ADAPT THE DATA, NOT THE MODEL: INPUT-SPACE ADAPTATION FOR FROZEN TIME-SERIES PREDICTORS

Anonymous authors
Paper under double-blind review

ABSTRACT

Conventional unsupervised domain adaptation (UDA) typically relies on updating the predictor's weights to optimize performance on the target domain using a labeled source dataset and unlabeled target data. However, a growing demand for secure deployments, such as closed inference APIs, firmware, or regulatory certification, means models are increasingly released with locked weights. We study domain adaptation in this locked-predictor regime for multivariate time-series models, where the deploying site holds its own labels but source training data may be inaccessible. We introduce PILOT (**P**er-timestep **I**nput-space adapter for **LO**cked predic**T**ors), a per-timestep input-space adapter trained against the frozen predictor by back-propagating the task loss using target labels. The framework supports two settings: a source-free variant for strict privacy constraints and a source-assisted variant when data sharing is permitted. We evaluate PILOT across five clinical time-series tasks (encompassing two domain shifts) and five AdaTime sensor benchmarks using a frozen 1D-CNN backbone. PILOT achieves state-of-the-art performance across all comparable frozen-backbone, end-to-end, and test-time adaptation baselines. It yields a $+15.3$ improvement in AUCPR for AKI, increases the mean Macro-F1 score by $+10.1$ over the leading test-time adaptation method on AdaTime and matches or exceeds the performance of natively trained target-domain models. Furthermore, training PILOT on a single architecture enables zero-shot transfer to other frozen predictors, allowing the same PILOT module to be reused across different architectures.

1 INTRODUCTION

When a clinical or industrial monitoring system runs in production on multivariate time-series inputs, two constraints often forbid retraining its predictor. First, the predictor is locked: medical devices regulated as Software-as-a-Medical-Device (SaMD) under the U.S. FDA framework, embedded sensors with firmware, and closed-API scoring services all expose only inference, and any modification to the model weights triggers a fresh approval cycle (U.S. Food and Drug Administration, 2021; 2024; Almarie et al., 2025). Second, the data the predictor was trained on may be inaccessible to the institution that runs it: vendor-shipped medical AI products keep their training corpora proprietary (Abràmoff et al., 2018); cross-hospital and cross-jurisdiction deployments are typically governed by privacy law and data-use agreements that do not let raw patient records leave their origin (Peloquin et al., 2020; El Emam et al., 2015); and large-scale clinical-data networks such as FDA Sentinel, OHDSI, NIH N3C, and EHDEN are architected from the outset so that patient-level records never leave the contributing site (Platt et al., 2018; Hripcsak et al., 2015; Haendel et al., 2021; Blacketer et al., 2025). Predictors deployed under these constraints degrade silently under distribution shift: across the five YAIB tasks (van de Water et al., 2024), MIMIC-IV-trained LSTMs lose up to 14.5 AUROC points on HiRID acute kidney injury and approximately 11 hours of mean-absolute-error (MAE) on HiRID length-of-stay, a pattern documented across both clinical and non-clinical AI deployments (Finlayson et al., 2021; Wong et al., 2021; Koh et al., 2021; Recht et al., 2019). The deploying institution typically holds its own retrospective labels, so the adapter is supervised by target labels and runs in either regime: with the source training data when policy permits its use, or without any source sample when it does not. Domain adaptation under these constraints, with a

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.

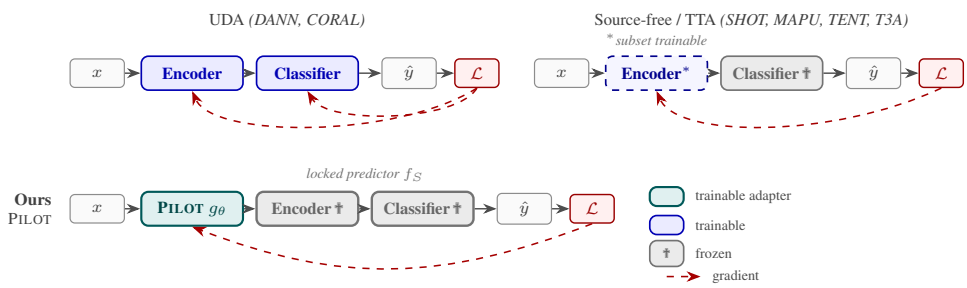

Figure 1: Prior UDA and source-free / test-time adaptation (top) update the predictor's weights or a subset of them. Our setting (bottom) freezes every predictor parameter; the task gradient flows through the locked predictor into an upstream PILOT adapter.

locked predictor, available target labels, and source data that may be entirely unavailable, sits between supervised domain adaptation, unsupervised domain adaptation (Ganin et al., 2016; Sun & Saenko, 2016), test-time adaptation (Wang et al., 2021; Niu et al., 2023), and source-free DA (Liang et al., 2020; Ragab et al., 2023b), and is largely unaddressed in published work. Classical UDA aligns model parameters to a target distribution, but locked weights take that mechanism off the table. TTA adapts a small parameter subset (BatchNorm affine, classifier head) at inference, which still mutates the certified model. Input-space adaptation precedents from vision (PixelDA (Bousmalis et al., 2017)) and clinical EHR (TATO (Qiu et al., 2026)) treat per-stay or per-cohort transformations of the inputs, but neither targets the per-timestep, retrieval-conditioned regime we address. Figure 1 contrasts where the task gradient lands across these families: prior methods write into the predictor's weights or a subset, while our setting freezes every predictor parameter and routes the gradient through the locked predictor into an upstream input adapter.

We align the data to the model instead. Our adapter, PILOT, is an instance-conditional input transformation supervised by the task loss back-propagated through the frozen predictor; an optional retrieval module fetches k source exemplars per timestep, fused via cross-attention (k-NN). In the adapter's source-free variant, the deploying site needs no source-domain training data at adapt time, requiring only retrospective target labels and inference access to the certified predictor; retraining requires write access to the certified weights, the original source training data, and target labels, so our prerequisites are a strict subset of what retraining demands. Source = the labelled distribution on which the predictor was trained (MIMIC-IV for our clinical results, the AdaTime source-domain for general time-series); target = the deployment distribution from which we hold retrospective labels (eICU or HiRID clinically; the AdaTime held-out domain otherwise). Evaluated against 18 comparable baselines spanning frozen-backbone DA, a per-feature affine simple-adapter, end-to-end fine-tuning, end-to-end DA, and test-time adaptation (Table 1; Appendix Table 19), our adapted predictions match or exceed the MIMIC-IV-LSTM-with-eICU-fine-tune baseline on all five tasks and reach or exceed the eICU-native LSTM on all five tasks. The framework extends input-space precedents from vision (Bousmalis et al., 2017) and time-series foundation models (Yang et al., 2021; Jin et al., 2024) to clinical and non-clinical per-timestep input adaptations, evaluated on EHR frozen predictors and on AdaTime's 1D-CNN. Furthermore, a single pretrained adapter transfers zero-shot to other frozen predictors across two target cohorts (Appendix D), so adapting one site to M tasks across N predictor families costs one adapter per task rather than one retrained predictor per (task, family) pair. Under the AdaTime protocol on five non-clinical benchmarks with a frozen 1D-CNN, the adapter beats thirteen end-to-end DA methods (mean $+2.6$ Macro-F1; Appendix C). A reduced-capacity adapter (Appendix E) matches the full adapter at predictor-matched parameter counts. **Our contributions are fourfold: Frozen predictors can be adapted through input-space learning, matching or exceeding end-to-end fine-tuning and natively trained models.** A target-labelled adapter trained by task-loss back-propagation through a strictly frozen predictor reshapes inputs, not weights, leaving regulatory certification of the predictor intact. To our knowledge, this is the first per-timestep input-space adapter for strictly frozen clinical time-series predictors.

Calibration improves as a structural property. The adapter reduces Brier on 9/9 classification cells and ECE in mean on 8/9 with no explicit calibration loss; post-hoc temperature scaling (Guo et al., 2017) composes on top with bit-exact AUROC and a further per-task ECE reduction (Appendix H).

The same pretrained adapter transfers across deployments, predictor families, and modalities. Phase 1 is pretrained once per dataset and reused across every task at that site; the adapter then transfers zero-shot to additional frozen predictor families, and a reduced-capacity adapter retains the gain at predictor-matched parameter counts.

The same adapter deploys with or without source training data. When source data is unavailable, removing every source-domain training signal leaves source-free adaptation that approaches or surpasses the source-assisted regime across all five clinical tasks (Section 4.2) and all AdaTime sensor benchmarks (Appendix K.2). Classical unsupervised domain adaptation requires labelled source data at adapt time; our adapter does not.

2 Related Work

Domain adaptation for time series. Feature-space DA aligns source and target distributions inside the predictor via adversarial discriminators (Ganin et al., 2016; Wilson et al., 2020), moment matching (Sun & Saenko, 2016; Gretton et al., 2012), or temporal-contrastive alignment (He et al., 2023; Ozyurt et al., 2023), and is benchmarked for time series by AdaTime (Ragab et al., 2023a). These methods retrain the feature extractor end-to-end with labelled source and unlabelled target. Unlike feature-space DA, our adapter leaves every predictor weight bit-identical to the certified backbone and adapts only the input stream.

Test-time and source-free adaptation. Test-time adaptation updates statistics or affine parameters at inference via entropy minimisation (Wang et al., 2021), sharpness-aware refinement (Niu et al., 2023), sample-efficient EATA (Niu et al., 2022), or template prototyping (Iwasawa & Matsuo, 2021); source-free methods such as SHOT (Liang et al., 2020) and time-series masked autoencoder pseudo-labelling (MAPU, Ragab et al., 2023b) retrain the feature extractor on pseudo-labels. The closest source-free time-series prior is MAPU, which adapts the feature extractor per-window in latent space. Unlike MAPU and BN-affine TTA, our adapter requires no norm-affine surface and leaves the deployed predictor bit-exact, rewriting the input stream at per-timestep granularity instead.

Parameter-efficient adaptation. Frozen-backbone PEFT injects trainable parameters inside or alongside the predictor: low-rank residuals (LoRA, Hu et al., 2022), bottleneck adapters (Houlsby et al., 2019; Pfeiffer et al., 2021), soft and visual prompts (Lester et al., 2021; Jia et al., 2022; Liu et al., 2022), distilled students (Hinton et al., 2015), and side-network ladders (Sung et al., 2022). Ladder Side-Tuning (Sung et al., 2022) keeps backbone weights bit-identical but composes a side network with backbone activations, still altering the deployed input-output map. Unlike PEFT, our adapter places its trainable parameters strictly upstream of the predictor, so the deployed forward pass remains a composition of the adapter and the unchanged frozen backbone.

Input-space adaptation and reprogramming. Input-space adaptation operates before the first layer of a frozen backbone: PixelDA re-renders images for a frozen classifier (Bousmalis et al., 2017), FDA swaps low-frequency Fourier components (Yang & Soatto, 2020), Voice2Series (Yang et al., 2021) and Time-LLM (Jin et al., 2024; Zhou et al., 2023) reprogram frozen acoustic and LLM backbones, and the reprogramming literature (Elsayed et al., 2019; Chen, 2024) unifies the framing. The closest contemporaneous work, TATO (Qiu et al., 2026), learns a global sequence-level transformation for foundation-model backbones. Unlike TATO, our adapter operates per-timestep and is supervised by the through-predictor task loss available at the deploying site, with instance-conditioning at every timestep rather than a global sequence-level transformation upstream of a foundation model.

Retrieval-augmented models. Retrieval augmentation has mostly been applied at the decoder or prediction head: kNN interpolation of a frozen language model with a datastore lookup (Khandelwal et al., 2020; 2021; Lewis et al., 2020; Borgeaud et al., 2022), with recent time-series and clinical variants keeping retrieval at the head and targeting LLM backbones (Ning et al., 2025; Xu et al., 2024; Cao et al., 2026). The closest priors are kNN-MT (Khandelwal et al., 2021), which retrieves at the output distribution, and EHR-RAG (Cao et al., 2026), which retrieves over textual records for an LLM. Unlike kNN-MT and EHR-RAG, we place k-NN retrieval with cross-attention fusion upstream of a strictly frozen predictor over a source-domain memory bank, so the same datastore conditions every downstream task that shares the input schema.

Cross-hospital EHR transfer. Cross-hospital EHR transfer is standardly evaluated on YAIB (van de Water et al., 2024) and its precursors (Purushotham et al., 2018; Harutyunyan et al., 2019); pretrained

EHR foundation models (Rasmy et al., 2021; Li et al., 2020; Burger et al., 2025) retrain large backbones on aggregated cohorts. Unlike foundation-model retraining, our adapter composes with any certified predictor without modifying its weights, so the cross-hospital transfer cost is paid once at the adapter and not at the backbone. Appendix Table 21 expands this placement into a nine-axis scorecard across twenty-seven methods; ours is the only row satisfying every axis.

3 METHOD

3.1 PROBLEM SETTING

Let $\mathcal{X}_S$ denote source-domain clinical time series (MIMIC-IV) drawn from P_S, with $\mathcal{D}_S$ the source training corpus, and $\mathcal{X}_T \sim P_T$ denote the target domain (eICU or HiRID). A clinical sample is the tuple

$$x \;=\; \big(\, X_{\text{val}} \in \mathbb{R}^{T\times F_{\text{dyn}}},\;\; X_{\text{miss}} \in \{0,1\}^{T\times F_{\text{dyn}}},\;\; x_{\text{static}} \in \mathbb{R}^{F_{\text{static}}},\;\; t \in \mathbb{R}^{T} \,\big), \tag{1}$$

where X_{val} holds the per-feature numerical values (forward-fill imputed and z-scored on the source training split) and X_{miss} is the per-feature observation-indicator channel (1 if the value at that step was observed pre-imputation, 0 otherwise). x_{static} is a vector of patient-level static covariates (age, sex, height, weight, F_{static}=4), and t is the timestamp index from which the inter-step delays $\Delta t_\tau = t_\tau - t_{\tau-1}$ are derived (with $\Delta t_1 = 0$). For all five EHR tasks F_{dyn}=48 harmonized clinical variables. AdaTime sensor benchmarks instantiate the same tuple with $X_{\text{miss}} \equiv 0$, $x_{\text{static}} \equiv 0$, $\Delta t_\tau \equiv \tau$ (positional index), recovering a generic time-series adapter as a special case. The same setup runs both with source data and, when source data is unavailable at adapt time, in a source-free regime (§4.2) in which no source sample contributes to the adapter's gradient. A predictor $f_S : \mathbb{R}^{T\times F_{\text{pred}}} \to \mathbb{R}^{T\times C}$, trained on $\mathcal{D}_S$, is *entirely frozen* with parameters θ_S satisfying $\nabla_{\theta_S} f_S = 0$ (per-task $F_{\text{pred}} \in \{52, 96, 100, 292\}$ values in Appendix B, Table 8). We learn an adapter $g_\theta : (X_{\text{val}}, X_{\text{miss}}, \Delta t, x_{\text{static}}) \mapsto \tilde{X}_{\text{val}} \in \mathbb{R}^{T\times F_{\text{dyn}}}$ that produces an adapted dynamic-feature stream; $\tilde{X}_{\text{val}}$ is reconstituted into the predictor's expected layout (re-attaching X_{miss}, broadcast statics, and any task-specific generated features; see App. B) before f_S consumes it, so that $f_S \circ g_\theta$ (with the reconstitution implicit in this composite) performs well on target labels y_T. Auxiliary heads (h_{aux} for $\mathcal{L}_{\text{tgt}}$, p_ξ for $\mathcal{L}_{\text{lp}}$) exist only during training; the deployed system is exactly $f_S(g_\theta(x_T))$, with f_S bit-identical to the certified source predictor. The Phase-2 (adaptation) objective is a weighted sum of six active terms, grouped by the three forces they impose on the adapter:

$$\mathcal{L}(\theta) \;=\; \underbrace{\mathcal{L}_{\text{task}}\big(f_S(g_\theta(x_T)), y_T\big) + \lambda_{\text{tgt}}\mathcal{L}_{\text{tgt}} + \lambda_{\text{lp}}\mathcal{L}_{\text{lp}}}_{\text{Utility: frozen predictor must succeed}} \;+\; \underbrace{\lambda_{\text{recon}}\mathcal{L}_{\text{recon}} + \lambda_{\text{range}}\mathcal{L}_{\text{range}}}_{\text{Faithfulness: clinical-signal preservation}} \;+\; \underbrace{\lambda_{\text{align}}\mathcal{L}_{\text{align}}\big(g_\theta(x_T), x_S\big)}_{\text{Coherence: input-space marginal alignment}} \tag{2}$$

Equation 2 balances three forces: utility (the frozen predictor must succeed on target labels), faithfulness (the adapter output must remain a plausible clinical signal), and coherence (the marginal distribution of adapted target inputs must not drift away from the source-input distribution that f_S was trained on). Per-task λ weights are tabulated in Appendix 4.

The six terms decompose as follows. $\mathcal{L}_{\text{task}}$ is cross-entropy (classification) or MSE (regression) through the frozen predictor; this is the only term that flows gradients through f_S. The auxiliary terms $\mathcal{L}_{\text{tgt}}, \mathcal{L}_{\text{lp}}$ are training-only: $\mathcal{L}_{\text{tgt}}$ applies task loss to an auxiliary head h_{aux} on $g_\theta(x_T)$ (non-vanishing gradient when f_S's decision surface is locally flat), and $\mathcal{L}_{\text{lp}}$ trains a linear probe on the adapter latent (never feeds f_S). $\mathcal{L}_{\text{recon}} = \|g_\theta(x_T) - X_{\text{val},T}\|_2^2$ is the task-dependent fidelity penalty (per-task weights in Appendix K); $\mathcal{L}_{\text{range}}$ penalises outputs outside the per-feature physiological envelope of $X_{\text{val},S}$; $\mathcal{L}_{\text{align}}$ is mini-batch MMD^2 between $\{g_\theta(x_T)\}$ and $\{x_S\}$ in input space (App. A.4.3). Full forms are in Appendix A.4. Per-task λ weights and the cross-attention depth L are listed in Appendix Table 4; the larger per-leaf knob form is in Appendix Table 27. Equation 2 governs Phase 2 (adaptation training); Phase 1 uses a separate reconstruction objective on source data only (§3.3).

The same adapter operates in two regimes that share the architecture and the objective. In the *source-assisted* regime the adapter is supervised by target labels with full access to source training data: the source-stat affine renormalisation, the source–target encoder MMD term ($\mathcal{L}_{\text{align}}$), the cross-domain reconstruction term, and the source-domain memory bank are all active. In the *source-free* regime no source sample contributes to the adapter's gradient: Phase 1 reconstruction is trained on the target cohort, the four source-tied loss terms are zeroed, and the cross-attention retrieval module is removed; only target labels and the frozen predictor's gradient remain. Both regimes are reported in Section 4.2.

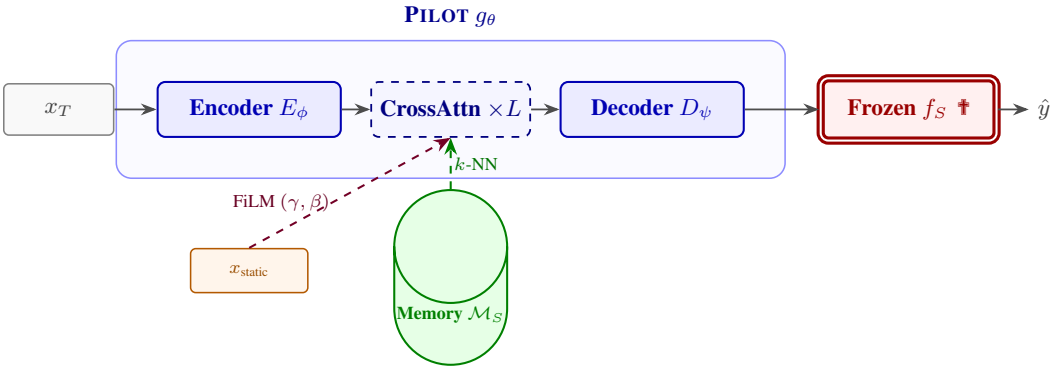

Figure 2: **PILOT pipeline.** A target window x_T enters the adapter g_θ (encoder $E_\phi \to$ optional cross-attention to a source memory bank $\mathcal{M}_S \to$ decoder D_ψ), whose output feeds the frozen predictor f_S (✝). Static covariates modulate every adapter block via FiLM (γ, β). Task loss flows through f_S into g_θ; f_S receives no updates.

3.2 ADAPTER ARCHITECTURE

Figure 2 summarizes the pipeline. The PILOT architecture comprises a shared encoder, a decoder, and, optionally, a source-domain memory bank with cross-attention fusion blocks. The memory bank is a conditional component, enabled or disabled per task by the decision tree in Appendix K. The encoder ingests the dynamic stream as a per-feature triplet $(X_{\text{val}}, X_{\text{miss}}, \Delta t) \in \mathbb{R}^{T \times F_{\text{dyn}} \times 3}$ projected to latent dim by a shared linear map, so the missing-indicator and time-delta channels are first-class inputs, not auxiliaries; static covariates x_{static} enter via feature-wise linear modulation (FiLM, Perez et al., 2018), producing per-layer scale/shift (γ_l, β_l) that modulate every encoder and decoder block, and are never concatenated into the dynamic stream. A shared encoder E_ϕ maps target and source windows to $z \in \mathbb{R}^{T \times F_{\text{dyn}} \times d}$; a decoder D_ψ maps back to input space. Each layer is an *axial attention block* factorising self-attention along both temporal and feature axes, capturing per-feature dynamics and cross-feature interactions (e.g., blood pressure with vasopressor dose) without collapsing the $T \times F_{\text{dyn}}$ tensor into a single token stream.

When retrieval is enabled, the source training split is encoded and segmented into windows of size W, each mean-pooled into a compact descriptor; the resulting memory bank $\mathcal{M}_S = \{E_\phi(x_S^j)\}_j$ resides on GPU and is rebuilt every R epochs. For each target timestep t, a causal-windowed query retrieves the k nearest source windows under Euclidean distance in latent space, matching local physiological states rather than full trajectories (k from grid search; Appendix A). Retrieved source context is integrated through L stacked cross-attention blocks. Each block computes a scaled dot-product attention from the target latent to its k nearest source latents, residually added to the target latent, followed by causal self-attention and a feed-forward network with residual connections and pre-LayerNorm (full form in Appendix A.5, Equation 8). Cross-attention allows selective, feature-level borrowing from source exemplars, so the model attends to different features from different neighbors rather than uniformly interpolating as in kNN-MT (Khandelwal et al., 2021). The decoder D_ψ maps the fused representations back to input space, $\tilde{X}_{\text{val}} = D_\psi(z_{\text{cross}}) \in \mathbb{R}^{T \times F_{\text{dyn}}}$. The cross-attention depth L is selected per task where retrieval helps (Appendix A).

3.3 TRAINING

Training follows a two-phase protocol. Per-task deltas for each phase are tabulated in Appendix C, Table 13. Each phase's necessity is ablated in Appendix C.5.

Phase 1 (Source Pretrain) trains E_ϕ, D_ψ on source data with identity reconstruction $\|D_\psi(E_\phi(x_S)) - X_{\text{val},S}\|_2^2$ (no masking). Phase 1 depends only on the data cohort, the architecture, and the seed, not on the downstream task, so a single checkpoint covers every task on that cohort; in the source-free regime the same checkpoint is built from the target cohort instead. **Phase 2 (Adaptation)** optimises $g_\theta = D_\psi \circ E_\phi$ (plus optional cross-attention) with Equation 2 on labeled target data; target features are first affinely renormalised to per-feature source statistics. Attention

is causal for per-timestep tasks (AKI, Sepsis, LoS) and bidirectional for per-stay tasks (Mortality, KF) to prevent leakage; AdaTime datasets use bidirectional attention as their windows carry a single window-level label. Per-task adapter configuration (loss weights, cross-attention depth, retrieval on/off) follows the decision tree in Appendix K, which serves as the complete deployment recipe.

4 EXPERIMENTS

4.1 SETUP

Datasets and tasks. Source $\mathcal{S}$ is MIMIC-IV; target $\mathcal{T}$ is eICU or HiRID, drawn from the YAIB benchmark cohorts (van de Water et al., 2024). Five clinical tasks span Mortality24 (in-ICU mortality from the first 24 hours), AKI and Sepsis (per-timestep classification), and Length-of-Stay and Kidney-Function (regression). Cohort sizes and per-task label structure are in Appendix B, Tables 6 and 7. For non-clinical generalisation we use the AdaTime suite (Ragab et al., 2023a) under its published cross-domain scenarios; we name each scenario explicitly when discussed in body text. Per-task deployment is determined by the PILOT playbook, a decision tree (Appendix K) that selects the exact hyperparameters and loss-term weights for each task.

Frozen predictors and baselines. The frozen predictor for each clinical task is the YAIB-released MIMIC-IV-trained LSTM; on AdaTime we reuse the published source-trained 1D-CNN. The clinical baseline ladder spans the frozen MIMIC-IV LSTM applied directly to eICU, a per-feature affine simple-adapter, end-to-end fine-tuning of the LSTM on eICU, two test-time adaptation methods (T3A (Iwasawa & Matsuo, 2021), SHOT (Liang et al., 2020)), seven end-to-end DA methods (DANN (Ganin et al., 2016), Deep CORAL (Sun & Saenko, 2016), CoDATS (Wilson et al., 2020), CDAN (Long et al., 2018), CLUDA (Ozyurt et al., 2023), ACON (Liu et al., 2024), and RAINCOAT (He et al., 2023)), and four DA-loss ablations of our adapter, in which the supervised translator loss is replaced by the DANN (Ganin et al., 2016), Deep CORAL (Sun & Saenko, 2016), CoDATS (Wilson et al., 2020), or ACON (Liu et al., 2024) domain-alignment objective while keeping the adapter architecture and the frozen MIMIC-IV-trained LSTM identical; this matched-architecture comparison isolates the contribution of the supervised target-label objective from any architectural advantage and shows the target loss is the right objective in the locked-predictor regime. TENT, SAR, and EATA re-estimate BatchNorm statistics at test time and apply only to architectures with BatchNorm; the YAIB LSTM has none, so we evaluate these methods on the AdaTime 1D-CNN where PILOT still wins on all five datasets (Appendix G). Input-space adaptation needs only that gradients flow through the frozen predictor; it does not need an adaptable normalisation surface.

Metrics. Classification reports AUROC ($\times 100$) with AUCPR companion deltas in Appendix Table 9. Regression reports MAE in target units (hours for LoS, mg/dL for KF) with z-scored MAE (denoted MAE-z) when comparing across regression tasks. Calibration is reported via Expected Calibration Error (ECE, 10-bin) and Brier on the held-out target test split, before any post-hoc temperature scaling (Appendix H). Native-LSTM numbers from the published YAIB benchmark are point estimates: we cite them without confidence intervals and treat any $\Delta < 0.01$ AUROC versus native (or $\Delta < 0.01$ mg/dL on KF) as a tie.

Statistical protocol. For each clinical task we report mean $\pm$ std across training seeds: five seeds for the source-assisted and for the source-free rows of Table 1, three seeds for the HiRID transfer. The headline ladder, the Bonferroni gate, and the HiRID transfer use paired bootstrap with 500 resamples on the held-out target test set; per-task bootstrap 95% CIs are tabulated in Table 1 (classification AUROC and regression MAE), Appendix Table 9 (classification AUCPR), and Appendix Table 10 (regression RMSE and R^2). Across the headline tables we report 20 paired contrasts (5 tasks $\times$ 4 baselines); per-contrast p-values on the HiRID transfer ladder (Appendix Table 11) are flagged against the Bonferroni-corrected threshold $\alpha = 0.05/20 = 0.0025$.

4.2 MAIN RESULTS

We report two regimes: source-assisted, with full access to source data, and source-free, where no source sample touches the adapter's gradient (§3.1). Table 1 reports our headline result on five clinical tasks: a frozen MIMIC-IV-trained predictor adapted to eICU with no change to the predictor's weights. The comparison ladder is the one defined in §4.1, augmented with a reference row for the in-domain natively-trained LSTM on eICU. Mean$\pm$std are over five training seeds; per-seed details

Table 1: Domain adaptation across five clinical tasks. Classification: AUROC ($\times 100$, ↑); regression: MAE (↓). † surpasses eICU-native LSTM. TTA rows (T3A, SHOT) run at test time (footnote §). T3A, SHOT and E2E methods are reported only on classification tasks (n/a on regression): those methods are defined for classifiers and have no standard regression formulation. End-to-end DA rows: full backbone unfrozen; numbers are best-hyperparameter within each method's published recommended range. The source-free row reports the strict source-free protocol (Phase 1 trained on the target deployment cohort, all source-coupling losses zeroed, retrieval disabled; Appendix K.1).

	CLASSIFICATION (AUROC ↑)			REGRESSION (MAE ↓)	
Method	Mortality	AKI	Sepsis	LoS (h)	KF (mg/dL)
Frozen predictor (our constraint)					
Frozen baseline (source only)	80.79	85.58	71.59	42.5	0.403
Per-feature affine adapter	80.85	86.55	71.02	40.8	0.417
Ours + DANN loss	84.38	88.74	73.23	n/a	n/a
Ours + CORAL loss	84.53	88.66	73.26	n/a	n/a
Ours + CoDATS discriminator	84.31	86.84	71.22	n/a	n/a
Ours + ACON contrastive	84.48	88.62	71.08	n/a	n/a
PILOT (source-assisted)	**85.64** †	**90.79** †	76.31 †	**37.1** †	0.287
±std	±0.09	±0.16	±0.40	±0.2	±0.004
[95% CI]	[84.65, 86.57]	[90.57, 91.11]	[76.14, 76.49]	[36.8, 37.3]	[0.280, 0.296]
PILOT (source-free)	84.73	90.50 †	76.58 †	**37.1** †	**0.286**
±std	±0.63	±0.23	±0.97	±0.1	±0.005
[95% CI]	[83.76, 85.79]	[90.20, 90.77]	[75.24, 77.98]	[36.90, 37.34]	[0.277, 0.292]
Reference: relaxes the frozen-predictor constraint					
Fine-tune LSTM (end-to-end)	84.61	90.48	74.59	38.5	0.304
DANN (E2E)	82.74	86.82	73.84	n/a	n/a
Deep CORAL (E2E)	82.79	86.94	73.41	n/a	n/a
CoDATS (E2E)	81.84	83.80	70.88	n/a	n/a
CDAN (E2E)	81.73	86.54	73.42	n/a	n/a
CLUDA (E2E)	81.81	87.35	**76.61**	n/a	n/a
RAINCOAT (E2E)	81.52	89.55	76.24	n/a	n/a
ACON (E2E)	82.23	70.79	75.30	n/a	n/a
T3A (TTA, frozen)[§]	80.61	82.18	72.24	n/a	n/a
SHOT (TTA, partial-freeze)[§]	80.21	65.03	70.49	n/a	n/a
Reference: natively trained in-domain models					
eICU-native LSTM	*85.5*	*90.2*	*74.0*	*39.2*	*0.28*

[§]TTA methods operate at test time only; SHOT partially unfreezes the LSTM feature extractor. TENT/SAR/EATA inapplicable to the YAIB LSTM (see §4.1); full TTA results on AdaTime are in Appendix G.

are in Appendix C.4. On non-clinical sensor benchmarks, we test whether the same recipe generalises beyond EHR. Table 2 reports results on AdaTime's five datasets (HAR, HHAR, WISDM, SSC, MFD) with a frozen 1D-CNN backbone, comparing our adapter against the strongest end-to-end DA method per dataset (selected from thirteen baselines: AdaTime's eleven plus CLUDA and RAINCOAT).

Source-assisted regime. The adapter matches or exceeds an eICU-native LSTM on all five clinical tasks, reaching native parity on KF within the ± 0.01 mg/dL tie threshold (§4.1). It also matches or exceeds the end-to-end fine-tuning upper bound on all five tasks, while keeping the certified weights bit-identical. On the sparse-label classification tasks (AKI and Sepsis) where positives are severely outnumbered, AUCPR is the more clinically informative metric: it weights rank-ordering of the rare true positives that drive alarm-triggering decisions. All three classification tasks gain in AUCPR, with the largest gain reaching $+15.3$ points on eICU AKI (per-task AUCPR in Appendix Table 9). Classification deltas improve by 4.93 AUROC points on average, and regression tasks by 0.021 MAE on z-scored targets. Per-task deltas appear in Table 1, full regression metrics in Appendix Table 10. On AdaTime, the same recipe wins on all five datasets at mean $+2.6$ Macro-F1.

Source-free regime. Appendix Table 22 reports the source-free results; the disabling protocol is described in Section 3.1 and detailed in Appendix K. Across all five clinical tasks (Appendix K.1, Table 22) and all five AdaTime datasets (Appendix K.2), source-free adaptation approaches or

Table 2: AdaTime (Macro-F1 $\times 100$). Frozen-backbone adapter vs. best of thirteen end-to-end methods (eleven from Ragab et al. (2023a) plus CLUDA and RAINCOAT). Mean$\pm$std over five seeds (HAR / HHAR / WISDM source-free) or three seeds (SSC / MFD source-free); all source-assisted cells use five seeds. Full per-method numbers for the eleven AdaTime baselines are in Ragab et al. (2023a) Table 4.

Dataset	Domain	Source	Best E2E	Ours, with source data	Δ_{src}	Ours, source-free	Δ_{free}
HAR	Wearable	80.0	93.7[a]	**94.1**$\pm$0.0	+0.4	**92.41**$\pm$0.79	-1.3
HHAR	Wearable	56.5	84.5[b]	**87.0**$\pm$0.7	+2.5	**87.36**$\pm$0.92	+2.9
WISDM	Wearable	50.0	66.3[b]	**70.3**$\pm$1.5	+4.0	**70.65**$\pm$4.67	+4.4
SSC	EEG	58.0	63.5[c]	**66.2**$\pm$0.2	+2.7	**65.69**$\pm$1.77	+2.2
MFD	Industrial	77.5	92.8[a]	**96.1**$\pm$0.1	+3.3	**96.93**$\pm$0.18	+4.1
Mean		64.4	80.2	**82.7**	+2.6	**82.61**	+2.4

[a]DIRT-T (Ragab et al., 2023a) [b]CoTMix (Ragab et al., 2023a) [c]MMDA (Ragab et al., 2023a)

surpasses the source-assisted regime. In absolute terms, source-free adaptation still matches or beats the YAIB eICU-native baseline on four of five clinical tasks (per-task absolutes in Appendix C); elaborated analysis is detailed in Appendix K.1.

Calibration improves as a byproduct. Across nine classification cells spanning the MIMIC-IV$\rightarrow$eICU headline, the MIMIC-IV$\rightarrow$eICU tiny-adapter re-evaluation, and the MIMIC-IV$\rightarrow$HiRID best-cell port, input-space adaptation reduces Brier on $9/9$ cells and ECE in mean on $8/9$ cells with no explicit calibration loss; the effect appears structural (per-cell Δ ECE in Appendix Table 20). Post-hoc temperature scaling composes with this on top of the eICU headline cells, yielding a further -0.013 ECE on Mortality and -0.033 on Sepsis with bit-exact AUROC; AKI saturates at $\bar{T}\approx 0.99$, consistent with its already-near-uniform confidence histogram (Appendix H.1). In contrast, frozen-backbone DA baselines trade calibration for discrimination: DANN and CoDATS worsen AKI calibration while improving AUROC. Full per-task Brier and ECE deltas (mean$\pm\sigma$ across five seeds) appear in Appendix H.

4.3 Cross-benchmark generalisation

We test whether the result generalises along four axes: deployment cohort (eICU vs. HiRID), data modality (clinical vs. non-clinical), predictor family (LSTM vs. GRU / TCN / 1D-CNN), and adapter capacity ($\leq 1\times$ vs. $> 1\times$ predictor parameters).

Cross-cohort generalisation. Each best-eICU recipe ports cleanly to HiRID (only data paths swap; full per-task results in Appendix Table 11). All 15 task–seed cells survive paired-bootstrap testing at $p < 0.001$; the AKI HiRID adapter beats native by $+1.7$ AUROC (82.7 vs 81.0), and HiRID LoS beats native by 1.7 hours of MAE (38.1 h vs 39.8 h, ΔMAE-z $= -0.0766 \pm 0.0048$). Adaptation deltas on HiRID exceed the corresponding eICU deltas on every classification task. The complete deployment recipe, including per-task fidelity-loss and cross-attention-depth selection, the source-free branch of §4.2, and the reduced-capacity adapter configuration, is the PILOT playbook (Appendix K).

Non-clinical sensor benchmarks. On the five AdaTime benchmarks (HAR, HHAR, WISDM, SSC, MFD; wearable, EEG, industrial) with a frozen 1D-CNN backbone, PILOT beats AdaTime's eleven published end-to-end DA methods plus CLUDA and RAINCOAT on all five datasets (mean $+2.6$ Macro-F1, Appendix C; trained under AdaTime's exact recipe (Ragab et al., 2023a)). Against the five strongest test-time adaptation baselines on the AdaTime backbone (T3A, SHOT, TENT, SAR, EATA), our adaptor wins on every dataset by up to $+10.1$ mean Macro-F1 (Appendix C). The same input-rewriting effect that helped on EHR forward-fill cadence transfers to wearable, EEG, and industrial sensor distributions.

Cross-architecture transfer. To our knowledge, no prior input-space adaptor has reported zero-shot transfer across predictor architectures on clinical time-series DA. An LSTM-trained adapter applied zero-shot to frozen GRU and TCN predictors retains 53–103% of the LSTM gain across five tasks (classification and regression) and two target cohorts (eICU and HiRID), with positive transfer in

all 16 task–architecture cells (Appendix Table 14); the signal the adapter learns is not specific to recurrent gating but generalises across recurrent, convolutional, and gated predictor families.

Reduced-capacity adapter. A reduced-capacity adapter approximately matched to the frozen predictor's parameter count (obtained by trimming d_{model}, d_{latent}, encoder/decoder depth, and cross-attention depth) matches or exceeds the full adapter on $8/10$ task–benchmark cells; exact per-task parameter counts and ratios are in Appendix E.1 (Tables 15 and 16).

4.4 Interpretability: distribution alignment is not the mechanism

A standard reading of domain adaptation says that pulling the target marginals towards the source training distribution is what enables transfer. Our adapter does the opposite: Wasserstein distance from adapted eICU to MIMIC-IV grows by $3.7\times$ on both AKI and Sepsis (MMD by $14\times$ and $22\times$, App. C), yet AUROC rises by four to six points and the adapted predictor matches or surpasses both end-to-end fine-tuning and the eICU-native LSTM (Table 1). Consistent with this divergence, the component ablation in Appendix Table 18 shows the gain is carried by the task gradient through the frozen predictor, not by the marginal-alignment term. What the adapter actually does is restore operational semantics: per-feature imputation fraction correlates strongly with the adaptation-induced lag-1 autocorrelation change $|\Delta\text{ACF}|$ (Spearman ρ=0.81 on AKI, 0.74 on Sepsis, $p < 10^{-9}$; App. Fig. 3). By reversing forward-fill artifacts in sparsely-sampled labs, the adapter rewrites each input channel into the temporal cadence the frozen LSTM was trained to consume, reusing the predictor's already-fit decision surface at its certified operating point rather than drifting its weights, which is why an input-space adapter can reach native parity and surpass end-to-end fine-tuning without ever touching the model's weights.

5 Conclusions and Limitations

Locked-weight, labelled-target domain adaptation has a clean operating point: adapt the data, not the model. A target-labelled input adapter trained by task-loss back-propagation through a strictly frozen predictor matches or surpasses all verified baselines without changing certified weights or requiring source training data in the source-free variant. The same adapter generalises across cohorts and predictor families, and a reduced-capacity variant retains the gain at predictor-matched parameter counts (Appendix E). Extending the regime to streaming targets without retrospective labels and to additional clinical or non-clinical modalities is the natural next step.

Limitations. While the adapter matches or surpasses all verified baselines without modifying certified weights, several limitations remain. **Supervised adaptation requires target labels.** We assume the deploying institution has retrospective target labels and a frozen, certified predictor it cannot retrain. Unsupervised DA and streaming TTA are out of scope: without target supervision the task loss has nothing to back-propagate. Deployments that lack retrospective target labels do not satisfy this assumption. **Single clinical labelled source.** Although the method is verified on the AdaTime five-dataset suite as well, every reported clinical headline uses MIMIC-IV as the only labelled source, so the clinical headline deltas were tested on the MIMIC-as-source regime; per-cell paired bootstraps are nonetheless statistically significant (Appendix C). **Empirically-derived per-task rules.** The methodological claim is supported across all reported cohorts and benchmarks (paired-bootstrap $p < 0.001$). The per-task playbook leaves in Appendix K are empirically inductive; if applied to a new label-density regime or a backbone outside LSTM / CNN / GRU / TCN, the ablations should be re-run before deployment. **The adapter is itself an ML component.** The adapter leaves the certified predictor bit-identical but is itself a separate ML component subject to its own regulatory obligations (U.S. Food and Drug Administration, 2024); it is trained once per (dataset, task) pair against publicly released frozen predictors, transfers across predictor architectures at that pair (Appendix D), and in the source-free variant requires neither the certified weights nor the source training data, so any obligation accrues once per pair on prerequisites strictly weaker than retraining.

AI use statement

In this work, we used generative AI tools to write parts of the experiment-orchestration and result-analysis code; all such code was reviewed, tested, and integrated by the authors, and the adapter architecture and training procedure were designed and implemented by the authors. We have not

used generative AI tools to generate synthetic data, to develop the theoretical framing or conceptual framework, to formulate or prove mathematical claims, to propose or refine hypotheses, to design or give feedback on the methodology or experiments, to clean or reformat datasets, or to interpret results; translation and qualitative or thematic data analysis are not applicable to this work. Additionally, we used generative AI tools to edit the manuscript for readability and to identify and summarise related literature. We have reviewed all AI-assisted work: every reference surfaced with AI assistance was verified against the original publication before citation, every edited passage was checked against the underlying experimental records, and AI-assisted code was covered by the unit tests shipped with the supplementary code archive. We take responsibility for the final content of this work, including text, claims, and artifacts produced with the aid of generative AI.

Ethics statement

This study uses only retrospective, de-identified intensive-care datasets (MIMIC-IV, eICU-CRD, and HiRID) that are distributed under PhysioNet credentialed-access data-use agreements, together with the public AdaTime sensor benchmarks; no new data were collected from human subjects and no raw patient records are redistributed. The method is intended to improve the reliability of predictors that are deployed under distribution shift, but the adapter is itself a learned component: as discussed in Section 5, it must be validated on the deploying site's own data before any clinical use, and our evaluation of fairness across patient subgroups is out of scope. We see no conflicts of interest or sponsorship concerns to declare.

Reproducibility statement

Appendix A lists every hyperparameter and the seed and split protocol used for each reported cell, and Appendix A.2 reports the hardware and wall-clock budgets. Appendix B describes the cohorts, tasks, and preprocessing, and Appendix A.4 gives the full form of every auxiliary loss. The anonymised code archive submitted as supplementary material contains the adapter and trainer implementation, the exact configuration file behind each cell of the main-body and appendix tables, the fixed split identifiers, the per-run result records that back the calibration and Phase-1 tables, a reproduction guide, and the configuration builder described in Appendix K.6.

References

Michael D. Abràmoff, Philip T. Lavin, Michele Birch, Nilay Shah, and James C. Folk. Pivotal trial of an autonomous AI-based diagnostic system for detection of diabetic retinopathy in primary care offices. *npj Digital Medicine*, 1:39, 2018. doi: 10.1038/s41746-018-0040-6.

Bassel Almarie, Luis Fernando Gonzalez-Gonzalez, Lucas Antônio dos Santos Barbosa, Amelie Lutz, Ulrich Grosse, and Felipe Fregni. Machine learning-enabled medical devices authorized by the US Food and Drug Administration in 2024: Regulatory characteristics, predicate lineage, and transparency reporting. *Biomedicines*, 13(12):3005, 2025. doi: 10.3390/biomedicines13123005.

Clair Blacketer, Martijn J. Schuemie, Maxim Moinat, Erica A. Voss, Montse Camprubi, Peter R. Rijnbeek, and Patrick B. Ryan. Advancing real-world evidence through a federated health data network (EHDEN): Descriptive study. *Journal of Medical Internet Research*, 27(1):e74119, 2025. doi: 10.2196/74119.

Sebastian Borgeaud, Arthur Mensch, Jordan Hoffmann, Trevor Cai, Eliza Rutherford, Katie Millican, George van den Driessche, Jean-Baptiste Lespiau, Bogdan Damoc, Aidan Clark, Diego de Las Casas, Aurelia Guy, Jacob Menick, Roman Ring, Tom Hennigan, Saffron Huang, Loren Maggiore, Chris Jones, Albin Cassirer, Andy Brock, Michela Paganini, Geoffrey Irving, Oriol Vinyals, Simon Osindero, Karen Simonyan, Jack W. Rae, Erich Elsen, and Laurent Sifre. Improving language models by retrieving from trillions of tokens. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2022.

Konstantinos Bousmalis, Nathan Silberman, David Dohan, Dumitru Erhan, and Dilip Krishnan. Unsupervised pixel-level domain adaptation with generative adversarial networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

Manuel Burger, Stephanie L. Hyland, Gunnar Ratsch, Bowen Lyu, Rita Kuznetsova, Hugo Yeche, Sasha Kobeissi, Alizee Pace, Lasse Spies, and Daphne Marx. A foundation model for intensive care. *medRxiv preprint*, 2025. ML4H 2025 Spotlight.

Lang Cao, Qingyu Chen, and Yue Guo. EHR-RAG: Bridging long-horizon structured electronic health records and large language models via enhanced retrieval-augmented generation. *arXiv preprint arXiv:2601.21340*, 2026. doi: 10.48550/arXiv.2601.21340.

Pin-Yu Chen. Model reprogramming: Resource-efficient cross-domain machine learning. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, 2024.

Khaled El Emam, Sam Rodgers, and Bradley Malin. Anonymising and sharing individual patient data. *BMJ*, 350:h1139, 2015. doi: 10.1136/bmj.h1139.

Gamaleldin F. Elsayed, Jascha Sohl-Dickstein, and Ian Goodfellow. Adversarial reprogramming of neural networks. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2019.

Samuel G. Finlayson, Adarsh Subbaswamy, Karandeep Singh, John Bowers, Annabel Kupke, Jonathan L. Zittrain, Isaac S. Kohane, and Suchi Saria. The clinician and dataset shift in artificial intelligence. *New England Journal of Medicine*, 385(3):283–286, 2021. doi: 10.1056/NEJMc2104626.

Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. Domain-adversarial training of neural networks. *Journal of Machine Learning Research (JMLR)*, 17:1–35, 2016.

Arthur Gretton, Karsten M. Borgwardt, Malte J. Rasch, Bernhard Schölkopf, and Alexander Smola. A kernel two-sample test. *Journal of Machine Learning Research (JMLR)*, 13:723–773, 2012.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. On calibration of modern neural networks. In *Proceedings of the 34th International Conference on Machine Learning (ICML)*, pp. 1321–1330, 2017.

Melissa A. Haendel, Christopher G. Chute, Tellen D. Bennett, David A. Eichmann, Justin Guinney, Warren A. Kibbe, Philip R. O. Payne, Emily R. Pfaff, Peter N. Robinson, Joel H. Saltz, Heidi Spratt, Christine Suver, John Wilbanks, Adam B. Wilcox, et al. The National COVID Cohort Collaborative (N3C): Rationale, design, infrastructure, and deployment. *Journal of the American Medical Informatics Association*, 28(3):427–443, 2021. doi: 10.1093/jamia/ocaa196.

Hrayr Harutyunyan, Hrant Khachatrian, David C. Kale, Greg Ver Steeg, and Aram Galstyan. Multitask learning and benchmarking with clinical time series data. *Scientific Data*, 6(96), 2019.

Huan He et al. Domain adaptation for time series under feature and label shifts. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2023.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin de Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. Parameter-efficient transfer learning for NLP. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2019.

George Hripcsak, Jon D. Duke, Nigam H. Shah, Christian G. Reich, Vojtech Huser, Martijn J. Schuemie, Marc A. Suchard, Rae Woong Park, Ian Chi Kei Wong, Peter R. Rijnbeek, Johan van der Lei, Nicole Pratt, G. Niklas Norén, Yu-Chuan Li, Paul E. Stang, David Madigan, and Patrick B. Ryan. Observational health data sciences and informatics (OHDSI): Opportunities for observational researchers. In *Studies in Health Technology and Informatics*, volume 216, pp. 574–578, 2015. doi: 10.3233/978-1-61499-564-7-574.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2022.

Yusuke Iwasawa and Yutaka Matsuo. Test-time classifier adjustment module for model-agnostic domain generalization. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021. T3A.

Menglin Jia, Luming Tang, Bor-Chun Chen, Claire Cardie, Serge Belongie, Bharath Hariharan, and Ser-Nam Lim. Visual prompt tuning. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2022.

Ming Jin, Shiyu Wang, Lintao Ma, Zhixuan Chu, James Y. Zhang, Xiaoming Shi, Pin-Yu Chen, Yuxuan Liang, Yuan-Fang Li, Shirui Pan, and Qingsong Wen. Time-LLM: Time series forecasting by reprogramming large language models. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2024.

Urvashi Khandelwal, Omer Levy, Dan Jurafsky, Luke Zettlemoyer, and Mike Lewis. Generalization through memorization: Nearest neighbor language models. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2020.

Urvashi Khandelwal, Angela Fan, Dan Jurafsky, Luke Zettlemoyer, and Mike Lewis. Nearest neighbor machine translation. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2021.

Pang Wei Koh, Shiori Sagawa, Henrik Marklund, Sang Michael Xie, Marvin Zhang, Akshay Balsubramani, Weihua Hu, Michihiro Yasunaga, Richard Lanas Phillips, Irena Gao, Tony Lee, Etienne David, Ian Stavness, Wei Guo, Berton A. Earnshaw, Imran S. Haque, Sara Beery, Jure Leskovec, Anshul Kundaje, Emma Pierson, Sergey Levine, Chelsea Finn, and Percy Liang. WILDS: A benchmark of in-the-wild distribution shifts. In *International Conference on Machine Learning (ICML)*, 2021.

Brian Lester, Rami Al-Rfou, and Noah Constant. The power of scale for parameter-efficient prompt tuning. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2021.

Patrick Lewis et al. Retrieval-augmented generation for knowledge-intensive NLP tasks. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

Yikuan Li, Shishir Rao, Jose Roberto Ayala Solares, Abdelaali Hassaine, Rema Ramakrishnan, Dexter Canoy, Yajie Zhu, Kazem Rahimi, and Gholamreza Salimi-Khorshidi. BEHRT: Transformer for electronic health records. *Scientific Reports*, 10, 2020.

Jian Liang, Dapeng Hu, and Jiashi Feng. Do we really need to access the source data? Source hypothesis transfer for unsupervised domain adaptation. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2020.

Mingyang Liu, Xinyang Chen, Yang Shu, Xiucheng Li, Weili Guan, and Liqiang Nie. Boosting transferability and discriminability for time series domain adaptation. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024. URL `https://proceedings.neurips.cc/paper_files/paper/2024/hash/b61da4f02b271cb7b5e3d538e2b78fb9-Abstract-Conference.html`.

Xiao Liu, Kaixuan Ji, Yicheng Fu, Weng Lam Tam, Zhengxiao Du, Zhilin Yang, and Jie Tang. P-tuning v2: Prompt tuning can be comparable to fine-tuning universally across scales and tasks. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (ACL, Short Papers)*, 2022.

Mingsheng Long, Zhangjie Cao, Jianmin Wang, and Michael I. Jordan. Conditional adversarial domain adaptation. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2018.

Manoj Kumar Mutnuri et al. Using domain adaptation and inductive transfer learning to improve patient outcome prediction in the intensive care unit. *JMIR Medical Informatics*, 2024.

Kailai Ning et al. TS-RAG: Retrieval-augmented generation based time series foundation models are stronger zero-shot forecaster. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025.

Shuaicheng Niu, Jiaxiang Wu, Yifan Zhang, Yaofo Chen, Shijian Zheng, Peilin Zhao, and Mingkui Tan. Efficient test-time model adaptation without forgetting. In *International Conference on Machine Learning (ICML)*, 2022. EATA.

Shuaicheng Niu, Jiaxiang Wu, Yifan Zhang, Zhiquan Wen, Yaofo Chen, Peilin Zhao, and Mingkui Tan. Towards stable test-time adaptation in dynamic wild world. In *International Conference on Learning Representations (ICLR)*, 2023. SAR.

OTTEHR Authors. Transport-based transfer learning on electronic health records. *Journal of the American Medical Informatics Association*, 33(1):15, 2025.

Yilmazcan Ozyurt, Stefan Feuerriegel, and Ce Zhang. Contrastive learning for unsupervised domain adaptation of time series. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2023.

David Peloquin, Michael DiMaio, Barbara E. Bierer, and Mark Barnes. Disruptive and avoidable: GDPR challenges to secondary research uses of data. *European Journal of Human Genetics*, 28 (6):697–705, 2020. doi: 10.1038/s41431-020-0596-x.

Ethan Perez, Florian Strub, Harm de Vries, Vincent Dumoulin, and Aaron Courville. FiLM: Visual reasoning with a general conditioning layer. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2018.

Jonas Pfeiffer, Aishwarya Kamath, Andreas Rücklé, Kyunghyun Cho, and Iryna Gurevych. AdapterFusion: Non-destructive task composition for transfer learning. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics (EACL)*, 2021. arXiv:2005.00247, 2020.

Richard Platt, Jeffrey S. Brown, Melissa Robb, Mark McClellan, Robert Ball, Michael D. Nguyen, and Rachel E. Sherman. The FDA Sentinel Initiative—An evolving national resource. *New England Journal of Medicine*, 379(22):2091–2093, 2018. doi: 10.1056/NEJMp1809643.

Sanjay Purushotham, Wilka Carvalho, Tanachat Nilanon, and Yan Liu. Variational recurrent adversarial deep domain adaptation. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2017.

Sanjay Purushotham et al. Benchmarking deep learning models on large healthcare datasets. *Journal of Biomedical Informatics*, 83:112–134, 2018.

Yunzhong Qiu, Zhiyao Cen, Zhongyi Pei, Chen Wang, and Jianmin Wang. Adapt data to model: Adaptive transformation optimization for domain-shared time series foundation models. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2026. URL `https://openreview.net/forum?id=uTK1SNgi1N`.

Mohamed Ragab, Emadeldeen Eldele, et al. ADATIME: A benchmarking suite for domain adaptation on time series data. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 2023a.

Mohamed Ragab et al. Source-free domain adaptation with temporal imputation for time series data. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, 2023b.

Laila Rasmy et al. Med-BERT: Pretrained contextualized embeddings on large-scale structured electronic health records for disease prediction. *npj Digital Medicine*, 4(86), 2021.

Benjamin Recht, Rebecca Roelofs, Ludwig Schmidt, and Vaishaal Shankar. Do ImageNet classifiers generalize to ImageNet? In *International Conference on Machine Learning (ICML)*, 2019.

Baochen Sun and Kate Saenko. Deep CORAL: Correlation alignment for deep domain adaptation. In *Proceedings of the European Conference on Computer Vision Workshops (ECCV-W)*, 2016.

Yi-Lin Sung, Jaemin Cho, and Mohit Bansal. LST: Ladder side-tuning for parameter and memory efficient transfer learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

U.S. Food and Drug Administration. Artificial intelligence/machine learning (AI/ML)-based software as a medical device (SaMD) action plan. https://www.fda.gov/medical-devices/software-medical-device-samd, 2021.

U.S. Food and Drug Administration. Marketing submission recommendations for a predetermined change control plan for artificial intelligence-enabled device software functions: Guidance for industry and FDA staff. Final Guidance, U.S. FDA, Center for Devices and Radiological Health, December 2024. Federal Register Doc. 2024-28361, finalised 4 December 2024.

Robin van de Water et al. Yet another ICU benchmark: A flexible multi-center framework for clinical ML. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2024.

Dequan Wang, Evan Shelhamer, Shaoteng Liu, Bruno Olshausen, and Trevor Darrell. Tent: Fully test-time adaptation by entropy minimization. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2021.

Garrett Wilson, Janardhan Rao Doppa, and Diane J. Cook. Multi-source deep domain adaptation with weak supervision for time-series sensor data. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, 2020.

Andrew Wong, Erkin Otles, John P. Donnelly, Andrew Krumm, Jeffrey McCullough, Olivia DeTroyer-Cooley, Justin Pestrue, Marie Phillips, Judy Konye, Carleen Penoza, Muhammad Ghous, and Karandeep Singh. External validation of a widely implemented proprietary sepsis prediction model in hospitalized patients. *JAMA Internal Medicine*, 181(8):1065–1070, 2021. doi: 10.1001/jamainternmed.2021.2626.

Ran Xu et al. RAM-EHR: Retrieval augmentation meets clinical predictions on electronic health records. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*, 2024.

Chao-Han Huck Yang, Yun-Yun Tsai, and Pin-Yu Chen. Voice2Series: Reprogramming acoustic models for time series classification. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2021.

Yanchao Yang and Stefano Soatto. FDA: Fourier domain adaptation for semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

Tian Zhou et al. One fits all: Power general time series analysis by pretrained LM. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

Appendix overview. This appendix is organised as twelve sections. Section A (Implementation Details) gathers hyperparameters, hardware, training schedule, computational cost, and full mathematical definitions of the auxiliary loss terms. Section B (Datasets and Tasks) describes the cohorts, per-task input-channel composition, the per-feature dynamic triplet, FiLM static-covariate handling, and the YAIB preprocessing pipeline. Section C (Extended Quantitative Results) holds the AUCPR, regression, HiRID, seed-variance, and Phase-1 ablation tables. Section D (Architecture-Agnostic Transfer) reports zero-shot transfer of an LSTM-trained adapter to frozen GRU and TCN predictors. Section E (Reduced-Capacity Adapter) sweeps adapter capacity below the frozen predictor. Section F (Component Ablation) gives the AKI per-component matrix. Section G (Test-Time Adaptation Baselines) reports T3A, SHOT, TENT, SAR, and EATA. Section H (Calibration) reports Brier and ECE. Section I (Imputation vs. ACF) gives the operational-semantics interpretation. Section J (Systematic Method Comparison) is the nine-axis cross-method scorecard. Section K (PILOT Playbook) is the prescriptive decision tree distilled from the ablations.

A Implementation Details

Code-vs-paper convention. The accompanying codebase (configs, variable names, docstrings) uses the opposite source/target convention from this paper: in the code, `source` refers to what the paper calls the *target* domain (eICU / HiRID) and `target` refers to what the paper calls the *source* domain (MIMIC-IV, the distribution the frozen LSTM was trained on). Reviewers browsing the repository should mentally swap the two labels when comparing to the paper's formulation.

A.1 HYPERPARAMETERS

Table 3 lists the architecture and optimisation settings that are shared across all five EHR tasks; Table 4 gives the per-task active loss weights and cross-attention depth L for the headline ablation cells. The values reported here drove every result in the main-body tables; eICU headline cells are averaged over five training seeds and HiRID transfer cells over three. Full per-leaf knobs (tied to the playbook in Appendix K) are tabulated in Table 27.

Table 3: Hyperparameters for the PILOT architecture.

Hyperparameter	**Value**
d_{model}	128
d_{latent}	128
n_{heads}	8
d_{ff}	512
$n_{\text{enc_layers}}$	4
$n_{\text{dec_layers}}$	3
$n_{\text{cross_layers}}$	2 (Mort., Sepsis) / 3 (AKI, LoS, KF)
k (neighbors)	16
W (retrieval window)	6 timesteps (matches the 6-hour reassessment cycle used for sepsis and AKI definitions)
Learning rate	5×10^{-4}
Optimizer	AdamW
Batch size	16
Phase 1 pretrain epochs	15
Phase 2 training epochs	30–35 (early stopping, patience 15)
Memory refresh (R)	5 epochs
Dropout	0.2

Table 4: Per-task active loss weights and cross-attention depth L for the headline ablation cells (compact form; full per-leaf knobs are in Appendix Table 27). The main task loss $\mathcal{L}_{\text{task}}$ (Sec. 3) is omitted from the columns because it is always active across every leaf: it is the only term that back-propagates gradients through the frozen predictor f_S and is the framework's central supervision signal. "n/a" denotes a disabled auxiliary term (weight set to zero).

Task	λ_{recon}	λ_{align}	λ_{tgt}	λ_{lp}	λ_{range}	L
Mortality	n/a	0.5	0.5	0.1	0.5	2
AKI	0.1	0.5	0.5	0.1	0.5	0
Sepsis	0.1	0.5	0.5	0.1	0.5	0
LoS	0.1	n/a	0.5	0.1	0.5	3
KF	n/a	n/a	n/a	n/a	0.5	3

Task-specific settings. Phase 1 trains the encoder E_ϕ and decoder D_ψ on source data with a reconstruction objective; cross-attention blocks (when present in the architecture) receive zero context during Phase 1 and only see retrieved source context in Phase 2 once the memory bank exists. Variable-length batching is used for per-timestep tasks and disabled for per-stay tasks.

Seeds and configuration files. eICU headline results use five training seeds $\{2222, 42, 7777, 1234, 9999\}$; HiRID transfer results use three training seeds $\{2222, 42, 7777\}$; the data split seed is fixed at 2222 across every experiment so that only the training-time stochasticity (initialisation, batch ordering, dropout) varies between seeds. AdaTime runs use the benchmark's standard five-seed protocol ($\{0, 1, 2, 3, 4\}$) with the same per-leaf adapter recipe. The exact JSON configuration that drove each cell in the main-body and appendix tables is shipped with the anonymised code release: per-task base configurations, the per-cell ablation configurations of Appendix F (one config per row of Table 18), and the HiRID variants are all included in the anonymised code release; the playbook-driven config builder (Appendix K.6) regenerates any leaf config from a single command. Per-run result JSONs that back Tables 20 and 13 are released alongside the configurations so those numeric cells can be re-derived from the snapshot without

Table 5: Training time per task over five seeds on the fastest GPU available per task (A6000-48GB where possible; L40S/A100 for LoS and KF, which were not run on A6000 or 3090). Phase 1 is pretrained once and reused across every experiment with matching architecture. Under the source-free protocol with the reduced-capacity adapter (no cross-attention, no memory bank), Phase 1 itself is structurally smaller and trains in approximately 27 minutes rather than 2.5 hours; per-seed end-to-end Mortality wall-clock on an RTX 6000 Ada-48GB is approximately 2 hours (three-seed range 1h43m–2h08m), with no Phase 1 amortization assumed.

Task	GPU	Phase 1 (h)	Phase 2 (h)	Total (h)
Mortality	A6000	2.5	5.0	7.5
AKI	A6000	7.0	16.3	23.3
Sepsis	3090	6.8	13.5	20.3
LoS	L40S	1.4	18.1	19.5
KF	A100	0.5	3.5	4.0

re-training. All other tables are reproduced by re-running the corresponding `(config, seed)` pair under §A.2 compute budgets.

AdaTime configuration. The same PILOT architecture is used with reduced capacity: $d_{\text{model}} = 64$, $d_{\text{latent}} = 64$, $d_{\text{ff}} = 128$, $n_{\text{enc_layers}} = 2$, $n_{\text{dec_layers}} = 1$, $n_{\text{cross_layers}} = 2$–3. The frozen backbone is AdaTime's standard 3-block 1D-CNN (∼200K parameters). Standard AdaTime adapter (d_{model}=64) uses 100K–450K parameters depending on dataset; a reduced ($\leq 1\times$ predictor) variant uses 68K–196K parameters and matches or exceeds the standard adapter on all five AdaTime datasets (Appendix E). All AdaTime runs (ours and every baseline) follow the benchmark's exact recipe: 40-epoch budget, Adam with $\beta = (0.5, 0.99)$, weight decay 10^{-4}, last-epoch model, no target-validation split.

A.2 HARDWARE AND COMPUTATIONAL COST

All reported EHR results run on a heterogeneous GPU pool: A6000-48GB (primary for ablations and multi-seed validation), V100S-32GB (early controls), RTX 3090-24GB (sepsis ablations), and L40S / A100-40GB SLURM nodes (LoS and KF). Cross-server variance is ± 0.003–0.006 AUROC. Training decomposes into a 15-epoch Phase-1 encoder pretrain on the source cohort (or the target cohort under the source-free protocol of §4.2) followed by a 30–35 epoch Phase-2 adaptation run. Phase 1 depends only on the (architecture, data cohort, seed) fingerprint, not on the downstream task: a single Phase-1 checkpoint serves every task on that cohort. With source data available, this checkpoint is reused across all target deployments; in the source-free regime it is reused across all tasks at the target institution. The deployment cost of adapting to a new task is therefore bounded by a single Phase-2 run (≈3.6–18 GPU-hours per task across A6000, 3090, L40S, and A100-40GB), comparable in order of magnitude to retraining the LSTM natively on the target cohort but with the frozen predictor preserved bit-identical.

Table 5 reports three-seed training time per task. Phase 1 is pretrained once and reused across every experiment with matching architecture, so the per-deployment cost reduces to one Phase-2 run.

Parameter counts. Adapter: 2.59–2.88M parameters depending on task (dominated by 7 axial attention blocks at 265K each, plus 2–3 cross-attention blocks). Tiny variant (Appendix E): 161K–1.43M on clinical tasks, 68K–196K on AdaTime (approximately matched to the frozen predictor; LoS at $1.06\times$, the other four EHR tasks at $\leq 1\times$). Frozen LSTMs: 170K (Sepsis) to 1.42M (AKI).

Inference overhead. At test time, the memory bank is pre-built once (∼6 minutes for AKI on an A6000). Per-stay inference latency on an RTX 6000 Ada (48 GB) is ∼27 ms for the full adapter and ∼18 ms for the tiny variant (Appendix E; 0.26–$0.34\times$ the full-adapter parameter count), both measured on Sepsis (∼1.28M test timesteps), including k-NN retrieval from a 569K-window memory bank. Memory-bank construction is a one-time pre-deployment cost: ∼222 s for the full adapter and ∼55 s for the tiny variant on the same GPU. The adapter is trained once per task and reused across downstream architectures (Appendix D), so this overhead is paid once per deployment.

Total compute budget. Full development required ∼2,400 GPU-hours across ∼700 experiments (architecture search, ablation studies, multi-seed validation, DA baselines, HiRID experiments). Minimum reproduction of the best configuration per task (all five EHR tasks plus five AdaTime

datasets) on an A6000: ∼74 GPU-hours. Hardware used across the project: A6000-48GB (primary headline), RTX 3090-24GB, L40S-48GB, A100-40GB, and V100S-32GB.

AdaTime. Per-dataset training at a single seed: ∼4 h for the full adapter across 10 cross-domain scenarios, ∼1.5 h for the tiny adapter. The full three-seed benchmark across five datasets takes ∼60 GPU-hours (full) or ∼22 GPU-hours (tiny) on a single A6000.

Licences for cited third-party assets. The YAIB benchmark is released under Apache 2.0 (van de Water et al., 2024); AdaTime is released under MIT (Ragab et al., 2023a). Source code dependencies (PyTorch, PyTorch Lightning, gin-config, polars, scikit-learn, NumPy, SciPy, Matplotlib, statsmodels) are used under their respective open-source licences (BSD-3, BSD-3, Apache 2.0, MIT, BSD-3, BSD-3, BSD-3, BSD, BSD). Datasets (MIMIC-IV, eICU-CRD, HiRID) are accessed via PhysioNet credentialed-access data-use agreements; no licence is granted to redistribute raw data. The released adapter code in this submission is open-sourced under the MIT licence as stated in the repository `LICENSE` file shipped with the supplementary code archive.

A.3 End-to-end DA baselines

AdaTime end-to-end baselines. Ragab et al. (2023a) benchmark eleven state-of-the-art end-to-end DA methods: DANN, Deep CORAL, CDAN, DSAN, HoMM, MMDA, CoDATS, AdvSKM, DIRT-T, CoTMix, and SASA. We also run CLUDA (Ozyurt et al., 2023) and RAINCOAT (He et al., 2023) under the same protocol.

A.4 Auxiliary-loss full definitions

The main-body §3.3 introduces six loss terms grouped under three forces (task-grounding, latent-shape, and adapter-output regularisation). Each term receives a one-line gloss in the main text. This appendix gives the full mathematical definition of the three terms whose form is non-obvious from the gloss: the target-task auxiliary head $\mathcal{L}_{\text{tgt}}$, the linear probe on the adapter latent $\mathcal{L}_{\text{lp}}$, and the latent MMD alignment term $\mathcal{L}_{\text{align}}$. We use the paper convention throughout: f_S is the frozen source-domain predictor (trained on MIMIC-IV); $x_T \in \mathcal{X}_T$ is a target-domain input (eICU or HiRID); $x_S \in \mathcal{X}_S$ is a source-domain input; g_θ is the input adapter; E_ϕ is the adapter encoder; $z_T = E_\phi(x_T)$ and $z_S = E_\phi(x_S)$ are encoder latents in $\mathbb{R}^d$; $y_T \in \mathcal{Y}$ is the target-domain label.

A.4.1 $\mathcal{L}_{\text{tgt}}$: target-task auxiliary head

Let $h_{\text{aux}} : \mathbb{R}^{T \times F_{\text{dyn}} \times d} \to \mathcal{Y}$ be a small MLP with a single hidden layer of width 64 and ReLU activation, followed by an output layer whose shape matches the output of f_S (a single logit for binary classification tasks; a single scalar for regression tasks). The auxiliary head is attached to the pooled encoder latent during training only and is parametrised independently of f_S and g_θ. For per-stay tasks (Mortality, KF) a mean pool over the time axis is applied before the head; per-timestep tasks evaluate h_{aux} at every timestep. $\mathcal{L}_{\text{recon}}$ is computed against the dynamic-feature component X_{val} of the input tuple defined in Eq. 1. The loss is the task-appropriate supervised criterion,

$$\mathcal{L}_{\text{tgt}} = \begin{cases} \text{CE}\big(h_{\text{aux}}(g_\theta(x_T)),\, y_T\big) & \text{(classification)}, \\ \text{MSE}\big(h_{\text{aux}}(g_\theta(x_T)),\, y_T\big) & \text{(regression)}, \end{cases} \tag{3}$$

where CE denotes binary cross-entropy on logits and MSE denotes mean squared error.

Training-only. h_{aux} never feeds f_S at any point in the pipeline. At inference, h_{aux} is discarded and the deployed system is exactly $f_S(g_\theta(x_T))$.

Purpose. The term $\mathcal{L}_{\text{tgt}}$ provides a non-vanishing gradient signal when the decision surface of f_S is locally flat with respect to $g_\theta(x_T)$, for example when f_S's logits saturate near 0 or 1 on mortality classification, so that $\nabla_{g_\theta(x_T)} f_S \approx 0$ and the through-f_S task gradient fails to update θ. Routing a parallel gradient through the auxiliary head bypasses this saturation while leaving f_S unchanged.

A.4.2 $\mathcal{L}_{\text{lp}}$: LINEAR PROBE ON ADAPTER LATENT

Let $p_\xi : \mathbb{R}^d \to \mathcal{Y}$ be a linear probe (a single affine map; no hidden layer) attached to the encoder output $z_T = E_\phi(x_T)$ of the adapter. The loss matches the task type,

$$\mathcal{L}_{\text{lp}} \;=\; \begin{cases} \mathrm{CE}\big(p_\xi(E_\phi(x_T)),\, y_T\big) & \text{(classification)},\\ \mathrm{MSE}\big(p_\xi(E_\phi(x_T)),\, y_T\big) & \text{(regression)}. \end{cases} \tag{4}$$

Training-only. Like h_{aux}, the probe p_ξ never feeds f_S and is discarded at inference; the deployed system remains $f_S(g_\theta(x_T))$.

Purpose. The probe loss keeps $z_T = E_\phi(x_T)$ linearly informative about y_T. Without this constraint, the encoder is free to collapse representational geometry (for instance, by mapping every input to a narrow region of latent space that the cross-attention block can still exploit through high-rank decoder weights), which empirically destabilises the retrieval and cross-attention pathways. Forcing linear separability of z_T is a cheap structural regulariser on encoder geometry that does not constrain the decoder.

A.4.3 $\mathcal{L}_{\text{align}}$: INPUT-SPACE MMD ALIGNMENT

For a target-domain minibatch $\{x_T^{(i)}\}_{i=1}^{B_T}$ pushed through the adapter, and a source-domain minibatch $\{x_S^{(j)}\}_{j=1}^{B_S}$ drawn uniformly at random from $\mathcal{X}_S$, the alignment loss is the squared Maximum Mean Discrepancy between the two *input-space* point clouds, taken over per-timestep feature vectors at every unpadded position:

$$\mathcal{L}_{\text{align}} \;=\; \mathrm{MMD}^2\Big(\,\big\{\, g_\theta(x_T^{(i)})_\tau \;:\; M_{\text{pad},\tau}^{(i)} = 0 \,\big\}_{i,\tau},\; \big\{\, (x_S^{(j)})_\tau \;:\; M_{\text{pad},\tau}^{(j)} = 0 \,\big\}_{j,\tau}\Big), \tag{5}$$

where each masked sample contributes a separate F_{dyn}-dimensional point per valid timestep and the kernel never sees the encoder latent. With $N_T = \sum_i T_i$ and $N_S = \sum_j T_j$ the total numbers of unpadded target/source timesteps in the minibatch, and $u^{(i,\tau)} \in \mathbb{R}^{F_{\text{dyn}}}$, $v^{(j,\tau)} \in \mathbb{R}^{F_{\text{dyn}}}$ the resulting per-timestep vectors, the loss is estimated using the standard unbiased U-statistic:

$$\begin{aligned} \widehat{\mathrm{MMD}}_u^2 \;=\; & \frac{1}{N_T(N_T-1)} \sum_{(i,\tau)\neq(i',\tau')} k\big(u^{(i,\tau)}, u^{(i',\tau')}\big) \;+\; \frac{1}{N_S(N_S-1)} \sum_{(j,\tau)\neq(j',\tau')} k\big(v^{(j,\tau)}, v^{(j',\tau')}\big) \\ & -\; \frac{2}{N_T N_S} \sum_{(i,\tau),(j,\tau')} k\big(u^{(i,\tau)}, v^{(j,\tau')}\big). \end{aligned} \tag{6}$$

$k(\cdot,\cdot)$ is a multi-bandwidth Gaussian kernel $k(u,v) = \sum_{b=1}^{B} \exp\big(-\|u-v\|^2/(2\sigma_b^2)\big)$ whose bandwidths $\{\sigma_b\}$ are set by the median heuristic over the pooled batch of pairwise distances at each step.

Source minibatch role. Source inputs x_S enter the alignment kernel as raw per-timestep feature vectors and are never passed through g_θ or f_S. The frozen predictor f_S is evaluated only on $g_\theta(x_T)$.

Purpose. As g_θ is optimised by the through-f_S task loss, its output stream can drift in input space such that the marginal distribution of adapted target inputs separates from the source-input distribution that f_S was trained to consume, even when the task loss on labelled target windows is small. This is the input-space analogue of the latent-drift failure mode reported for adapter-only objectives. $\mathcal{L}_{\text{align}}$ pulls the marginal of $g_\theta(x_T)$ toward the marginal of x_S in $\mathbb{R}^{F_{\text{dyn}}}$ without requiring access to source-domain labels.

A.4.4 $\mathcal{L}_{\text{range}}$: PHYSIOLOGICAL ENVELOPE PENALTY

Let $[m_f, M_f]$ denote the per-feature physiological envelope on the source training corpus, set to the 0.5 and 99.5 percentiles of $X_{\text{val},S}$ in z-scored space (clipped to $[-6, 6]$ for any feature whose tail exceeds the 0.5% trim threshold). For an adapter output $\tilde{x} = g_\theta(x_T) \in \mathbb{R}^{T \times F_{\text{dyn}}}$ the range loss is the squared one-sided hinge:

$$\mathcal{L}_{\text{range}} \;=\; \frac{1}{T \cdot F_{\text{dyn}}} \sum_{t,f} \Big[\max(0,\, m_f - \tilde{x}^{t,f})^2 \;+\; \max(0,\, \tilde{x}^{t,f} - M_f)^2\Big]. \tag{7}$$

Values inside the envelope contribute zero; the squared form gives a smooth gradient at the boundary. The penalty is evaluated on adapter outputs only; padded timesteps are masked out before the average.

A.4.5 FROZEN-INTEGRITY RESTATEMENT

All auxiliary heads (h_{aux}, p_ξ) and source-domain minibatches enter only the training loss; at inference the system is exactly $f_S(g_\theta(x_T))$, with f_S bit-identical to the certified source-domain predictor (see §3.1).

A.5 CROSS-ATTENTION BLOCK (FULL FORM)

Let $\bar{z}_T^t \in \mathbb{R}^d$ be the per-timestep query obtained by per-feature mean-pooling the encoder latent $z_T \in \mathbb{R}^{T \times F_{\text{dyn}} \times d}$ along the feature axis at timestep t, and let $\mathcal{N}_k(\bar{z}_T^t) \in \mathbb{R}^{d \times k}$ denote the column-stacked matrix of the k nearest source-window descriptors. Each of the L stacked cross-attention blocks computes:

$$z_{\text{cross}}^t \;=\; \text{softmax}\!\left(\frac{(W_Q\,\bar{z}_T^t)^\top\,(W_K\,\mathcal{N}_k(\bar{z}_T^t))}{\sqrt{d}}\right)\,(W_V\,\mathcal{N}_k(\bar{z}_T^t))^\top \;+\; \bar{z}_T^t, \tag{8}$$

producing a $1 \times k$ attention score row that is mixed against the value matrix; this is followed by causal self-attention and a feed-forward network, both with residual connections and pre-LayerNorm.

B DATASETS AND TASKS

Cohorts. All EHR experiments use the YAIB (Yet Another ICU Benchmark) harmonisation pipeline (van de Water et al., 2024), which exposes three ICU cohorts under a single feature schema: MIMIC-IV (predictor-training distribution), eICU, and HiRID (adaptation domains). The five tasks (Mortality24, AKI, Sepsis, LoS, KF) inherit YAIB's cohort definitions and per-task labels. Table 6 summarises the three databases, and Table 7 summarises the five clinical tasks.

Table 6: ICU databases used in this study. All data is preprocessed via the YAIB benchmark (van de Water et al., 2024) with standardized cohort definitions, feature extraction, and train/validation/test splits.

Dataset	Role	Stays	Hosp.	Country	Source
MIMIC-IV	Source	73,181	1	US	Academic med. ctr.
eICU	Target	200,859	208	US	Telehealth net.
HiRID	Target 2	33,905	1	Switz.	Univ. hospital

Table 7: Clinical prediction tasks. MI = missing indicator features (binary flags for whether a value was measured or forward-filled). Per-timestep tasks produce a prediction at every hourly timestep; per-stay tasks produce a single prediction per ICU stay.

Task	Type	Label	Granularity	Feature Dim	Pos. Rate
Mortality24	Classification	ICU death within 24 h	Per-stay	96 (48 + 48 MI)	∼5.5%
AKI	Classification	KDIGO Stage ≥1	Per-timestep	100 (48 + 48 MI + 4 static)	∼12%
Sepsis	Classification	Sepsis-3 criteria	Per-timestep	100 (48 + 48 MI + 4 static)	∼1.1%
LoS	Regression	Hours remaining	Per-timestep	52 (48 + 4 static)	N/A
KF	Regression	Serum creatinine (mg/dL)	Per-stay	292 (48 + 48 MI + 192 gen + 4 static)	N/A

The 48 clinical features include vital signs (heart rate, blood pressure, SpO_2, temperature, respiratory rate) and laboratory values (metabolic panels, blood gases, liver enzymes, cardiac markers, coagulation studies). The 4 static features encode patient demographics (age, sex, height, weight). KF uses 192 additional generated features: cumulative minimum, maximum, mean, and count statistics for each of the 48 clinical features, which must maintain internal consistency after adaptation. LoS omits missing indicators because the YAIB LoS LSTM was trained on a 52-feature schema.

Per-feature triplet $(X_{\textbf{val}}, X_{\textbf{miss}}, \Delta t)$**.** Each dynamic clinical variable arrives as a triple of channels at every timestep: the numerical value X_{val} (last-observation-carried-forward imputed by YAIB and

z-scored on the source training split, per feature), the binary observation-indicator X_{miss} (1 if the value at that step was observed before imputation, 0 otherwise), and the inter-step delay $\Delta t_\tau = t_\tau - t_{\tau-1}$ (with $\Delta t_1 = 0$, computed from the YAIB sequence index inside the adapter forward pass). The triplet is stacked along a new last axis to give an $(B, T, F_{\text{dyn}}, 3)$ tensor and projected to encoder latent dimension d by a shared Linear$(3 \to d)$ map.

Static covariates and FiLM. The four patient-level static covariates (x_{static} = age, sex, height, weight) feed an MLP whose outputs are per-layer scale/shift pairs (γ_l, β_l) applied via FiLM (Perez et al., 2018) inside every encoder and decoder block; they are never concatenated into the dynamic stream on the adapter side. Whether the *frozen predictor* sees broadcast static columns is independently determined by the certified predictor's preprocessor (Table 8); on AKI/Sepsis/LoS/KF the YAIB tensor consumed by f_S has the four statics broadcast across time, while Mortality24's predictor was trained on dynamics-only.

Predictor-side reconstitution. The adapter outputs an adapted dynamic stream $\tilde{X}_{\text{val}} \in \mathbb{R}^{T \times F_{\text{dyn}}}$. Before f_S consumes it, the schema resolver writes $\tilde{X}_{\text{val}}$ back into the predictor's expected layout, leaving the unchanged X_{miss} columns and any broadcast static columns in place. For Kidney Function the predictor was trained with $192 = 48 \times 4$ generated cumulative features (per-feature running min, max, count, mean); after adaptation these are recomputed from $\tilde{X}_{\text{val}}$ inside the resolver so that the predictor sees self-consistent feature engineering rather than stale cumulatives derived from the un-adapted source values.

Table 8: Per-task input-channel composition. $F_{\text{dyn}} = 48$ for all five EHR tasks; the adapter operates on F_{dyn}-dimensional dynamic streams plus a 4-dimensional FiLM static vector. The frozen-predictor input dimension F_{pred} is task-specific because each pretrained YAIB LSTM was trained with a different combination of {missing-indicator, broadcast statics, generated cumulatives}. Composition columns are read from the per-task pretrained-LSTM `train_config.gin` files. AdaTime is included for contrast: same adapter code, but $X_{\text{miss}} \equiv 0$, $x_{\text{static}} \equiv 0$, $\Delta t_\tau \equiv \tau$ (the delay channel is repurposed as a positional index in place of the unit step, since AdaTime windows are uniformly sampled).

Task	F_{dyn}	MI in F_{pred}	Generated	Static (broadcast)	F_{pred}
Mortality24	48	48	0	0	96
AKI	48	48	0	4	100
Sepsis	48	48	0	4	100
LoS	48	0	0	4	52
Kidney Function	48	48	192	4	292
AdaTime (HAR)	9	0	0	0	9
AdaTime (HHAR/WISDM)	3	0	0	0	3
AdaTime (SSC/MFD)	1	0	0	0	1

Why F_{pred} varies across tasks. The five MIMIC YAIB LSTM checkpoints were trained under different YAIB preprocessor flags:

- **Mortality24**: `preprocess.use_static = False` (dynamics-only).
- **AKI / Sepsis**: `use_100_features = True` with statics broadcast.
- **LoS**: `use_missing_indicator = False` (dynamics + statics, no MI).
- **Kidney Function**: `generate_features = True` with `use_100_features = True` (yielding 48 dyn + 48 MI + 192 generated cumulatives + 4 statics).

The adapter is unaware of these per-task differences (they are absorbed by `SchemaResolver` at the predictor boundary) and operates on the same (F_{dyn}=48, F_{static}=4) interface across all five tasks.

Input pipeline (YAIB). Raw EHR records contain irregular, sparse event-level measurements (lab draws at hours apart, vitals at varying frequencies). The YAIB pipeline converts these into the hourly feature matrix consumed by the frozen predictor through five steps: (i) hourly binning: measurements in each one-hour bin are aggregated (last value for vitals, mean for labs); (ii) forward-fill imputation: unmeasured features at hour t inherit the last observed value, producing long flat runs for sparsely measured labs; (iii) missing indicators (MI): a parallel binary feature flags whether

each value at hour t was newly measured or forward-filled, letting the LSTM distinguish fresh from stale readings; (iv) static broadcast: the 4 demographic features are repeated at every timestep; (v) cumulative statistics (KF only): running min/max/mean/count of each vital are concatenated, adding 192 features that must remain internally consistent after adaptation. The resulting stay is a $T \times F$ matrix with $F \in \{52, 96, 100, 292\}$. Cross-domain normalization (§3.3) is applied on top. The adapter's empirical tendency to disrupt forward-fill runs on sparsely measured labs is documented in Appendix I.

C Extended Quantitative Results

This section collects the per-task tables that the main body cites for AUCPR, regression metrics, HiRID transfer, multi-seed variance, and the Phase-1 ablation. Every cell here is averaged across five training seeds for source-assisted eICU pools and three training seeds for HiRID transfer; spread columns and bootstrap CIs use 500 resamples on the held-out target test split.

C.1 AUCPR Results (MIMIC-IV → eICU)

Table 9 reports AUCPR for the three classification tasks. Adaptation produces a strictly larger improvement on AUCPR than on AUROC for AKI ($+15.3$ AUCPR vs $+5.21$ AUROC), reflecting precision-recall gains at clinically operating points where the positive class is the rate-limiting factor.

Table 9: AUCPR results ($\times 100$) for classification tasks across five seeds. Bootstrap 95% CIs (500 replicates).

	Frozen Baseline	**Pilot** [95% CI]	**Δ**
Mortality	29.66	33.85 [31.73, 36.46]	+4.19
AKI	56.79	72.06 [71.43, 72.71]	+15.27
Sepsis	2.97	4.79 [4.31, 5.51]	+1.82

C.2 Regression Extended Results (MIMIC-IV → eICU)

Table 10 gives the full multi-metric breakdown for the two regression tasks averaged across five seeds: MAE, RMSE, and R^2. We report all three because MAE is the primary clinical metric (median error in hours / mg dL), RMSE is sensitive to tail-of-distribution mistakes, and R^2 summarises how much variance in the target is explained.

Table 10: Regression results across multiple metrics (five seeds). Bootstrap 95% CIs (500 replicates). MAE-z and RMSE-z: z-scored target (lower is better, ↓); R^2: higher is better (↑). Native-unit MAE values for LoS (hours) and KF (mg/dL) appear in Table 1.

Task	**Metric**	**Baseline**	**Pilot** [95% CI]	**Δ**
LoS	MAE-z (↓)	0.2528	0.2205 [0.2190, 0.2216]	−0.0323
LoS	RMSE-z (↓)	0.3124	0.3062 [0.3046, 0.3078]	−0.0062
LoS	R^2 (↑)	0.1132	0.1481 [0.1427, 0.1548]	+0.0349
KF	MAE-z (↓)	0.0330	0.0236 [0.0229, 0.0242]	−0.0094
KF	RMSE-z (↓)	0.0600	0.0471 [0.0444, 0.0495]	−0.0129
KF	R^2 (↑)	0.7488	0.8456 [0.8304, 0.8590]	+0.0968

Denormalised MAE values for LoS and KF are reported in §4.2. We attribute the remaining KF gap to low kidney-function variability over short stays and to KF's reliance on 192 cumulative-statistic features that must remain internally consistent after adaptation. KF R^2 improves from 0.749 to 0.846, nearly doubling the explained variance in serum creatinine predictions. Dropping the fidelity loss removes fidelity gradient dominance in KF's 292-feature cumulative-statistics regime.

C.3 MIMIC-IV → HiRID Extended Results

Table 11 reports the same five tasks under a second target distribution (Swiss-ICU HiRID) at three seeds. Each row is a strict port of the best eICU recipe with only the data paths swapped.

Table 11: MIMIC-IV → HiRID results across three seeds. All five tasks statistically significant.

Task	**Metric**	**Δ vs Frozen** mean ± std (three seeds)	p
Mortality	AUROC	$+0.0589 \pm 0.0059$	<0.001
Mortality	AUCPR	$+0.0502 \pm 0.0084$	<0.001
AKI	AUROC	$+0.0753 \pm 0.0029$	<0.001
AKI	AUCPR	$+0.1471 \pm 0.0058$	<0.001
Sepsis	AUROC	$+0.0689 \pm 0.0113$	<0.001
Sepsis	AUCPR	$+0.0525 \pm 0.0046$	<0.001
LoS	MAE	-0.0766 ± 0.0048 z-score (≈ -12.9 h)	<0.001
KF	MAE	-0.0017 ± 0.0002	<0.001

HiRID results show that the adapter works across target domains with very different data collection practices (Swiss vs. US ICUs). AKI and Sepsis improvements on HiRID ($+7.5$ and $+6.9$ AUROC) exceed those on eICU ($+5.1$ and $+4.9$); we hypothesise that greater domain distance provides more signal for adaptation. All five tasks are statistically significant; the smaller magnitude on KF reflects HiRID's smaller cohort and the cumulative-statistic feature regime.

C.4 Seed Variance (MIMIC-IV → eICU)

Table 12 summarises five-seed variance for the headline cells. Spread is reported as max − min across the five seeds and is a stricter summary than σ for samples of size 5.

Table 12: Multi-seed variance across five seeds. Classification: mean ΔAUROC ($\times 100$); regression: mean ΔMAE on the task's native scale. Spread = max − min across seeds.

Task	**Adapter**	**Mean Δ**	**Spread**
Mortality	classification	$+4.85$	0.21
AKI	classification	$+5.21$	0.43
Sepsis	classification	$+4.72$	0.85
LoS	regression	-0.0323	0.0029
KF	regression	-0.0094	0.0012

AKI, Mortality, and LoS show low variance (five-seed spreads below one-third of the mean gain), well within noise. Sepsis exhibits wider spread driven by its 1.1% positive rate: a single seed can shift the AUROC by up to 0.72 points through resampling of the small positive set; this variance is inherent to the task rather than method-specific, as DA baselines show comparable variance. KF has the tightest proportional spread; the remaining gap to eICU-native is systematic rather than stochastic.

C.5 Phase-1 ablation: per-task deltas

Both pretraining phases are necessary: removing Phase 1 destabilises every clinical task across seeds (Sepsis collapses, the other four widen variance two-to-forty-fold), and disabling Phase 2 erases or sign-flips the adaptation gain on every clinical task. Per-task deltas in Table 13.

Table 13 backs the §3.3 claim that Phase 1 reconstruction pretraining is the spine of adaptation. Two complementary cells isolate the contribution of each phase: *Phase-1-only* (Phase 2 disabled; predictor is fed the autoencoder's reconstruction without any task-supervised adaptation) and *no-pretrain* (Phase 1 disabled, Phase 2 unchanged; the no-pretrain ablation cell from the component-ablation matrix in Appendix F). Per-seed result JSONs are included in the released code snapshot; all cells are reported as three-seed pools.

Table 13: Phase-1 ablation deltas relative to the frozen MIMIC-IV LSTM baseline (eICU → MIMIC-IV). Classification rows are reported as ΔAUROC (×100) to match Tables 1 and 18; regression rows are ΔMAE-z. Negative AUROC and positive MAE-z indicate degradation. *Phase-1-only* runs the autoencoder without any task-supervised Phase 2; *no-pretrain* runs Phase 2 from random initialisation. Phase-1-only collapses dense-label classification (AKI, sepsis); no-pretrain is structurally negative on sparse classification (sepsis). Removing either Phase 1 or Phase 2 reduces the adaptation gain on every task: through mean collapse (Sepsis no-pretrain; AKI Phase-1-only at -11 AUROC points) or through marked seed instability (no-pretrain seed-σ rises $2.6\times$ to $40\times$ over the full pipeline). Mortality / AKI no-pretrain show wide seed-to-seed spread, so they are not classed as structural drops.

Task	Metric	Phase-1-only (Δ)	No-pretrain (Δ)	n
Mortality	ΔAUROC (×100)	-0.09 ± 0.19	-1.90 ± 8.40 (wide)	3 / 3
AKI	ΔAUROC (×100)	$\mathbf{-11.22 \pm 0.07}$	-0.21 ± 6.06 (wide)	3 / 3
Sepsis	ΔAUROC (×100)	-3.75 ± 0.14	$\mathbf{-8.48 \pm 0.35}$	3 / 3
LoS	ΔMAE-z	-0.0010 ± 0.0029	-0.0051 ± 0.0150	3 / 3
KF	ΔMAE-z	$+0.0052 \pm 0.0002$	-0.0035 ± 0.0037	3 / 3

D Architecture-Agnostic Transfer

To test whether the adapter learns a domain-invariant input mapping rather than LSTM-specific artifacts, we apply an LSTM-trained adapter *zero-shot* to frozen GRU and TCN architectures with no retraining. Each cell is reported as mean ± standard deviation across three training seeds.

Table 14: Architecture-agnostic transfer: LSTM-trained adapter applied zero-shot to frozen GRU and TCN predictors. Improvement over each architecture's frozen baseline ($\Delta \times 100$, mean ± std across three seeds). For classification tasks the metric is AUROC; for LoS and KF it is the (negated) z-scored MAE, where more negative is better. Percentage in parentheses is the fraction of the LSTM gain retained on the same task.

Source	**Task**	**LSTM Δ**	**GRU $\Delta \pm \sigma$ (%)**	**TCN $\Delta \pm \sigma$ (%)**
eICU	Mortality	+4.91	+3.00 ± 1.02 (61%)	+2.83 ± 0.74 (58%)
	AKI	+5.14	+2.70 ± 0.38 (53%)	+2.75 ± 0.25 (53%)
	Sepsis	+4.88	+3.13 ± 0.46 (64%)	+3.58 ± 0.79 (73%)
	LoS (ΔMAE z)	−3.23	−2.38 ± 0.19 (74%)	−3.32 ± 0.14 (103%)
	KF (ΔMAE z)	−0.68	−0.67 ± 0.18 (98%)	−0.42 ± 0.05 (62%)
HiRID	Mortality	+5.89	+4.04 ± 0.63 (69%)	+3.88 ± 0.32 (66%)
	AKI	+7.53	+4.76 ± 0.67 (63%)	+5.61 ± 0.67 (75%)
	Sepsis	+6.89	+3.94 ± 0.17 (57%)	+5.20 ± 0.37 (75%)

All 16 task–architecture cells show positive transfer in the direction of the LSTM gain, with retention ranging from 53% to 103% of the LSTM improvement. The strongest retention occurs for TCN on eICU LoS, where the transferred adapter slightly exceeds the LSTM gain (103%), and for GRU on eICU KF (98%); both regression tasks therefore preserve the adapter benefit nearly unchanged across architectures. Across all 16 cells the across-seed standard deviation stays below 0.011, indicating that the input-space correction is consistent rather than seed-fragile. Across the 16 cells the per-cell retention has mean $\approx 66\%$ and median 65% of the LSTM gain; ten of the sixteen cells fall in the 60–75% band, consistent with a shared input-space correction modulated by predictor-family residual sensitivity. The adapter resolves distributional mismatches at the input level (e.g., forward-fill artefacts), so its output benefits predictors trained on the source distribution whether they use recurrent or convolutional gating.

E Reduced-Capacity (Tiny) Adapter

We test whether the adapter can be scaled below the frozen predictor's parameter count while retaining the gain. All tiny adapters use a reduced d_{model}, d_{latent}, and depth; n_{cross} is preserved. Tables 15 and 16

compare the tiny adapter to the per-task headline cell of Table 1 across three seeds, on the EHR tasks and on the AdaTime suite respectively. Table 17 gives the per-cell capacity sweep that backs Threat 3 in §4.3.

Table 15: Tiny adapter approximately matched to the frozen predictor's parameter count (per-task ratios in the table body; LoS at $1.06\times$, the other four tasks at $\leq 1\times$) across three seeds. Clinical tasks: ΔAUROC or ΔMAE. Ratio is adapter parameters divided by frozen-predictor parameters. The "vs headline" column reports the tiny-adapter delta as a fraction of the per-task headline cell of Table 1; "matches" denotes within multi-seed noise of that cell.

Task	Tiny params	Ratio	Δ	vs headline
Mortality	416K	$0.86\times$	$+0.0475 \pm 0.0013$	matches
AKI	1.43M	$1.00\times$	$+0.0409 \pm 0.0063$	78%
Sepsis	161K	$0.95\times$	$+0.0194 \pm 0.0070$	41%
LoS	1.42M	$1.06\times$	-0.0305 ± 0.0003	95%
KF	552K	$0.98\times$	-0.0086 ± 0.0004	matches

Table 16: Tiny adapter on AdaTime across three seeds with `pretrain_epochs=0`. The "vs headline" column reports tiny-adapter Macro-F1 minus the per-dataset headline cell of Table 1 (full-capacity adapter under the same recipe; reference values listed in Table 17). Cells whose tiny-adapter Macro-F1 exceeds the headline cell are in **bold**.

Dataset	Tiny params	Ratio	MF1 $\pm$ std	vs headline
HAR	196K	$0.98\times$	$\mathbf{0.9438 \pm 0.000}$	$\mathbf{+0.31}$
HHAR	196K	$0.98\times$	$\mathbf{0.9093 \pm 0.003}$	$\mathbf{+3.89}$
WISDM	196K	$0.98\times$	$\mathbf{0.7595 \pm 0.016}$	$\mathbf{+5.68}$
SSC	196K	$0.99\times$	0.6635 ± 0.001	$\approx$ tie
MFD	68K	$\mathbf{0.34\times}$	$\mathbf{0.9696 \pm 0.002}$	$\mathbf{+0.88}$

Tiny adapters match or exceed the full adapter on 3/5 clinical task–benchmark cells (Mortality, LoS, KF) and exceed on all five AdaTime datasets; on MFD, a $0.34\times$ adapter (68K parameters) achieves 0.9696 Macro-F1, the best MFD result we have observed on this benchmark. This indicates that the adapter's effective capacity is substantially lower than the frozen predictor in most settings, and that the gains reported in the main body do not depend on over-parameterisation.

E.1 Per-cell capacity sweep

Table 17 backs Threat 3 in §4.3 (the four trimmed knobs are listed there): the tiny variant approximately matched to the frozen predictor (per-task ratios in Table 17) is reported on every (task $\times$ benchmark) cell, three seeds per cell.

F Component Ablation Study

We ablate the main components of the adapter across all five EHR tasks; the cells are three-seed means with † marking five-seed cells matched to Table 1. Each row modifies exactly one setting from the control configuration; the per-row ablation codes are defined in the caption and back the playbook rules in Appendix K.

Pretraining is essential for EHR tasks. Skipping Phase 1 collapses adaptation by up to $\sim$11 AUROC points relative to the full method, indicating Phase 1 is required for stable transfer. On AdaTime, in contrast, top configurations use `pretrain_epochs=0`: the pre-trained CNN backbone already provides sufficient latent structure. Removing the memory bank is within multi-seed noise on AKI (no-retrieval $+5.21 \pm 0.16$ vs the full method $+4.97 \pm 0.52$), within multi-seed noise on Sepsis (no-retrieval $+4.72 \pm 0.40$ vs full $+4.75 \pm 0.38$), and slightly harmful on Mortality (no-retrieval $+4.20 \pm 0.37$ vs full $+4.68 \pm 0.26$). On both regression tasks removing retrieval is beneficial: LoS reaches -0.0247 ± 0.0033 versus full -0.0241 ± 0.0029, and KF reaches -0.0066 ± 0.0003 versus full -0.0030 ± 0.0025 (Table 18). Removing fidelity is catastrophic for Sepsis ($\Delta = -0.042 \pm 0.022$

Table 17: Per-cell capacity-sweep across three seeds. Ratio is adapter parameters divided by frozen-predictor parameters. EHR rows show ΔAUROC or ΔMAE-z relative to the frozen baseline. AdaTime rows show absolute Macro-F1 alongside the headline-cell value. The tiny adapter matches or exceeds the headline cell on $8/10$ cells; the two exceptions (AKI, Sepsis) are tasks where the headline adapter retains a $> 2\times$-predictor capacity advantage that the tiny variant cannot replicate.

Cell	Tiny variant	Params	Ratio	Tiny vs headline
EHR (5 tasks, Δ relative to frozen MIMIC-IV LSTM)				
Mortality	`mort_cap_S`	415,938	0.86×	$+0.0475 \pm 0.0013$ AUROC (matches headline)
AKI	`aki_cap_T`	1,425,538	1.00×	$+0.0409 \pm 0.0063$ AUROC (78% of headline)
Sepsis	`sepsis_cap_T`	160,818	0.95×	$+0.0194 \pm 0.0070$ AUROC (41% of headline)
LoS	`los_cap_T`	1,419,394	1.06×	-0.0305 ± 0.0003 MAE-z (95% of headline)
KF	`kf_cap_T`	551,826	0.98×	-0.0086 ± 0.0004 MAE-z (matches headline)
AdaTime (5 datasets, absolute Macro-F1)				
HAR	`cap_T_drop10`	∼196K	0.98×	0.9438 (beats 0.9407 headline)
HHAR	`cap_T_p0`	195,658	0.98×	0.9093 ± 0.003 (beats 0.8704 headline)
WISDM	`cap_T_p0`	195,658	0.98×	0.7595 ± 0.016 (beats 0.7027 headline)
SSC	`cap_T`	195,514	0.99×	0.6635 ± 0.001 (matches 0.6624 headline)
MFD	`cap_T`	195,514	0.99×	0.9626 ± 0.002 (beats 0.9608 headline)

Table 18: Per-task component ablation. Each row removes exactly one component from the full retrieval adapter and is named for that removal. Classification tasks are reported in ΔAUROC×100, regression tasks in ΔMAE on z-scored targets. Cells marked with † are five-seed means and match the corresponding per-task headline ablation cell in Table 1 (Mortality: No fidelity loss; AKI: No retrieval; Sepsis: No retrieval; LoS: No MMD alignment); KF's headline cell is the source-pretrain leaf reported in Table 25 and is not a row of this table. All unmarked cells are three-seed means.

Configuration	Mortality ΔAUROC×100	AKI ΔAUROC×100	Sepsis ΔAUROC×100	LoS ΔMAE (z)	KF ΔMAE (z)
Full method (control)	$+4.68 \pm 0.26$	$+4.97 \pm 0.52$	$+4.75 \pm 0.38$	-0.0241 ± 0.0029	-0.0030 ± 0.0025
No retrieval	$+4.20 \pm 0.37$	$+5.21 \pm 0.16$ †	$+4.72 \pm 0.40$ †	-0.0247 ± 0.0033	-0.0066 ± 0.0003
No MMD alignment	$+4.51 \pm 0.34$	$+4.99 \pm 0.18$	$+4.26 \pm 0.91$	-0.0323 ± 0.0012 †	-0.0067 ± 0.0008
No target-task loss	$+3.96 \pm 0.35$	$+4.50 \pm 0.38$	$+4.74 \pm 0.37$	-0.0132 ± 0.0106	-0.0042 ± 0.0011
No fidelity loss	$+4.85 \pm 0.09$ †	$+4.83 \pm 0.96$	-4.20 ± 2.15	$+0.0026 \pm 0.0006$	-0.0051 ± 0.0040
No pretrain (skip Phase 1)	-1.90 ± 8.40	-0.21 ± 6.06	-8.48 ± 0.35	-0.0051 ± 0.0150	-0.0035 ± 0.0037
Residual output	$+4.07 \pm 0.45$	$+2.44 \pm 1.49$	$+3.36 \pm 0.48$	-0.0113 ± 0.0044	-0.0016 ± 0.0005

at n_{cross}=2; $\Delta = -0.078 \pm 0.008$ at n_{cross}=3) and mildly harmful for LoS, but beneficial for Mortality and neutral-to-beneficial for KF. Sparse-label classification requires the fidelity anchor; dense-label classification and low-noise regression can drop it. The residual-output ablation on AKI has the widest seed-to-seed spread of any cell in the matrix (2.65 AUROC points).

G Test-Time Adaptation Baselines

We evaluate five TTA methods against our adapter: T3A (Iwasawa & Matsuo, 2021), SHOT (Liang et al., 2020), TENT (Wang et al., 2021), SAR (Niu et al., 2023), and EATA (Niu et al., 2022). TTA methods operate at test time only (no training-time supervision) and typically require specific architectural features (BatchNorm for TENT/EATA; any norm-affine layer (BN/LN/GN) for SAR). All EHR cells are averaged over three training seeds; AdaTime cells follow the benchmark's standard 5-seed convention.

Clinical (LSTM). T3A and SHOT are applicable to the YAIB LSTM; TENT/EATA require BatchNorm and SAR requires a norm-affine layer (BN/LN/GN), all absent from this architecture. Three seeds, ΔAUROC vs the frozen source-only baseline: T3A Mortality -0.18, AKI -3.40, Sepsis $+0.65$; SHOT Mortality -0.58, AKI -20.55, Sepsis -1.10. All classification tasks are net-negative or near-zero (Table 19, left columns).

AdaTime (1D-CNN). The AdaTime backbone contains BatchNorm, enabling all five TTA methods. Mean Macro-F1 over three seeds across ten cross-domain scenarios per dataset, against the source-

only baseline of 65.1: T3A 69.1, SHOT 67.7 (bimodal across datasets: HAR -19.3, HHAR $+17.2$), TENT 72.6, SAR 72.5, EATA 72.5. Our adapter reaches 82.7 (Macro-F1 Δ: T3A $+4.0$, SHOT $+2.7$, TENT/SAR/EATA $+7.5/+7.4/+7.4$, ours $+17.6$). TENT, SAR, and EATA are tied within noise, reflecting the BatchNorm-affine regime ($\sim$448 updated parameters). Table 19 consolidates these numbers alongside the LSTM cells.

Table 19: Test-time adaptation baselines vs. our input-space adapter. Clinical columns report ΔAUROC ($\times 100$, $\uparrow$) on the frozen YAIB MIMIC-IV-trained LSTM (eICU target, three training seeds). AdaTime column reports mean Macro-F1 ($\times 100$, $\uparrow$) over three seeds across the five-dataset suite (HAR, HHAR, WISDM, SSC, MFD; absolute, not Δ). TENT, SAR, and EATA re-estimate BatchNorm statistics at test time and are inapplicable to the BN-free YAIB LSTM (n/a cells); SHOT exhibits dataset-level bimodality on AdaTime (HAR -19.3, HHAR $+17.2$ Macro-F1).

Method	**Mortality (ΔAUROC)**	**AKI (ΔAUROC)**	**Sepsis (ΔAUROC)**	**AdaTime MF1 (absolute)**
Source-only	0.00	0.00	0.00	65.1
T3A	-0.18	-3.40	$+0.65$	69.1
SHOT	-0.58	-20.55	-1.10	67.7
TENT	n/a	n/a	n/a	72.6
SAR	n/a	n/a	n/a	72.5
EATA	n/a	n/a	n/a	72.5
Ours	**+4.91**	**+5.14**	**+4.90**	**82.7**

Frozen-backbone TTA methods plateau $\sim$10 Macro-F1 below our adapter on AdaTime, and on the BN-free YAIB LSTM only T3A and SHOT apply at all, and both are net-negative or near-zero across Mortality, AKI, and Sepsis. Input-space adaptation requires only that gradients flow through the frozen predictor; BatchNorm-based TTA additionally requires the predictor to expose an adaptable normalisation surface, which the YAIB LSTM does not.

H CALIBRATION

Table 20 reports Brier score and ECE (10 equal-width bins, computed via the standard NumPy histogram protocol described in App. A.4) across the three-seed paired-seed pool used throughout the paper. All deltas are adaptor-minus-frozen-baseline on the held-out target test cohort (eICU or HiRID); negative values indicate improved calibration. The eICU rows are reported twice – once for the five-seed headline winner (the same cell cited for AUROC/AUCPR in Table 2) and once for the three-seed tiny-adapter re-evaluation cell (Appendix E) – to demonstrate that the calibration improvement is robust to the ablation-cell choice rather than a property of the headline-cell hyperparameters.

Comparison with DA baselines (MIMIC$\rightarrow$eICU). Deep CORAL achieves the best Mortality Brier (0.1095 vs our headline 0.1031 ± 0.0065) via explicit covariance matching, but its AUROC is lower ($+0.0374$ vs our $+0.0485$), and its AKI run did not converge. Our adapter is the only method that improves Brier consistently across all three tasks while also yielding the largest AUROC gain.

H.1 TEMPERATURE SCALING COMPOSES

Post-hoc temperature scaling (Guo et al., 2017) (a one-parameter $T > 0$ fit on validation logits) scales logits without touching any predictor parameter and is therefore fully compatible with the frozen-predictor constraint: applying the adapter and then a temperature T at inference time leaves the certified backbone bit-exact. We fit T per-seed on the eICU validation split by minimising $\mathrm{BCE}(\sigma(z/T), y)$ via L-BFGS, then apply T to the held-out eICU test logits of the headline $n{=}5$ cells (Mortality *no-fidelity*, AKI *no-retrieval*, Sepsis *no-retrieval*; 15 cell $\times$ seed fits). Two empirical facts follow.

(i) Rank-ordering is preserved. AUROC and AUCPR drift between pre-T and post-T probabilities is bit-exact zero across all 15 fits, so applying T does not alter Table 1.

(ii) Calibration improves further, but the structural effect dominates. Per-task incremental ΔECE on top of the structural gain in Table 20 is -0.013 on Mortality ($\bar{T} = 0.87 \pm 0.09$, $n{=}5$), -0.001

Table 20: Classification calibration (Brier, ECE, mean $\pm\sigma$) before and after input-space adaptation. The MIMIC-IV→eICU headline panel aggregates five paired training seeds (2222, 42, 7777, 1234, 9999); the tiny-adapter re-evaluation and MIMIC-IV→HiRID port aggregate three paired seeds each (per-panel pools in §H.2). Brier improves in mean on 9/9 cells and ECE improves in mean on 8/9 cells. Per-seed Brier sign agrees with the mean on every cell. Per-seed ECE sign agrees with the mean on every cell except AKI eICU headline ECE at n=5, where 4/5 seeds improve and s_{9999} adapted ECE (0.193) is $+0.002$ above the deterministic baseline (0.191). AKI remains the weakest task at the ECE bin scale (eICU tiny-adapter re-evaluation mean Δ ECE $= -0.0027$ with 1/3 per-seed sign; HiRID port mean Δ ECE $= +0.0036$ with 1/3 per-seed sign), reflecting that the predictor's AKI confidence histogram is already near-uniform on the chosen bin grid. The Orig columns are deterministic across adapter seeds (frozen predictor on un-adapted source data); σ is therefore reported only on the Adapted and Δ columns. All values extracted from per-seed result JSONs released with the code snapshot.

Target	Task	Br$_o$	Adapted Br	Δ Br	ECE$_o$	Adapted ECE	Δ ECE
eICU: five-seed headline winners (training seeds 2222, 42, 7777, 1234, 9999)							
eICU	AKI	.137	.117±.005	−.019±.005	.191	.185±.007	−.006±.007
eICU	Mortality	.145	.106±.010	−.039±.010	.240	.186±.025	−.054±.025
eICU	Sepsis	.184	.124±.019	−.060±.019	.322	.257±.025	−.065±.025
eICU: tiny-adapter re-evaluation (training seeds 2222, 42, 7777)							
eICU	AKI	.137	.120±.008	−.016±.008	.191	.189±.011	−.003±.011
eICU	Mortality	.145	.116±.003	−.029±.003	.240	.204±.021	−.037±.021
eICU	Sepsis	.184	.107±.010	−.077±.010	.322	.249±.014	−.073±.014
HiRID: three-seed best-cell port (training seeds 2222, 42, 7777, same ablation cell as eICU headline)							
HiRID	AKI	.204	.179±.012	−.025±.012	.271	.275±.022	+.004±.022
HiRID	Mortality	.242	.128±.024	−.114±.024	.350	.172±.025	−.178±.025
HiRID	Sepsis	.309	.167±.017	−.143±.017	.458	.279±.022	−.178±.022
Mean Δ ECE across 9 cells							−0.066

on AKI ($\bar{T} = 0.99 \pm 0.04$), and -0.033 on Sepsis ($\bar{T} = 0.73 \pm 0.04$); paired-bootstrap 95% CIs across 500 replicates ($\mathrm{RandomState}(42)$, N_{cap}=100k) are tabulated in the released summary JSON. The structural ΔECE in Table 20 is $4\times$ the post-hoc Δ on Mortality, $6\times$ on AKI, and $2\times$ on Sepsis, so input-space adaptation is the dominant calibration mechanism with temperature scaling a small composable polish on top.

The AKI cell's near-unit $\bar{T}$ is the direct empirical confirmation of Table 20's caption: AKI's confidence histogram is already near-uniform on the 10-bin grid, leaving little headroom for any monotone re-mapping of test probabilities.

H.2 Calibration seed pools and source files

Each row in Table 20 is the mean $\pm$ sample-stdev over the per-seed result JSONs released alongside the code snapshot. The eICU headline panel pools five training seeds (2222, 42, 7777, 1234, 9999); the tiny-adapter re-evaluation and HiRID port panels pool three (2222, 42, 7777). The three groups of rows correspond to:

- **eICU headline** (Mortality, AKI, Sepsis): the five training seeds under each task's headline ablation cell (Mortality no-fidelity, AKI no-retrieval, Sepsis no-retrieval; Appendix F).
- **eICU tiny-adapter re-evaluation** (Mortality, AKI, Sepsis): the same three training seeds under the matching tiny / reduced-capacity adapter (Appendix E), demonstrating that the calibration improvement is a property of input-space adaptation rather than the headline-cell hyperparameters.
- **HiRID three-seed best-cell port** (Mortality, AKI, Sepsis): the three training seeds under the eICU-equivalent ablation cell ported to HiRID source data.

The released code snapshot ships every per-seed result JSON alongside the corresponding configuration so each cell in Table 20 can be re-derived from the snapshot without re-training.

I INTERPRETABILITY: IMPUTATION VS. ACF CHANGE

The mechanism behind PILOT's gain on per-timestep classification is operational-semantics restoration, not marginal alignment. Per-feature imputation fraction, the proportion of timesteps filled by last-observation-carried-forward rather than fresh measurements, predicts the magnitude of the adaptation-induced change in lag-1 autocorrelation ($|\Delta\text{ACF}|$) on both AKI (Figure 3) and Sepsis (Figure 4).

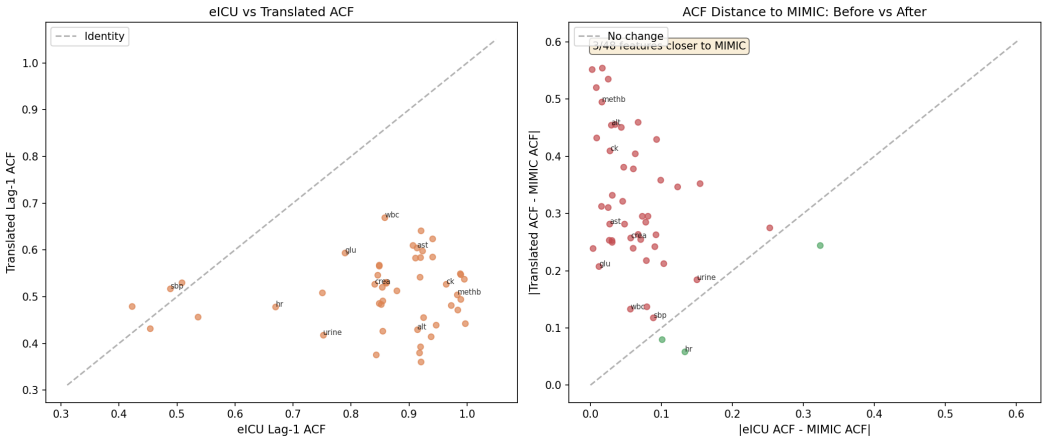

Figure 3: AKI: per-feature lag-1 autocorrelation before adaptation (x-axis, eICU) versus after adaptation (y-axis, translated). Points coloured by feature imputation fraction; the adaptation-induced change $|\Delta\text{ACF}|$ is the off-diagonal displacement of each point. Spearman rank correlation between feature imputation fraction and $|\Delta\text{ACF}|$ is ρ=0.81, $p < 10^{-9}$. Sparsely sampled laboratory values (high imputation, upper region of the colour scale) undergo the largest off-diagonal displacement; continuously monitored vitals (low imputation, lower-left region) hug the diagonal.

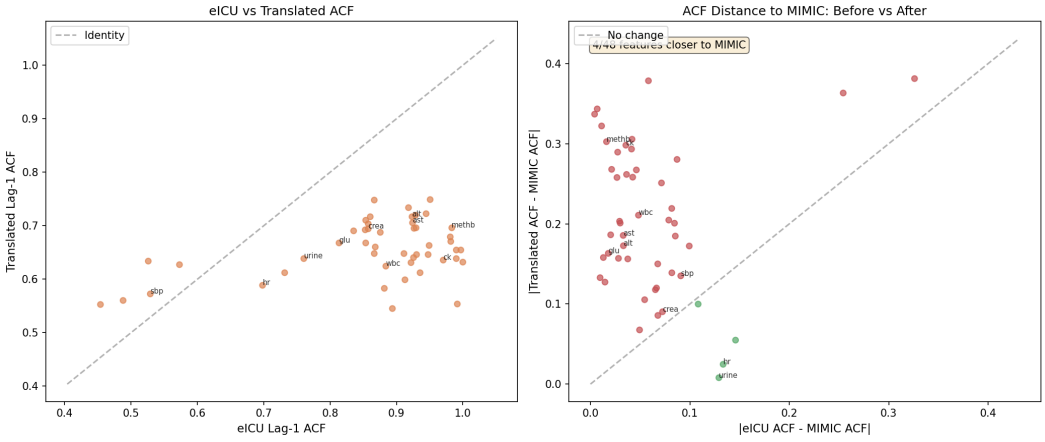

Figure 4: Sepsis: per-feature lag-1 autocorrelation before adaptation (x-axis, eICU) versus after adaptation (y-axis, translated). Points coloured by feature imputation fraction; $|\Delta\text{ACF}|$ is the off-diagonal displacement. Spearman rank correlation between feature imputation fraction and $|\Delta\text{ACF}|$ is ρ=0.74, $p < 10^{-9}$. The same imputation-driven temporal restructuring observed on AKI reproduces here, with a marginally weaker rank correlation reflecting Sepsis's sparser per-timestep label signal.

Read jointly, the two panels show a single mechanism. The Spearman rank correlation between imputation fraction and $|\Delta\text{ACF}|$ across the n=48 dynamic features is ρ=0.81 on AKI and ρ=0.74 on Sepsis, both with $p < 10^{-9}$. Sparsely sampled laboratory values populate the upper region

of both panels; creatinine, bilirubin, and lactate carry imputation fractions above 0.9 and show the largest $|\Delta\mathrm{ACF}|$. Continuously monitored vitals such as heart rate, SpO_2, and mean arterial pressure cluster in the lower-left region, with imputation fractions below 0.3 and near-zero $\Delta\mathrm{ACF}$. The adapter rewrites the temporal-correlation signature of forward-filled labs and leaves genuinely sampled channels alone; the marginally weaker rank correlation on Sepsis is consistent with its $\sim$1% per-timestep label density, since sparser supervision gives the adapter less per-feature task signal to redistribute even though the direction of the effect is identical to AKI's.

This pattern matches what the frozen MIMIC-IV-trained LSTM was trained to expect. In MIMIC-IV, a long flat run in a lab channel encodes a specific measurement cadence typical of an academic medical center. In eICU, a visually identical flat run reflects the different ordering and telehealth-assisted cadence of community ICUs, so the same input pattern carries a different clinical meaning. The adapter disrupts the eICU forward-fill artefact so that stale channels appear stale to the predictor in the positions where staleness should matter, without attempting to match MIMIC-IV's marginal feature statistics. Wasserstein distance increases because it measures marginal distance, which is orthogonal to the temporal operational cue the frozen LSTM was trained on.

J Systematic Method Comparison

Table 21 expands the nine-axis comparison introduced in §2, adding per-method scores across twenty-seven entries.

Axis-coding nuances. BN-free tracks a general method property: ✓ covers methods compatible with any norm-affine layer (BN/LN/GN) or none. SAR (Niu et al., 2023) is BN-free in this sense (designed for GN/LN backbones); its inapplicability to the YAIB LSTM stems from the absence of *any* norm-affine parameters, not BN specifically. **Predictor frozen** requires the deployed backbone weights to be bit-exact: reprogramming methods (VPT, Voice2Series, Time-LLM) augment the input stream with learnable tokens but leave backbone weights identical (✓); RAG methods that fine-tune an adaptive retrieval module on top of a frozen foundation backbone (TS-RAG) are partial ($\sim$); methods that co-train any predictor weights (RAM-EHR) are trainable (✗). **No source labels** discriminates the source-free axis from the unsupervised-DA axis: classical UDA methods (DANN, CORAL, RAINCOAT, CLUDA, . . .) train end-to-end with labeled source and unlabeled target, so they require source labels at adaptation (✗); source-free TTA, PEFT, reprogramming, and our adapter all assume the source-trained predictor is given and adapt using only target-side or pre-built-artifact information (✓). Retrieval methods are coded ✓ when the datastore is a fixed pre-built artifact rather than fresh source supervision. **Per-step** discriminates temporal granularity from adaptation space: input-space methods that emit a single global perturbation per window or clip (VPT, Voice2Series) score ✗; per-patch reprogramming where each token is a fixed sub-window (Time-LLM) is partial ($\sim$); representation-level adversarial alignment imposed at every recurrent latent (VRADA) is partial ($\sim$) because the per-timestep statistic is not an input rewrite; output-space methods that interpolate the next-step distribution at every decoded position (kNN-MT) score ✓; our adapter rewrites the input at every timestep (✓).

K Pilot Playbook

We distill the ablation evidence into a prescriptive decision tree mapping dataset and task properties to optimal adapter configuration. The tree splits first by predictor regime (frozen recurrent + composite loss (EHR) versus frozen 1D-CNN + balanced softmax (AdaTime)), then by task-class within each branch. The EHR branch instantiates 7 rules (R1–R7) and 5 leaves (L1–L5); the AdaTime branch instantiates 3 rules (A1–A3) and 3 leaves (LA1–LA3), with full per-rule supporting evidence in Tables 25 and 26.

K.1 Source-free branch: term-by-term recipe

Under the strict source-free protocol the adapter is built and trained at the deploying site without ever reading a source sample. Phase 1 (autoencoder pretrain) is performed on the target deployment cohort (eICU or HiRID, the only data available at the deploying site under this regime). Phase 2 (adapter training) zeroes every source-coupling loss term, namely the MMD alignment λ_{align}, the source-pretrain reconstruction λ_{recon}, the source-side target-task regulariser, the linear-probe term λ_{lp},

Table 21: Systematic comparison of domain adaptation, frozen-model, retrieval-augmented, and foundation-model methods across nine applicability axes for clinical time-series adaptation with a frozen predictor. ✓ = supported; ✗ = not supported; ∼ = partial. **Predictor frozen**: deployed forward pass bit-identical to the certified backbone (✓), partially fixed (∼), or trainable (✗). **No source labels**: adaptation requires no access to source-domain data or labels at adaptation time (canonical source-free setting). **Adapt. space**: Input (I), Representation (R), Weight (W), or Output (O). **Per-step**: the adaptation produces a distinct response at every individual timestep, as opposed to a single global transformation per window or per record. **Sample-cond.**: the adaptation transformation depends on the input instance. **Retrieval**: uses a domain-specific datastore for per-instance conditioning (k-NN over a memory bank). **BN-free**: requires no BatchNorm statistics (the YAIB LSTM has none). **Var.-len. TS**: handles variable-length time-series natively. **Clinical EHR**: number of distinct clinical prediction tasks evaluated (0 = non-clinical only).

Method	Predictor frozen	No source labels	Adapt. space	Per-step	Sample-cond.	Retrieval	BN-free	Var.-len. TS	Clinical EHR
Clinical DA									
VRADA (Purushotham et al., 2017)	✗	✗	R	∼	✓	✗	✓	✓	1
OTTEHR (OTTEHR Authors, 2025)	✗	✗	R	✗	✗	✗	✓	✗	3
Mutnuri et al. (Mutnuri et al., 2024)	✗	✗	W	✗	✓	✗	✓	✓	3
General time-series DA									
DANN (Ganin et al., 2016)	✗	✗	R	✗	✓	✗	✓	✓	∼
Deep CORAL (Sun & Saenko, 2016)	✗	✗	R	✗	✓	✗	✓	✓	∼
CoDATS (Wilson et al., 2020)	✗	✗	R	✗	✓	✗	✓	✗	0
RAINCOAT (He et al., 2023)	✗	✗	R	✗	✓	✗	✓	✗	0
CLUDA (Ozyurt et al., 2023)	✗	✗	R	✗	✓	✗	✓	✗	0
TTA & source-free									
TENT (Wang et al., 2021)	✗	✓	W	✗	✗	✗	✗	∼	0
SAR (Niu et al., 2023)	✗	✓	W	✗	✗	✗	✓	∼	0
EATA (Niu et al., 2022)	✗	✓	W	✗	✗	✗	✗	∼	0
T3A (Iwasawa & Matsuo, 2021)	✗	✓	O	✗	✓	✗	✓	∼	0
SHOT (Liang et al., 2020)	✗	✓	W	✗	✗	✗	✓	∼	0
MAPU (Ragab et al., 2023b)	✗	✓	W	✗	✗	✗	✓	∼	0
PEFT (backbone-coupled adapters)									
LoRA (Hu et al., 2022)	✗	✓	W	✗	✗	✗	✓	∼	0
AdapterFusion (Pfeiffer et al., 2021)	✗	✓	W	✗	✗	✗	✓	∼	0
P-Tuning (Liu et al., 2022)	✗	✓	W	✗	✗	✗	✓	∼	0
LST (Sung et al., 2022)	✗	✓	W	✗	✗	✗	✓	∼	0
Frozen-model & reprogramming (input-space)									
VPT (Jia et al., 2022)	✓	✓	I	✗	✗	✗	✓	✗	0
Voice2Series (Yang et al., 2021)	✓	✓	I	✗	✗	✗	✓	✗	0
Time-LLM (Jin et al., 2024)	✓	✓	I	∼	✗	✗	✓	✗	0
TATO (Qiu et al., 2026)	✓	∼	I	✗	∼	✗	✓	✓	0
Retrieval-augmented prediction heads									
kNN-MT (Khandelwal et al., 2021)	✓	✓	O	✓	✓	✓	✓	✓	0
TS-RAG (Ning et al., 2025)	∼	✓	R	✗	✓	✓	✓	✗	0
EHR-RAG (Cao et al., 2026)	∼	✓	O	✗	✓	✓	✓	∼	∼
RAM-EHR (Xu et al., 2024)	✗	✓	R	✗	✓	✓	✓	∼	∼
Foundation model									
ICareFM (Burger et al., 2025)	✗	✓	W	✗	✓	✗	✓	✓	7
PILOT (Ours)	✓	✓	**I**	✓	✓	✓	✓	✓	**5**

Notes. See Axis-coding nuances for BN-free, Predictor frozen, and No source labels.

and the importance regulariser. The cross-attention retrieval module is removed, so no source memory bank is constructed or queried at inference, and the source-stat affine renormalisation is disabled. After these adjustments, the adapter's training objective contains no source-domain term, no source sample contributes a gradient, and no source sample is read at inference. The same protocol applied to the AdaTime sensor benchmarks approaches or surpasses the source-using adapter (Appendix K.2). Five seeds per task; split seed fixed.

Strict source-free vs. source-assisted detail. The strict protocol zeros every source-coupling loss term ($\lambda_{\text{align}}, \lambda_{\text{recon}}, \lambda_{\text{lp}}$, source-side target-task regulariser, importance regulariser) and disables

Table 22: Strict source-free adaptation across all five clinical tasks, mean ± std over five training seeds. Recipe: Phase 1 autoencoder pretrains on the target deployment cohort (eICU or HiRID); Phase 2 zeroes every source-coupling loss ($\mathcal{L}_{\text{align}}, \mathcal{L}_{\text{recon}}, \mathcal{L}_{\text{lp}}$, source-side target-task regulariser, importance regulariser), disables the source-stat affine renormalisation, and removes the cross-attention retrieval module; no source sample is read at inference. The source-assisted comparator is the per-task headline ablation cell from Table 1 (Mortality, AKI, Sepsis, LoS: cells marked † in Table 18; KF: source-pretrain leaf from Table 25). Per-seed values are in Table 23.

Task	Metric	Source-free Δ (mean ± std, n=5)	Source-assisted Δ (best cell, n=5)
Mortality24	AUROC	$+0.0394 \pm 0.0063$	$+0.0485 \pm 0.0009$
AKI	AUROC	$+0.0492 \pm 0.0023$	$+0.0521 \pm 0.0016$
Sepsis	AUROC	$+0.0499 \pm 0.0097$	$+0.0472 \pm 0.0040$
LoS	MAE-z	-0.0321 ± 0.0006	-0.0323 ± 0.0012
KF	MAE-z	-0.0096 ± 0.0004	-0.0094 ± 0.0004

Table 23: Per-seed strict source-free deltas, all five clinical tasks at five seeds. Seeds anonymised; the seed-to-cohort split is held fixed across all 25 cells, only the training seed varies.

Task	Metric	Seed 1	Seed 2	Seed 3	Seed 4	Seed 5	Mean ± std
Mortality24	ΔAUROC	$+0.0349$	$+0.0448$	$+0.0414$	$+0.0310$	$+0.0451$	$+0.0394 \pm 0.0063$
AKI	ΔAUROC	$+0.0516$	$+0.0485$	$+0.0512$	$+0.0459$	$+0.0488$	$+0.0492 \pm 0.0023$
Sepsis	ΔAUROC	$+0.0488$	$+0.0414$	$+0.0407$	$+0.0640$	$+0.0546$	$+0.0499 \pm 0.0097$
LoS	ΔMAE-z	-0.0313	-0.0327	-0.0319	-0.0325	-0.0319	-0.0321 ± 0.0006
KF	ΔMAE-z	-0.0099	-0.0091	-0.0100	-0.0096	-0.0095	-0.0096 ± 0.0004

retrieval; for each task the source-free row is built from this recipe rather than from a partial-removal ablation. Under the source-free protocol the seed-to-seed standard deviation on KF is tight at $\sigma_{\text{KF}} = 0.0004$ MAE-z; the per-seed Mortality spread of 0.0063 AUROC overlaps the source-assisted CI on the same task; the regression cells (LoS $\sigma_{\text{MAE-z}} = 0.0006$, KF $\sigma_{\text{MAE-z}} = 0.0004$) and AKI ($\sigma_{\text{AUROC}} = 0.0023$) keep source-free adaptation at or above the source-assisted regime, with Mortality the largest clinical gap.

K.2 Source-free protocol on the AdaTime sensor benchmarks

We replicate the source-free protocol of §4.2 on all five AdaTime datasets (HAR, WISDM, HHAR, SSC, MFD) where the adapter never reads source inputs or labels at any stage of training: the frozen 1D-CNN is source-trained in an earlier stage and held bit-identical to the with-source experiments, the Phase 1 autoencoder is pretrained on target inputs, and Phase 2 adapter training uses only target inputs. Across the five datasets the source-free adapter approaches or surpasses the source-using adapter (mean Macro-F1 difference ≈ 0 across the five datasets; HAR, WISDM, HHAR over five seeds; SSC and MFD over three seeds, the released source-free pool). On the three datasets that admit a residual-versus-absolute output-mode comparison (HAR, WISDM, HHAR), residual-output adapters beat absolute-output adapters by approximately 21 Macro-F1 points on HAR and 13 on WISDM with all other settings held fixed, while HHAR is within multi-seed noise of either choice. Table 24 reports the per-dataset source-free deltas.

Holding all other settings fixed and comparing residual against absolute output on the three datasets that admit such a comparison, paired Wilcoxon signed-rank tests favour the residual-output adapter on HAR (effect size ≈ 1.5, $p < 10^{-5}$) and WISDM (≈ 0.7, $p \approx 10^{-5}$); HHAR is a within-noise tie (≈ -0.1, $p \approx 0.08$). The same direction is observed in the with-source AdaTime runs reported in Table 2, so the output-mode preference is regime-stable.

K.3 Decision tree

```
ROOT (new task)
+-- EHR  (frozen recurrent + composite loss)  -- R1-R7
|   +-- classification
|   |   +-- per-stay                  -> L3 (Mortality-class)
```

Table 24: Source-free adaptation on the five AdaTime sensor benchmarks. Mean ± std (HAR, WISDM, HHAR over five seeds; SSC and MFD over three seeds, the released source-free pool); absolute Macro-F1 (MF1) for the source-free adapter, alongside the source-only frozen 1D-CNN baseline and the source-free delta. Bolded source-free MF1 cells indicate datasets where the source-free adapter beats its source-assisted counterpart in Table 2. Statistical significance of the residual-versus-absolute output-mode comparison (with all other settings held fixed) is reported in the surrounding prose for HAR, WISDM, and HHAR; SSC and MFD use a single output mode without a paired residual-versus-absolute toggle.

Dataset	Source-only MF1	Source-free MF1	Δ MF1
HAR	79.97	92.41 ± 0.79	$+12.44 \pm 0.79$
WISDM	47.78	$\mathbf{70.65 \pm 4.67}$	$+22.87 \pm 5.51$
HHAR	59.07	$\mathbf{87.36 \pm 0.92}$	$+28.30 \pm 3.20$
SSC	57.85	65.69 ± 1.77	$+7.84 \pm 3.90$
MFD	77.04	$\mathbf{96.93 \pm 0.18}$	$+19.89 \pm 2.56$

```
|   |   +-- per-timestep
|   |       +-- dense (>=10% pos)     -> L4 (AKI-class)
|   |       +-- sparse (<5%  pos)     -> L5 (Sepsis-class)
|   +-- regression
|       +-- cumulative-stat schema    -> L1 (KF-class)
|       +-- otherwise                 -> L2 (LoS-class)
+-- AdaTime (frozen 1D-CNN + balanced softmax) -- A1-A3
    +-- controlled single-protocol multi-channel -> LA1 (HAR-like)
    +-- 1-channel quasi-stationary               -> LA2 (SSC/MFD-like)
    +-- multi-subject/device heterogeneous       -> LA3 (HHAR/WISDM-like)
```

K.4 Rules with supporting evidence

K.5 Leaf configurations

Each leaf in the decision tree corresponds to a concrete adapter configuration. Table 27 lists the load-bearing knobs; unlisted knobs take the codebase defaults (see the released code's training-configuration whitelist), including the always-on main task loss $\mathcal{L}_{\text{task}}$ that back-propagates gradients through the frozen predictor in every leaf (so columns showing $\lambda_{\text{recon}} = \lambda_{\text{align}} = \lambda_{\text{tgt}} = 0$ for the kidney-function leaf disable only the auxiliary terms).

K.6 Reproducibility tool: `build_input_adapter_config.py`

The CLI tool `scripts/build_input_adapter_config.py` mechanises the decision tree above: it walks the user's dataset/task properties to a leaf, loads the matching on-disk template config, and applies only the playbook-documented overrides. No fields are invented; schema fidelity is inherited from the template. Required arguments:

- `-predictor-regime {ehr,adatime}`: top-of-tree split.
- `-task-type {classification,regression}`: EHR branch only.
- `-granularity {per-stay,per-timestep}`: EHR classification only.
- `-feature-schema {standard,cumulative-stat,mi-optional}`: EHR regression only.
- `-label-density {dense,sparse,borderline}`: EHR per-timestep classification only.
- `-heterogeneity {controlled,single-channel,heterogeneous}`: AdaTime branch only.
- `-output-mode {auto,residual,absolute}`: defaults to leaf's per-regime default (EHR→absolute / AdaTime→residual).
- User fields: `-source-data-dir`, `-target-data-dir` (EHR) or `-dataset`, `-data-path` (AdaTime); `-seed`, `-output`.

Table 25: EHR-branch rules (R1–R7). Each rule is supported by a multi-seed ablation cell (three or five seeds; per-row counts are stated in the supporting-cell column) or a hard-guardrail mechanism. Cohort labels reference the playbook's Cohort-stability scorecard (§6).

ID	Rule	Supporting cell
R1	Drop fidelity ($\lambda_{\text{recon}} = 0$) on Mortality-class	Mortality no-fidelity ablation: eICU $+0.0485 \pm 0.0009$ across five seeds; HiRID $+0.0589 \pm 0.0059$ across three seeds (3/3 paired seeds $p < 0.001$)
R2a	On KF (source-pretrain regime), zero *all* Phase-2 task losses except λ_{range}: an autoencoder pretrained on the source cohort plus a range-regularised near-identity adapter is sufficient	KF source-pretrain leaf (only λ_{range} active in Phase 2): eICU -0.0094 ± 0.0004 across five seeds; HiRID -0.0017 ± 0.0002 (three seeds)
R2b	Drop MMD on LoS-class regression	LoS strict-reconstruction (no-MMD) ablation: eICU -0.0323 ± 0.0012 MAE-z (five seeds)
R3	`output_mode = "absolute"` on EHR	5/5 EHR tasks across three seeds (strict-toggle ablation)
R4	Cosine $\times$ density rule for λ_{recon}: weak directional after the five-task cosine pilot; four regimes (cooperative-cos $\geq +0.5$; borderline; orthogonal-sparse; orthogonal-dense), with LoS as a known sign-failure (cos -0.32, near-null at $\lambda_{\text{recon}} = 0$); per-task three-seed evidence overrides the cosine sign	Mortality cos $+0.84 \rightarrow +0.0485 \pm 0.0009$ at five seeds; Sepsis cos $-0.21 \rightarrow -0.0690 \pm 0.0167$ collapse on the dropped-fidelity ablation; LoS cos $-0.32 \rightarrow +0.0026 \pm 0.0006$ near-null at $\lambda_{\text{recon}} = 0$
R5	`n_cross_layers` is task-specific: 0 for sparse-label classification (AKI, Sepsis), 2 for Mortality, 3 for LoS and KF. Retrieval is at-most-tied with no-retrieval on classification	The AKI no-retrieval ablation tightens σ $2.5\times$ vs the full pipeline; per-task settings match the leaf table (Table 27)
R6	Pretrain mandatory on sparse-label classification; otherwise no-pretrain destabilises across seeds	No-pretrain, three seeds: Sepsis -0.0848 ± 0.0035 (mean collapse); Mortality -0.0190 ± 0.0840 (σ $40\times$ full pipeline); AKI -0.0021 ± 0.0606 ($12\times$); LoS -0.0051 ± 0.0150 ($5\times$); KF -0.0035 ± 0.0037 ($2.6\times$)
R7	Drop $\lambda_{\text{target_task}}$ on the KF source-pretrain leaf; keep elsewhere, including on Sepsis	The Sepsis five-seed headline (no-retrieval cell, $+0.0472 \pm 0.0040$ AUROC) retains $\lambda_{\text{target_task}} = 0.5$; the no-target-task ablation lands at $+4.74 \pm 0.37$, within multi-seed noise of the all-components-on control $+4.75 \pm 0.38$ but below the no-retrieval headline. KF: source-pretrain leaf with $\lambda_{\text{target_task}} = 0$ alongside dropped fidelity and MMD attains -0.0094 ± 0.0004 MAE-z across five seeds
R8	Source-domain Phase 1 pretrain on KF regression (champion candidate, cohort verification on HiRID outstanding)	KF source-pretrain five-seed mean -0.0094 ± 0.0004 MAE-z; LoS source-pretrain ties the target-pretrain reference at -0.0320 ± 0.0009

Example invocation for a Mortality-class task on a new ICU cohort (leaf L3):

```
python scripts/build_input_adapter_config.py \
    --predictor-regime ehr \
    --task-type classification \
    --granularity per-stay \
```

Table 26: AdaTime-branch rules (A1–A3). All three are tier-stable across three seeds on the 5-dataset corpus.

ID	Rule	Supporting cell
A1	`pretrain_epochs = 0` is universal AdaTime default	5/5 datasets prefer $p = 0$; HHAR five-seed reverse $+1.70$ MF1 ($z = +3.27\sigma$); WISDM five-seed reverse $+3.83$ MF1 ($z = +2.95\sigma$)
A2	Tiny capacity ($\leq 1\times$ predictor params) $\geq$ Full at every measured step	5/5 datasets across three seeds; magnitudes $+0.0011$ (SSC) to $+0.057$ (WISDM) MF1
A3	Residual-output adapter wins or ties at every measured AdaTime cell. AdaTime follows the released TKDD protocol (Ragab et al., 2023a) with no held-out target-validation split, so A3 is reported as a descriptive observation across held-out target test sets at three or more seeds rather than a prescriptive rule for unseen datasets	HAR controlled comparison (three seeds, all other settings held fixed) residual advantage $+30.49 \pm 4.91$ MF1, paired Wilcoxon $p < 10^{-5}$; WISDM low-fidelity controlled comparison (three seeds) residual advantage $+17.47 \pm 5.94$ MF1, $p \approx 4\times10^{-4}$; HHAR controlled comparison a within-noise tie at three seeds; Pearson r between λ_{fid} and the residual advantage ≈ -0.93 ($n = 4$, $p \approx 0.07$)

```
--source-data-dir /data/new_cohort \
--target-data-dir /data/mimic_iv \
--seed 2222 \
--output configs/new_cohort_mortality.json
```

The output is a single JSON config that drops directly into the training CLI: `python run.py train_translator -config configs/new_cohort_mortality.json`. The `-explain` flag prints the resolved leaf rationale and override list without writing a config; `-dry-run` prints the resulting JSON to stdout. The tool currently exposes 5 EHR leaves (L1–L5) and 3 AdaTime leaves (LA1–LA3); adding a new leaf requires a new `LeafTemplate` entry plus a template configuration corresponding to the new leaf (the released ablation and HiRID configuration libraries already cover the existing eight leaves).

Table 27: Per-leaf adapter knobs. Pretrain epochs "p" refers to the Phase-1 autoencoder pretrain length (15 EHR / 0 AdaTime). n_x denotes `n_cross_layers`. "out" is `output_mode`. "FG" is the feature gate, a learnable per-feature sigmoid loss-weighting module (Appendix F.3). Each leaf cites its supporting ablation cell.

Leaf	p	out	n_x	λ_{recon}	λ_{align}	λ_{tgt}	FG	Supporting cell
L1 (KF)	15	abs	3	0.0	0.0	0.0	on	KF source-pretrain leaf (Phase 1 trained on the source cohort), five-seed mean -0.0094 ± 0.0004 MAE-z; cohort fallback recipe (low encoder learning rate, dropped fidelity, dropped MMD) at -0.0086 ± 0.0014
L2 (LoS)	15	abs	3	0.1	0.0	0.5	on	LoS strict-reconstruction (no-MMD) ablation, three-seed mean -0.0323 ± 0.0016
L3 (Mortality)	15	abs	2	0.0	0.5	0.5	on	Mortality no-fidelity ablation, five-seed mean $+0.0485 \pm 0.0009$ AUROC
L4 (AKI)	15	abs	0	0.1	0.5	0.5	on	AKI no-retrieval ablation, five-seed mean $+0.0521 \pm 0.0016$
L5 (Sepsis)	15	abs	0	0.1	0.5	0.5	on	Sepsis no-retrieval ablation, five-seed mean $+0.0472 \pm 0.0040$
LA1 (HAR-like)	0	res	2–3	0.1	n/a	n/a	n/a	AdaTime HAR tiny-capacity cell, +0.13–0.14 MF1 vs source-only
LA2 (SSC/MFD-like)	0	res	2	0.1	n/a	n/a	n/a	AdaTime SSC tiny-capacity cell, +0.08; MFD tiny-capacity (extra-tiny variant), +0.20
LA3 (HHAR/WISDM)	0	res	0	0.1	n/a	n/a	n/a	AdaTime HHAR and WISDM tiny-capacity, no-pretrain cells, +0.30–0.35 MF1