LEARNING WHERE TO ADAPT: LEARNED SUBSPACE TEST-TIME ADAPTATION FOR MEDICAL TIME SERIES

Anonymous authors
Paper under double-blind review

ABSTRACT

Test-time adaptation (TTA) promises to personalize medical time series classifiers to a new subject from a few unlabeled recordings. Its standard recipe, entropy minimization over normalization parameters, follows the model's uncertainty rather than the ways in which subjects actually differ, and with few windows it can turn correct predictions into errors. We argue that where to adapt should itself be learned. Because inter-subject shift is plausibly driven by a few shared physiological and recording factors, we propose Learned Subspace Test-Time Adaptation (LSTA), which meta-learns from past subjects a low-rank space of LayerNorm updates and restricts each new subject's adaptation to it. During meta-training, unlabeled support windows determine an entropy step within this space, and the classification loss on labeled query windows from the same subject shapes it. At test time the space is fixed, the parameters outside LayerNorm remain frozen, and each subject adapts from at most eight unlabeled windows. On five EEG and ECG benchmarks with subject-exclusive splits, LSTA obtains the highest mean accuracy and macro-F1 among seven TTA baselines, and a learned subspace outperforms a random one in 22 of 25 dataset-seed pairs. Across all eight datasets we evaluate, LSTA is the only method whose macro-F1 never falls below that of the frozen model, and under contaminated support it degrades far less than direct LayerNorm adaptation and turns fewer correct predictions into errors. On clean data, its gain over direct LayerNorm adaptation is concentrated on one dataset. Code is available at https://anonymous.4open.science/r/Med-458D/.

1 INTRODUCTION

Deep learning models for medical time series (MedTS), such as electrocardiography (ECG) and electroencephalography (EEG), are increasingly accurate on benchmark splits (Wang et al., 2024; Yang et al., 2023a; Guoqi et al., 2026). In deployment, however, every prediction concerns a subject the model has never seen. Recordings from a new subject differ from the training population in ways that have little to do with diagnosis, such as amplitude gain, baseline level and the orientation of the heart's electrical axis relative to the electrodes. A recording also contains many windows from the same subject, so a few unlabeled windows are usually available before the model has to classify the rest. Test-time adaptation (TTA) is designed to exploit exactly this opportunity.

Figure 1 summarizes the problem, our approach and the main results.

The standard TTA recipe minimizes the entropy of predictions on unlabeled test data by updating normalization parameters (Wang et al., 2020; Niu et al., 2023). This recipe was designed for large test batches from a single shifted domain, and for a new subject it faces two difficulties. First, the entropy gradient points where the model is uncertain, which need not be where this subject differs from others; with only a handful of windows it can also follow noise or the class content of the windows themselves. Second, the update is unconstrained, so every normalization parameter may move. In our experiments this appears as harm: direct LayerNorm adaptation turns 3.45% of the correct predictions on PTB into errors, and when half of its support windows come from another subject its macro-F1 on PTB falls to 66.7, compared with 81.6 for the frozen model.

Our starting point is a different view of what distinguishes one subject from another. The physiological and recording factors listed above are few and shared across people, so we hypothesize that the

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.

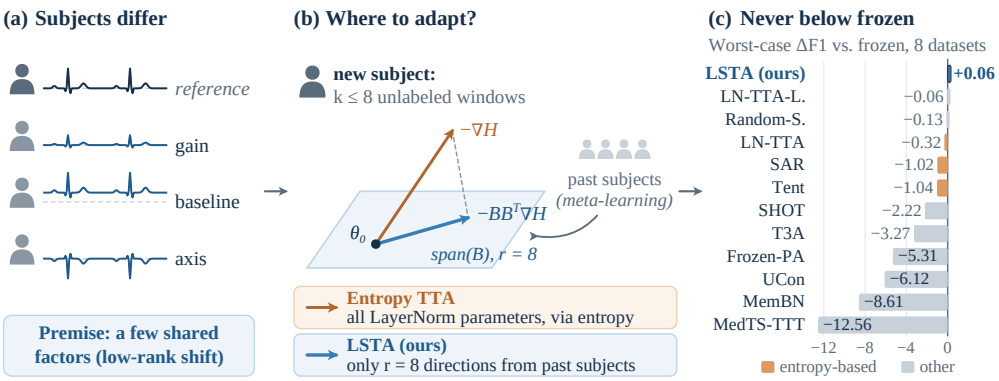


Figure 1: **Overview of LSTA.** (a) Premise: recordings from different subjects differ through a few shared factors such as gain, baseline and electrical axis (schematic; traces are illustrative, not measured data). (b) Entropy-based test-time adaptation moves every LayerNorm parameter along the model's uncertainty, whereas LSTA restricts the update of a new subject to a low-rank subspace meta-learned from past subject. (c) Measured worst-case change in macro-F1 relative to the frozen backbone across the eight datasets (Tables 1, 2, 5 and 7); LSTA is the only method that never falls below the frozen model.

shift between subjects occupies a low-dimensional space of parameter changes, whereas uncertainty-driven updates do not. This leads to our central question: *can we learn, from past subjects, where a new subject is allowed to differ, and adapt only there?*

An analogy from recommender systems makes the idea concrete. A new user with only a few ratings is not given a profile learned from scratch; the system instead places the user in a low-dimensional space of latent factors learned from many earlier users (Koren et al., 2009). We propose Learned Subspace Test-Time Adaptation (LSTA), which plays the same role for subjects. LSTA meta-learns a shared basis B of LayerNorm updates from training subjects while the pretrained classifier stays fixed. During meta-training, unlabeled support windows determine an entropy step within the span of B, and the classification loss on labeled query windows from the same subject shapes B. At test time B is frozen, and a new subject only chooses $r = 8$ coordinates, estimated from at most eight unlabeled windows.

We evaluate LSTA on eight EEG and ECG datasets, six in which all windows of a subject share one label and two whose windows carry their own labels, against seven TTA methods run on the same backbone checkpoints. LSTA gives the highest mean macro-F1 and accuracy among these methods on the five standard subject-labeled benchmarks, and a learned basis outperforms a random one in 22 of 25 dataset-seed pairs. Its clearest advantage, however, is safety. LSTA is the only method we evaluate whose macro-F1 is never below that of the frozen model on any of the eight datasets, it is stable across 1 to 32 inference steps and across a range of step sizes, and with contaminated support it degrades far less than direct LayerNorm adaptation and turns fewer correct predictions into errors. On clean subject-labeled data its gain over direct LayerNorm adaptation is small outside ADFTD, and we report these limits explicitly. Our contributions are as follows:

1. We identify a mismatch in test-time adaptation for medical time series: entropy updates follow model uncertainty and allows updates across the full parameter space, whereas subjects differ through a few shared factors. We show that unconstrained entropy updates are harmful when support windows are few or unreliable.
2. We propose LSTA, which meta-learns a low-rank LayerNorm update space from past subjects so that each new subject adapts only within it, from a few unlabeled windows and without source recordings or target labels. A one-step training rule avoids second-order gradients.
3. We evaluate LSTA against published classifiers and seven TTA methods on identical checkpoints, including datasets in which a subject's windows belong to different classes, and we isolate the role of the learned subspace through component ablations, support contamina-

tion, harmful prediction changes, signal corruption, and sweeps over support size, rank, number of steps and step size.

2 Related Work

Test-time adaptation. Test-time adaptation (TTA) updates a pretrained model with unlabeled test data. Tent (Wang et al., 2020) minimizes prediction entropy over normalization parameters, and SAR (Niu et al., 2023) stabilizes this objective with sharpness-aware, sample-selective updates. Entropy-based adaptation has been applied to EEG decoding across subjects and sessions (Wimpff et al., 2024; Li et al., 2023), and backpropagation-free variants combine predictions from transformed inputs (Li et al., 2026). These methods adapt along the direction chosen by the unsupervised objective, whereas LSTA constrains where adaptation may occur.

Learning how to adapt. MT3 (Bartler et al., 2022) and adaptive risk minimization (Zhang et al., 2021) meta-learn models that adapt from unlabeled test data, Meta-Curvature (Park & Oliva, 2019) learns gradient transformations, and BOLT (Park et al., 2026) adapts within a fixed low-rank subspace extracted from fine-tuned models. For physiological signals, SuLoRA (Klein et al., 2025) learns subject-specific adapters with labels, TRANSIT-EEG (Ahuja & Sethia, 2025) fine-tunes LoRA modules for new subjects, and Jin et al. (2024) align test data to source subsets. LSTA keeps the source model fixed, treats each subject as an unlabeled group, and meta-learns one shared low-rank LayerNorm subspace through which every new subject adapts.

Medical time series classification. Medformer (Wang et al., 2024), MedGNN (Fan et al., 2025) and TeCh (Guoqi et al., 2026) model temporal and cross-channel dependencies, and pretrained models such as BIOT (Yang et al., 2023a) and LaBraM (Jiang et al., 2024) learn transferable representations. These models are evaluated with subject-exclusive splits but classify every window with fixed parameters; LSTA is complementary and adapts such a backbone to each new subject. An extended discussion is given in Appendix A.2.

3 Methodology

Figure 2 provides an overview of our framework, including adaptation subspace learning and subject-specific test-time adaptation.

3.1 Problem Formulation

Let $\mathcal{D}_s = \{(x_{s,i}, y_{s,i})\}_{i=1}^{n_s}$ denote the windows associated with subject s, where $x_{s,i} \in \mathbb{R}^{C\times T}$ contains C channels and T time points. We adopt a subject-independent setting in which the training, validation, and test subject sets are pairwise disjoint:

$$\mathcal{U}_a \cap \mathcal{U}_b = \emptyset, \qquad a \neq b, \quad a, b \in \{\text{train}, \text{val}, \text{test}\}. \tag{1}$$

We consider tasks with a single diagnostic label y_s for each subject and assign this label to all corresponding windows, so that $y_{s,i} = y_s$. Consequently, each subject's support set is class-homogeneous, although its label is unavailable during adaptation. Predicting the same class for all support windows is therefore not inherently incorrect; the concern is that entropy minimization may increase confidence in the wrong class. Query supervision during meta-training guides the adaptation space according to classification performance on separate windows, but does not guarantee correct adaptation for every unseen subject.

Let f_{ψ_0,ϕ_0} denote a classifier pretrained using labeled windows from $\mathcal{U}_{\text{train}}$. Here, $\phi_0 \in \mathbb{R}^D$ concatenates the affine weights and biases of all LayerNorm modules, while ψ_0 contains the remaining model parameters. We restrict subject-specific adaptation to the LayerNorm parameters, keeping ψ_0 fixed.

For each adaptation episode involving subject s, we select an unlabeled support set $\mathcal{S}_s$ and a separate query set $\mathcal{Q}_s$, with no window appearing in both sets. The support set contains up to k windows and is used to obtain subject-specific parameters ϕ_s, initialized from ϕ_0. The adapted classifier f_{ψ_0,ϕ_s}

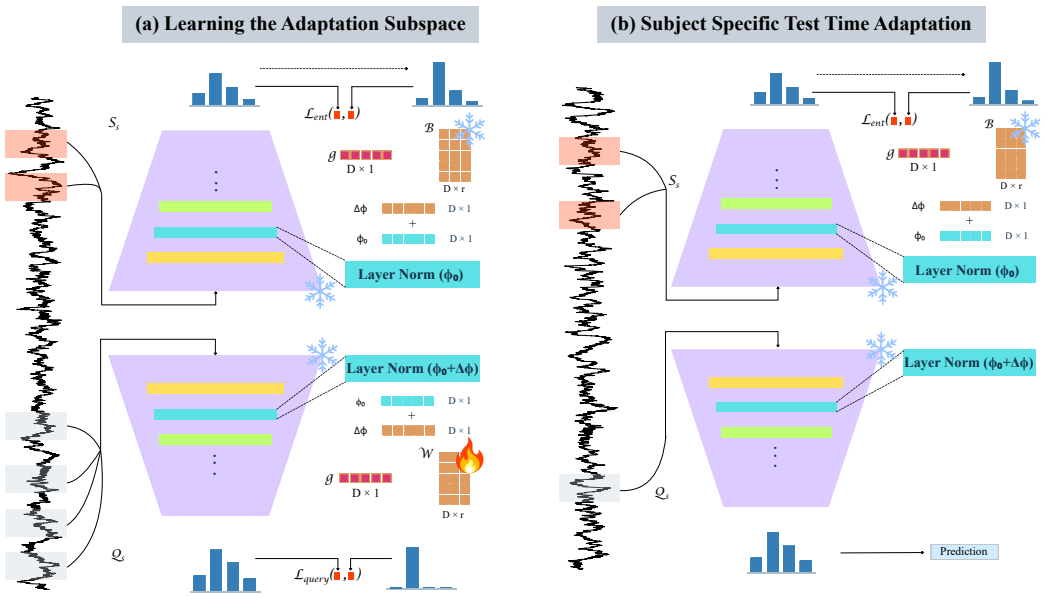


Figure 2: Overview of the proposed framework LSTA. (a) Learning the adaptation subspace: unlabeled support windows determine an entropy-based LayerNorm update within a shared low-dimensional space. Classification loss on separate labeled query windows trains the adaptation basis, while the pretrained source parameters remain fixed. (b) Subject-specific test-time adaptation: the learned basis is frozen, and unlabeled support windows from an unseen subject determine the LayerNorm update used to classify that subject's query windows.

then predicts the labels of the query windows. Each episode starts from the source parameters, preventing adaptation updates from carrying across subjects or episodes.

Our objective is to learn an adaptation mechanism from training subjects that enables useful parameter updates from an unseen subject's unlabeled support windows. During this additional training stage, support inputs determine the adaptation update, while labeled query windows provide supervision for learning the adaptation mechanism. The source parameters ψ_0 and ϕ_0 remain fixed throughout this stage.

At test time, the learned mechanism is fixed, and subject-specific updates use only the support inputs, without access to source recordings or target labels. Performance is measured on the separate query windows.

3.2 Learned Adaptation Subspace

This parameterization encodes the premise of Section 1: a new subject may differ from the source model only along directions learned from past subjects. We constrain subject-specific LayerNorm updates to a shared low-dimensional space. Specifically, we express the adapted parameters for subject s as

$$\phi_s = \phi_0 + Bz_s, \qquad B \in \mathbb{R}^{D\times r}, \quad z_s \in \mathbb{R}^r, \tag{2}$$

where $r \ll D$. The matrix B defines the allowable update directions and is shared across subjects, while z_s determines the adjustment for an individual subject. Each column of B represents a coordinated change across the LayerNorm weights and biases, restricting adaptation to a space with at most r dimensions.

We parameterize B using a learnable matrix $W \in \mathbb{R}^{D\times r}$, initialized as the orthonormal factor of the QR decomposition of a random Gaussian matrix. After each outer optimizer step, we apply QR retraction to maintain $W^\top W = I_r$. The forward pass retains the normalization

$$B(W) = \frac{\sqrt{r}\,W}{\max(\|W\|_F, \epsilon)}, \tag{3}$$

where ϵ is a small constant for numerical stability. Since $W^\top W = I_r$ implies $\|W\|_F = \sqrt{r}$, this normalization leaves W unchanged, giving $B(W) = W$ and $B^\top B = I_r$ up to numerical precision.

For a given unlabeled support set $\mathcal{S}_s$, we adapt the model by minimizing the mean entropy of its temperature-scaled predictions:

$$\mathcal{L}_{\text{ent}}(\phi; \mathcal{S}_s) = -\frac{1}{|\mathcal{S}_s|} \sum_{x \in \mathcal{S}_s} \sum_{c=1}^{K} \pi_c^{(\tau)}(x; \phi) \log \pi_c^{(\tau)}(x; \phi), \tag{4}$$

where $\pi_c^{(\tau)}(x; \phi)$ is the probability of class c obtained by applying softmax to the model logits divided by temperature τ, and K is the number of classes. Temperature scaling is applied only to the support set training objective.

Each adaptation episode starts from $z_s^{(0)} = 0$, corresponding to the source parameters ϕ_0. Let g_s denote the support entropy gradient at this initial state:

$$g_s = \nabla_\phi \mathcal{L}_{\text{ent}}(\phi; \mathcal{S}_s)|_{\phi=\phi_0}. \tag{5}$$

A single gradient step with respect to the low-dimensional coordinates gives

$$z_s^{(1)} = -\eta B^\top g_s, \qquad \phi_s^{(1)} = \phi_0 - \eta B B^\top g_s, \tag{6}$$

where η is the adaptation learning rate.

Because B has orthonormal columns, $BB^\top$ projects the support gradient onto the learned subspace without increasing its norm.

3.3 Training and Test-Time Adaptation

We learn the adaptation matrix using episodes sampled from training subjects. Each optimizer step averages the losses from four episodes. For each episode, we sample a subject uniformly, use two disjoint pools of that subject's windows, and randomly assign their support and query roles. Starting from ϕ_0, we apply one entropy-based update using up to k support windows and compute the classification loss on up to 32 query windows:

$$\mathcal{L}_{\text{query}}(W) = -\frac{1}{|\mathcal{Q}_s|} \sum_{(x,y) \in \mathcal{Q}_s} \log \pi_y(x; \phi_s^{(1)}(W)). \tag{7}$$

We minimize the episode-averaged query loss with respect to W, keeping ψ_0 and ϕ_0 fixed. Support inputs determine the subject-specific update, while query labels supervise whether that update improves classification on separate windows. For this single-step training rule, g_s is evaluated at the fixed source parameters and is independent of W. The outer gradient therefore differentiates through $B(W)B(W)^\top$ and the query forward pass, without requiring a derivative of the support gradient with respect to W. This property is specific to the stated single-step training rule.

We optimize W using Adam, clipping its gradient norm to 1 before each optimizer step. After each step, we apply QR retraction with column-sign correction to keep the basis columns orthogonal and unit length. Retraction is performed outside the gradient computation and is required only during training, since the basis remains fixed at test time.

At test time, B is frozen. For each test subject, we start from ϕ_0 and adapt using up to k unlabeled support windows:

$$\phi_s^{(j+1)} = \phi_s^{(j)} - \eta B B^\top \nabla_\phi \mathcal{L}_{\text{ent}}(\phi_s^{(j)}; \mathcal{S}_s), \qquad \phi_s^{(0)} = \phi_0. \tag{8}$$

The support set remains unchanged across steps. In all experiments we use a single adaptation step at inference, matching the training rule. We then use the adapted model to classify the query windows.

For evaluation, we adapt on support windows sampled from one pool and predict all windows in the other. Then we reverse the pool roles and repeat, starting again from ϕ_0. Every retained window is therefore evaluated once per support draw, without being included in the adaptation used to predict it. Updates are reset between subjects and directions. This evaluation uses windows from the same recording without a fixed order. Test labels are used only to compute the final metrics.

Table 1: Macro-F1 (%, mean ± s.d. over five backbone seeds) of test-time adaptation methods on the five subject-labeled datasets. All methods adapt the same frozen TeCh checkpoints with the same support draws and query windows, except MedTS-TTT, which uses its own backbone. Best and second-best means are in **bold** and underlined. Accuracy, precision, recall, AUROC and AUPRC are reported in Appendix Table 8.

Method	ADFTD	APAVA	TDBrain	PTB	PTB-XL	Mean
Frozen backbone	48.44±1.72	85.71±0.49	92.36±1.27	81.60±2.31	62.39±0.30	74.10
Tent (Wang et al., 2020)	52.70±3.00	88.17±2.00	95.14±2.05	81.54±2.63	64.10±0.45	76.33
T3A (Iwasawa & Matsuo, 2021)	49.95±2.60	87.64±1.28	92.66±1.32	81.70±2.94	64.44±0.42	75.28
SAR (Niu et al., 2023)	50.54±3.47	87.25±2.52	97.18±0.56	81.68±2.71	63.81±0.38	76.09
MemBN (Kang et al., 2024)	45.93±1.84	81.05±0.72	89.42±1.17	72.99±3.66	57.95±0.49	69.47
SHOT (Liang et al., 2020)	50.73±3.21	88.64±1.94	94.22±1.69	80.99±2.78	63.97±0.46	75.71
UCon (Xu et al., 2025)	51.70±3.36	89.05±3.64	92.92±3.35	82.02±1.72	64.62±0.44	76.06
MedTS-TTT (Chen et al., 2026)	48.80±2.79	73.15±9.25	96.88±2.86	73.89±4.49	63.33±1.00	71.21
LSTA (ours)	**55.32**±1.86	**93.18**±3.60	**98.88**±0.76	**84.09**±3.53	**64.66**±0.51	**79.23**

4 EXPERIMENTS

4.1 DATASETS

We use eight datasets with subject-exclusive splits (Appendix Table 4). In six of them every window inherits a label shared by all windows of its subject: APAVA (Escudero et al., 2006), TDBrain (Van Dijk et al., 2022) and ADFTD (Miltiadous et al., 2023) (EEG) and PTB (Goldberger et al., 2000) and PTB-XL (Wagner et al., 2020) (ECG), for which we follow the splits of Guoqi et al. (2026), and HPP (Shilo et al., 2021; Reicher et al., 2025), a cohort of 7,139 participants with 12-lead resting ECGs labeled by diagnostic interpretation category. In MIT-BIH (Moody & Mark, 2001) and Sleep-EDF (Kemp et al., 2000), each window has its own beat-type or sleep-stage label, so a subject's support windows belong to different classes. Descriptions are given in Appendix A.3.

4.2 EXPERIMENTAL SETTINGS

Protocol. For each dataset we train five TeCh checkpoints (Guoqi et al., 2026) (seeds 42–46) with the published configuration, and all their parameters stay fixed. For each test subject, the windows are split into two disjoint pools. Up to $k = 8$ support windows are drawn from one pool, the adapted model classifies every window of the other pool, and the roles are then reversed, each time starting from the source checkpoint. We repeat this for five support draws and report the mean and standard deviation over backbone seeds. Test labels are used only to compute metrics.

LSTA and baselines. LSTA uses rank $r = 8$ and a single entropy step in meta-training and at inference; the step size and temperature are selected on validation subjects (Appendix A.4). We compare with seven TTA methods run with the same support draws and query windows: Tent (Wang et al., 2020), SAR (Niu et al., 2023), T3A (Iwasawa & Matsuo, 2021), MemBN (Kang et al., 2024), SHOT (Liang et al., 2020), UCon (Xu et al., 2025) and MedTS-TTT (Chen et al., 2026), and with frozen prediction aggregation (Frozen-PA), which averages the frozen model's predictions over the support windows. Comparisons with eleven published classifiers are given in Appendix A.1.

4.3 MAIN RESULTS

Table 1 compares LSTA with test-time adaptation methods under an identical protocol. LSTA obtains the highest macro-F1 on each of the five subject-labeled datasets, raising the average from 74.10 for the frozen backbone to 79.23, compared with 76.33 for the strongest competitor (Tent); it also obtains the highest accuracy on every dataset (Appendix Table 8). The margin over the best competing method is 2.62 points on ADFTD, 4.13 on APAVA, 1.70 on TDBrain, 2.07 on PTB and 0.04 on PTB-XL; on PTB-XL the methods are within one standard deviation of each other. Several methods fall below the frozen backbone on some datasets, and MemBN does so on all five, which illustrates that adaptation from a few windows can hurt. Relative to the published TeCh results, LSTA

Table 2: Component ablation (test F1, $\times 100$, mean$\pm$std over 5 seeds). Top: components added in turn. Bottom: LSTA's meta-learned rank-8 subspace replaced. Bold: best variant per column.

Variant	APAVA	TDBrain	PTB	ADFTD	PTB-XL	Mean
Frozen Backbone	85.71±0.49	92.36±1.27	81.60±2.31	48.44±1.72	62.39±0.30	74.10
LN-TTA	92.36±3.68	98.91±0.79	83.56±3.27	51.42±3.71	**64.71**±0.45	78.19
LSTA	**93.18**±3.60	98.88±0.76	**84.09**±3.53	**55.32**±1.86	64.66±0.51	**79.23**
Random-Subspace	91.07±5.96	98.65±0.98	81.47±4.31	51.08±3.46	64.06±0.44	77.26
LN-TTA-Learnable	92.41±3.75	**98.93**±0.79	83.60±3.25	52.51±3.39	**64.71**±0.46	78.43

improves macro-F1 by 2.1 to 6.9 points (Appendix Table 3); because LSTA also uses unlabeled windows from each subject, we treat that comparison as indicative only.

Appendix Table 5 reports the two window-labeled datasets and HPP. When a subject's support windows belong to different classes, aggregating frozen predictions becomes harmful: Frozen-PA lowers macro-F1 relative to the frozen model on Sleep-EDF (67.13 vs. 72.36) and MIT-BIH (36.33 vs. 41.64). On Sleep-EDF every other adaptation method also lowers macro-F1, whereas LSTA keeps it at the level of the frozen model (72.42). On MIT-BIH, LSTA obtains the highest macro-F1 (42.51, compared with 42.26 for Tent and 41.64 for the frozen model), although the margins are small relative to the seed variation. On HPP, LSTA obtains the highest accuracy and macro-F1 (53.85 vs. 48.54 for the frozen model), while Frozen-PA attains the highest AUROC. Taken together, LSTA is the only method whose macro-F1 is never below that of the frozen model on any of the eight datasets (Figure 1c); for several competitors the shortfall is small, but for others it reaches several points.

4.4 Ablation Study

Table 2 separates the effect of adaptation from that of the learned subspace, using test macro-F1 on identical checkpoints. Entropy adaptation of all LayerNorm parameters (LN-TTA) already improves over the frozen backbone, raising mean macro-F1 from 74.10 to 78.19 and winning 24 of 25 dataset-seed pairs. Restricting the update to LSTA's meta-learned rank-8 subspace raises the mean to 79.23. The largest gain over LN-TTA occurs on ADFTD (55.32 vs. 51.42), with smaller gains on APAVA (93.18 vs. 92.36) and PTB (84.09 vs. 83.56), while on TDBrain and PTB-XL the two methods differ by at most 0.05 points. LSTA outperforms LN-TTA in 17 of 25 pairs, which is not significant under a one-sided sign test ($p \approx 0.054$).

The lower block replaces the learned basis. With a random rank-8 basis (Random-Subspace), mean macro-F1 drops to 77.26 and LSTA wins 22 of 25 pairs ($p < 0.001$), so the improvement is not explained by the low-rank constraint alone. LN-TTA-Learnable, which uses the same meta-training procedure as LSTA to learn a positive scale for each LayerNorm gradient , reaches 78.43, and LSTA wins 16 of 25 pairs against it ($p \approx 0.11$). Learning the update directions therefore helps most clearly on ADFTD, while on the other datasets its benefit is small relative to the variation across backbone seeds. These sign tests treat each dataset-seed pair as one observation and do not account for variation across subjects.

The support-size sweep (Appendix Figure 7) shows that LSTA gives the highest macro-F1 on ADFTD, APAVA, PTB and TDBrain once four or more support windows are available, while SAR is better on PTB-XL at most support sizes. On HPP, LSTA is also the best variant (53.85 vs. 52.36 for LN-TTA), whereas on MIT-BIH and Sleep-EDF all variants lie within 1.2 points of the frozen model (Appendix Table 7).

4.5 Adapting Without Harm

An adaptation method is useful in practice only if it does not damage predictions when the unlabeled support set is unreliable. We therefore evaluate LSTA in three settings in which entropy-based updates can go wrong. All methods keep the settings selected on clean validation data.

Contaminated support. We replace 25% or 50% of a subject's eight support windows with windows from another subject of a different diagnostic class, while the query windows remain

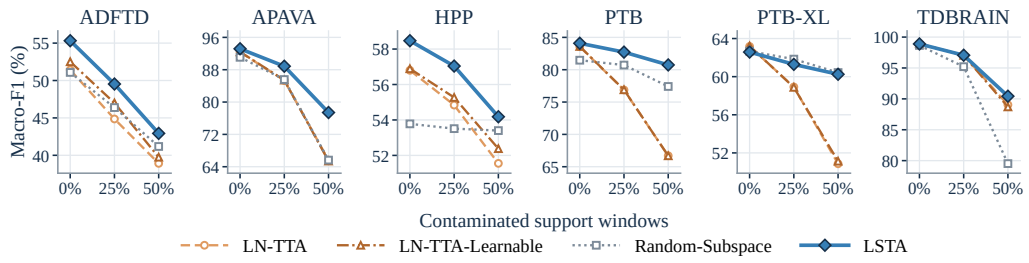


Figure 3: Robustness to support contamination across six subject-labeled datasets. We replace 0%, 25%, or 50% of the support windows with windows from another subject of a different diagnostic class, keeping query windows and adaptation settings unchanged. Curves show mean macro-F1 across five backbone seeds and five support draws per seed. Among the plotted variants, LSTA achieves the highest macro-F1 at 50% contamination on all six datasets.

unchanged (Figure 3). At 50% contamination, LSTA obtains the highest macro-F1 on all six subject-labeled datasets among the adaptation variants of Table 2 (LN-TTA, LN-TTA-Learnable and Random-Subspace). On APAVA, PTB and PTB-XL it reaches 77.4, 80.7 and 60.2, compared with 65.6, 66.7 and 50.9 for LN-TTA; on HPP it stays above the frozen model (54.18 vs. 53.16), whereas LN-TTA falls below it (51.55; Appendix Table 6). Heavy contamination still lowers LSTA below the frozen model on several datasets, so the learned subspace limits the damage from foreign support windows rather than removing it.

Harmful prediction changes. Adaptation can also turn correct predictions into errors. The fraction of initially correct query predictions that become incorrect after adaptation falls from 3.87% with LN-TTA to 2.53% with LSTA on APAVA, from 3.45% to 1.28% on PTB, and from 16.93% to 12.32% on ADFTD. Consistent with this, LSTA preserves more of the source representation than LN-TTA on all four datasets evaluated: centered kernel alignment between frozen and adapted features rises from 0.40 to 0.82 on APAVA and from 0.83 to 0.99 on PTB (Appendix A.8).

Signal corruption. Under noise, baseline wander, powerline interference and channel dropout applied at test time, the change in macro-F1 relative to the frozen model is larger for LSTA than for Tent, SAR, SHOT and LN-TTA on most datasets, particularly when support windows are less corrupted than query windows (Appendix Figure 10).

Step count and step size. LSTA is meta-trained with one step, yet its macro-F1 changes little between 1 and 32 inference steps, and its advantage over Tent, SAR, SHOT and UCon holds when each method's validation-selected step size is scaled by 0.25 to 2 on four of the five datasets (Appendix Figures 8 and 9). The gains are therefore not tied to a particular step size.

Inference cost. LSTA adds little cost to entropy minimization (Figure 4). On TDBrain it reaches the highest macro-F1 while needing roughly 15 to 30 ms per test episode, comparable to Tent and T3A, because it performs a single projected gradient step on the LayerNorm parameters. SHOT, UCon and MemBN are about an order of magnitude slower (about 230 to 420 ms) and also less accurate.

Together, these results suggest that the main advantage of the learned subspace over direct LayerNorm adaptation lies less in average accuracy on clean data than in adapting without disrupting predictions that the source model already gets right.

5 Conclusion

We argued that test-time adaptation for medical time series should learn where to adapt, rather than follow every direction of model uncertainty. LSTA implements this idea by meta-learning a low-rank LayerNorm update space from past subjects and adapting each new subject only within it. Across subject-labeled EEG and ECG benchmarks, LSTA improves over the frozen backbone and over seven adaptation methods on identical checkpoints, and a learned subspace outperforms a random one. Its most consistent benefit is robustness: with contaminated support it degrades far less

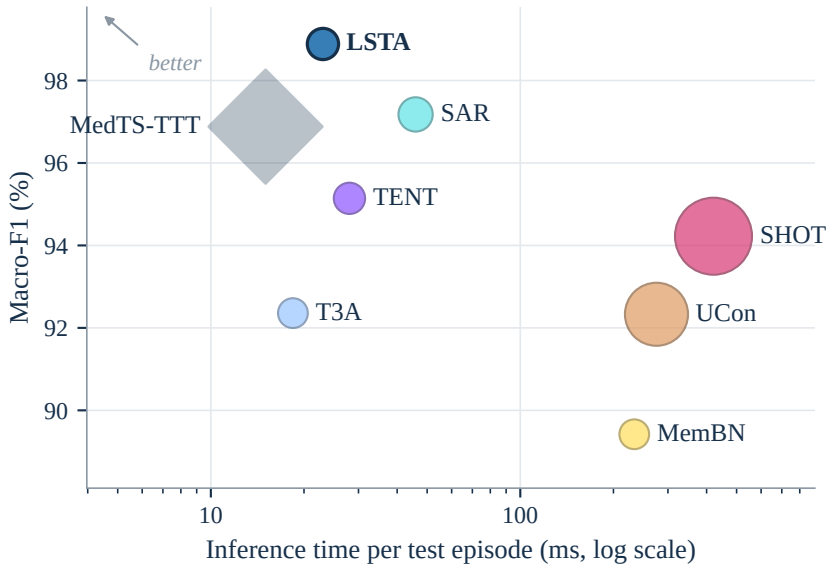


Figure 4: Performance–efficiency comparison on TDBRAIN. Test macro-F1 (y-axis) is plotted against inference time per test episode (x-axis, logarithmic scale). Bubble size indicates the FLOPs of each method, and the upper-left corner is better. LSTA achieves the highest macro-F1 among the compared methods while maintaining low inference time.

than direct LayerNorm adaptation, it changes fewer correct predictions into errors and preserves the source representation, and it is the only method we evaluate whose macro-F1 never falls below that of the frozen model on any of the eight datasets. On clean subject-labeled data its advantage over direct LayerNorm adaptation is large on ADFTD but small elsewhere, and under heavy contamination it limits, but does not remove, the loss relative to the frozen model.

Limitations. We have not measured directly whether inter-subject shift is low-rank; this premise motivates the design, and the evidence for it is indirect. Our protocol adapts offline within a single recording, uses one backbone family, and reports variability across backbone seeds rather than paired tests over subjects. When all windows of a subject share one label, aggregating frozen predictions over the support set is a strong alternative that LSTA does not always exceed. Chronological adaptation, other backbones and subject-level statistical testing remain for future work.

6 POSTSCRIPT

AI USE STATEMENT

In this work, we used generative AI tools, including ChatGPT, Claude, and GitHub Copilot to assist with developing and refining the adaptation framework, discussing its mathematical formulation, designing experiments, implementing individual functions, and organizing and debugging the codebase. These tools also assisted with manuscript structuring and proofreading and to baseline implementation. We have reviewed all AI-assisted work; for example, LLM-modified code was verified and tested for correctness. We take responsibility for the final manuscript, code, experimental claims, and other AI-assisted artifacts.

REPRODUCIBILITY STATEMENT

The Methodology and Experimental Settings sections (Sections 3 and A.4) describe the adaptation procedure, hyperparameters, random seeds, model selection, and evaluation protocol. We will release the codebase together with the exact configurations, commands and checkpoints used to obtain the final reported results.

REFERENCES

Chirag Ahuja and Divyashikha Sethia. Transit-eeg—a framework for cross-subject classification with subject specific adaptation. *IEEE Transactions on Cognitive and Developmental Systems*, 17 (4):923–937, 2025.

Hamdi Altaheri, Ghulam Muhammad, Mansour Alsulaiman, Syed Umar Amin, Ghadir Ali Altuwaijri, Wadood Abdul, Mohamed A Bencherif, and Mohammed Faisal. Deep learning techniques for classification of electroencephalogram (eeg) motor imagery (mi) signals: A review. *Neural Computing and Applications*, 35(20):14681–14722, 2023.

Yara Badr, Usman Tariq, Fares Al-Shargie, Fabio Babiloni, Fadwa Al Mughairbi, and Hasan Al-Nashash. A review on evaluating mental stress by deep learning using eeg signals. *Neural Computing and Applications*, 36(21):12629–12654, 2024.

Alexander Bartler, Andre Bühler, Felix Wiewel, Mario Döbler, and Bin Yang. Mt3: Meta test-time training for self-supervised test-time adaption. In *International Conference on Artificial Intelligence and Statistics*, pp. 3080–3090. PMLR, 2022.

Mingzhi Chen, Yiyu Gui, and Guibo Luo. Medts-ttt: Test-time training for medical time series classification. *arXiv preprint arXiv:2606.21329*, 2026.

Yuanchao Dai, Jing Wu, Yuanzhao Fan, Jin Wang, Jianwei Niu, Fei Gu, and Shigen Shen. Mseva: A musculoskeletal rehabilitation evaluation system based on emg signals. *ACM Transactions on Sensor Networks*, 19(1):1–23, 2022.

Philip De Chazal, Maria O'Dwyer, and Richard B Reilly. Automatic classification of heartbeats using ecg morphology and heartbeat interval features. *IEEE transactions on biomedical engineering*, 51(7):1196–1206, 2004.

J Escudero, Daniel Abásolo, Roberto Hornero, Pedro Espino, and Miguel López. Analysis of electroencephalograms in alzheimer's disease patients with multiscale entropy. *Physiological measurement*, 27(11):1091–1106, 2006.

Jiahao Fan, Chenglu Sun, Meng Long, Chen Chen, and Wei Chen. Eognet: A novel deep learning model for sleep stage classification based on single-channel eog signal. *Frontiers in neuroscience*, 15:573194, 2021.

Wei Fan, Jingru Fei, Dingyu Guo, Kun Yi, Xiaozhuang Song, Haolong Xiang, Hangting Ye, and Min Li. Towards multi-resolution spatiotemporal graph learning for medical time series classification. In *Proceedings of the ACM on Web Conference 2025*, pp. 5054–5064, 2025.

Ary L Goldberger, Luis AN Amaral, Leon Glass, Jeffrey M Hausdorff, Plamen Ch Ivanov, Roger G Mark, Joseph E Mietus, George B Moody, Chung-Kang Peng, and H Eugene Stanley. Physiobank, physiotoolkit, and physionet: components of a new research resource for complex physiologic signals. *circulation*, 101(23):e215–e220, 2000.

Yu Guoqi, Juncheng Wang, Chen Yang, Jing Qin, Angelica Aviles-Rivero, and Shujun Wang. Decentralized attention fails centralized signals: Rethinking transformers for medical time series. In *International Conference on Learning Representations*, volume 2026, pp. 108935–108955, 2026.

Yusuke Iwasawa and Yutaka Matsuo. Test-time classifier adjustment module for model-agnostic domain generalization. *Advances in Neural Information Processing Systems*, 34:2427–2440, 2021.

Mahboobeh Jafari, Afshin Shoeibi, Marjane Khodatars, Sara Bagherzadeh, Ahmad Shalbaf, David López García, Juan M Gorriz, and U Rajendra Acharya. Emotion recognition in eeg signals using deep learning methods: A review. *Computers in biology and medicine*, 165:107450, 2023.

Wei-Bang Jiang, Liming Zhao, and Bao-Liang Lu. Large brain model for learning generic representations with tremendous eeg data in bci. In *International Conference on Learning Representations*, volume 2024, pp. 16405–16426, 2024.

Ming Jin, Danni Zhang, Gangming Zhao, Changde Du, and Jinpeng Li. Evofa: Evolvable fast adaptation for eeg emotion recognition. *arXiv preprint arXiv:2409.15733*, 2024.

Juwon Kang, Nayeong Kim, Jungseul Ok, and Suha Kwak. Membn: Robust test-time adaptation via batch norm with statistics memory. In *European Conference on Computer Vision*, pp. 467–483. Springer, 2024.

Bob Kemp, Aeilko H Zwinderman, Bert Tuk, Hilbert AC Kamphuisen, and Josefien JL Oberye. Analysis of a sleep-dependent neuronal feedback loop: the slow-wave microcontinuity of the eeg. *IEEE Transactions on Biomedical Engineering*, 47(9):1185–1194, 2000.

Nikita Kitaev, Łukasz Kaiser, and Anselm Levskaya. Reformer: The efficient transformer. *arXiv preprint arXiv:2001.04451*, 2020.

Timon Klein, Piotr Minakowski, Sebastian Sager, and Steffen Schotthöfer. Mitigating subject dependency in eeg decoding with subject-specific low-rank adapters. *arXiv preprint arXiv:2510.08059*, 2025.

Yehuda Koren, Robert Bell, and Chris Volinsky. Matrix factorization techniques for recommender systems. *Computer*, 42(8):30–37, 2009.

Siyang Li, Ziwei Wang, Hanbin Luo, Lieyun Ding, and Dongrui Wu. T-time: Test-time information maximization ensemble for plug-and-play bcis. *IEEE Transactions on Biomedical Engineering*, 71(2):423–432, 2023.

Siyang Li, Jiayi Ouyang, Zhenyao Cui, Ziwei Wang, Tianwang Jia, Feng Wan, and Dongrui Wu. Backpropagation-free test-time adaptation for lightweight eeg-based brain-computer interfaces. *arXiv preprint arXiv:2601.07556*, 2026.

Jian Liang, Dapeng Hu, and Jiashi Feng. Do we really need to access the source data? source hypothesis transfer for unsupervised domain adaptation. In *International conference on machine learning*, pp. 6028–6039. PMLR, 2020.

Xinwen Liu, Huan Wang, Zongjin Li, and Lang Qin. Deep learning in ecg diagnosis: A review. *Knowledge-Based Systems*, 227:107187, 2021.

Yong Liu, Haixu Wu, Jianmin Wang, and Mingsheng Long. Non-stationary transformers: Exploring the stationarity in time series forecasting. *Advances in neural information processing systems*, 35: 9881–9893, 2022.

Yong Liu, Tengge Hu, Haoran Zhang, Haixu Wu, Shiyu Wang, Lintao Ma, and Mingsheng Long. itransformer: Inverted transformers are effective for time series forecasting. In *International conference on learning representations*, volume 2024, pp. 11116–11140, 2024.

Andreas Miltiadous, Katerina D Tzimourta, Theodora Afrantou, Panagiotis Ioannidis, Nikolaos Grigoriadis, Dimitrios G Tsalikakis, Pantelis Angelidis, Markos G Tsipouras, Euripidis Glavas, Nikolaos Giannakeas, et al. A dataset of scalp eeg recordings of alzheimer's disease, frontotemporal dementia and healthy subjects from routine eeg. *Data*, 8(6):95, 2023.

George B Moody and Roger G Mark. The impact of the mit-bih arrhythmia database. *IEEE engineering in medicine and biology magazine*, 20(3):45–50, 2001.

Elon Musk et al. An integrated brain-machine interface platform with thousands of channels. *Journal of medical Internet research*, 21(10):e16194, 2019.

Yuqi Nie, Nam H Nguyen, Phanwadee Sinthong, and Jayant Kalagnanam. A time series is worth 64 words: Long-term forecasting with transformers. *arXiv preprint arXiv:2211.14730*, 2022.

Shuaicheng Niu, Jiaxiang Wu, Yifan Zhang, Zhiquan Wen, Yaofo Chen, Peilin Zhao, and Mingkui Tan. Towards stable test-time adaptation in dynamic wild world. *arXiv preprint arXiv:2302.12400*, 2023.

Eunbyung Park and Junier B Oliva. Meta-curvature. *Advances in neural information processing systems*, 32, 2019.

Junghwan Park, Woojin Cho, Junhyuk Heo, Darongsae Kwon, and Kookjin Lee. Basis-oriented low-rank transfer for few-shot and test-time adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 860–870, 2026.

Lee Reicher, Smadar Shilo, Anastasia Godneva, Guy Lutsker, Liron Zahavi, Saar Shoer, David Krongauz, Michal Rein, Sarah Kohn, Tomer Segev, et al. Deep phenotyping of health–disease continuum in the human phenotype project. *Nature Medicine*, 31(9):3191–3203, 2025.

Smadar Shilo, Noam Bar, Ayya Keshet, Yeela Talmor-Barkan, Hagai Rossman, Anastasia Godneva, Yaron Aviv, Yochai Edlitz, Lee Reicher, Dmitry Kolobkov, et al. 10 k: a large-scale prospective longitudinal study in israel. *European journal of epidemiology*, 36(11):1187–1194, 2021.

Akara Supratak, Hao Dong, Chao Wu, and Yike Guo. Deepsleepnet: A model for automatic sleep stage scoring based on raw single-channel eeg. *IEEE transactions on neural systems and rehabilitation engineering*, 25(11):1998–2008, 2017.

Siyi Tang, Jared A Dunnmon, Khaled Saab, Xuan Zhang, Qianying Huang, Florian Dubost, Daniel L Rubin, and Christopher Lee-Messer. Self-supervised graph neural networks for improved electroencephalographic seizure analysis. *arXiv preprint arXiv:2104.08336*, 2021.

Hanneke Van Dijk, Guido Van Wingen, Damiaan Denys, Sebastian Olbrich, Rosalinde Van Ruth, and Martijn Arns. The two decades brainclinics research archive for insights in neurophysiology (tdbrain) database. *Scientific data*, 9(1):333, 2022.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Patrick Wagner, Nils Strodthoff, Ralf-Dieter Bousseljot, Dieter Kreiseler, Fatima I Lunze, Wojciech Samek, and Tobias Schaeffter. Ptb-xl, a large publicly available electrocardiography dataset. *Scientific data*, 7(1):154, 2020.

Dequan Wang, Evan Shelhamer, Shaoteng Liu, Bruno Olshausen, and Trevor Darrell. Tent: Fully test-time adaptation by entropy minimization. *arXiv preprint arXiv:2006.10726*, 2020.

Yihe Wang, Nan Huang, Taida Li, Yujun Yan, and Xiang Zhang. Medformer: A multi-granularity patching transformer for medical time-series classification. *Advances in Neural Information Processing Systems*, 37:36314–36341, 2024.

Martin Wimpff, Mario Döbler, and Bin Yang. Calibration-free online test-time adaptation for electroencephalography motor imagery decoding. In *2024 12th International Winter Conference on Brain-Computer Interface (BCI)*, pp. 1–6. IEEE, 2024.

Haixu Wu, Jiehui Xu, Jianmin Wang, and Mingsheng Long. Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting. *Advances in neural information processing systems*, 34:22419–22430, 2021.

Qiao Xiao, Khuan Lee, Siti Aisah Mokhtar, Iskasymar Ismail, Ahmad Luqman bin Md Pauzi, Qiuxia Zhang, and Poh Ying Lim. Deep learning-based ecg arrhythmia classification: A systematic review. *Applied Sciences*, 13(8):4964, 2023.

Gezheng Xu, Hui Guo, Li Yi, Charles Ling, Boyu Wang, and Grace Yi. Revisiting source-free domain adaptation: a new perspective via uncertainty control. In *International Conference on Learning Representations*, volume 2025, pp. 92900–92939, 2025.

Chaoqi Yang, M Westover, and Jimeng Sun. Biot: Biosignal transformer for cross-data learning in the wild. *Advances in Neural Information Processing Systems*, 36:78240–78260, 2023a.

Chaoqi Yang, M Brandon Westover, and Jimeng Sun. Manydg: Many-domain generalization for healthcare applications. *arXiv preprint arXiv:2301.08834*, 2023b.

Marvin Zhang, Henrik Marklund, Nikita Dhawan, Abhishek Gupta, Sergey Levine, and Chelsea Finn. Adaptive risk minimization: Learning to adapt to domain shift. *Advances in neural information processing systems*, 34:23664–23678, 2021.

Yitian Zhang, Liheng Ma, Soumyasundar Pal, Yingxue Zhang, and Mark Coates. Multi-resolution time-series transformer for long-term forecasting. In *International conference on artificial intelligence and statistics*, pp. 4222–4230. PMLR, 2024.

Haoyi Zhou, Shanghang Zhang, Jieqi Peng, Shuai Zhang, Jianxin Li, Hui Xiong, and Wancai Zhang. Informer: Beyond efficient transformer for long sequence time-series forecasting. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pp. 11106–11115, 2021.

Tian Zhou, Ziqing Ma, Qingsong Wen, Xue Wang, Liang Sun, and Rong Jin. Fedformer: Frequency enhanced decomposed transformer for long-term series forecasting. In *International conference on machine learning*, pp. 27268–27286. PMLR, 2022.

A Appendix

A.1 Comparison with Published Classifiers

Table 3 compares LSTA with the published classifiers. LSTA obtains the highest mean accuracy, macro-precision, macro-recall and macro-F1 on all five subject-labeled datasets. Relative to the published TeCh results, macro-F1 increases by 6.48 points on ADFTD (48.84 to 55.32), 6.88 on APAVA (86.30 to 93.18), 5.68 on TDBrain (93.20 to 98.88), 2.12 on PTB (81.97 to 84.09) and 2.22 on PTB-XL (62.44 to 64.66). The gains are not uniform across metrics: on PTB-XL, AUROC (88.66 vs. 90.03) and AUPRC (66.18 vs. 67.19) are lower than those of TeCh. Because LSTA also uses unlabeled windows from each subject, this comparison is indicative only; Table 1 gives the comparison under an identical protocol.

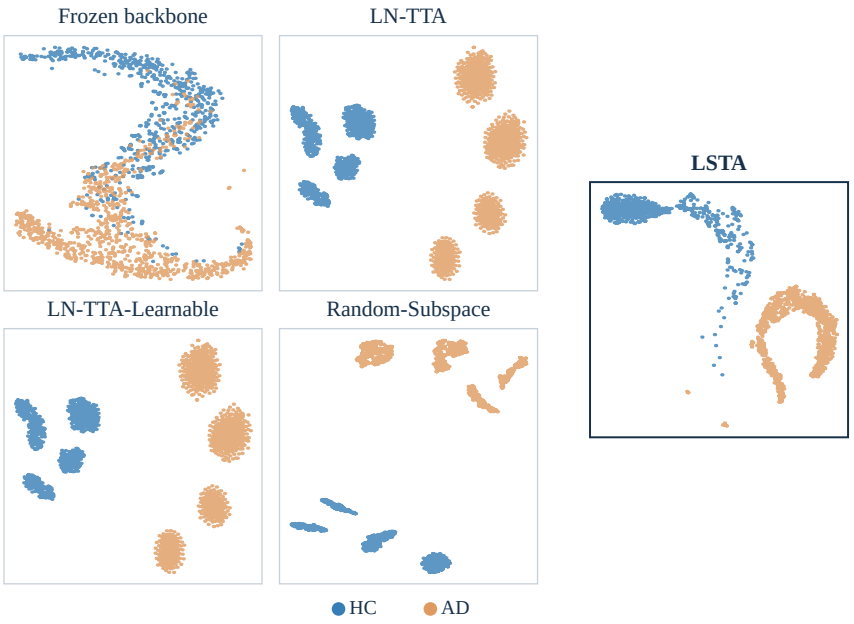


Figure 5: t-SNE visualization of pre-classifier features for Frozen and LN-TTA (top row), LN-TTA-Learnable and Random subspace (bottom row), and LSTA (right). Each point represents a query window, coloured by diagnosis. A shared t-SNE embedding is fitted jointly across all five methods for the same seed and query windows. LSTA retains a visual structure closer to the frozen backbone, whereas the other adaptation variants produce more compact, disconnected clusters.

A.2 Extended Related Work

Test-Time Adaptation and Meta-Learning. Test-time adaptation (TTA) adapts a pretrained model to distribution shifts using unlabeled test observations. Tent (Wang et al., 2020) minimizes prediction entropy by optimizing the parameters of normalization layers, while SAR (Niu et al., 2023) improves stability through sharpness-aware optimization and selective entropy minimization that excludes samples with unreliable gradients. For physiological time series, Wimpff et al. (2024) investigate a calibration-free approach for EEG motor imagery decoding by minimizing entropy to

Table 3: Results on five MedTS datasets with subject-exclusive training, validation, and test splits. Published baseline results are taken from Guoqi et al. (2026). The highest and second-highest means are bold and underlined, respectively. Avg is the unweighted mean of the six metrics. Published methods classify each window independently, whereas LSTA additionally uses up to $k = 8$ unlabeled windows from the same subject; see Appendix Table 8 for adaptation methods under an identical protocol.

Datasets	Models	Accuracy	Precision	Recall	F1 score	AUROC	AUPRC	Avg
ADFTD (3 classes)	Autoformer	45.25±1.48	43.67±1.94	42.96±2.03	42.59±1.85	61.02±1.82	43.10±2.30	46.43±1.90
	FEDformer	46.30±0.59	46.05±0.76	44.22±1.38	43.91±1.37	62.62±1.75	46.11±1.44	48.20±1.22
	Informer	48.45±1.96	46.54±1.68	46.06±1.84	45.74±1.38	65.87±1.27	47.60±1.30	50.04±1.57
	iTransformer	52.60±1.59	46.79±1.27	47.28±1.29	46.79±1.13	67.26±1.16	49.53±1.21	51.71±1.28
	MTST	45.60±2.03	44.70±1.33	45.05±1.30	44.31±1.74	62.50±0.81	45.16±0.85	47.89±1.34
	Nonformer	49.95±1.05	47.71±0.97	47.46±1.50	46.96±1.35	66.23±1.37	47.33±1.78	50.94±1.34
	PatchTST	44.37±0.95	42.40±1.13	42.06±1.48	41.97±1.37	60.08±1.50	42.49±1.79	45.56±1.37
	Reformer	50.78±1.17	49.64±1.49	49.89±1.67	47.94±0.69	69.17±1.58	51.73±1.94	53.19±1.42
	Transformer	50.47±2.14	49.13±1.83	48.01±1.53	48.09±1.59	67.93±1.59	48.93±2.02	52.09±1.78
	Medformer	53.27±1.54	51.02±1.57	50.71±1.55	50.65±1.51	70.93±1.19	51.21±1.32	54.63±1.45
	TeCh	54.54±0.70	53.02±0.87	49.25±1.01	48.84±1.72	68.67±1.05	50.62±1.26	54.16±1.10
	LSTA	**62.30**±1.49	**64.05**±5.36	**56.26**±1.41	**55.32**±1.86	**74.83**±2.86	**58.08**±3.91	**61.81**±2.53
APAVA (2 classes)	Autoformer	68.64±1.82	68.48±2.10	68.77±2.27	68.06±1.94	75.94±3.61	74.38±4.05	70.71±2.63
	FEDformer	74.94±2.15	74.59±1.50	73.56±3.55	73.51±3.39	83.72±1.97	82.94±2.37	77.21±2.49
	Informer	73.11±4.40	75.17±6.06	69.17±4.56	69.47±5.06	70.46±4.91	70.75±5.27	71.36±5.04
	iTransformer	74.55±1.66	74.77±2.10	71.76±1.72	72.30±1.79	85.59±1.55	84.39±1.57	77.23±1.73
	MTST	71.14±1.59	79.30±0.97	65.27±2.28	64.01±3.16	68.87±2.34	71.06±1.60	69.94±1.99
	Nonformer	71.89±3.81	71.80±4.58	69.44±3.56	69.74±3.84	70.55±2.96	70.78±4.08	70.70±3.81
	PatchTST	67.03±1.65	78.76±1.28	59.91±2.02	55.97±3.10	65.65±0.28	67.99±0.76	65.89±1.52
	Reformer	78.70±2.00	82.50±3.95	75.00±1.61	75.93±1.82	73.94±1.40	76.04±1.14	77.02±1.99
	Transformer	76.30±4.72	77.64±5.95	73.09±5.01	73.75±5.38	72.50±6.60	73.23±7.60	74.42±5.88
	Medformer	78.74±0.64	81.11±0.84	75.40±0.66	76.31±0.71	83.20±0.91	83.66±0.92	79.74±0.78
	TeCh	86.86±1.09	86.85±1.29	86.10±1.00	86.30±1.06	94.02±0.52	93.79±0.56	88.99±0.92
	LSTA	**93.70**±3.23	**95.23**±2.15	**92.41**±4.03	**93.18**±3.60	**97.58**±1.94	**97.36**±2.14	**94.91**±2.76
TDBrain (2 classes)	Autoformer	87.33±3.79	88.06±3.56	87.33±3.79	87.26±3.84	93.81±2.26	93.32±2.42	89.52±3.28
	FEDformer	78.13±1.98	78.52±1.91	78.13±1.98	78.04±2.01	86.56±1.86	86.48±1.99	80.98±1.96
	Informer	89.02±2.50	89.43±2.14	89.02±2.50	88.98±2.54	96.64±0.68	96.75±0.63	91.64±1.83
	iTransformer	74.67±1.06	74.71±1.06	74.67±1.06	74.65±1.06	83.37±1.14	83.73±1.27	77.63±1.11
	MTST	76.96±3.76	77.24±3.59	76.96±3.76	76.88±3.83	85.27±4.46	82.81±5.64	79.35±4.17
	Nonformer	87.88±2.48	88.86±1.84	87.88±2.48	87.78±2.56	97.05±0.68	96.99±0.68	91.07±1.79
	PatchTST	79.25±3.79	79.60±4.09	79.25±3.79	79.20±3.77	87.95±4.96	86.36±6.67	81.94±4.51
	Reformer	87.92±2.01	88.64±1.40	87.92±2.01	87.85±2.08	96.30±0.54	96.40±0.45	90.84±1.42
	Transformer	87.17±1.67	87.99±1.68	87.17±1.67	87.10±1.68	96.28±0.92	96.34±0.81	90.34±1.41
	Medformer	89.62±0.81	89.68±0.78	89.62±0.81	89.62±0.81	96.41±0.35	96.51±0.33	91.91±0.65
	TeCh	93.21±0.61	93.39±0.58	93.21±0.61	93.20±0.61	98.68±0.19	98.72±0.17	95.07±0.46
	LSTA	**98.88**±0.76	**98.95**±0.69	**98.88**±0.76	**98.88**±0.76	**99.98**±0.01	**99.98**±0.01	**99.26**±0.50
PTB (2 classes)	Autoformer	73.35±2.10	72.11±2.89	63.24±3.17	63.69±3.84	78.54±3.48	74.25±3.53	70.86±3.17
	FEDformer	76.05±2.54	77.58±3.61	66.10±3.55	67.14±4.37	85.93±4.31	82.59±5.42	75.90±3.97
	Informer	78.69±1.68	82.87±1.02	69.19±2.90	70.84±3.47	92.09±0.53	90.02±0.60	80.62±1.70
	iTransformer	83.89±0.71	88.25±1.18	76.39±1.01	79.06±1.06	91.18±1.16	90.93±0.98	84.95±1.02
	MTST	76.59±1.90	79.88±1.90	66.31±2.95	67.38±3.71	86.86±2.75	83.75±2.84	76.80±2.68
	Nonformer	78.66±0.49	82.77±0.86	69.12±0.87	70.90±1.00	89.37±2.51	86.67±2.38	79.58±1.35
	PatchTST	74.74±1.62	76.94±1.51	63.89±2.71	64.36±3.38	88.79±0.91	83.39±0.96	75.35±1.85
	Reformer	77.96±2.13	81.72±1.61	68.20±3.35	69.65±3.88	91.13±0.74	88.42±1.30	79.51±2.17
	Transformer	77.37±1.02	81.84±0.66	67.14±1.80	68.47±2.19	90.08±1.76	87.22±1.68	78.69±1.52
	Medformer	83.50±2.01	85.19±0.94	77.11±3.39	79.18±3.31	92.81±1.48	90.32±1.54	84.69±2.11
	TeCh	85.96±2.52	89.92±0.74	79.43±4.13	81.97±4.07	94.57±0.70	94.36±0.66	87.70±2.14
	LSTA	**87.48**±2.46	**91.45**±1.48	**81.39**±3.72	**84.09**±3.53	**95.07**±0.31	**95.17**±0.34	**89.11**±1.92
PTB-XL (5 classes)	Autoformer	61.68±2.72	51.60±1.64	49.10±1.52	48.85±2.27	82.04±1.44	51.93±1.71	57.53±1.88
	FEDformer	57.20±9.47	52.38±6.09	49.04±7.26	47.89±8.44	82.13±4.17	52.31±7.03	56.83±7.08
	Informer	71.43±0.32	62.64±0.60	59.12±0.47	60.44±0.43	88.65±0.09	64.76±0.17	67.84±0.35
	iTransformer	69.28±0.22	59.59±0.45	54.62±0.18	56.20±0.19	86.71±0.10	60.27±0.21	64.45±0.23
	MTST	72.14±0.27	63.84±0.72	60.01±0.81	61.43±0.38	88.97±0.33	65.83±0.51	68.70±0.50
	Nonformer	70.56±0.55	61.57±0.66	57.75±0.72	59.10±0.66	88.32±0.36	63.40±0.79	66.78±0.62
	PatchTST	73.23±0.25	65.70±0.64	60.82±0.76	62.61±0.34	89.74±0.19	**67.32**±0.22	69.90±0.40
	Reformer	71.72±0.43	63.12±1.02	59.20±0.75	60.69±0.18	88.80±0.24	64.72±0.47	68.04±0.52
	Transformer	70.59±0.44	61.57±0.65	57.62±0.35	59.05±0.25	88.21±0.16	63.36±0.29	66.73±0.36
	Medformer	72.87±0.23	64.14±0.42	60.60±0.46	62.02±0.37	89.66±0.13	66.39±0.22	69.28±0.31
	TeCh	73.53±0.07	65.92±0.52	60.61±0.59	62.44±0.27	**90.03**±0.12	67.19±0.25	69.95±0.30
	LSTA	**75.33**±0.35	**68.51**±0.87	**62.67**±0.65	**64.66**±0.51	88.66±0.33	66.18±0.46	**71.00**±0.37

address shifts between subjects and sessions. Similarly, Li et al. (2023) adapt to unlabeled trials from unseen subjects by conditional entropy minimization and adaptive marginal distribution regularization for calibration-free BCIs. More recently, Li et al. (2026) propose backpropagation-free adaptation for EEG decoding, combining predictions from transformed inputs through a reliability-ranking module trained on source data. Beyond applying unsupervised objectives directly at inference, MT3 (Bartler et al., 2022) uses meta-learning to train models for self-supervised test-time adaptation for corrupted image classification.

Related work explores low-rank parameter updates and meta-learning. SuLoRA (Klein et al., 2025) jointly learns shared weights and subject specific adapters, whereas LSTA keeps the source parameters frozen and meta-learns a shared update space for unlabeled adaptation to unseen subjects. TRANSIT-EEG (Ahuja & Sethia, 2025) combines diffusion-based EEG data augmentation with LoRA fine-tuning for new subjects. Beyond adapter parameterization, Meta-Curvature (Park & Oliva, 2019) learns gradient transformations for adaptation to new tasks, while BOLT (Park et al., 2026) adapts to new tasks within a fixed low-rank subspace extracted from previously fine-tuned models. Meta-learning has also been studied for physiological signals. Jin et al. (2024) combine few-shot learning with test-time alignment to source-data subsets. These studies motivate learning where adaptation should occur and how its updates should be shaped. Adaptive risk minimization (Zhang et al., 2021) also meta-learns how to adapt from unlabeled groups of test points; LSTA treats each subject as such a group but adapts only through a low-rank LayerNorm subspace. Our approach learns a shared low-dimensional LayerNorm update space specifically for unlabeled subject adaptation. With the pretrained source parameters fixed, we optimize the basis using classification loss on separate labeled query windows after an entropy-based support update. At inference, the basis remains fixed, and subject-specific coefficients are obtained from unlabeled support windows without access to source recordings or target labels.

Medical Time Series Classification. Medical time series record physiological signals over time and support disease diagnosis (Liu et al., 2021; Xiao et al., 2023), health monitoring (Jafari et al., 2023; Badr et al., 2024), and brain-computer interfaces (BCIs) (Musk et al., 2019; Altaheri et al., 2023). Common modalities include EEG (Tang et al., 2021; Yang et al., 2023b), ECG (Xiao et al., 2023), EMG (Dai et al., 2022) and EOG (Fan et al., 2021), each capturing distinct information for diagnosis. MedTS classification aims to extract relevant patterns from these signals, including those associated with neurological conditions and cardiac abnormalities.

Recent approaches improve classification by learning temporal and cross-channel dependencies within input windows. Medformer (Wang et al., 2024) combines cross-channel patching with multi-granularity self-attention to capture patterns at different scales. MedGNN (Fan et al., 2025) uses multi-resolution spatiotemporal graph learning. TeCh (Guoqi et al., 2026) introduces core token attention aggregation and redistribution to model interactions among tokens. Another direction learns transferable representations through pretraining. BIOT (Yang et al., 2023a) enables cross-data learning with mismatched channels, variable lengths, and missing values, while LaBram (Jiang et al., 2024) uses vector-quantized neural spectrum prediction to train a semantically rich neural tokenizer that encodes continuous raw EEG channel patches into compact neural codes.

Evaluating these representations on unseen subjects requires subject-disjoint splits. Medformer and TeCh adopt subject-independent evaluation, separating subject across training, validation, and test splits (Wang et al., 2024; Guoqi et al., 2026). Under their standard inference protocol, individual windows are classified using fixed model parameters. Our work complements these representation-learning approaches by incorporating subject-specific support observations through test-time adaptation, with predictions evaluated on separate windows from the same individual.

A.3 Datasets

We evaluate on eight medical time series datasets in two settings. All splits are subject-exclusive, so no subject contributes windows to more than one split. Table 4 summarizes the datasets and whether labels are assigned per subject or per window.

Subject-labeled datasets. In five benchmarks every window inherits the diagnosis of its subject, and we adopt the splits of Guoqi et al. (2026). **APAVA** (Escudero et al., 2006) is an EEG dataset with a binary label indicating whether the subject has Alzheimer's disease. For **TDBrain** (Van Dijk

et al., 2022), we use a binary Parkinson's disease subset of the EEG dataset. **ADFTD** (Miltiadous et al., 2023) is an EEG dataset with three classes: healthy, frontotemporal dementia and Alzheimer's disease. **PTB** (Goldberger et al., 2000) is an ECG dataset with a binary label for myocardial infarction. **PTB-XL** (Wagner et al., 2020) is an ECG dataset with five diagnostic classes. We additionally evaluate on the ECG subset of the **Human Phenotype Project (HPP)** (Reicher et al., 2025; Shilo et al., 2021), extending the evaluation to a separate cohort with five machine-generated ECG interpretation categories: normal ECG, ECG without significant abnormalities, normal variant of ECG, possibly abnormal ECG, and abnormal ECG. We selected participants whose category is consistent across visits and construct subject independant splits, as detailed in the Appendix A.3.

Window-labeled datasets. To test adaptation when a subject's support windows belong to different classes, we add two datasets in which each window carries its own label. **MIT-BIH** (Moody & Mark, 2001; Goldberger et al., 2000) is a two-lead ECG arrhythmia dataset with four beat classes. **Sleep-EDF** (Kemp et al., 2000; Goldberger et al., 2000) contains polysomnography recordings in which each window is labeled with a sleep stage.

Table 4: Dataset summary. Windows are the input samples; length denotes the number of time points per window. Label indicates whether all windows of a subject share one label (subject) or each window has its own label (window).

Dataset	Subjects	Windows	Classes	Channels	Length	Label
ADFTD	88	69,752	3	19	256	subject
APAVA	23	5,967	2	16	256	subject
TDBrain	72	6,240	2	33	256	subject
PTB	198	64,356	2	15	300	subject
PTB-XL	17,596	191,400	5	12	250	subject
MIT-BIH	43	98,755	4	2	300	window
Sleep-EDF	78	195,469	5	3	3,000	window
HPP	7,139	150,860	5	12	250	subject

A.4 Experimental Settings

Backbone and checkpoints. For each dataset, we train TeCh checkpoints using its published architecture configuration and the training epochs and early-stopping patience specified in its code repository. We use five independently trained checkpoints per dataset, with seeds 42–46.

Hyperparameter selection. We fix the subspace rank to $r = 8$ and use up to $k = 8$ support windows per subject, or all available windows when a subject has fewer. Because windows are extracted with a fixed stride, we skip intermediate windows when sampling the support set so that support and query windows do not overlap. We use a single entropy step both in adaptation training and at test time. The meta-training learning rate is fixed to 10^{-4}, and the adaptation step size is selected from $\eta \in \{2, 4, 8, 16\}$ for each dataset and backbone seed with the validation criterion described below; the same grid is used for all datasets. The entropy temperature is set to $\tau = 5$ for datasets whose validation macro F1 exceeds 0.7 and to $\tau = 1$ otherwise. Due to larger number of subjects in PTB-XL and HPP datasets we used fixed subset of 300 subjects to do evaluations for the sections Appendix A.7 and A.8. For checkpoint selected we used a fixed subset of 400 subjects from the same two datasets. Random seed 0 used for both selection.

Adaptation training. For each backbone checkpoint, we learn a separate adaptation matrix while all pretrained model parameters remain fixed. We optimize the matrix with Adam for up to 1,000 steps and evaluate every 100 steps. Each episode uses up to 32 query windows and one support adaptation step. Each evaluation averages five support draws (seeds 1000–1004), and we keep the checkpoint with the highest validation macro-F1. For PTB-XL and HPP, datasets which have larger number of subject count, validation uses 400 subjects sampled with seed 0.

Test-time evaluation. For each test subject, we divide the retained windows into two fixed, disjoint pools. We sample up to k support windows from one pool and evaluate the adapted classifier

on all windows in the other set, then reverse the pool roles. Each direction starts independently from the source checkpoint. We repeat this procedure for five support draws with seeds 1000–1004, to measure sensitivity to support selection. Also we keep the pools fixed for fair evaluation. Test labels are used only to compute evaluation metrics.

Baselines. We compare against two groups of methods. The first group consists of eleven published classifiers, Autoformer (Wu et al., 2021), FEDformer (Zhou et al., 2022), Informer (Zhou et al., 2021), iTransformer (Liu et al., 2024), MTST (Zhang et al., 2024), Nonformer (Liu et al., 2022), PatchTST (Nie et al., 2022), Reformer (Kitaev et al., 2020), the vanilla Transformer (Vaswani et al., 2017), Medformer (Wang et al., 2024) and TeCh (Guoqi et al., 2026), whose results we take from Guoqi et al. (2026). These methods classify each window independently, whereas LSTA also uses up to k unlabeled windows from the same subject, so this comparison indicates overall performance rather than isolating the effect of adaptation. The second group consists of adaptation methods that we run on the same TeCh checkpoints, support draws and query windows as LSTA: Tent (Wang et al., 2020), SAR (Niu et al., 2023), T3A (Iwasawa & Matsuo, 2021), MemBN (Kang et al., 2024), SHOT (Liang et al., 2020), UCon (Xu et al., 2025) and MedTS-TTT (Chen et al., 2026). We also report frozen prediction aggregation (Frozen-PA), which averages the frozen model's predicted probabilities over the support windows and compute the average with prediction for query window. Implementation details are given in Appendix A.

Implementation. We report the mean and standard deviation across five independently trained backbone checkpoints (seeds 42–46). For each checkpoint, we learn a separate adaptation matrix with basis initialization seed 0 and episode sampling seed 1. All adaptation experiments run on an NVIDIA RTX PRO 6000 Blackwell GPU.

A.5 Data preprocessing

We obtained preprocessed APAVA, ADFTD, TDBrain, PTB and PTB-XL datasets from **Medformer** (Wang et al., 2024).

A.5.1 APAVA Preprocessing

The Alzheimer's Patients' Relatives Association of Valladolid (APAVA) dataset[1], referenced in the paper (Escudero et al., 2006), is a public EEG time series dataset with 2 classes and 23 subjects, including 12 Alzheimer's disease patients and 11 healthy control subjects. On average, each subject has 30.0 ± 12.5 trials, with each trial being a 5-second time sequence consisting of 1280 timestamps across 16 channels. Before further preprocessing, each trial is scaled using the standard scaler. Subsequently, we segment each trial into 9 half-overlapping samples, where each sample is a 1-second time sequence comprising 256 timestamps. This process results in 5,967 samples. Each sample has a subject ID to indicate its originating subject. For the training, validation, and test set splits, we employ the Subject-Independent setup. Samples with subject IDs 15,16,19,20 and 1,2,17,18 are assigned to the validation and test sets, respectively. The remaining samples are allocated to the training set.

A.5.2 TDBRAIN Preprocessing

TDBrain dataset[2] that is referenced in the paper (Van Dijk et al., 2022), is a large EEG time series dataset recording brain activities of 1274 subjects with 33 channels. Each subject has two trials: one under eye open and one under eye closed setup. The dataset includes a total of 60 labels, with each subject potentially having multiple labels indicating multiple diseases simultaneously. In this paper, we utilize a subset of this dataset containing 25 subjects with Parkinson's disease and 25 healthy controls, all under the eye-closed task condition. Each eye-closed trial is segmented into non-overlapping 1-second samples with 256 timestamps, and any samples shorter than 1 second are discarded. This process results in 6,240 samples. Each sample is assigned a subject ID to indicate its originating subject. For the training, validation, and test set splits, we employ the Subject-Independent setup. Samples with subject IDs 18,19,20,21,46,47,48,49 are assigned to the

[1] https://osf.io/jbysn/

[2] https://brainclinics.com/resources/

validation set, while samples with subject IDs 22,23,24,25,50,51,52,53 are assigned to the test set. The remaining samples are allocated to the training set

A.5.3 ADFTD PREPROCESSING

The Alzheimer's Disease and FronTotemporal Dementia (ADFTD) dataset[3], referenced in the papers (Miltiadous et al., 2023), is a public EEG time series dataset with 3 classes, including 36 Alzheimer's disease (AD) patients, 23 Frontotemporal Dementia (FTD) patients, and 29 healthy control (HC) subjects. Dataset has 19 channels, and collected at the sampling rate of 500Hz. Each subject has a trial, with durations of approximately 13.5 minutes for AD subjects (min=5.1, max=21.3), 12 minutes for FD subjects (min=7.9, max=16.9), and 13.8 minutes for HC subjects (min=12.5, max=16.5). To mitigate the noisy components bandpass filter between 0.5-45Hz is applied to each trial. We downsample each trial to 256Hz and segment them into non-overlapping 1-second samples with 256 timestamps, discarding any samples shorter than 1 second. This process results in 69,752 samples. For the training, validation, and test set splits, we employ the Subject-Independent setup by allocating 60%, 20%, and 20% of total subjects with their corresponding samples into the training, validation, and test sets, respectively.

A.5.4 PTB PREPROCESSING

The PTB dataset[4], referenced in the paper (Goldberger et al., 2000), is a publicly accessible ECG time series recording from 290 subjects, with 15 channels and a total of 8 labels representing 7 heart diseases and 1 healthy control. The raw sampling rate is 1000Hz. For this paper, we utilize a subset of 198 subjects, including patients with Myocardial infarction and healthy control subjects. We first downsample the sampling frequency to 250Hz and normalize the ECG signals using standard scalers. Subsequently, we process the data into single heartbeats through several steps. We identify the R-Peak intervals across all channels and remove any outliers. Each heartbeat is then sampled from its R-Peak position, and we ensure all samples have the same length by applying zero padding to shorter samples, with the maximum duration across all channels serving as the reference. This process results in 64,356 samples. For the training, validation, and test set splits, we employ the Subject-Independent setup. Specifically, we allocate 60%, 20%, and 20% of the total subjects, along with their corresponding samples, into the training, validation, and test sets, respectively.

A.5.5 PTB-XL PREPROCESSING

The PTB-XL dataset[5], referenced in the paper (Wagner et al., 2020), is a large public ECG time series dataset recorded from 18,869 subjects, with 12 channels and 5 labels representing 4 heart diseases and 1 healthy control category. Each subject may have one or more trials. To ensure consistency, we discard subjects with varying diagnosis results across different trials, resulting in 17,596 subjects remaining. The raw trials consist of 10-second time intervals, with sampling frequencies of 100Hz and 500Hz versions. For our paper, we utilize the 500Hz version, then we downsample to 250Hz and normalize using standard scalers. Subsequently, each trial is segmented into non overlapping 1-second samples with 250 timestamps, discarding any samples shorter than 1 second. This process results in 191,400 samples. For the training, validation, and test set splits, we employ the Subject-Independent setup. Specifically, we allocate 60%, 20%, and 20% of the total subjects, along with their corresponding samples, into the training, validation, and test sets, respectively.

A.5.6 SLEEP-EDF PREPROCESSING

The Sleep-EDF dataset[6], referenced in the paper (Kemp et al., 2000; Goldberger et al., 2000), is a public polysomnography dataset. We use its Sleep Cassette study, which contains 153 whole-night recordings from 78 healthy subjects aged 25–101, with two nights for most subjects. We use three channels, EEG Fpz-Cz, EEG Pz-Oz and horizontal EOG, all recorded at 100Hz, so each 30-second epoch becomes one sample with 3,000 timestamps. Following standard sleep-staging practice (Supratak et al., 2017), we merge stages 3 and 4 into N3, which gives five classes (W, N1,

[3] https://openneuro.org/datasets/ds004504/versions/1.0.6

[4] https://physionet.org/content/ptbdb/1.0.0/

[5] https://physionet.org/content/ptb-xl/1.0.3/

[6] https://physionet.org/content/sleep-edfx/1.0.0/

N2, N3 and REM). We discard movement and unscored epochs, and keep only the 30 minutes of wake before the first and after the last sleep epoch of each night. This process results in 195,469 samples (W 65,941; N1 21,522; N2 69,132; N3 13,039; REM 25,835), and each sample is normalized per channel using standard scalers. For the training, validation, and test set splits, we employ the Subject-Independent setup. Specifically, we allocate 60%, 20%, and 20% of the 78 subjects (46, 16, and 16 subjects; 89, 32, and 32 nights), along with their corresponding samples, into the training, validation, and test sets, respectively. Both nights of a subject always fall into the same split. Unlike the other datasets, every sample carries its own label, and one night contains all sleep stages.

A.5.7 MIT-BIH PREPROCESSING

The MIT-BIH Arrhythmia dataset[7] (Moody & Mark, 2001; Goldberger et al., 2000) contains 48 half-hour, two-lead ambulatory ECG recordings from 47 subjects, sampled at 360 Hz, with cardiologist-provided beat annotations. Following the preprocessing convention of De Chazal et al. (2004), we exclude the four recordings containing paced, fusion-of-paced, and unclassifiable beats, as well as ventricular flutter wavesbeats (102, 104, 107, and 217). Recordings 201 and 202 belong to the same subject. We therefore exclude recording 201 from training while retaining recording 202 in testing, leaving 43 recordings and ensuring subject-exclusive splits. We use a modified version of the DS1/DS2 split of De Chazal et al. (2004). All 22 DS2 recordings form the test set. From DS1, we assign recordings 118, 124, 205, 215, and 223 to validation, ensuring that every class is represented. After excluding recording 201, the remaining 16 recordings form the training set. Unlike subject-level diagnosis tasks, each sample has its own beat-class label, which can vary within the same subject.

We resample both leads to 250 Hz and standardize each channel over the full recording. Each heartbeat is extracted as a window centred on its annotated R peak, with its length determined by the recording's median R–R interval, and zero-padded to 300 time points. Beats are grouped into four AAMI classes: N (N, L, R, e, j), S (A, a, J, S), V (V, E), and F (F). Paced, fusion-of-paced, and unclassifiable beats are discarded. This produces 98,755 samples: 88,490 N, 2,653 S, 6,811 V, and 801 F. Each sample is then standardized independently per channel.

A.5.8 HPP DATASET PREPROCESSING

The Human Phenotype Project (HPP) dataset,, is a large permission-accessible ECG time series dataset collected within a prospective longitudinal cohort in Israel (Reicher et al., 2025; Shilo et al., 2021). We use the ECG subset of the Human Phenotype Project (HPP), extends our evaluation to an additional cohort with thousands of participants and a different classification target: overall ECG interpretation categories rather than specific disease diagnoses. Each recording is a 12-lead resting ECG sampled at 1000 Hz, and each recording is accompanied by a diagnostic conclusion, which we map to 5 labels: normal ECG, ECG without significant abnormalities, normal variant of ECG, possibly abnormal ECG, and abnormal ECG. Of the 9,441 available waveform files, 9,428 are matched to a labelled recording, and the remaining 13 are discarded. Since some participants attended repeat visits, we discard participants whose diagnosis differs across visits, resulting in 7,139 of 7,879 subjects and 7,940 recordings. We downsample each recording to 250 Hz and normalize each lead of each recording using standard scaler. Each recording is then segmented into 1-second samples with 250 timestamps using a 50% overlap. We employ the Subject-Independent setup, allocating 4,998, 714, and 1,427 subjects to the training, validation, and test sets, which contain 105,165, 15,276, and 30,419 samples, respectively. The label distribution is imbalanced: the validation set contains 379, 173, 98, 37, and 27 subjects, and the test set contains 757, 347, 197, 73, and 53 subjects, in the order of the five classes listed above.

A.6 IMPLEMENTATION DETAILS

A.6.1 IMPLEMENTATION DETAILS OF LSTA

The adaptation basis $W \in \mathbb{R}^{D\times r}$ with $r = 8$ is initialized as the orthonormal factor of a QR decomposition of a Gaussian matrix (basis seed 0) and trained with Adam at learning rate 10^{-4} for up to 1,000 steps, with gradient norms clipped to 1 and QR retraction with column-sign correction

[7] https://physionet.org/content/mitdb/1.0.0/

after every step. Each step averages the query loss over four episodes. Each episode samples a training subject uniformly (episode seed 1), draws up to $k = 8$ support windows and up to 32 query windows from disjoint pools, and applies one entropy step with the selected step size η and temperature τ. We evaluate on validation subjects every 100 steps, averaging five support draws, and keep the basis with the highest validation macro-F1. At test time the basis is frozen and each subject receives a single projected entropy step from the source parameters.

A.6.2 Ablation Controls

LN-TTA. LN-TTA directly updates all LayerNorm scale and bias parameters using the support entropy gradient, without a learned transformation or low-rank constraint. It uses the same entropy temperature as LSTA, with its adaptation step size selected on validation data. This control isolates the benefit of learning and constraining the update directions.

Random-Subspace. Random-Subspace replaces the learned basis with LSTA's initial random orthonormal basis of the same rank. The basis remains fixed without meta-training, and support entropy gradients are projected onto this random subspace. This control tests whether restricting updates to a low-dimensional space is sufficient without learning its directions.

LN-TTA-Learnable. LN-TTA-Learnable meta-learns a separate update scale for each LayerNorm parameter using labeled query feedback. At test time, these scales remain fixed and multiply the support entropy gradient elementwise. Unlike LSTA, this variant does not couple parameter updates through a shared low-dimensional basis. It tests whether learning individual parameter scales can provide the same benefit as learning a low-rank adaptation space.

A.7 Hyperparameter Sensitivity

Subspace Rank

Figure 6 examines the tradeoff between the adaptation flexibility (rank) and the effect of the subspace constraint. A larger rank allows more degrees of freedom allowing more directions of parameter changes, but these additional directions does not necessarily improve performance. This is particularly relevant to ADFTD where rank-8 LSTA achieves 55.32 macro-F1 compared with 51.42 for unrestricted LN-TTA. So this comparison suggesting that restricting update directions can be more useful.

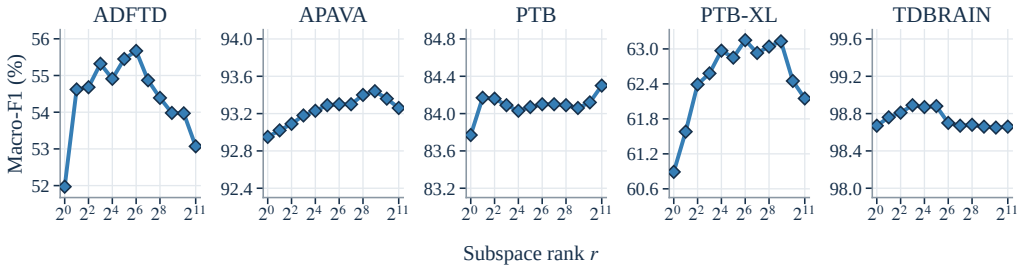


Figure 6: Effect of adaptation subspace rank on macro-F1 across five datasets. Lines show means across five seeds, with $k = 8$ support windows and one adaptation step. Panels use different y-axis scales.

A rank of one is too restrictive on ADFTD (51.97) and PTB-XL (60.89), and very large ranks lose accuracy again on these two datasets (53.07 and 62.15 at $r = 2^{11}$), as the subspace grows toward the full LayerNorm parameter space. On APAVA, PTB and TDBrain, macro-F1 varies by at most 0.5 points for $r \geq 2$. The default $r = 8$ lies within 0.6 points of the best rank on every dataset and is the best rank on TDBrain, so the results do not depend on a finely tuned rank.

Number of Support windows Figure **??** examines how the number of unlabeled support windows affects classification performance on separate query windows from the same subject. This evaluates

how effectively each method uses additional support data and how well it performs when only a few windows are available.

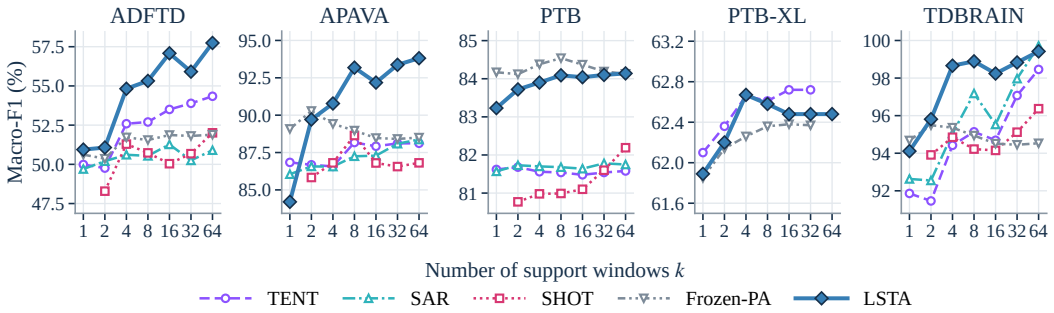


Figure 7: Effect of the number of unlabeled support windows on macro-F1 across five datasets. From four support windows onward, LSTA achieves the highest mean macro-F1 among the plotted methods on ADFTD and APAVA, and on TDBrain up to $k = 32$; Frozen-PA is slightly higher on PTB. SAR and SHOT are not plotted for PTB-XL. Panels use different y-axis scales.

LSTA improves as more support windows become available: between $k = 1$ and $k = 64$, macro-F1 rises from 50.94 to 57.74 on ADFTD, from 84.20 to 93.81 on APAVA and from 94.11 to 99.42 on TDBrain. A single window is not enough for a reliable update; with $k = 1$, LSTA falls below the frozen backbone on APAVA (84.20 vs. 85.71) and below Frozen-PA on APAVA, PTB and TDBrain. From $k = 4$ onward, LSTA has the highest macro-F1 among the plotted methods on ADFTD and APAVA, and on TDBrain up to $k = 32$ (SAR is higher at $k = 64$, 99.71 vs. 99.42). On PTB, Frozen-PA, which averages the frozen predictions over a subject's windows, stays slightly above LSTA for k between 1 and 32 (for example 84.54 vs. 84.09 at $k = 8$), which is expected when all windows of a subject share one label. On PTB-XL, LSTA is within 0.3 points of the best plotted method at every support size.

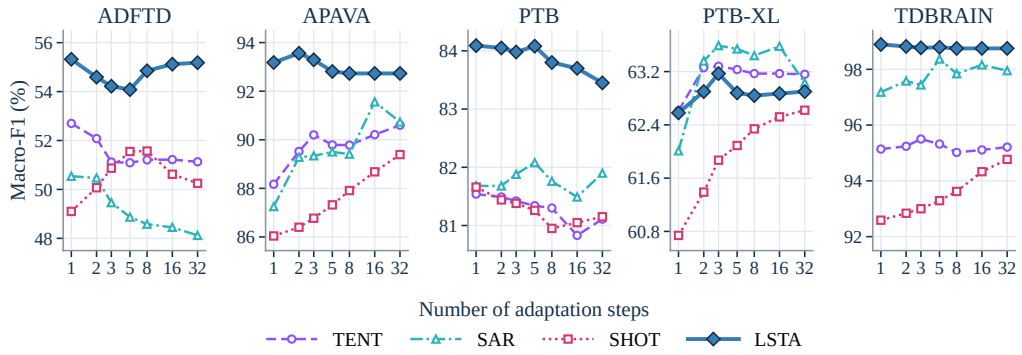


Figure 8: Effect of the number of test-time adaptation steps on macro-F1 across five datasets. LSTA is meta-trained using one adaptation step; the same learned basis is kept fixed when evaluating additional steps at inference.

Number of Steps Although LSTA is meta-trained with a single adaptation step, its performance remains relatively stable when the same basis is used for multiple step sizes during the inference (Figure 8). Across the tested step count its macro-F1 changes by at most 1.3 points on any dataset (ADFTD 54.08 to 55.32, APAVA 92.73 to 93.56, PTB 83.45 to 84.09, PTB-XL 62.58 to 63.17, TDBrain 98.75 to 98.89), so performance remains relatively stable when it is iterated. Other methods show larger changes: SAR decreases from 50.54 to 48.12 on ADFTD, while SHOT improves by 3.4 points on APAVA and 2.2 points on TDBrain. On PTB-XL, SAR and Tent outperform LSTA by up to 0.7 points when two or more steps are used. Additional steps therefore offer no consistent benefit for LSTA, supporting the single-step configuration used in our main experiments.

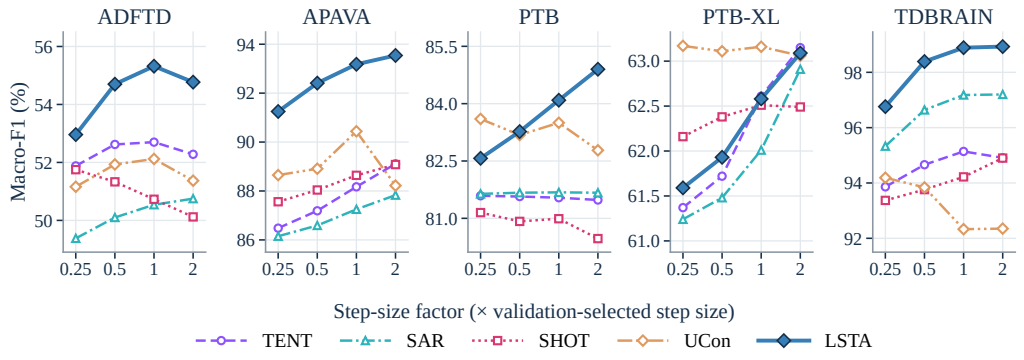


Figure 9: Sensitivity of macro-F1 to the adaptation step size across five datasets. Each method's validation-selected step size is multiplied by a factor in $\{0.25, 0.5, 1, 2\}$, where a factor of 1 corresponds to the selected value. Panels use different y-axis scales.

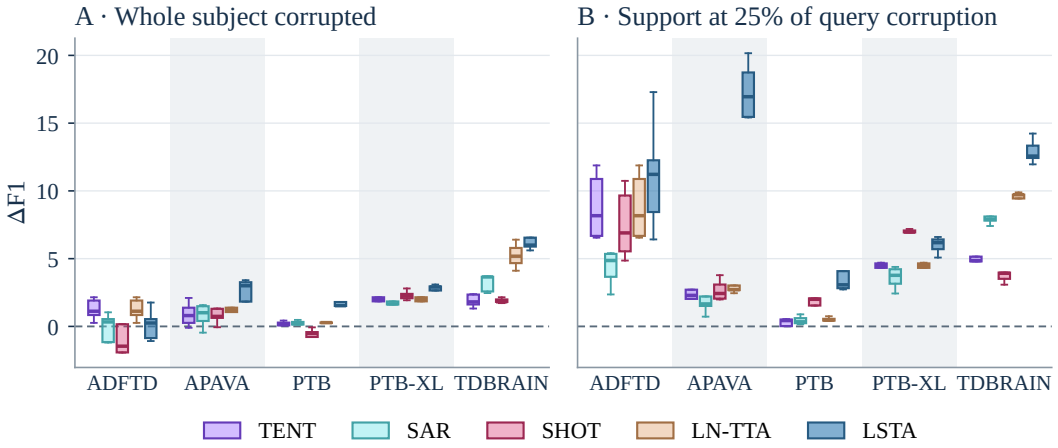


Figure 10: Robustness to test-time signal corruption. Change in test F1 relative to the source model, averaged over 12 corruption conditions (Gaussian noise, baseline wander, powerline interference, channel dropout; 3 severities each), mean ± std over 5 seeds of the per-seed difference. Scenario A: support and query windows equally corrupted. Scenario B: support windows corrupted at 25% of the query's severity. The source model's mean corrupted F1 is given under each dataset name. PTB-XL is evaluated on a fixed 300-subject test subset. No hyperparameters are re-tuned for corruption.

Step size. LSTA is least accurate at a factor of 0.25 on every dataset. Its macro-F1 increases up to a factor of 1 on ADFTD (52.96 to 55.32) and up to a factor of 2 on APAVA, PTB, PTB-XL and TDBrain (for example 82.57 to 84.90 on PTB). LSTA has the highest macro-F1 at all four factors on ADFTD, APAVA and TDBrain, and at factors of 0.5 to 2 on PTB, where UCon is higher at 0.25 (83.60 vs. 82.57). On PTB-XL, UCon is highest at factors up to 1, and Tent, LSTA and UCon are within 0.1 points of one another at a factor of 2. The steady improvement of LSTA up to a factor of 2 on four datasets suggests that the validation-selected step is conservative for the projected update; a larger step moves the parameters only within the learned subspace.

Signal corruption. When the whole subject is corrupted (Scenario A), LSTA has the largest median gain over the source model on APAVA, PTB, PTB-XL and TDBrain (3.00, 1.53, 2.91 and 6.00 points), while on ADFTD its median gain (0.25) is below that of Tent and LN-TTA (1.12). When the support windows are less corrupted than the query windows (Scenario B), the gains generally grow, and LSTA reaches the largest median gain on four of the five datasets, including 16.95 points on APAVA and 12.57 on TDBrain; on PTB-XL, SHOT is higher (6.96 vs. 6.18). LSTA therefore

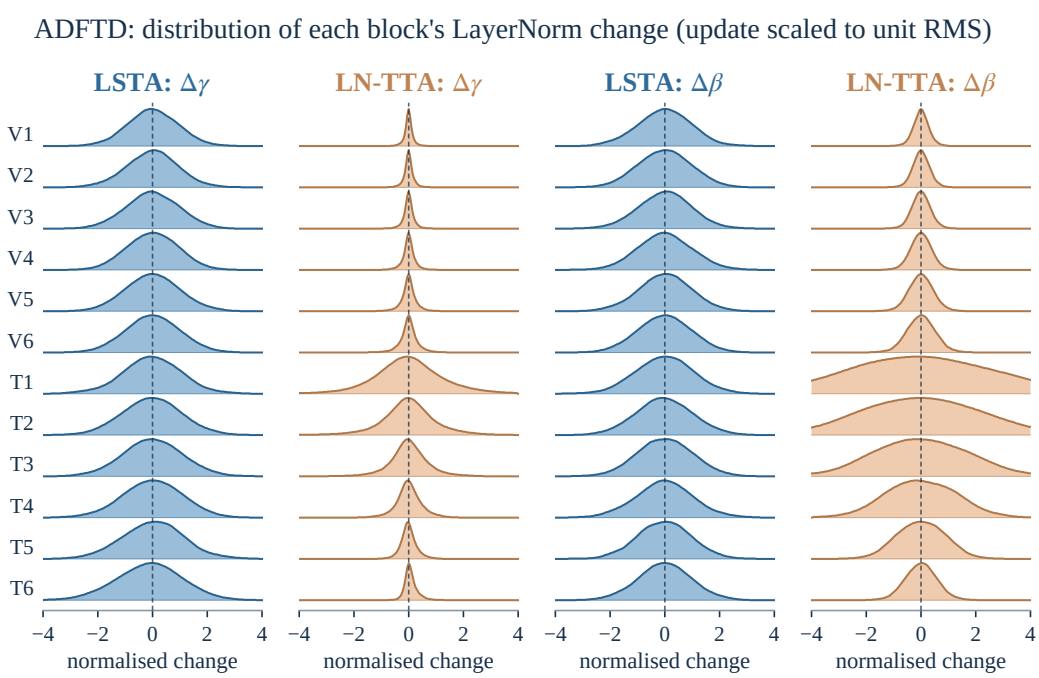


Figure 11: Distributions of LayerNorm scale ($\Delta\gamma$) and bias $\Delta\beta$) updates for LSTA and LN-TTA. Rows represent channel and temporal encoder blocks. Updates are normalized to unit RMS for visualization.

makes better use of a cleaner support set than the entropy-based baselines, although not uniformly across datasets.

A.8 Robustness and Adaptation Analysis

Robustness to contaminated support. Results are reported in Section 4.5 and Figure 3; per-seed results on HPP are given in Table 6.

Representation preservation. We compare frozen and adapted features using centered kernel alignment (CKA) and the fraction of within-subject nearest neighbours retained after adaptation. LSTA preserves more of the source representation structure than LN-TTA across all four evaluated datasets. On APAVA, CKA increases from 0.40 for LN-TTA to 0.82 for LSTA, while 10-neighbour retention increases from 0.25 to 0.45. On PTB, the corresponding values increase from 0.83 to 0.99 and from 0.77 to 0.96.

This greater preservation is accompanied by fewer harmful prediction changes. The proportion of initially correct predictions that become incorrect decreases from 3.87% to 2.53% on APAVA, from 3.45% to 1.28% on PTB, and from 16.93% to 12.32% on ADFTD. Together, these results suggest that the learned subspace enables adaptation while limiting disruption of the source representation and its successful predictions.

Distribution of LayerNorm updates. Figure 11 shows, for each channel block (V1 to V6) and temporal block (T1 to T6) on ADFTD, the distribution of LayerNorm scale and bias updates after rescaling each update to unit RMS. LSTA produces broad, bell-shaped distributions of similar width in every block, for both scale and bias. LN-TTA instead produces sharply peaked scale updates in the channel blocks, where most parameters barely move and a few move far, and much wider scale and bias distributions in the first temporal blocks. The learned basis therefore spreads the update evenly across normalization layers, whereas the unconstrained entropy gradient concentrates it in a few parameters and blocks.

A.9 VISUALIZATION

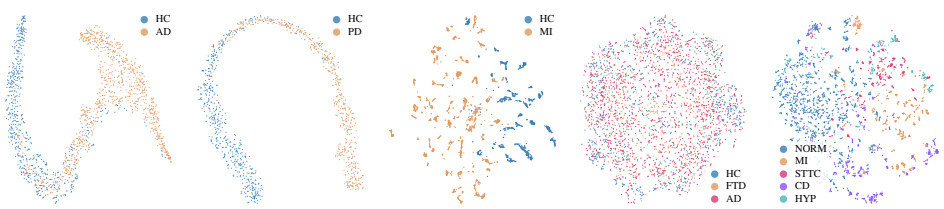
Feature geometry. Figures 5, 13 and 12 show t-SNE embeddings of pre-classifier features. On APAVA (Figure 5), the frozen backbone places the two diagnostic groups on a continuous curved structure with a region of overlap. LN-TTA and LN-TTA-Learnable break this structure into several compact, well-separated clusters per class, and Random-Subspace produces thin disconnected fragments. LSTA keeps a continuous layout that resembles the frozen embedding while separating the HC and AD windows more clearly considerably to the frozen backbone which has learned a generalized representation over large training examples. The same pattern appears on ADFTD (Figure 13): the unconstrained and random updates fragment the embedding and lower macro-F1 for this seed (47.3, 47.6 and 44.7, against 49.1 for the frozen backbone), whereas LSTA keeps a single cloud and reaches 52.3. Despite their apparent separation, these clusters still contain mixed diagnoses, and macro-F1 decreases from 49.1 for the frozen backbone to 47.3, 47.6 and 44.7, respectively. LSTA retains a layout resembling the source embedding while increasing macro-F1 to 52.3. This example suggests that useful adaptation need not considerably reorganize the representation: the learned subspace can improve predictions while limiting disruption of existing feature structure.

Across the five subject-labeled datasets (Figure 12), Tent and T3A leave the embeddings close to the frozen ones, while UCon, MedTS-TTT and LSTA change them more visibly on some datasets. We read these plots only as qualitative support for the preservation measures in Appendix A.8; t-SNE distances are not directly interpretable, and compact clusters may reflect subject identity as well as class.

A.10 ADDITIONAL RESULTS

Table 5: Performance on three additional medical time series datasets. Test results ($\times 100$) are mean $\pm$ s.d. over five backbone seeds with $k = 8$ support windows. The highest and second-highest means within each dataset, including the frozen model, are shown in **bold** and underlined. LSTA rows are shaded.

Dataset	Method	Accuracy	Precision	Recall	F1	AUROC	AUPRC	Avg
MIT-BIH (4 Classes)	Frozen	85.03±3.66	43.74±3.71	**47.73**±5.48	41.64±1.70	80.22±3.46	44.76±4.56	57.19±2.35
	Frozen-PA	85.66±3.18	44.16±8.25	36.95±5.36	36.33±4.11	78.02±3.31	42.61±4.51	53.95±3.77
	Tent	86.60±3.60	43.47±1.47	47.50±5.61	42.26±1.92	80.44±3.05	44.47±3.76	57.46±2.01
	T3A	85.52±3.48	42.40±3.11	47.69±5.62	42.13±2.72	80.55±3.89	**46.38**±4.61	57.44±2.86
	SAR	86.44±3.69	42.99±1.87	47.35±5.58	42.13±2.02	80.44±3.06	44.35±3.79	57.28±2.09
	MemBN	**88.77**±1.61	42.69±2.41	44.04±4.22	41.58±1.67	76.96±4.32	42.12±2.82	56.03±2.17
	SHOT	84.56±4.08	43.80±5.77	43.26±4.15	40.08±2.11	76.56±1.97	42.49±3.59	55.12±1.81
	UCon	80.68±4.46	44.31±5.29	38.63±4.87	35.52±4.59	73.12±4.20	38.78±3.97	51.84±2.85
	MedTS-TTT	86.43±4.09	45.32±9.12	40.01±8.25	39.24±6.15	77.39±5.18	43.03±5.71	55.24±5.99
	LSTA	87.10±3.25	**46.90**±3.33	47.13±6.16	**42.51**±2.10	**81.68**±2.72	45.80±1.98	**58.52**±1.09
Sleep-EDF (5 Classes)	Frozen	79.68±0.14	73.07±1.10	72.03±0.62	72.36±0.32	**94.37**±0.08	**78.21**±0.21	78.29±0.22
	Frozen-PA	75.78±0.37	72.66±0.97	64.31±1.16	67.13±0.42	89.55±0.16	71.43±0.23	73.48±0.19
	Tent	79.39±0.11	72.36±0.97	70.85±0.63	71.32±0.38	93.93±0.11	77.25±0.33	77.52±0.25
	T3A	77.21±0.40	70.33±0.87	68.47±0.91	69.09±0.44	93.15±0.19	74.03±0.37	75.38±0.35
	SAR	79.39±0.11	72.37±0.97	70.88±0.64	71.34±0.35	93.98±0.12	77.31±0.34	77.54±0.24
	MemBN	79.57±0.24	72.53±1.09	**72.08**±0.51	72.07±0.38	94.31±0.05	78.04±0.11	78.10±0.20
	SHOT	78.13±0.38	69.08±0.93	72.04±0.36	70.14±0.66	93.76±0.15	76.15±0.34	76.55±0.38
	UCon	78.40±0.33	70.53±1.20	70.70±0.66	70.39±0.65	93.90±0.17	76.48±0.50	76.73±0.47
	MedTS-TTT	78.89±0.69	70.99±1.54	70.50±1.57	70.26±1.10	93.72±0.38	76.36±1.02	76.79±0.78
	LSTA	**80.02**±0.13	**73.35**±0.53	71.78±0.51	**72.42**±0.22	94.35±0.09	**78.21**±0.21	**78.36**±0.16
HPP (5 Classes)	Frozen	61.77±0.17	58.24±1.09	45.00±0.73	48.54±0.64	77.60±0.26	49.61±0.60	56.79±0.47
	Frozen-PA	65.05±0.29	**66.24**±1.33	47.25±0.96	51.85±1.02	**81.46**±0.30	**55.80**±0.32	**61.28**±0.56
	Tent	63.26±0.23	61.83±0.89	46.51±0.64	50.37±0.69	78.28±0.18	51.38±0.46	58.60±0.39
	T3A	64.29±0.26	65.96±0.70	46.49±0.83	51.21±0.75	76.97±1.64	51.99±1.24	59.48±0.75
	SAR	63.27±0.20	61.60±0.60	46.62±0.71	50.36±0.64	78.23±0.20	51.24±0.44	58.55±0.34
	MemBN	60.03±0.17	56.55±1.31	39.84±0.75	43.69±0.71	75.23±0.32	44.67±0.68	53.34±0.50
	SHOT	64.05±0.44	62.11±1.47	47.26±0.88	50.39±0.73	76.77±0.31	49.77±0.69	58.39±0.65
	UCon	64.42±0.77	65.77±1.39	46.77±0.95	51.11±1.11	76.76±0.63	50.49±0.82	59.22±0.78
	MedTS-TTT	63.65±1.02	65.19±0.53	45.43±0.74	50.73±0.63	79.76±1.16	53.50±1.34	59.71±0.71
	LSTA	**65.63**±0.39	65.21±1.47	**49.74**±0.64	**53.85**±0.86	76.50±0.63	52.03±0.92	60.49±0.72



(a) Frozen Backbone



(b) TENT (Wang et al., 2020)



(c) T3A (Iwasawa & Matsuo, 2021)



(d) UCon (Xu et al., 2025)



(e) MedTS-TTT (Chen et al., 2026)



(f) LSTA (Ours)

Figure 12: t-SNE visualizations of representations produced by the frozen backbone and selected adaptation methods. Rows correspond to methods. Column 1: APAVA; Column 2: TDBrain; Column 3: PTB; Column 4: ADFTD; Column 5: PTB-XL.

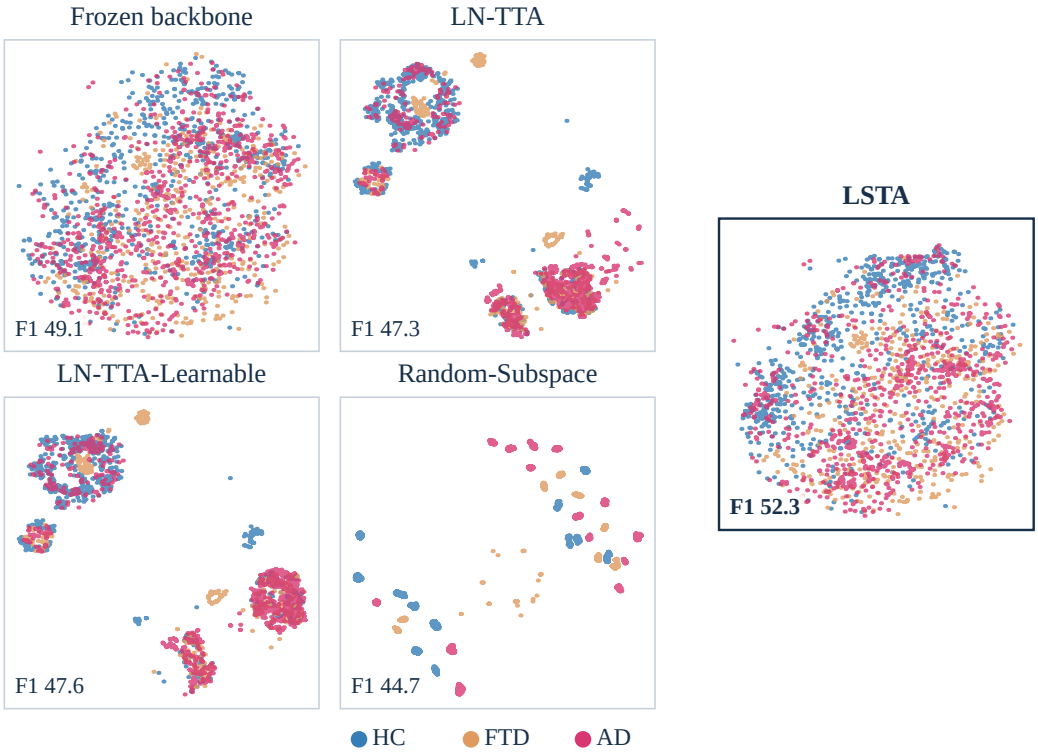


Figure 13: t-SNE visualization of ADFTD pre-classifier features on a selected random seed. Left block, top row: Frozen Backbone and LN-TTA; bottom row: LN-TTA-Learnable and Random-Subspace. Right: LSTA. LSTA largely preserves the frozen model's feature structure and achieves the highest macro-F1 in this comparison, while the other updates produce more fragmented clusters. Macro-F1 scores are shown for each method.

Window-labeled datasets and HPP. Table 5 covers MIT-BIH, Sleep-EDF and HPP. On Sleep-EDF, no method improves meaningfully on the frozen backbone (72.36): LSTA reaches 72.42, MemBN 72.07, and the other methods lose between 1.0 and 5.2 points. On MIT-BIH, UCon and Frozen-PA lose more than five points (35.52 and 36.33 against 41.64). Averaging predictions over a subject is harmful when the subject's windows carry different labels, whereas on HPP, where all windows of a subject share one label, Frozen-PA improves macro-F1 (51.85 vs. 48.54) and has the highest AUROC (81.46). LSTA has the highest macro-F1 on all three datasets (42.51, 72.42 and 53.85), but its margins on MIT-BIH and Sleep-EDF are smaller than the seed variation.

Contamination on HPP. Table 6 gives the per-seed results behind the HPP curve in Figure 3. LSTA has the highest mean macro-F1 at every contamination level (58.47, 57.04 and 54.18 at 0%, 25% and 50%) and stays above the frozen backbone on every seed at 50%. LN-TTA and LN-TTA-Learnable degrade faster and fall below the frozen backbone at 50% (51.55 and 52.38 against 53.16). Random-Subspace stays close to the frozen backbone at every level (53.78 to 53.40). This table uses a fixed evaluation set of 300 subjects and the best validation checkpoint per seed, so its absolute values differ from those in Table 5.

Component ablation beyond the main benchmarks. Table 7 repeats the component ablation on HPP, MIT-BIH and Sleep-EDF. On HPP, LN-TTA already improves macro-F1 over the frozen backbone (52.36 vs. 48.54), LSTA improves it further (53.85), and Random-Subspace stays close to the frozen model (49.01), so a random low-rank basis removes most of the benefit of adaptation. On MIT-BIH, LN-TTA lowers macro-F1 (41.32 vs. 41.64), while both learned variants raise it (42.80 for LN-TTA-Learnable and 42.51 for LSTA). On Sleep-EDF, all variants are within 0.13 points of one another, so the choice of update space matters little when the frozen model is already stable.

Table 6: F1 scores (%) on HPP under different contamination levels. Results are reported over five seeds. We used a fixed set of 300 subjects for evaluation choosing best performing checkpoint on validation set for each seed.

Dataset	Cont. (%)	Variant	F1 Mean ± Std	Seed 42	Seed 43	Seed 44	Seed 45	Seed 46
HPP	0	Frozen Backbone	53.16 ± 0.86	53.75	53.72	51.45	53.46	53.41
		LN-TTA	56.80 ± 3.00	59.87	59.13	51.31	56.95	56.74
		LN-TTA-Learnable	56.89 ± 3.05	59.92	59.15	51.22	57.07	57.07
		Random-Subspace	53.78 ± 1.10	54.59	54.60	51.68	54.29	53.76
		LSTA	**58.47 ± 1.52**	60.05	59.48	55.82	57.82	59.20
	25	Frozen Backbone	53.16 ± 0.86	53.75	53.72	51.45	53.46	53.41
		LN-TTA	54.84 ± 3.83	58.06	57.98	47.60	55.70	54.88
		LN-TTA-Learnable	55.27 ± 3.86	58.17	58.24	47.80	56.27	55.88
		Random-Subspace	53.51 ± 1.09	54.26	54.41	51.44	54.00	53.46
		LSTA	**57.04 ± 1.35**	58.55	58.39	54.93	56.33	57.01
	50	Frozen Backbone	53.16 ± 0.86	53.75	53.72	51.45	53.46	53.41
		LN-TTA	51.55 ± 5.99	55.05	54.69	39.64	55.08	53.28
		LN-TTA-Learnable	52.38 ± 6.21	55.62	55.49	39.95	55.35	55.47
		Random-Subspace	53.40 ± 0.96	54.09	54.10	51.54	53.82	53.46
		LSTA	**54.18 ± 1.35**	54.70	55.30	51.56	54.30	55.03

Table 7: Component ablation on HPP, MIT-BIH, and Sleep-EDF. Values are mean ± standard deviation (%) over five backbone seeds, after averaging over support draws. Precision, recall, F1, AUROC, and AUPRC are macro-averaged. Avg is the unweighted mean of the six metrics. **Bold** and underline indicate the best and second-best distinct displayed means, respectively, within each dataset. LSTA rows are shaded.

Dataset	Method	Accuracy	Precision	Recall	F1	AUROC	AUPRC	Avg
HPP (5 classes)	Frozen backbone	61.77±0.16	58.24±1.09	45.00±0.73	48.54±0.64	77.60±0.26	49.62±0.60	56.79±0.47
	LN-TTA	65.45±0.62	**65.75**±2.62	48.61±0.86	52.36±1.33	76.70±1.92	51.75±1.87	60.10±1.39
	LN-TTA-Learnable	65.09±0.58	64.93±2.33	48.59±0.88	52.41±1.32	76.78±2.10	51.65±1.87	59.91±1.34
	Random-Subspace	62.09±0.24	59.05±1.11	45.39±0.68	49.01±0.68	**77.76**±0.26	50.09±0.60	57.23±0.46
	LSTA	**65.63**±0.39	65.21±1.47	**49.74**±0.64	**53.85**±0.85	76.51±0.63	**52.03**±0.92	**60.49**±0.71
MIT-BIH (4 classes)	Frozen backbone	85.03±3.66	43.74±3.71	**47.73**±5.48	41.64±1.70	80.22±3.46	44.76±4.56	57.19±2.35
	LN-TTA	86.24±3.74	43.06±3.76	44.87±5.07	41.32±2.01	78.71±3.66	43.51±4.01	56.29±2.57
	LN-TTA-Learnable	**87.14**±3.52	**47.60**±3.70	47.11±5.45	**42.80**±1.74	**82.29**±1.96	45.71±2.25	**58.78**±1.01
	Random-Subspace	86.12±3.37	43.75±3.96	47.38±5.89	42.16±1.96	79.54±2.99	44.39±4.08	57.22±2.17
	LSTA	87.10±3.25	46.90±3.33	47.13±6.16	42.51±2.10	81.68±2.72	**45.80**±1.98	58.52±1.09
Sleep-EDF (5 classes)	Frozen backbone	79.68±0.14	73.07±1.10	**72.03**±0.62	72.36±0.32	**94.37**±0.08	**78.21**±0.21	78.29±0.22
	LN-TTA	79.69±0.14	73.05±1.07	71.91±0.60	72.29±0.31	94.33±0.08	78.16±0.22	78.24±0.21
	LN-TTA-Learnable	79.72±0.15	73.07±1.05	71.91±0.60	72.30±0.30	94.34±0.08	78.17±0.21	78.25±0.20
	Random-Subspace	79.68±0.15	73.06±1.09	72.00±0.64	72.34±0.31	**94.37**±0.08	78.20±0.22	78.27±0.22
	LSTA	**80.02**±0.13	**73.35**±0.53	71.78±0.51	**72.42**±0.22	94.35±0.09	**78.21**±0.21	**78.36**±0.16

Full comparison on subject-labeled datasets. Table 8 reports all six metrics for the TTA and source-free baselines. LSTA has the highest accuracy and macro-F1 on all five subject-labeled datasets, and the highest AUROC on ADFTD, APAVA, TDBrain and PTB. On PTB-XL its AUROC (88.66) is below that of the frozen backbone (89.85) and MedTS-TTT (90.56), so its gain there is limited to threshold-dependent metrics. Several baselines fall below the frozen backbone: MemBN lowers macro-F1 on all five datasets, and MedTS-TTT does so on APAVA and PTB, which shows that adaptation from a few windows can hurt.

Table 8: Comparison of LSTA with test-time adaptation and source-free adaptation baselines under subject-exclusive splits. All methods start from the same TeCh backbone checkpoints (Frozen) and are evaluated on the same query windows. Values are mean ± s.d. (%) over five backbone checkpoints, after averaging over support draws. Precision, recall, F1, AUROC and AUPRC are macro-averaged; Avg is the unweighted mean of the six metrics. **Bold**: best mean; underline: second-best mean within each dataset. LSTA rows are shaded.

Dataset	Method	Accuracy	Precision	Recall	F1	AUROC	AUPRC	Avg
ADFTD (3 classes)	Frozen	54.54±0.70	53.02±0.87	49.25±1.01	48.44±1.72	68.67±1.05	50.62±1.26	54.09±0.90
	Tent	61.11±1.32	65.20±1.76	54.86±1.45	52.70±3.00	74.40±2.23	57.12±3.03	60.90±1.60
	T3A	57.33±1.14	58.89±2.27	51.21±1.56	49.95±2.60	71.22±2.44	54.25±2.51	57.14±1.63
	SAR	57.27±2.55	54.97±3.17	52.09±2.46	50.54±3.47	71.54±1.83	53.64±2.48	56.68±2.60
	MemBN	52.28±0.99	50.82±0.74	46.97±1.05	45.93±1.84	66.51±0.98	48.44±1.06	51.82±0.90
	SHOT	57.92±2.02	61.92±4.91	52.77±1.89	50.73±3.21	71.37±2.04	55.46±2.25	58.36±2.12
	UCon	60.86±1.81	**66.58**±1.78	54.43±1.94	51.70±3.36	73.18±3.76	56.07±3.18	60.47±1.79
	MedTS-TTT	52.82±1.92	48.92±3.07	49.21±2.72	48.80±2.79	68.21±2.60	50.05±2.04	53.00±2.43
	LSTA	**62.30**±1.49	64.05±5.36	**56.26**±1.41	**55.32**±1.86	**74.83**±2.86	**58.08**±3.91	**61.81**±2.53
APAVA (2 classes)	Frozen	86.35±0.55	86.40±1.08	85.38±0.57	85.71±0.49	93.26±0.58	92.99±0.57	88.35±0.56
	Tent	88.73±1.83	88.92±1.66	87.82±2.31	88.17±2.00	94.74±1.57	94.53±1.63	90.49±1.80
	T3A	88.15±1.24	88.16±1.45	87.38±1.34	87.64±1.28	94.45±0.88	93.75±0.99	89.92±1.01
	SAR	87.88±2.32	88.09±2.21	86.90±2.81	87.25±2.52	93.83±2.51	93.57±2.65	89.59±2.46
	MemBN	82.14±0.52	82.44±0.98	80.54±1.07	81.05±0.72	89.45±0.55	88.72±0.59	84.06±0.58
	SHOT	89.12±1.87	89.21±2.06	88.34±2.00	88.64±1.94	95.21±1.47	95.04±1.50	90.93±1.78
	UCon	89.59±3.48	90.44±3.50	88.65±3.72	89.05±3.64	94.12±1.78	93.96±1.81	90.97±2.86
	MedTS-TTT	75.19±9.10	76.33±10.96	72.48±8.85	73.15±9.25	75.96±12.18	75.09±12.71	74.70±10.47
	LSTA	**93.70**±3.23	**95.23**±2.15	**92.41**±4.03	**93.18**±3.60	**97.58**±1.94	**97.36**±2.14	**94.91**±2.76
TDBrain (2 classes)	Frozen	92.37±1.25	92.70±1.01	92.38±1.25	92.36±1.27	98.44±0.14	98.49±0.14	94.45±0.84
	Tent	95.16±2.03	95.44±1.82	95.16±2.03	95.14±2.05	98.82±0.80	98.82±0.81	96.42±1.56
	T3A	92.67±1.30	92.98±1.05	92.67±1.30	92.66±1.32	97.93±0.69	97.27±1.32	94.37±1.12
	SAR	97.18±0.56	97.33±0.52	97.18±0.56	97.18±0.56	99.68±0.19	99.67±0.22	98.04±0.41
	MemBN	89.45±1.15	89.86±0.92	89.45±1.15	89.42±1.17	96.42±0.30	96.45±0.35	91.84±0.82
	SHOT	94.24±1.67	94.50±1.40	94.24±1.67	94.22±1.69	98.96±0.41	98.97±0.43	95.85±1.20
	UCon	92.98±3.30	93.83±2.83	92.98±3.30	92.92±3.35	97.82±2.31	97.76±2.45	94.71±2.71
	MedTS-TTT	96.90±2.83	97.16±2.37	96.90±2.83	96.88±2.86	**99.88**±0.12	**99.88**±0.12	97.93±1.82
	LSTA	**98.88**±0.76	**98.95**±0.69	**98.88**±0.76	**98.88**±0.76	**99.98**±0.01	**99.98**±0.01	**99.26**±0.50
PTB (2 classes)	Frozen	85.67±1.44	89.88±0.57	78.88±2.38	81.60±2.31	94.73±0.42	94.61±0.39	87.56±1.07
	Tent	85.65±1.65	89.93±0.71	78.82±2.68	81.54±2.63	94.73±0.42	94.63±0.38	87.55±1.24
	T3A	85.81±1.91	90.32±0.98	78.95±2.96	81.70±2.94	94.57±0.49	94.21±0.77	87.60±1.59
	SAR	85.74±1.71	90.00±0.74	78.96±2.76	81.68±2.71	94.75±0.41	94.66±0.37	87.63±1.29
	MemBN	80.22±2.02	85.49±1.26	70.97±3.15	72.99±3.66	91.52±0.60	90.52±0.88	81.95±1.73
	SHOT	85.31±1.74	89.64±0.80	78.36±2.83	80.99±2.78	91.52±1.19	91.64±1.35	86.24±1.73
	UCon	85.86±1.12	89.35±0.87	79.43±1.84	82.02±1.72	87.21±2.64	87.04±2.84	85.15±1.67
	MedTS-TTT	81.23±2.58	88.92±1.25	71.71±3.89	73.89±4.49	94.38±1.29	94.27±1.13	84.07±1.74
	LSTA	**87.48**±2.46	**91.45**±1.48	**81.39**±3.72	**84.09**±3.53	**95.07**±0.31	**95.17**±0.34	**89.11**±1.92
PTB-XL (5 classes)	Frozen	73.21±0.26	65.37±0.84	60.73±0.62	62.39±0.30	89.85±0.12	67.00±0.25	69.76±0.25
	Tent	74.62±0.25	67.53±0.88	62.18±0.79	64.10±0.45	89.39±0.38	67.09±0.63	70.82±0.37
	T3A	74.91±0.21	68.03±0.78	62.44±0.74	64.44±0.42	82.94±0.20	60.85±0.20	68.93±0.25
	SAR	74.43±0.16	67.17±0.88	61.92±0.70	63.81±0.38	89.53±0.29	67.40±0.49	70.71±0.30
	MemBN	70.78±0.46	63.70±0.65	56.02±0.76	57.95±0.49	88.06±0.20	63.12±0.40	66.61±0.34
	SHOT	74.41±0.40	67.00±1.14	62.16±0.70	63.97±0.46	84.46±0.73	59.94±0.92	68.66±0.57
	UCon	74.84±0.40	67.65±0.93	**62.75**±0.66	64.62±0.44	84.50±0.51	61.39±0.60	69.29±0.46
	MedTS-TTT	74.41±0.38	66.73±1.25	62.15±1.33	63.33±1.00	**90.56±0.37**	**68.54±0.82**	70.95±0.50
	LSTA	**75.33**±0.35	**68.51**±0.87	62.67±0.65	**64.66**±0.51	88.66±0.33	66.18±0.46	**71.00**±0.37