTransformer / Weather-96	
	MSE	MAE	MSE	MAE	MSE	MAE
Uniform responsibility	0.3538	0.3938	0.1986	0.2757	0.1741	0.2140
Instant outcome ($\beta = 0$)	0.3433	0.3864	0.1935	0.2717	0.1715	0.2116
Forecast-only responsibility	0.3538	0.3939	0.1986	0.2757	0.1740	0.2139
Full cMoE	**0.3403**	**0.3822**	**0.1921**	**0.2704**	**0.1680**	**0.2091**

Because these comparisons change both trained models and forecast weights, a paired intervention tests whether the same weights become more useful under responsibility-weighted training. Branches share a checkpoint and retain identical historical forecast weights, using equal-channel or responsibility-weighted training. On each trained model, shared-weight MSE minus channel-specific MSE measures the benefit of channel-specific prediction; shared weights average the original weights across channels. Responsibility-weighted training increases this benefit by 0.00360 ± 0.01431, 0.00352 ± 0.00102, and 0.00534 ± 0.00039 for DLinear, FITS, and iTransformer, respectively (mean $\pm$ sample standard deviation over three paired runs; Supplementary Table S9). DLinear varies more, with positive differences in two of three runs.

We refit shared and channel-specific weights on validation data for frozen component-ablation models to test whether the benefit of channel-specific combination remains training-dependent (Supplementary Table S10). Historical training increases the mean benefit of channel-specific combination over instant-outcome training in all three settings, although the equal-channel model still has lower test MSE than the historical model after channel-specific refitting on DLinear.

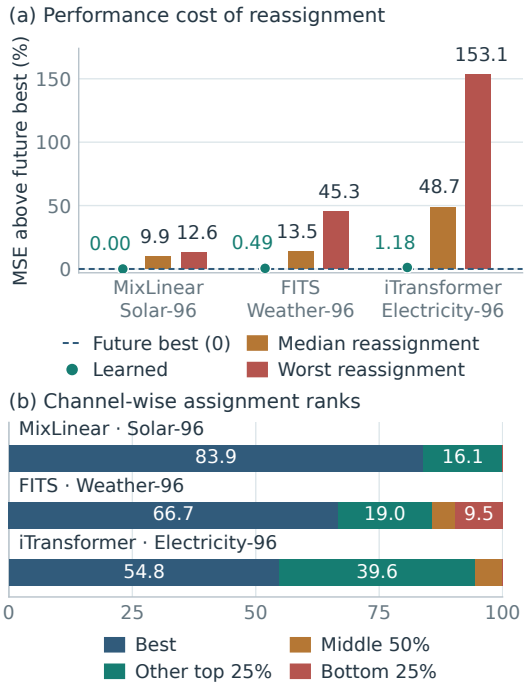


Figure 4: Learned expert identities remain near the best feasible assignment. With soft weights and active-expert counts fixed per channel, only expert identities change. (a) Aggregate MSE excess over the future-best assignment. (b) Channel-wise rank distribution of learned assignments among feasible reassignments. Future-best is a post-hoc validation reference; routing uses historical outcomes.

To isolate memory's training effect, ETTm2-96 branches start from the same training state and differ only in β; forecast weights are refitted afterward (Supplementary Table S11). History lowers iTransformer's test MSE and increases its benefit from channel-specific combination. DLinear remains nearly tied, with uniform responsibilities at the historical branch's final training checkpoint. These results support a training-dependent value of channel-specific prediction; outcome memory's added benefit varies across settings.

4.4 Expert Specialization Analysis

The analysis of responsibility-weighted learning characterizes how channel weighting changes expert-parameter dispersion under virtual gradient steps (Eq. 4; diagnostics in Supplementary Section F). We next examine whether the trained pools offer complementary channel-level competence for channel-specific forecast combination. The analysis uses MixLinear/Solar-96, FITS/Weather-96, and iTransformer/Electricity-96, spanning linear, frequency, and attention backbones.

For the responsibility maps, we additionally use Weather's 21 channels with six DLinear experts and five experts each for FITS and iTransformer to keep individual channel roles legible. Figure 3 describes competence and responsibility in separate saved pools. In panel (a), different channels favor different experts in the nine-expert DLinear/Weather-720 pool, with several experts remaining competitive. In panel (b), experts attain the lowest validation MSE on different proportions of channels. Panel (c) shows how channels contribute to multiple experts, with rows denoting experts and columns denoting channels; each expert's objective normalizes these weights by its total responsibility, as in Eq. 2. Expert indices are local to each pool. Together, these results show channel-relative complementarity. Matched same-start comparisons in Supplementary Section F further show that responsibility-weighted training increases endpoint selection advantage over equal-channel training, linking this complementarity to the training allocation.

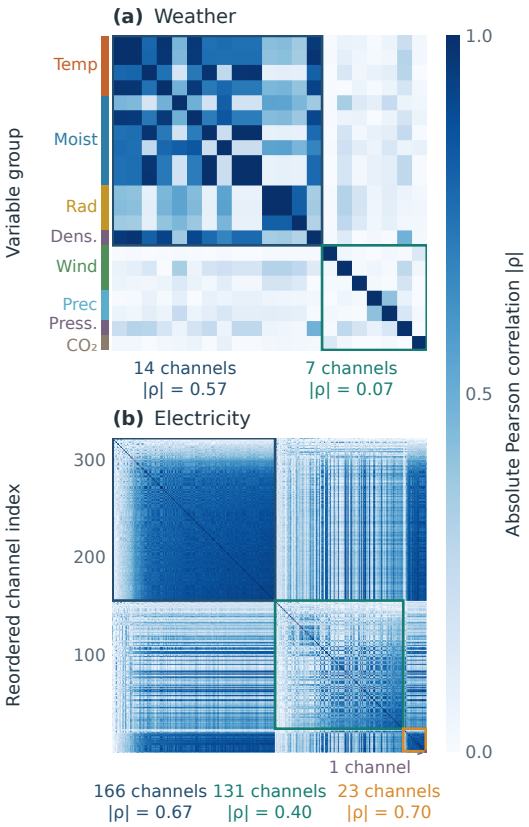


Figure 5: Post-hoc channel-structure analysis. Channels are ordered by learned responsibility groups; outlines mark groups and labels report group size and within-group mean absolute correlation. Correlations are not used by the router.

4.5 Expert Assignment Analysis

We next test whether learned responsibility captures informative channel–expert assignments beyond the magnitudes of the mixture weights. For each channel, we fix the trained expert pool, nonzero soft weights, and active-expert count, then evaluate all feasible assignments of these weights to expert identities on held-out validation windows (Figure 4).

The learned assignment has 0.0%, 0.5%, and 1.2% excess MSE over the future-best assignment on MixLinear/Solar-96, FITS/Weather-96, and iTransformer/Electricity-96. Panel (b) places the learned identities within the top quarter for 100%, 85.7%, and 94.4% of channels, respectively.

The historical signal is further assessed through the expert-loss advantage bound in Eq. 6. We evaluate its empirical coverage on training trajectories. With $H = 32$ and 32 non-overlapping calibration windows, coverage of this lower bound is 96.56%, 96.98%, and 96.50% on DLinear/Weather-720, FITS/Electricity-96, and iTransformer/Weather-96, respectively (nominal 95%). The supplement reports the calibration audit, four-combination MSE/MAE results, and run-level uncertainty.

4.6 Channel Structure Analysis

In Figure 5, Weather channels are grouped by active expert set and Electricity channels by highest-weight expert. The router receives neither physical labels nor correlations. Thus, learned responsibility can align with observable channel structure without using correlation information explicitly.

5 Conclusion

cMoE uses historical channel–expert errors to guide both expert training and forecast aggregation, without predefined groups or a separately trained router. Paired interventions on linear, frequency, and attention backbones show that responsibility-weighted training increases the mean forecasting benefit of the same historical channel weights. Across 180 settings with five backbones and nine datasets, cMoE improves each backbone and dataset on average. These results indicate that the value of channel-specific forecast weights depends on the competence developed through training.

AI USE STATEMENT

Generative AI tools were used to assist with manuscript polishing and figure preparation. The authors take responsibility for the final content of this work.

LIMITATIONS

Full channel–expert evaluation makes training costs grow with the expert count. On the three profiled settings, cMoE takes 2.77–$3.64\times$ the single-backbone training time; Supplementary Section C reports memory, parameter counts, and same-pool costs. Abrupt regime changes can make Scoreboard memory stale. The current implementation retains all experts at inference and freezes responsibility; sparse deployment and online adaptation remain practical extensions.

REFERENCES

Jialin Chen, Jan Eric Lenssen, Aosong Feng, Weihua Hu, Matthias Fey, Leandros Tassiulas, Jure Leskovec, and Rex Ying. From similarity to superiority: Channel clustering for time series forecasting. In *Advances in Neural Information Processing Systems*, 2024a.

Peng Chen, Yingying Zhang, Yunyao Cheng, Yang Shu, Yihang Wang, Qingsong Wen, Bin Yang, and Chenjuan Guo. Pathformer: Multi-scale transformers with adaptive pathways for time series forecasting. In *International Conference on Learning Representations*, 2024b.

Jinliang Deng, Xiusi Chen, Renhe Jiang, Xuan Song, and Ivor W. Tsang. ST-Norm: Spatial and temporal normalization for multi-variate time series forecasting. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 269–278, 2021. doi: 10.1145/3447548.3467330.

Zheng Dong, Renhe Jiang, Haotian Gao, Hangchen Liu, Jinliang Deng, Qingsong Wen, and Xuan Song. Heterogeneity-informed meta-parameter learning for spatiotemporal time series forecasting. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 631–641, 2024. doi: 10.1145/3637528.3671961.

Btissame El Mahtout and Florian Ziel. Fast training of mixture-of-experts for time series forecasting via expert loss integration. *arXiv preprint arXiv:2605.10330*, 2026.

Wei Fan, Shun Zheng, Pengyang Wang, Rui Xie, Kun Yi, Qi Zhang, Jiang Bian, and Yanjie Fu. IN-Flow: Instance normalization flow for non-stationary time series forecasting. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.1*, pp. 295–306, 2025. doi: 10.1145/3690624.3709260. URL https://doi.org/10.1145/3690624.3709260.

William Fedus, Barret Zoph, and Noam Shazeer. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. *Journal of Machine Learning Research*, 23(120):1–39, 2022.

Yoav Freund and Robert E. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, 55(1):119–139, 1997.

Rentao Gu, Yihang Ding, Junjie Li, Yi Ding, Weijing Sang, Xiaoli Huo, Xin Qin, and Yuefeng Ji. FM-LLM: A frequency-enhanced mixture-of-experts framework for adapting LLMs to time series forecasting. *Knowledge-Based Systems*, 341:115776, 2026. doi: 10.1016/j.knosys.2026.115776.

Abner Guzman-Rivera, Dhruv Batra, and Pushmeet Kohli. Multiple choice learning: Learning to produce multiple structured outputs. In *Advances in Neural Information Processing Systems*, volume 25, 2012.

Lu Han, Xu-Yang Chen, Han-Jia Ye, and De-Chuan Zhan. SOFTS: Efficient multivariate time series forecasting with series-core fusion. In *Advances in Neural Information Processing Systems*, 2024.

Robert A. Jacobs, Michael I. Jordan, Steven J. Nowlan, and Geoffrey E. Hinton. Adaptive mixtures of local experts. *Neural Computation*, 3(1):79–87, 1991.

Renhe Jiang, Zhaonan Wang, Jiawei Yong, Puneet Jeph, Quanjun Chen, Yasumasa Kobayashi, Xuan Song, Shintaro Fukushima, and Toyotaro Suzumura. Spatio-temporal meta-graph learning for traffic forecasting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pp. 8078–8086, 2023. doi: 10.1609/aaai.v37i7.25976.

Michael I. Jordan and Robert A. Jacobs. Hierarchical mixtures of experts and the EM algorithm. *Neural Computation*, 6(2):181–214, 1994.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations*, 2015. URL https://arxiv.org/abs/1412.6980.

M. P. Kumar, Benjamin Packer, and Daphne Koller. Self-paced learning for latent variable models. In *Advances in Neural Information Processing Systems*, volume 23, 2010.

Guokun Lai, Wei-Cheng Chang, Yiming Yang, and Hanxiao Liu. Modeling long- and short-term temporal patterns with deep neural networks. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*, pp. 95–104. ACM, 2018. doi: 10.1145/3209978.3210006.

Stefan Lee, Senthil Purushwalkam Shiva Prakash, Michael Cogswell, Viresh Ranjan, David Crandall, and Dhruv Batra. Stochastic multiple choice learning for training diverse deep ensembles. In *Advances in Neural Information Processing Systems*, volume 29, 2016.

Dmitry Lepikhin, HyoukJoong Lee, Yuanzhong Xu, Dehao Chen, Orhan Firat, Yanping Huang, Maxim Krikun, Noam Shazeer, and Zhifeng Chen. GShard: Scaling giant models with conditional computation and automatic sharding. In *International Conference on Learning Representations*, 2021.

Dongyuan Li, Shun Zheng, Chang Xu, Jiang Bian, and Renhe Jiang. Routing channel-patch dependencies in time series forecasting with graph spectral decomposition. In *International Conference on Learning Representations*, 2026a.

Qianyang Li, Xingjun Zhang, Shaoxun Wang, Tao Peng, and Jia Wei. FAME: Forecastability-aware mixture of experts for heterogeneous time series forecasting. *arXiv preprint arXiv:2606.08896*, 2026b.

Nick Littlestone and Manfred K. Warmuth. The weighted majority algorithm. *Information and Computation*, 108(2):212–261, 1994.

Hangchen Liu, Zheng Dong, Renhe Jiang, Jiewen Deng, Jinliang Deng, Quanjun Chen, and Xuan Song. Spatio-temporal adaptive embedding makes vanilla transformer SOTA for traffic forecasting. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*, pp. 4125–4129, 2023a.

Xu Liu, Juncheng Liu, Gerald Woo, Taha Aksu, Yuxuan Liang, Roger Zimmermann, Chenghao Liu, Junnan Li, Silvio Savarese, Caiming Xiong, and Doyen Sahoo. Moirai-MoE: Empowering time series foundation models with sparse mixture of experts. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 38940–38962. PMLR, 2025a.

Yong Liu, Tengge Hu, Haoran Zhang, Haixu Wu, Shiyu Wang, Lintao Ma, and Mingsheng Long. iTransformer: Inverted transformers are effective for time series forecasting. In *International Conference on Learning Representations*, 2024.

Zhiding Liu, Mingyue Cheng, Zhi Li, Zhenya Huang, Qi Liu, Yanhu Xie, and Enhong Chen. Adaptive normalization for non-stationary time series forecasting: A temporal slice perspective. In *Advances in Neural Information Processing Systems*, volume 36, pp. 14273–14292, 2023b. doi: 10.52202/075280-0628. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/2e19dab94882bc95ed094c4399cfda02-Abstract-Conference.html.

Zhiding Liu, Jiqian Yang, Qingyang Mao, Yuze Zhao, Mingyue Cheng, Zhi Li, Qi Liu, and Enhong Chen. DisenTS: Disentangled channel evolving pattern modeling for multivariate time series forecasting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2026. doi: 10.1109/TPAMI.2026.3697127. URL https://doi.org/10.1109/TPAMI.2026.3697127.

Zhining Liu, Ze Yang, Xiao Lin, Ruizhong Qiu, Tianxin Wei, Yada Zhu, Hendrik Hamann, Jingrui He, and Hanghang Tong. Breaking silos: Adaptive model fusion unlocks better time series forecasting. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 40022–40042. PMLR, 2025b.

Ziqi Liu. FreqMoE: Enhancing time series forecasting through frequency decomposition mixture of experts. In *Proceedings of the 28th International Conference on Artificial Intelligence and Statistics*, volume 258, pp. 3430–3438. PMLR, 2025. URL https://proceedings.mlr.press/v258/liu25i.html.

Aitian Ma, Dongsheng Luo, and Mo Sha. MixLinear: Extreme low-resource multivariate time series forecasting with 0.1k parameters. In *International Conference on Learning Representations*, 2026.

Xiaowen Ma, Shuning Ge, Fan Yang, Xiangyu Li, Yun Chen, Mengting Ma, Wei Zhang, and Zhipeng Liu. TimeExpert: Boosting long time series forecasting with temporal mix of experts. *arXiv preprint arXiv:2509.23145*, 2025.

Pablo Montero-Manso, George Athanasopoulos, Rob J. Hyndman, and Thiyanga S. Talagala. FFORMA: Feature-based forecast model averaging. *International Journal of Forecasting*, 36(1):86–92, 2020.

Ronghao Ni, Zinan Lin, Shuaiqi Wang, and Giulia Fanti. Mixture-of-linear-experts for long-term time series forecasting. In *Proceedings of The 27th International Conference on Artificial Intelligence and Statistics*, volume 238 of *Proceedings of Machine Learning Research*, pp. 4672–4680. PMLR, 2024.

Yuqi Nie, Nam H. Nguyen, Phanwadee Sinthong, and Jayant Kalagnanam. A time series is worth 64 words: Long-term forecasting with transformers. In *International Conference on Learning Representations*, 2023.

Liran Nochumsohn, Raz Marshanski, Hedi Zisling, and Omri Azencot. Super-Linear: A lightweight pretrained mixture of linear experts for time series forecasting. *Transactions on Machine Learning Research*, 2026. URL https://openreview.net/forum?id=av8niDGYMk.

Evandro S. Ortigossa and Eran Segal. Seg-moe: Multi-resolution segment-wise mixture-of-experts for time series forecasting transformers. *arXiv preprint arXiv:2601.21641*, 2026.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. PyTorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems*, volume 32, 2019. URL https://proceedings.neurips.cc/paper_files/paper/2019/file/bdbca288fee7f92f2bfa9f7012727740-Paper.pdf.

Xiangfei Qiu, Xingjian Wu, Yan Lin, Chenjuan Guo, Jilin Hu, and Bin Yang. DUET: Dual clustering enhanced multivariate time series forecasting. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2025. doi: 10.1145/3690624.3709325.

Xiangfei Qiu, Hanyin Cheng, Xingjian Wu, Junkai Lu, Jilin Hu, Chenjuan Guo, Christian S. Jensen, and Bin Yang. A comprehensive survey of deep learning for multivariate time series forecasting: A channel strategy perspective. In *Proceedings of the Thirty-Fifth International Joint Conference on Artificial Intelligence*, pp. 7995–8003, 2026. doi: 10.24963/ijcai.2026/887. URL https://www.ijcai.org/proceedings/2026/887.

Adrian E. Raftery, Miroslav Kárný, and Pavel Ettler. Online prediction under model uncertainty via dynamic model averaging: Application to a cold rolling mill. *Technometrics*, 52(1):52–66, 2010. doi: 10.1198/TECH.2009.08104.

Ozan Sener and Vladlen Koltun. Multi-task learning as multi-objective optimization. In *Advances in Neural Information Processing Systems*, volume 31, 2018.

Noam Shazeer, Azalia Mirhoseini, Krzysztof Maziarz, Andy Davis, Quoc Le, Geoffrey Hinton, and Jeff Dean. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. In *International Conference on Learning Representations*, 2017.

Lifeng Shen, Weiyu Chen, and James T. Kwok. Multi-resolution diffusion models for time series forecasting. In *International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=mmjnr0G8ZY.

Xiaoming Shi, Shiyu Wang, Yuqi Nie, Dianqi Li, Zhou Ye, Qingsong Wen, and Ming Jin. Time-MoE: Billion-scale time series foundation models with mixture of experts. In *International Conference on Learning Representations*, 2025.

Yanru Sun, Zongxia Xie, Emadeldeen Eldele, Dongyue Chen, Qinghua Hu, and Min Wu. Learning pattern-specific experts for time series forecasting under patch-level distribution shift. In *Advances in Neural Information Processing Systems*, volume 38, pp. 91810–91844, 2025. doi: 10.52202/085713-3071. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/8491a7fcc218946b471b600a915c8b02-Abstract-Conference.html.

Rui Wang, Renhao Xue, Ray Razi, Huan Song, and Hannah R. Marlowe. AME-TS: Anchored mixture-of-experts for time series forecasting. *arXiv preprint arXiv:2605.25166*, 2026.

Shiyu Wang, Haixu Wu, Xiaoming Shi, Tengge Hu, Huakun Luo, Lintao Ma, James Y. Zhang, and Jun Zhou. TimeMixer: Decomposable multiscale mixing for time series forecasting. In *International Conference on Learning Representations*, 2024a.

Shiyu Wang, Jiawei Li, Xiaoming Shi, Zhou Ye, Baichuan Mo, Wenze Lin, Shengtong Ju, Zhixuan Chu, and Ming Jin. TimeMixer++: A general time series pattern machine for universal predictive analysis. In *International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=1CLzLXSFNn.

Yuxuan Wang, Haixu Wu, Jiaxiang Dong, Guo Qin, Haoran Zhang, Yong Liu, Yunzhong Qiu, Jianmin Wang, and Mingsheng Long. TimeXer: Empowering transformers for time series forecasting with exogenous variables. In *Advances in Neural Information Processing Systems*, 2024b.

Haixu Wu, Jiehui Xu, Jianmin Wang, and Mingsheng Long. Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting. In *Advances in Neural Information Processing Systems*, 2021.

Haixu Wu, Tengge Hu, Yong Liu, Hang Zhou, Jianmin Wang, and Mingsheng Long. TimesNet: Temporal 2d-variation modeling for general time series analysis. In *International Conference on Learning Representations*, 2023.

Zhijian Xu, Ailing Zeng, and Qiang Xu. FITS: Modeling time series with 10k parameters. In *International Conference on Learning Representations*, 2024.

Weiwei Ye, Songgaojun Deng, Qiaosha Zou, and Ning Gui. Frequency adaptive normalization for non-stationary time series forecasting. In *Advances in Neural Information Processing Systems*, volume 37, 2024.

Kun Yi, Qi Zhang, Wei Fan, Shoujin Wang, Pengyang Wang, Hui He, Ning An, Defu Lian, Longbing Cao, and Zhendong Niu. Frequency-domain MLPs are more effective learners in time series forecasting. In *Advances in Neural Information Processing Systems*, volume 36, pp. 76656–76679, 2023. doi: 10.52202/075280-3349. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/f1d16af76939f476b5f040fd1398c0a3-Abstract-Conference.html.

Tianhe Yu, Saurabh Kumar, Abhishek Gupta, Sergey Levine, Karol Hausman, and Chelsea Finn. Gradient surgery for multi-task learning. In *Advances in Neural Information Processing Systems*, volume 33, pp. 5824–5836, 2020.

Wenzhen Yue, Yong Liu, Hao Wang, Haoxuan Li, Xianghua Ying, Ruohao Guo, Bowei Xing, and Ji Shi. OLinear: A linear model for time series forecasting in orthogonally transformed domain. In *Advances in Neural Information Processing Systems*, 2025.

Ailing Zeng, Muxi Chen, Lei Zhang, and Qiang Xu. Are transformers effective for time series forecasting? In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2023.

Huanyao Zhang, Jiaye Lin, Wentao Zhang, Haitao Yuan, and Guoliang Li. AdaMixT: Adaptive weighted mixture of multi-scale expert transformers for time series forecasting. In *Proceedings of the Thirty-Fourth International Joint Conference on Artificial Intelligence*, pp. 3633–3641, 2025. doi: 10.24963/ijcai.2025/404.

Lingzheng Zhang, Lifeng Shen, Yimin Zheng, Shiyuan Piao, Ziyue Li, and Fugee Tsung. LeMoLE: LLM-enhanced mixture of linear experts for time series forecasting. *arXiv preprint arXiv:2412.00053*, 2024. doi: 10.48550/arXiv.2412.00053. URL https://arxiv.org/abs/2412.00053.

Haoyi Zhou, Shanghang Zhang, Jieqi Peng, Shuai Zhang, Jianxin Li, Hui Xiong, and Wancai Zhang. Informer: Beyond efficient transformer for long sequence time-series forecasting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2021.

A Theory and Supporting Analysis

This appendix supports the link between training responsibility, expert competence, and forecast combination. Theorem 1 below is the central theoretical result: it provides an exact update-level identity for the effect of channel composition on expert dispersion. The remaining results are supporting consequences, measurement definitions, or experimental diagnostics for the corresponding regimes.

A.1 Core Update-Level Specialization Result

Theorem 1 is exact for the matched one-step comparison. The propositions and lemmas that follow make its leading term measurable, relate it to persistent channel gradients, and describe how to evaluate the same mechanism over finite windows.

We suppress the training-step index in the single-update derivation and retain the main text's pressure notation $\mathcal{P}$. Here $w_{c,k}$ denotes the main text's $w_{t,c}^{k}$. Write $\theta_k = \bar{\theta} + \delta_k$, with $K^{-1}\sum_k \delta_k = 0$. Assume positive total responsibility for each expert in this comparison. Let $q_{c,k} = w_{c,k}/\sum_{c'} w_{c',k}$ be the unclipped responsibility normalized within expert k and define

$$Q_{c,k} = q_{c,k} - \frac{1}{C}, \qquad \sum_c Q_{c,k} = 0. \tag{S1}$$

$Q_{c,k}$ is expert k's outcome-written shift in channel composition from equal-channel training. This normalization matches the implemented objective when each expert's total responsibility is at least 10^{-6}; below that threshold, the implementation floors the denominator.

Theorem 1 (exact responsibility–gradient specialization identity). Let r_c be the current channel training loss, so $r_c(\theta_k) = \ell_{t,c}^{\mathrm{train},k}$ in the main text, and let $g_{c,k} = \nabla r_c(\theta_k)$. Compare one responsibility-weighted gradient update with an equal-channel update from the same expert state. Let $\eta > 0$ be their common step size and define

$$\mathcal{P} = -\sum_{c,k} Q_{c,k}\delta_k^{\top} g_{c,k} \tag{S2}$$

and $D = K^{-1}\sum_k \|\theta_k - K^{-1}\sum_j \theta_j\|^2$. The one-step dispersion lift is

$$L = D_+^{\mathrm{outcome}} - D_+^{\mathrm{equal}} = \frac{2\eta}{K}\mathcal{P} + R^{(2)}, \tag{S3}$$

where $R^{(2)}$ is the exact difference between the quadratic update terms. This identity requires neither a linear parameter-to-loss approximation nor a particular backbone.

Proof. The gradient difference for expert k is $G_k^{\mathrm{outcome}} - G_k^{\mathrm{equal}} = \sum_c Q_{c,k} g_{c,k}$. For either update, expanding the squared distance from the updated expert mean gives a linear term $-2\eta K^{-1}\sum_k \delta_k^{\top} G_k$ and a quadratic term in the centered expert gradients. Subtracting the two updates yields Eq. S3, with their quadratic difference collected in $R^{(2)}$. □

Proposition 1 (pre-update directional certificate). Define $\widehat{L} = 2\eta\mathcal{P}/K$. After the current target is observed and before the optimizer step, $\widehat{L}$ and $R^{(2)}$ can be evaluated from the current parameters, responsibility, and per-channel gradients. By Theorem 1, $|L - \widehat{L}| = |R^{(2)}|$. Therefore,

$$|\widehat{L}| > |R^{(2)}| \quad \Longrightarrow \quad \mathrm{sign}(L) = \mathrm{sign}(\widehat{L}). \tag{S4}$$

The condition certifies the direction of the pending specialization lift without reading post-update parameters. It characterizes the current labeled update; future unlabeled batches are outside this local update test.

Lemma 1 (Scoreboard outcome-contrast expansion). For an auxiliary softmax applied directly to raw Scoreboard entries, let

$$w_{c,k} = \frac{\exp(-S_{c,k}/\tau)}{\sum_j \exp(-S_{c,j}/\tau)} \tag{S5}$$

be formed from the Scoreboard with temperature $\tau > 0$. Define the doubly centered outcome contrast

$$A_{c,k} = S_{c,k} - \bar{S}_{c\cdot} - \bar{S}_{\cdot k} + \bar{S}_{\cdot}. \tag{S6}$$

The bars denote averages over the indicated indices; $\bar{S}$ is the grand mean. Let $\varepsilon_S = \max_{c,k} |S_{c,k} - \bar{S}_{c\cdot}|/\tau$ measure the full within-channel score spread. As $\varepsilon_S \to 0$,

$$Q_{c,k} = -\frac{A_{c,k}}{C\tau} + O\left(\frac{\varepsilon_S^2}{C}\right). \tag{S7}$$

Proof. Softmax expansion gives

$$w_{c,k} = \frac{1}{K}\left[1 - \frac{S_{c,k} - \bar{S}_{c\cdot}}{\tau} + O(\varepsilon_S^2)\right].$$

Summing over channels and expanding the reciprocal of that sum yields Eq. S7. The remainder includes expert-wide score offsets, which double centering removes only from the leading coefficient. □

This auxiliary map omits row-relative normalization, score scaling, balancing, and concentration. Theorem 1 instead uses the final responsibility and remains exact under these transformations. For this auxiliary map, substitution into Theorem 1's linear term gives the approximation

$$\frac{2\eta}{K}\mathcal{P} = \frac{2\eta}{KC\tau}\sum_{c,k} A_{c,k}\delta_k^\top g_{c,k} + O\left(\frac{\eta\varepsilon_S^2}{KC}\sum_{c,k}|\delta_k^\top g_{c,k}|\right). \tag{S8}$$

Pre-update interpretation and alignment audit. For a labeled training step, the Scoreboard contrast, current expert deviations, and per-channel gradients can be evaluated before the optimizer changes the parameters; the audit explicitly materializes these gradients. The leading expression in Eq. S8 approximates the pending linear lift. Its error includes the routing expansion and, for the full lift, $R^{(2)}$.

We test whether the prediction contains channel–expert alignment information rather than only total expert weight or expert distance. The audit contains 36 consecutive-batch windows from the linear, frequency, and attention settings. These probes use recorded outcomes and gradients at the common expert mean. This common-mean diagnostic is separate from the exact expert-state identity in Theorem 1.

Across these probes, ε_S ranges from 2.25×10^{-5} to 0.0563. Direct softmax evaluation gives composition errors below $0.441\varepsilon_S^2/C$ in every probe; this is an empirical scale check.

In this common-mean diagnostic, the aligned predictor obtains 36/36 correct signs, a Spearman correlation of 1.0, and a family-normalized mean absolute error of 0.00137. In each of 10,000 randomizations, one channel permutation or one expert permutation is fixed for all four windows of a backbone seed trajectory, preserving within-trajectory dependence and the marginal contrast values while breaking their identity correspondence. Channel permutations have a mean sign accuracy of 0.486 and a maximum of 0.833; expert permutations have a mean of 0.488 and a maximum of 0.917. Their maximum Spearman correlations are 0.707 and 0.811, respectively. None of the 10,000 randomizations matches the aligned predictor on sign accuracy, rank correlation, or normalized error. A nonnegative magnitude-only control obtains 26/36 correct signs and a Spearman correlation of 0.107. Thus, the empirical prediction depends on the signed correspondence among outcome contrast, expert deviation, and channel gradient.

A sufficient local condition. Continue with the auxiliary raw-score map and assume that the Scoreboard and training use the same channel loss. Suppose $S_{c,k} = r_{t-1,c}(\bar{\theta} + \delta_k)$ and each channel

risk is twice continuously differentiable near $\bar{\theta}$. With $g_{t,c} = \nabla r_{t,c}(\bar{\theta})$, $\bar{g}_t = C^{-1}\sum_c g_{t,c}$, and $\rho = \max_k \|\delta_k\|$,

$$A_{c,k} = (g_{t-1,c} - \bar{g}_{t-1})^\top \delta_k + O(\rho^2). \tag{S9}$$

Consequently, Eq. S8 becomes

$$\frac{2\eta}{K\tau}\sum_k \delta_k^\top \operatorname{Sym}(B_t)\delta_k + O(\eta\rho^3), \qquad B_t = \frac{1}{C}\sum_c (g_{t,c} - \bar{g}_t)(g_{t-1,c} - \bar{g}_{t-1})^\top. \tag{S10}$$

Here $\operatorname{Sym}(B_t) = (B_t + B_t^\top)/2$ is the symmetric part of B_t. Thus, persistent channel-gradient covariance is a sufficient local explanation of the more general responsibility–gradient identity. Since $\varepsilon_S = O(\rho)$ here, the corrected routing remainder contributes $O(\eta\rho^3)$. The predictor requires near-uniform softmax scores but not linear reconstruction of past losses from parameters.

A.2 Supporting Finite-Window Interpretation

For a same-state comparison at step t, define

$$L_t(\theta) = D(F_{S,t}(\theta)) - D(F_{E,t}(\theta))\,, \tag{S11}$$

where $F_{S,t}$ is the outcome-written update and $F_{E,t}$ is the equal-channel update with the same total weight per expert. Let $L_t^{(1)} = (2\eta/K)\mathcal{P}_t$ denote the exact linear term in Eq. S3 and define the measurable remainder $R_t = L_t - L_t^{(1)}$. Then, for any consecutive window,

$$\sum_{t=0}^{T-1} L_t = \sum_{t=0}^{T-1} L_t^{(1)} + \sum_{t=0}^{T-1} R_t, \tag{S12}$$

and

$$\sum_{t=0}^{T-1} L_t \geq \sum_{t=0}^{T-1} L_t^{(1)} - \sum_{t=0}^{T-1} |R_t|. \tag{S13}$$

This statement is an exact summation identity: the responsibility–gradient pressure and its quadratic update remainder are measured separately.

For two real branches initialized from the same state, let $\Delta_t = D(\theta_t^S) - D(\theta_t^E)$ and

$$\rho_t = D\big(F_{E,t}(\theta_t^S)\big) - D\big(F_{E,t}(\theta_t^E)\big)\,. \tag{S14}$$

The one-step relation is $\Delta_{t+1} = L_t(\theta_t^S) + \rho_t$. Here ρ_t carries the existing branch separation through the control update. Subtracting Δ_t before summation gives

$$\Delta_T = \Delta_0 + \sum_{t=0}^{T-1} L_t(\theta_t^S) + \sum_{t=0}^{T-1} (\rho_t - \Delta_t). \tag{S15}$$

If the dispersion-change map $\theta \mapsto D(F_{E,t}(\theta)) - D(\theta)$ is M_t-Lipschitz on the segment joining the branch states, then

$$|\rho_t - \Delta_t| \leq M_t \|\theta_t^S - \theta_t^E\|. \tag{S16}$$

Thus, the real-branch gap combines local lifts with the propagation corrections $\rho_t - \Delta_t$. Endpoint accumulation is determined by the measured or bounded sum of both terms. □

For optimizers with memory, the same telescoping relation applies to full training states, including optimizer buffers. Its local lifts must use those optimizer maps; the virtual gradient probe is a separate measurement.

Channel-relative competence interpretation. If $g_{t,c} = g_{t-1,c}$ locally, then $B_t = \Sigma_t \succeq 0$, where

$$\Sigma_t = \frac{1}{C}\sum_c (g_{t,c} - \bar{g}_t)(g_{t,c} - \bar{g}_t)^\top. \tag{S17}$$

Under the sufficient local condition above, the gradient-covariance form is nonnegative and is strictly positive whenever some δ_k has a component outside the null space of Σ_t. To express the same

one-step comparison in channel-relative risk coordinates, hold the local metric Σ_t fixed during the two candidate updates and define

$$V_t(\delta) = \frac{1}{K}\sum_k \delta_k^\top \Sigma_t \delta_k = \frac{1}{CK}\sum_{c,k}\big[(g_{t,c} - \bar{g}_t)^\top \delta_k\big]^2. \tag{S18}$$

If $\delta_+^{\text{outcome}}$ and δ_+^{equal} denote the centered expert perturbations after the two updates, then $V_t(\delta_+^{\text{outcome}}) - V_t(\delta_+^{\text{equal}})$ equals

$$\frac{2\eta}{K\tau}\sum_k \delta_k^\top \Sigma_t^2 \delta_k + O(\eta\rho^3 + \eta^2\rho^2). \tag{S19}$$

Its leading term is nonnegative and acts in directions that change channel-relative losses. Positivity of the full finite step requires that this term dominates the remainder.

Under the stated loss-based Scoreboard construction, identical parameters also give identical responsibilities and updates. Initialization or optimization differences are therefore needed to start this feedback.

A.3 A FIXED-FEATURE POPULATION BRIDGE

The update identity above is local and applies to arbitrary backbones. We additionally provide a fixed-feature population result that connects a frozen responsibility matrix directly to the final mixture forecast. It characterizes the fixed-feature population regime, while the update identity applies to jointly trained deep cMoE.

Let $x \in \mathbb{R}^d$ be a fixed-feature vector with a second-moment matrix $\Sigma = \mathbb{E}[xx^\top] \succeq 0$ shared across channels. Channel c has a target $y_c = a_c^\top x + \varepsilon_c$, where $\mathbb{E}[x\varepsilon_c] = 0$. Let $W \in \mathbb{R}^{C\times K}$ be a fixed row-stochastic responsibility matrix, let $m = W^\top \mathbf{1}$ with $m_k > 0$, and define

$$P = W\,\mathrm{diag}(m)^{-1}W^\top. \tag{S20}$$

Each expert minimizes its responsibility-normalized population squared loss. Let $A \in \mathbb{R}^{C\times d}$ contain $a_c^\top$ in row c, let $\bar{a} = C^{-1}\sum_c a_c$, and define the centered coefficient matrix $A_0 = A - \mathbf{1}\bar{a}^\top$.

Theorem 2 (fixed-feature mixture gap). One population-optimal coefficient of expert k is

$$b_k = \frac{1}{m_k}\sum_c W_{c,k}a_c.$$

Consequently, the responsibility-weighted mixture has a coefficient matrix PA, whereas equal-channel training has a coefficient matrix $\mathbf{1}\bar{a}^\top$. Their excess population risks satisfy

$$R_{\text{equal}} - R_{\text{mix}} = \frac{1}{C}\,\mathrm{tr}\big[A_0^\top(2P - P^2)A_0\Sigma\big] \geq 0. \tag{S21}$$

If $\Sigma \succ 0$, the inequality is strict exactly when $PA_0 \neq 0$.

Proof. The first-order condition gives $\Sigma(b_k - m_k^{-1}\sum_c W_{c,k}a_c) = 0$, so the displayed b_k is a minimizer. If Σ is singular, other minimizers differ only in their null space and give the same population predictions. Mixing these coefficients with W gives PA. Equal-channel training gives the same coefficient $\bar{a}$ to every expert. Since W is row-stochastic, P is symmetric and $P\mathbf{1} = \mathbf{1}$; because it is also positive semi-definite, its eigenvalues lie in $[0, 1]$. Hence $2P - P^2 \succeq 0$. Expanding the two centered squared risks and using $P\mathbf{1} = \mathbf{1}$ gives Eq. S21. Strictness follows from the null space of $2P - P^2$, which equals the null space of P, when $\Sigma \succ 0$. The noise terms cancel from the difference. □

The theorem identifies the measurable condition behind the gain: the responsibility operator must preserve some channel-level coefficient heterogeneity. If all rows of W are identical, then $PA_0 = 0$ and the theorem gives equality. The result does not cover changing W, jointly trained feature extractors, finite-step Adam, or the shared residual path; the main text therefore retains the arbitrary-backbone update identity as its primary result.

As an algebra audit, 100 random finite-dimensional instances gave a maximum identity error of 5.64×10^{-14}, no negative predicted gaps, and responsibility-operator eigenvalues in $[-2.33 \times 10^{-16}, 1.00 + 10^{-15}]$. Under uniform responsibility, the predicted and directly expanded gaps were both below 10^{-15} in magnitude.

Table S1: Competence contrast between responsibility-weighted and matched-mass channel-shuffled training, using the seed 2023 recorded window for each setting. Ω_t is defined in Eq. S22; Γ is the cMoE-minus-control gap.

Setting	Steps	Mean Γ	95% CI	Positive	Specialist gap
DLinear / Weather-720	24	0.06425	[0.06264, 0.06591]	100%	+4.621 pp
FITS / Electricity-96	25	0.00773	[0.00716, 0.00812]	100%	+3.572 pp
iTransformer / Weather-96	25	0.01806	[0.01410, 0.02124]	100%	+5.332 pp

A.4 Measuring Channel-Relative Competence

The parameter-dispersion identity describes a local update effect, but parameter distance serves as an update-level specialization diagnostic. We therefore evaluate the channel–expert loss matrix directly. For a recorded step t, define

$$\Omega_t = \frac{1}{C}\sum_c \left(\frac{1}{K}\sum_k \ell_{t,c,k} - \min_k \ell_{t,c,k}\right). \tag{S22}$$

This nonnegative gap measures expert-loss heterogeneity relative to equal-expert assignment, without depending on parameter coordinates. It is zero when all experts have the same loss on every channel. Channel-specific complementarity additionally requires different channels to favor different experts; the gap can be positive even when one expert is best on every channel. Let Ω be the gap computed from expected channel–expert losses, taking the minimum after expectation; $\widehat{\Omega}$ uses evaluation-set mean losses.

Proposition 2 (finite-sample estimation of competence contrast). Let $r_{c,k}$ be the expected loss of expert k on channel c. Suppose its estimate $\hat{r}_{c,k}$ is the mean of n independent losses in $[0, B]$, and let $\widehat{\Omega}$ be Eq. S22 computed from these empirical risks. Then, with probability at least $1-\delta$,

$$|\widehat{\Omega} - \Omega| \le 2B\sqrt{\frac{\log(2CK/\delta)}{2n}}. \tag{S23}$$

Indeed, a union bound and Hoeffding's inequality control every $|\hat{r}_{c,k} - r_{c,k}|$ by the square-root term without the leading factor two. Both the equal-expert average and the channel-wise minimum are one-Lipschitz in this maximum error, giving Eq. S23. Thus, expert-loss contrast can be estimated directly from the recorded loss matrix.

For each setting, we compare responsibility-weighted training with a matched-mass channel-shuffled branch. The control preserves each expert's total training weight while shuffling channel identity. Both branches start from the same pre-activation checkpoint and are evaluated at common recorded steps. Let $\Gamma_t = \Omega_t^S - \Omega_t^{\mathrm{shuf}}$, where S and shuf denote these two branches. We measure whether this competence contrast stays positive over the recorded window:

The routed branch has greater expert-loss contrast at every shared step in these three trajectories. The intervals use 20,000 paired moving-block bootstrap samples with a block length of four; lower bounds remain positive for block lengths from two to eight. This diagnostic measures differences among expert losses. Table S13 compares channel-wise expert choice with the best shared expert. Main Figure 3 describes channel competence and responsibility; Main Figure 4 tests expert identities while keeping the mixture weights fixed.

A.5 Supporting Allocation Certificate

The method uses a soft responsibility, so reliability need not require one expert to remain the hard winner. We instead certify the loss saved by the complete responsibility matrix over a fixed future window.

Let $\bar{\ell}_{t,H,c,k} = H^{-1}\sum_{j=1}^{H} \ell_{t+j,c,k}$. Define the future responsibility gain

$$G_{t,H} = \frac{1}{C}\sum_c \left(\frac{1}{K}\sum_k \bar{\ell}_{t,H,c,k} - \sum_k W_{t,c,k}\bar{\ell}_{t,H,c,k}\right). \tag{S24}$$

A positive value means that W_t gives a lower average expert loss than equal-expert assignment during the next H batches.

The Scoreboard predicts the same quantity without reading future outcomes:

$$\widehat{G}_t = \frac{1}{C}\sum_c \left(\frac{1}{K}\sum_k S_{t,c,k} - \sum_k W_{t,c,k} S_{t,c,k}\right). \tag{S25}$$

Let $E_{t,H} = \widehat{G}_t - G_{t,H}$. Then $G_{t,H} = \widehat{G}_t - E_{t,H}$ holds exactly.

At step t, the future residual is unknown. We calibrate it with B past residuals whose complete future windows have already been observed. These windows are mutually non-overlapping and remain in the same routing phase as the current window.

Let q_t be the rth order statistic of these residuals, where $r = \lceil (B+1)(1-\alpha) \rceil \le B$. For the experiments, $H = B = 32$ and $\alpha = 0.05$ give $r = 32$. Here $\alpha \in (0,1)$ is the target error probability. The main-text bound is $\mathcal{B}_t = \widehat{G}_t - q_t$. A competence regime is a contiguous interval with a fixed routing phase, Scoreboard update rule, expert pool, and responsibility transformation. Exchangeability within this interval means that permuting the calibration residuals and next-window residuals leaves their joint distribution unchanged.

Proposition 3 (future soft-responsibility certificate). Assume that the next aggregate residual and the B selected calibration residuals are exchangeable within the current competence regime. Then

$$\Pr\Big(G_{t,H} \ge \widehat{G}_t - q_t\Big) \ge \frac{r}{B+1} \ge 1-\alpha. \tag{S26}$$

Proof. Under exchangeability, the rank of $E_{t,H}$ among the $B+1$ residuals is uniform with random tie breaking; using the non-strict inequality is conservative in the presence of ties. Therefore $\Pr(E_{t,H} \le q_t) \ge r/(B+1)$. Substituting the exact identity gives Eq. S26. □

The bound is computed before the future window begins. Coverage is marginal for each eligible window under the stated exchangeability condition, with non-overlapping windows defining the calibration blocks. Observed coverage evaluates calibration on the recorded dependent trajectories.

To test whether the learned channel correspondence matters, we define a *permuted-channel assignment*. It permutes the channel rows of W_t while leaving the expert dimension unchanged. Every permutation preserves the total weight assigned to each expert exactly:

$$\sum_c \widetilde{W}_{t,c,k} = \sum_c W_{t,c,k} \quad \text{for every } k. \tag{S27}$$

We average over 32 deterministic derangements, each of which changes every channel row. Replacing equal-expert loss in Eq. S24 by the loss under this permuted assignment tests whether the learned channel–expert correspondence improves over an assignment with the same weight values and the same total weight per expert.

The total weight assigned to each expert is preserved to numerical precision for all three settings. Thus, the second comparison removes the total-weight explanation while preserving the observed future advantage of the original channel correspondence.

Table S2: Post-activation soft-responsibility certificates. "Cert." is the fraction of eligible steps with a positive lower bound; "Pos." is the future-positive rate among certified steps.

Setting	Reference	Cert.	Pos.	Coverage	Future gain
DLinear/Weather-720	Uniform responsibility	100.00	100.00	96.56	0.11579
FITS/Electricity-96	Uniform responsibility	100.00	100.00	96.98	0.01733
iTransformer/Weather-96	Uniform responsibility	100.00	100.00	96.50	0.07344
DLinear/Weather-720	Permuted channels	99.70	100.00	95.79	0.10805
FITS/Electricity-96	Permuted channels	100.00	100.00	97.07	0.01604
iTransformer/Weather-96	Permuted channels	100.00	100.00	96.54	0.07298

A.6 Equal-Channel Control

For expert k, let $g_{c,k} = \nabla \ell_{c,k}^{\text{train}}$, where $\ell_{c,k}^{\text{train}}$ denotes the main text's $\ell_{t,c}^{\text{train},k}$ with t suppressed, and let two assignments W, W' give it the same total weight $M_k = \sum_c W_{c,k} = \sum_c W'_{c,k}$. Their normalized gradient difference is

$$G_k(W) - G_k(W') = \frac{1}{M_k} \sum_c (W_{c,k} - W'_{c,k}) g_{c,k}. \tag{S28}$$

With $\Delta_c = W_{c,k} - W'_{c,k}$ and channel-gradient Gram matrix $Q_{cc'} = \langle g_{c,k}, g_{c',k} \rangle$,

$$\|G_k(W) - G_k(W')\|_2^2 = \frac{1}{M_k^2} \Delta^\top Q \Delta. \tag{S29}$$

Equal total weight removes differences in objective scaling while retaining channel composition. The updates differ whenever Δ is outside the null space of Q. The paired training and prediction interventions evaluate the forecasting effect of this change in composition.

B Experimental Protocol and Configuration Overview

Main-benchmark runs use the data preprocessing, forecasting loss, and checkpoint selection protocols of the corresponding backbone. The main table contains five backbones, nine multivariate datasets, and four horizons per dataset. Dedicated diagnostics use the protocols specified in their respective sections.

Each *profile* specifies a backbone–dataset pair and a horizon group. There are 45 backbone–dataset pairs, each with a short and a long profile, giving 90 profiles. Short profiles cover horizons 96 and 192; long profiles cover 336 and 720. For ILI, the corresponding pairs are 24/36 and 48/60. This count refers to horizon-group configurations, not separate baseline and cMoE entries.

ETTh1/ETTh2 use the first 12 months for training, the next four for validation, and the following four for testing; ETTm1/ETTm2 use the same temporal boundaries at 15-minute resolution. Electricity, Exchange, Solar, Weather, and ILI use chronological 70%/10%/20% splits. Each channel is standardized using training-split statistics. Validation and test windows may use preceding observations as input context, while their forecast targets remain within the respective split.

Routing transformations and schedules. The Scoreboard starts with all entries equal to one. For each channel, row-relative normalization subtracts its minimum score and divides by the absolute minimum, floored at 10^{-6}. The negative normalized scores are multiplied by the profile's score scale and divided by its temperature before softmax. For balancing, a copy of these weights alternates row normalization to one and column normalization to C/K for the configured number of iterations, then renormalizes its rows. The balance strength is the convex mixing weight between this copy and the original softmax weights. Once top-k_{top} is active, each row retains its largest k_{top} entries and renormalizes them. Temperature stays at its initial value during the configured warm-up epochs, then changes linearly to its final value over the annealing interval. Top-k_{top} has a separate activation epoch. The EMA records losses from the first batch; the 50-step warm-up counter controls cached hard assignments, not the start of soft Scoreboard updates.

Training procedure. The following steps specify the batch-level soft-routing path. The backbone initializer and configured expert perturbations initialize the expert pool and shared residual path; Adam updates their parameters.

1. Compute responsibility from the stored Scoreboard using the transformations above. Save these weights for the current batch; the Scoreboard and its weights are detached from gradient computation.
2. Evaluate every expert on every channel. Compute the profile's channel-wise training losses and MSE/MAE Scoreboard losses before adding the shared residual. Training losses use the configured L1 or L2 criterion and any horizon weights; Scoreboard losses average over the batch and forecast horizon. Form the routed forecast with the saved weights.

Table S3: Experimental configuration overview. Short and long horizon groups define 90 profiles across 45 backbone–dataset pairs. Ranges summarize the recorded configuration fields. The training procedure below defines the routing transformations.

Component	Configuration
Forecasting protocol	Lookback length 96; horizons $\{96, 192, 336, 720\}$, with $\{24, 36, 48, 60\}$ for ILI; MSE and MAE are reported.
Expert pool	Same-backbone experts with separate parameters; $K \in [2, 11]$ across formal commands.
Scoreboard	Full-information, batch-level loss audit; update schedule 1:1; 50 warm-up steps; momentum $\beta \in \{0.90, 0.95, 0.97\}$; routing score is MSE or MAE according to the frozen profile.
Score transformation	Row-relative normalization with $\epsilon = 10^{-6}$; score scale 0.35–2.0; softmax temperature schedule $\tau_{\text{start}} \in \{0.5, 5.0\}$ and $\tau_{\text{end}} \in \{0.05, 0.20, 0.55, 0.80, 1.0, 1.2, 1.5\}$.
Soft routing	Soft top-k_{top} with $k_{\text{top}} \in \{2, 3, 4\}$, where k_{top} is the number of retained experts per channel; activation after 1–8 warm-up epochs; 0 or 8 balancing iterations with blend strength in $[0, 1]$; annealing lasts 4–74 epochs.
Expert training	Routed responsibility with expert-mass normalization; the shared residual path is enabled and trained jointly with the routed experts.
Optimization profile	Adam; learning rate 2.68×10^{-5}–6.25×10^{-2}; batch size 8–256; epoch cap 1–100; early-stopping patience 1–28; the backbone's prescribed learning-rate schedule is retained.
Checkpoint and reporting	Checkpoints minimize the profile's validation forecasting loss (L1 or L2, including any horizon weights). Main Table 1 reports three-run means; component ablations and paired interventions report their own matched-run means.

3. For each expert with positive total responsibility, divide its weighted channel-loss sum by that total, floored at 10^{-6}. Sum the expert objectives. If experts have zero total responsibility, add their unweighted mean loss over the inactive experts, channels, batch, and horizon.
4. Form the final forecast by adding the scaled shared residual to a detached copy of the routed forecast. Its auxiliary forecasting loss, scaled by the configured shared-loss coefficient, updates only the shared path and its learned scale; the routed experts receive the gradients of their own channel-weighted objectives.
5. Update the Scoreboard by EMA from the saved, pre-update expert predictions without gradients, and take the Adam step on the combined objectives. The new Scoreboard weights apply to the next batch.
6. Select the checkpoint by the configured validation forecasting loss. At inference, retain its expert parameters and responsibility; do not update the Scoreboard from evaluation targets.

For the main benchmark, validation selects the checkpoint, and the test split is evaluated once for each finalized run. Each record includes the command, seed, GPU, checkpoint, validation epoch, test metrics, and source hash. Every cell in the 180-setting performance matrix reports the mean of three finalized runs for the same backbone–dataset–horizon setting. The benchmark comparisons in Tables S7 and S15 use these Main Table 1 values. Component ablations and paired interventions report the means of their matched runs.

C Capacity and Efficiency Audit

The single backbone is the deployment reference, while the uniform-responsibility control has the same expert pool as cMoE and isolates the additional cost of writing and using the Scoreboard. Each profile uses the configured command for its setting, one warm-up epoch, and two timed epochs; the three profiling repeats use a rotated execution order on otherwise isolated GPUs.

Table S4 reports parameter count, timed training cost, reserved training memory, and model-evaluation time as ratios. The final column isolates the additional training cost of writing and using the

Table S4: Capacity and efficiency audit on three representative settings. Ratios compare cMoE with the single-backbone reference, except the final column, which compares cMoE with the same expert pool using uniform responsibility. Values are medians over three profiling repeats; all measurements use the current implementation and no test metric is used for this audit.

Setting	K	Params ×	Train ×	Train mem. ×	Eval ×	Audit train ×
DLinear / Weather-720	9	10.00	2.77	5.68	1.05	1.18
FITS / Electricity-96	7	8.00	3.64	2.95	1.37	1.21
iTransformer / Weather-96	5	6.00	3.52	4.42	1.68	1.10

Table S5: Resource scaling with the routed expert count on DLinear / Weather-720. The shared residual path and all other settings are fixed. Timings use one warm-up epoch followed by two measured epochs.

Routed experts	Parameters	Train memory (MiB)	Epoch time (s)
1	0.279M	92	8.11
3	0.559M	178	12.23
5	0.838M	234	13.26
7	1.117M	334	14.79
9	1.397M	386	17.83

Scoreboard after the expert pool has already been introduced. Inference remains dense: every expert is evaluated, and the learned responsibility is frozen after training.

Table S5 gives the corresponding resource scaling as the routed expert count changes. This pilot measures capacity and resource growth only; its short training budget serves as a resource-scaling diagnostic alongside the main 180-setting results.

D Input-Ambiguity Diagnostic

The following controlled diagnostic gives two channels identical normalized routing inputs but different future trajectories. It illustrates how channel-indexed outcome history can distinguish channels that the current routing input cannot distinguish.

D.1 Controlled construction

We call two channels *input-ambiguous* when their normalized lookback windows are identical, although their future trajectories differ. For a deterministic Input Gate g, $x_a = x_b$ implies $g(x_a) = g(x_b)$, so both channels receive the same responsibility.

We construct two triangle-wave regimes with periods 64 and 256 and unit slopes $+1$ and -1. Raw channel offsets vary, but per-channel normalization makes every 12-step routing input identical. The two regimes then produce different 48-step futures.

Input Gate and Scoreboard use the same two-expert pool, optimizer, learning rate, and 500-epoch budget. Channel order is randomized. Warm-up assignments are balanced and independent of the regime, and each outcome updates only the next epoch's Scoreboard.

Figure S1 shows the complete construction. The Input Gate must assign the same weights to both regimes and learn a shared compromise. The channel-indexed Scoreboard recovers separate responsibilities and the corresponding specialist forecasts.

Table S6: Three-seed results for the input-ambiguity construction. Specialist accuracy is permutation-invariant: expert indices may exchange without changing the recovered partition.

Seed	Input Gate MSE	Scoreboard MSE	Input specialist acc.	Scoreboard specialist acc.
2023	32.5315	0.000163	0.50	1.00
2024	32.5315	0.000094	0.50	1.00
2025	32.5315	0.000152	0.50	1.00

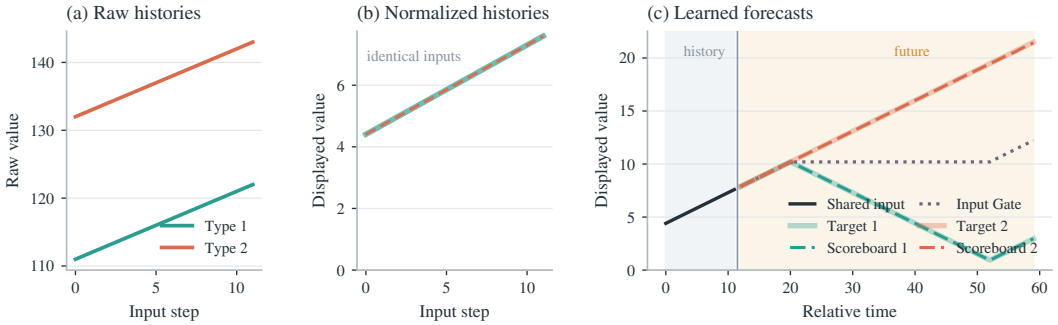


Figure S1: A trained input-ambiguity example. (a) Raw histories have different offsets. (b) Per-channel normalization produces identical routing inputs. (c) Input Gate therefore produces one shared forecast, whereas the future-safe Scoreboard recovers two responsibilities. A common display offset is used in panels (b–c); training and evaluation use the unshifted normalized values.

Here, permutation-invariant specialist accuracy is the largest assignment accuracy over permutations of expert labels. Across all three seeds, the maximum difference between Input Gate weights for the two regimes is exactly zero, while the Scoreboard recovers the complete two-regime partition.

The prospective reliability audit in Section A.5 uses permuted-channel assignment as a negative control. It preserves the learned weight values and the total weight assigned to every expert while breaking the channel–expert correspondence. A positive gap over this control therefore indicates channel-specific outcome information and separates it from global expert strength.

E Alternative Responsibility Controls

Main Table 2 tests historical smoothing and the use of responsibility during training. The controls here compare ways of obtaining channel weights or incorporating expert losses.

The static-gate control learns one free logit for every channel–expert pair. The outcome-supervised control trains the same logits toward a delayed target derived from the historical expert-loss matrix. Neither control directly reuses that matrix as the responsibility applied to prediction and expert updates.

The ELI-style control combines forecast loss with mean expert-specific loss while using an input gate, with coefficients 0.25 and 0.75, respectively. All controls inherit the corresponding main-table backbone, expert count, shared residual, data split, optimizer, learning rate, batch size, epoch limit, early-stopping rule, temperature schedule, balancing, and soft top-k_{top} settings. Only the responsibility mechanism and its associated loss change; checkpoints minimize the profile's validation forecasting loss.

The panel covers three backbones on ETTm2-96, Electricity-96, and Weather-96. cMoE uses the three-run means in Main Table 1; controls report three-run means and standard deviations. cMoE has a lower reported mean MSE than both learned-gate controls in all nine settings and than ELI-style in seven. ELI-style has a lower mean MSE and MAE on DLinear/Electricity-96 and FITS/Electricity-96. The reported precision does not establish a strict MAE ordering between cMoE and the static gate on FITS/Weather-96.

Table S7: Direct responsibility controls on nine 96-step settings. cMoE uses the three-run benchmark means in Main Table 1; controls report mean ± standard deviation over seeds 2023, 2024, and 2025. Lower is better.

Setting	Method	MSE ↓	MAE ↓
DLinear / ETTm2-96	cMoE (main table)	**0.183**	**0.274**
	Static gate	0.190856 ± 0.003456	0.286790 ± 0.004083
	Outcome-supervised gate	0.189520 ± 0.000256	0.285386 ± 0.001229
	ELI-style	0.183792 ± 0.003007	0.276655 ± 0.007313
DLinear / Electricity-96	cMoE (main table)	0.189	0.273
	Static gate	0.192997 ± 0.000376	0.275583 ± 0.001152
	Outcome-supervised gate	0.192022 ± 0.000197	0.275182 ± 0.000544
	ELI-style	**0.186178 ± 0.000109**	**0.270585 ± 0.000418**
DLinear / Weather-96	cMoE (main table)	**0.161**	**0.227**
	Static gate	0.187276 ± 0.004721	0.252584 ± 0.010371
	Outcome-supervised gate	0.180833 ± 0.000108	0.248706 ± 0.000389
	ELI-style	0.191011 ± 0.002109	0.256452 ± 0.003885
FITS / ETTm2-96	cMoE (main table)	**0.177**	**0.258**
	Static gate	0.181266 ± 0.001164	0.264327 ± 0.001466
	Outcome-supervised gate	0.180135 ± 0.000308	0.262386 ± 0.000391
	ELI-style	0.179371 ± 0.000453	0.261651 ± 0.001013
FITS / Electricity-96	cMoE (main table)	0.192	0.270
	Static gate	0.197491 ± 0.000036	0.274383 ± 0.000193
	Outcome-supervised gate	0.196928 ± 0.000102	0.274197 ± 0.000140
	ELI-style	**0.190538 ± 0.000318**	**0.269299 ± 0.000268**
FITS / Weather-96	cMoE (main table)	**0.181**	0.229
	Static gate	0.183939 ± 0.001619	0.229135 ± 0.000367
	Outcome-supervised gate	0.186805 ± 0.003910	0.231771 ± 0.001454
	ELI-style	0.195187 ± 0.000422	0.234767 ± 0.000231
iTransformer / ETTm2-96	cMoE (main table)	**0.179**	**0.261**
	Static gate	0.181204 ± 0.001595	0.264709 ± 0.004226
	Outcome-supervised gate	0.182070 ± 0.002143	0.265943 ± 0.003619
	ELI-style	0.181274 ± 0.001024	0.264817 ± 0.002741
iTransformer / Electricity-96	cMoE (main table)	**0.142**	**0.235**
	Static gate	0.146747 ± 0.001316	0.239095 ± 0.001349
	Outcome-supervised gate	0.145277 ± 0.000899	0.237265 ± 0.000898
	ELI-style	0.145101 ± 0.000407	0.236875 ± 0.000324
iTransformer / Weather-96	cMoE (main table)	**0.166**	**0.207**
	Static gate	0.171858 ± 0.003304	0.211771 ± 0.003007
	Outcome-supervised gate	0.171828 ± 0.003624	0.211693 ± 0.003501
	ELI-style	0.172034 ± 0.002197	0.211758 ± 0.002675

E.1 Channel Specificity Across Routing Mechanisms

In a separate prediction-intervention series, we evaluate three saved, validation-selected models per mechanism on the complete test split. The shared-start, frozen-responsibility branches used below serve a further intervention. We retain each model's expert predictions and shared residual, changing only the forecast weights. Channel-averaged weights preserve each expert's total prediction weight. For the input-dependent ELI-style gate, this average is computed separately within each input window, retaining its input-dependent expert totals. Test targets are not used to construct prediction weights; they enter only into the reported test-error and oracle diagnostics.

The cMoE rows in Table S8 use the three repeated runs of this prediction-intervention series. Its average MSE gains from channel specificity are 5.74%, 5.04%, and 1.85%. These are within-model reductions. Selection advantage uses each effective expert forecast, including the unchanged shared residual. It measures available competence, whereas the prediction intervention measures its use by

Table S8: Frozen prediction interventions on the alternative mechanisms. The two gains are relative test-error reductions from channel-specific rather than channel-averaged weights. Selection advantage compares the best expert per channel with the best shared expert and uses test labels only as an oracle diagnostic. All columns report three-run mean ± standard deviation (%).

Method	MSE gain	MAE gain	Selection advantage
DLinear / Weather-720			
cMoE	5.74 ± 0.31	5.40 ± 0.17	7.77 ± 0.66
Static gate	1.93 ± 0.55	1.63 ± 0.49	5.99 ± 1.35
Outcome-supervised gate	4.64 ± 0.59	4.62 ± 0.36	8.30 ± 0.25
ELI-style	0.92 ± 0.08	0.63 ± 0.16	1.41 ± 0.34
FITS / Electricity-96			
cMoE	5.04 ± 0.27	2.59 ± 0.07	5.80 ± 0.39
Static gate	0.83 ± 0.06	0.50 ± 0.05	2.18 ± 0.12
Outcome-supervised gate	1.01 ± 0.05	0.62 ± 0.02	2.47 ± 0.04
ELI-style	2.90 ± 0.21	1.51 ± 0.13	0.96 ± 0.08
iTransformer / Weather-96			
cMoE	1.85 ± 0.63	0.91 ± 0.35	7.58 ± 0.47
Static gate	1.07 ± 1.85	0.85 ± 1.46	1.10 ± 0.86
Outcome-supervised gate	1.56 ± 2.70	1.04 ± 1.80	1.27 ± 1.33
ELI-style	0.19 ± 0.10	0.18 ± 0.03	1.37 ± 0.57

the model's own weights. The paired training intervention below tests how historical responsibility changes the usefulness of those weights.

E.2 Paired Training and Prediction Interventions

For the equal-channel paired-training experiment, we use three backbone–dataset settings: DLinear with Weather-720, FITS with Electricity-96, and iTransformer with Weather-96. The paired branches start from the same pre-activation checkpoint and use seeds 2023, 2024, and 2025. They share the expert pool, prediction route, optimization budget, and normalization by each expert's total responsibility. cMoE and equal-channel training differ only in the channel composition of the expert updates.

We use paired training runs to test whether training composition changes the value of channel-specific prediction. Each pair starts from the same epoch-4 model and Scoreboard checkpoint, retains identical historical prediction weights, and trains through epoch 10. Both branches reinitialize Adam and restart the random generators with the same run seed; they do not resume the optimizer or random generator states of the epoch-4 training process. The expert and shared-residual parameters continue to train. We evaluate each branch's existing validation-selected checkpoint and report the three runs per setting, with nine experts for DLinear, seven for FITS, and five for iTransformer.

Each saved pool is evaluated with two prediction choices. Channel-specific weights retain the original responsibility. Shared weights average it over channels and use that vector on every channel, preserving the total prediction weight assigned to each expert. Both choices keep the expert predictions and shared residual fixed. This gives four combinations of training composition and prediction weights. All test windows and channels are included, and no weights are fitted to test targets.

Table S9 reports the errors of the combined forecasts, not responsibility-weighted averages of expert losses. After equal-channel training, the mean relative MSE gain from channel-specific prediction is below 0.01% for DLinear and FITS, and -1.27% for iTransformer. After responsibility-weighted training, these gains become 1.02%, 1.77%, and 1.86%. The corresponding MAE gains become 1.13%, 1.12%, and 0.88%. Relative gains are computed within each run before averaging.

Within each branch, the MSE reduction is the shared-weight error minus the channel-specific error. We subtract the equal-channel branch's reduction from the responsibility-weighted branch's reduction, then average these paired differences over the three runs. The additional MSE reductions are 0.00360 ± 0.01431, 0.00352 ± 0.00102, and 0.00534 ± 0.00039 for DLinear, FITS, and iTransformer, respectively (mean ± sample standard deviation of the paired differences). The differences are positive in two of the three DLinear runs and all three FITS and iTransformer runs. This comparison measures

Table S9: Paired training and prediction interventions. Test MSE/MAE are mean ± sample standard deviation over three runs. Each pair uses its existing validation-selected checkpoints; both prediction choices are evaluated on the same saved pool.

Training	Forecast weights	MSE ↓	MAE ↓
DLinear / Weather-720			
Equal-channel	Shared	0.3500 ± 0.0043	0.3883 ± 0.0058
Equal-channel	Channel-specific	0.3500 ± 0.0043	0.3883 ± 0.0058
Responsibility-weighted	Shared	0.3518 ± 0.0016	0.3930 ± 0.0035
Responsibility-weighted	Channel-specific	0.3482 ± 0.0138	0.3885 ± 0.0097
FITS / Electricity-96			
Equal-channel	Shared	0.1987 ± 0.0001	0.2757 ± 0.0004
Equal-channel	Channel-specific	0.1987 ± 0.0001	0.2757 ± 0.0004
Responsibility-weighted	Shared	0.1995 ± 0.0003	0.2764 ± 0.0005
Responsibility-weighted	Channel-specific	0.1959 ± 0.0007	0.2733 ± 0.0003
iTransformer / Weather-96			
Equal-channel	Shared	0.1701 ± 0.0007	0.2095 ± 0.0004
Equal-channel	Channel-specific	0.1722 ± 0.0006	0.2108 ± 0.0006
Responsibility-weighted	Shared	0.1713 ± 0.0002	0.2101 ± 0.0006
Responsibility-weighted	Channel-specific	0.1681 ± 0.0008	0.2082 ± 0.0013

how training responsibility changes the average value of the fixed forecast weights in the trained system. Table S9 reports the four forecast errors and their variation; Main Figure 4 tests expert identities on a fixed pool.

Prediction reconstruction and direct inference agree to floating-point precision. The reevaluated native MSE and MAE reproduce the saved test results; checkpoint parameters and Scoreboard states remain unchanged throughout the intervention.

E.3 Training History and Channel-specific Combination

We examine whether training history changes the value of channel-specific forecast combination. For each of the three ablation settings, we freeze the equal-channel, instant-outcome, and historical-responsibility models from three matched runs, retaining each model's shared residual. Shared combination fits one expert-weight vector for all channels; channel-specific combination fits a separate vector for each channel. Both minimize combined-forecast MSE over all validation windows, with nonnegative weights summing to one and all experts available. Weights are fixed before test evaluation. The convex quadratic fits use active-set enumeration; the maximum relative duality gap is 2.1×10^{-15}. This gap bounds the distance from the fitted validation objective to its optimum. These optimized shared weights differ from the channel-averaged historical weights used in Table S9.

Table S10 separates the Main Table 1 benchmark means from the frozen-model diagnostics. With channel-specific fitted weights, historical training reduces mean MSE relative to instant-outcome training by 1.35%, 0.37%, and 2.83% on DLinear, FITS, and iTransformer. Moving from shared to channel-specific combination reduces mean MSE by 8.50%, 3.65%, and 1.33% after instant-outcome training, versus 8.72%, 4.28%, and 4.64% after historical training. These percentages use the ratio of the reported means. Historical training therefore increases the mean benefit of channel-specific combination in these settings. Relative to fitted equal-channel models, fitted historical models have lower errors on FITS and iTransformer, while the equal-channel model remains better on DLinear.

We further isolate memory through a shared-start intervention on ETTm2-96, using DLinear with two experts and iTransformer with four. For each of the seeds 2023, 2024, and 2025, both branches start from the same epoch-4 state, including expert and shared parameters, the Scoreboard, Adam state, and random and data-loader states. Historical training retains $\beta = 0.9$; instant-outcome training changes only β to zero from epoch 5. Both continue to the fixed epoch-10 endpoint with identical batches, learning-rate and temperature schedules, and per-setting soft top-k configuration. We then fit shared and channel-specific prediction weights using the procedure above; all twelve models' fits are saved before test evaluation. This intervention measures the effect of memory on subsequent training of the complete system, including its shared residual.

Table S10: Validation-fitted forecast combination on frozen ablation models. Benchmark rows use the original cMoE weights and Main Table 1 three-run means. Other rows report test means ± sample standard deviations over three runs, with shared or channel-specific weights fitted on validation data.

	Shared	Channel-specific	
Training	MSE	MSE	MAE
DLinear / Weather-720			
cMoE benchmark	—	0.339	0.380
Equal-channel	0.353829 ± 0.001362	0.353810 ± 0.001355	0.393821 ± 0.001130
Instant outcome	0.395480 ± 0.003277	0.361881 ± 0.004183	0.412427 ± 0.008866
Historical responsibility	0.391112 ± 0.002273	0.356994 ± 0.002778	0.409054 ± 0.003249
FITS / Electricity-96			
cMoE benchmark	—	0.192	0.270
Equal-channel	0.198600 ± 0.000012	0.198598 ± 0.000012	0.275714 ± 0.000063
Instant outcome	0.199216 ± 0.000145	0.191940 ± 0.000043	0.270320 ± 0.000356
Historical responsibility	0.199775 ± 0.000200	0.191228 ± 0.000059	0.270185 ± 0.000254
iTransformer / Weather-96			
cMoE benchmark	—	0.166	0.207
Equal-channel	0.173013 ± 0.001921	0.172887 ± 0.002219	0.212486 ± 0.003008
Instant outcome	0.172351 ± 0.002673	0.170067 ± 0.004319	0.210318 ± 0.003975
Historical responsibility	0.173297 ± 0.000925	0.165253 ± 0.000187	0.207048 ± 0.001457

Table S11: Shared-start memory intervention on ETTm2-96. Branches share the epoch-4 state and differ only in subsequent Scoreboard memory. Both are evaluated at epoch 10 after validation fitting of forecast weights. Entries are test means ± sample standard deviations over three paired runs.

	Shared	Channel-specific	
Training after epoch 4	MSE	MSE	MAE
DLinear / ETTm2-96			
Instant outcome	0.197841 ± 0.002487	0.197841 ± 0.002487	0.297387 ± 0.002932
Historical responsibility	0.197841 ± 0.002487	0.197841 ± 0.002487	0.297387 ± 0.002932
iTransformer / ETTm2-96			
Instant outcome	0.180508 ± 0.000225	0.179881 ± 0.000331	0.262422 ± 0.000210
Historical responsibility	0.180301 ± 0.000369	0.178098 ± 0.000431	0.259264 ± 0.000335

Table S11 shows that historical training lowers iTransformer's mean MSE by 0.99% and MAE by 1.20% under channel-specific fitted combination, with the same direction in all three runs. Its MSE benefit from channel-specific rather than shared combination exceeds the instant-outcome branch's benefit by 0.001577 ± 0.000137 across runs. DLinear's two branches are nearly tied: the mean channel-specific MSE difference is 5.8×10^{-10}, and the historical branch's channel weights are $0.5/0.5$ at epoch 10. The paired results identify a memory effect on the value of channel-specific combination in the attention setting, alongside the near-uniform linear setting.

F Additional Specialist Diagnostics

This section provides component-ablation statistics, update and competence diagnostics, and the protocols for Main Figures 3–5. The equal-channel paired-training and forecast-weight intervention is specified in Section E.2.

Component ablation diagnostics. Main Table 2 averages 36 fresh-training runs: four variants on three settings, each with seeds 2023, 2024, and 2025. Variants share the per-setting configuration and seed, changing the responsibility mechanism specified in the main text. They are trained from initialization, with epoch caps of 30, 80, and 50 for DLinear/Weather-720, FITS/Electricity-96, and iTransformer/Weather-96, respectively, and validation-based early stopping. These full-budget ablations and the shared-checkpoint interventions in Section E.2 are separate experiments.

Table S12: Component ablation diagnostics: growth in channel-wise expert-choice advantage (Spec., percentage points) and weighted expert-loss reduction (Alloc., %). Higher is better.

Variant	DLinear / W-720		FITS / ECL-96		iTransformer / W-96		Average	
	Spec. gain ↑	Alloc. gain ↑	Spec. gain ↑	Alloc. gain ↑	Spec. gain ↑	Alloc. gain ↑	Spec. gain ↑	Alloc. gain ↑
Uniform responsibility	-0.06	-0.00	-0.32	-0.00	0.82	-0.00	0.15	-0.00
Instant outcome ($\beta = 0$)	12.05	9.57	5.41	5.92	3.33	2.87	6.93	6.12
Forecast-only responsibility	-0.05	-0.01	-0.32	0.00	0.90	0.51	0.18	0.17
Full cMoE	**12.72**	**9.77**	**6.12**	**8.19**	**9.21**	**16.62**	**9.35**	**11.53**

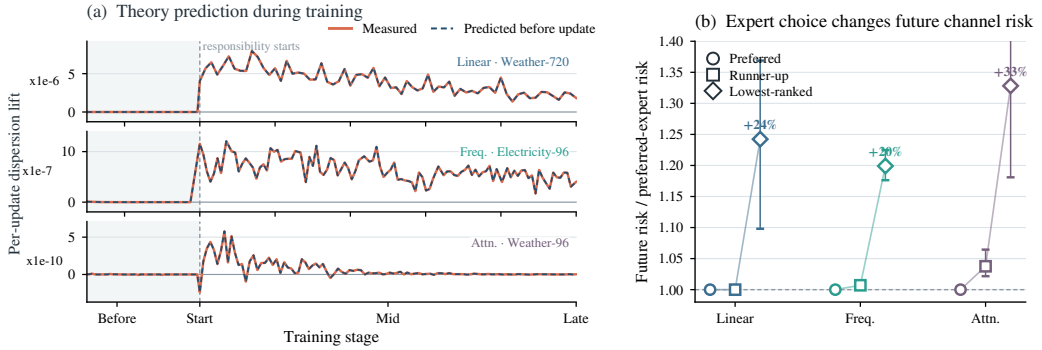


Figure S2: Update-level and future-ranking diagnostics. (a) Pre-update specialization pressure and measured per-update dispersion lift. (b) Individual-expert validation losses after ranking experts for each channel by the frozen Scoreboard. The ranking uses stored outcome scores.

Table S14 reports the corresponding means and sample standard deviations, recomputed from the saved per-run test results.

Table S12 complements the final forecasting errors in main Table 2 without repeating those errors. Specialization Gain is the growth in the advantage of selecting the best expert separately for each channel instead of one global expert. Allocation Gain is the reduction in responsibility-weighted expert loss relative to equal-expert assignment. The dataset abbreviations W and ECL denote Weather and Electricity.

Update trajectories and future expert ranking. Figure S2 shows the update-level and future-ranking diagnostics. The pressure comparison concerns virtual one-step parameter updates; the ranking comparison concerns individual-expert validation losses, while the final mixture is evaluated separately.

The lift is small before responsibility activation and becomes positive afterwards in the displayed trajectories. Its later decline concerns the effect of individual updates; accumulated competence is evaluated by the endpoint diagnostics. The exact identity and its finite-step correction are given in Section A.

In the separate frozen-expert ranking diagnostic, the runner-up is nearly tied with the preferred expert on DLinear and FITS and has 3.8% higher loss on iTransformer. The lowest-ranked expert has 24%, 20%, and 33% higher loss, respectively. These results show that several experts can remain competent for the same channel.

Common-mean gradient probe. The local-specialization diagnostic uses DLinear/Weather-720, FITS/Electricity-96, and iTransformer/Weather-96 with seeds 2023, 2024, and 2025. For each seed, five non-overlapping loader batch pairs at offsets $0, 2, 4, 6, 8$ are evaluated and averaged before seed-level analysis. The initialization phase instantiates the official model and performs no parameter or Scoreboard update. The checkpoint phase loads the epoch-4 pre-activation state. In both phases, the probe saves the expert differences, temporarily replaces every expert by their parameter mean,

Table S13: Paired-training endpoint selection advantage (%); full precision for values below 0.01.

Setting	Seed	Equal-channel	Responsibility-weighted
DLinear / Weather-720	2023	0.000001	3.841
	2024	0.000014	4.124
	2025	0.000051	6.553
FITS / Electricity-96	2023	0.005501	3.557
	2024	0.001321	4.091
	2025	0.002998	3.511
iTransformer / Weather-96	2023	0.231573	5.842
	2024	2.754717	4.881
	2025	1.183603	5.765

computes per-channel gradients on two successive optimization batches, projects those gradients onto the saved differences, and restores the state exactly. The reported feedback cosine is the normalized cross-batch agreement of these centered projections. FITS complex parameters use their equivalent real coordinates and the real Hermitian inner product.

Main Figure 4 identity-reassignment protocol. For MixLinear/Solar-96, FITS/Weather-96, and iTransformer/Electricity-96, each channel keeps its learned soft responsibility values and the number of active experts. We enumerate feasible permutations of those responsibility values across expert identities and evaluate the resulting mixtures on the held-out validation windows. The future-best, learned, median, and worst assignments are the minimum, recorded, median, and maximum losses in this finite assignment set. This is a post-hoc test of identity informativeness; routing uses historical outcome scores.

Main Figure 3 panel protocols. Panel (a) uses the epoch-10 DLinear/Weather-720 responsibility-weighted branch with seed 2023 and nine experts, preserving the original channel and expert order. For channel c and expert k, let $L_{c,k}$ be the validation MSE. The plotted gap is $100\left(\frac{L_{c,k}}{\min_j L_{c,j}} - 1\right)$, where j ranges over experts; the color scale saturates at 100%, and dots mark the lowest-error expert in each channel.

Panel (b) uses frozen seed 2023 validation exports for MixLinear on Solar-96, FITS on Weather-96, and iTransformer on Electricity-96. Each channel is counted for its lowest-MSE expert, and the bars report these counts as percentages of all channels. An expert with no lowest-error channel receives a zero count. This winner count is a competence diagnostic, not the soft responsibility used for training or prediction.

Panel (c) uses separate saved Weather responsibility maps: DLinear at horizon 720 with six experts, and FITS and iTransformer at horizon 96 with five experts each. Rows are experts and columns are the 21 Weather channels; weights are normalized across experts within each channel, and larger circles indicate larger weights. These maps describe soft allocations at their own saved states rather than the paired-training contrast.

Paired-training endpoint competence. The matched branches start from the same epoch-4 checkpoint and keep historical prediction weights fixed through epoch 10. For each endpoint validation loss matrix L, selection advantage is $100[1 - \mathrm{mean}_c \min_k L_{c,k} / \min_k \mathrm{mean}_c L_{c,k}]$. This compares channel-wise expert choice with the best single expert shared by all channels. Table S13 reports each run separately from the channel counts in Main Figure 3(b).

Channel-level competence detail. Figure S3 reports fixed-epoch expert competence; Table S9 reports the paired branches' combined-forecast errors.

Figure S3(a) uses the matched training branches at epochs 5, 7, and 10. Within each branch, the shared best expert minimizes mean validation MSE over all channels. Each channel's expert-choice gain is its relative MSE reduction from that expert to its own lowest-error expert. Gains are averaged across the three matched runs before channels are counted in the descriptive intervals $[0, 1]$, $(1, 5]$,

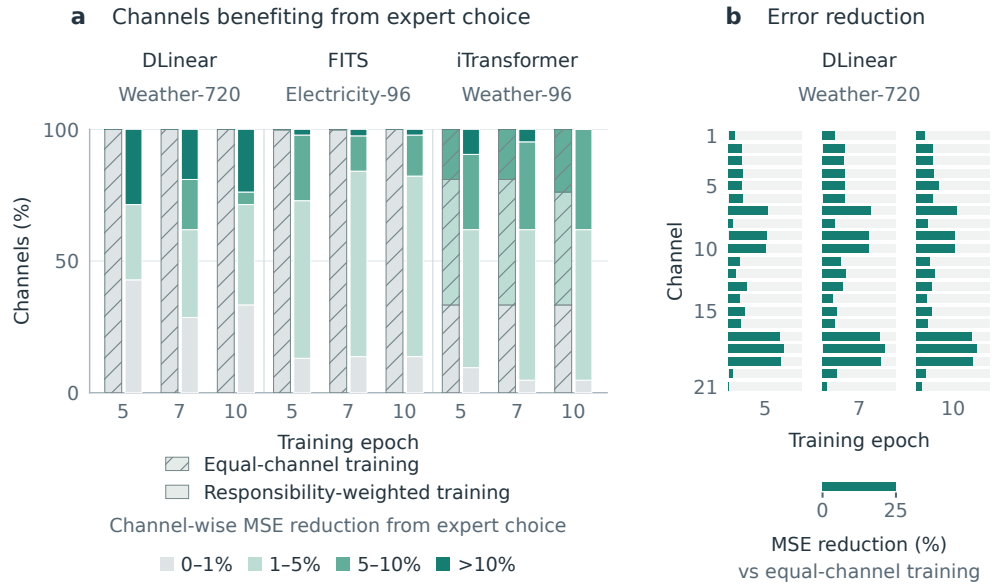


Figure S3: Channel-level competence during paired training. (a) Gains from channel-wise expert choice within each branch; hatching denotes equal-channel training. (b) Reduction in the lowest expert MSE per channel after responsibility-weighted training on DLinear/Weather-720. These panels evaluate individual-expert validation losses.

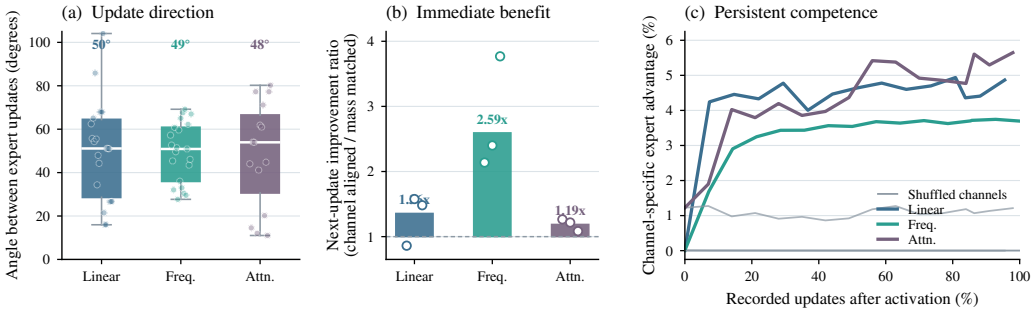


Figure S4: Update and training-trajectory diagnostics on DLinear/Weather-720, FITS/Electricity-96, and iTransformer/Weather-96. (a) Expert-gradient angles between channel-aligned and mass-matched updates. (b) Next-update specialist improvement relative to a matched-mass channel-shuffled control. (c) Recorded specialist-advantage trajectories for responsibility-weighted and matched-mass channel-shuffled training.

$(5, 10]$, and $(10, 100)$ percent. Every column includes every channel. This panel is a descriptive choice diagnostic; branch-level forecasting comparisons are reported separately.

Figure S3(b) compares the lowest expert MSE of the two branches for each channel. Relative reductions are computed within each paired run and then averaged. The DLinear/Weather-720 illustration retains all 21 channels in their original order across the same three epochs. Other settings also include negative reductions. At epoch 10, the numbers of channels with positive mean relative reductions are 21/21, 312/321, and 14/21 for DLinear/Weather, FITS/Electricity, and iTransformer/Weather, respectively. These counts are descriptive, not independent-run significance tests.

Update and long-window branch diagnostics. Figure S4(a) reports mean expert-gradient angles of $50°$, $49°$, and $48°$. In panel (b), next-update specialist improvements are $1.36\times$, $2.59\times$, and $1.19\times$ those of the matched-mass channel-shuffled control. Panel (c) ends with specialist-advantage gaps of

Table S14: Run-level variation for the component ablation in Main Table 2. Entries are mean ± sample standard deviation over seeds 2023, 2024, and 2025, using the same 36 runs as the main ablation.

Variant	MSE	MAE
DLinear / Weather-720		
Uniform responsibility	0.353818 ± 0.001361	0.393820 ± 0.001137
Instant outcome ($\beta = 0$)	0.343319 ± 0.006326	0.386442 ± 0.006901
Forecast-only responsibility	0.353848 ± 0.001374	0.393866 ± 0.001197
Full cMoE	0.340258 ± 0.003240	0.382197 ± 0.003999
FITS / Electricity-96		
Uniform responsibility	0.198604 ± 0.000014	0.275722 ± 0.000062
Instant outcome ($\beta = 0$)	0.193534 ± 0.000214	0.271656 ± 0.000472
Forecast-only responsibility	0.198602 ± 0.000013	0.275718 ± 0.000062
Full cMoE	0.192061 ± 0.000325	0.270447 ± 0.000416
iTransformer / Weather-96		
Uniform responsibility	0.174086 ± 0.002195	0.213993 ± 0.002744
Instant outcome ($\beta = 0$)	0.171481 ± 0.004331	0.211551 ± 0.004518
Forecast-only responsibility	0.173999 ± 0.002066	0.213934 ± 0.002661
Full cMoE	0.167957 ± 0.001350	0.209108 ± 0.002263

Table S15: External positioning against the three-run cMoE benchmark means in Main Table 1. External runs use sequence length 96 and seed 2023. The cMoE backbone is DLinear for the MoLE-DLinear comparison and OLinear for Pathformer.

External	Dataset	Horizon	Ext. MSE	cMoE MSE	Ext. MAE	cMoE MAE
MoLE-DLinear	ETTm2	96	0.200	0.183	0.291	0.274
MoLE-DLinear	ETTm2	720	0.850	0.433	0.569	0.438
MoLE-DLinear	Weather	336	0.340	0.257	0.383	0.320
Pathformer	ETTm2	96	0.171	0.168	0.251	0.246
Pathformer	ETTm2	720	0.393	0.388	0.390	0.386
Pathformer	Weather	336	0.259	0.254	0.282	0.276

4.62, 3.57, and 5.33 percentage points. These trajectories correspond to Table S1; Table S13 reports the separate epoch-10 equal-channel comparison.

Channel-structure protocol. Main Figure 5 uses DLinear on Weather-720 and iTransformer on Electricity-720. Weather channels are grouped by their common active expert set, using a weight threshold of 10^{-6}; Electricity channels are grouped by their highest-weight expert. The correlation matrices use training-split observations. Within-group means exclude diagonal entries, so a singleton has no pairwise mean. Every channel is retained, and Electricity indices indicate reordered positions counted from the bottom. Physical labels and correlations are used only for the displayed analysis.

The Weather strips denote temperature (Temp), moisture (Moist), radiation (Rad), air density (Dens.), wind, precipitation (Prec), pressure (Press.), and carbon dioxide (CO_2). These physical categories organize labels within the displayed responsibility groups; they are not provided during training.

G External Positioning Panel

Table S15 provides a descriptive positioning reference. External methods use a sequence length of 96 and a seed of 2023; cMoE values are three-run benchmark means. With only one external run per setting, this panel does not estimate variability across external runs.