Predictive Surprise as Self-Grounding Concept Bottleneck for Interpretable Time Series

Sachith Abeywickrama[1,2] **Emadeldeen Eldele**[3] **Min Wu**[2] **Xiaoli Li**[4] **Chau Yuen**[1]

[1]Nanyang Technological University, Singapore
[2]A*STAR Institute for Infocomm Research, Singapore
[3]Khalifa University, UAE [4]Singapore University of Technology and Design, Singapore

e240203@e.ntu.edu.sg emadeldeenahmed.eldele@ku.ac.ae
wumin@i2r.a-star.edu.sg xiaoli_li@sutd.edu.sg chau.yuen@ntu.edu.sg

Abstract

Time-series classifiers deployed in risk-sensitive domains such as healthcare, wearable sensing, and industrial monitoring must be accurate, interpretable, and capable of abstaining when evidence is weak. No existing method delivers all three, and the obstacle is structural, rather than logistical. Meaningful temporal patterns cannot be defined without knowing segment boundaries, yet meaningful boundaries cannot be placed without knowing which patterns to expect. Existing approaches break this circularity through fixed windows or expert-annotated vocabularies, sacrificing at least one of the three properties. We introduce CONCEPTTIME, which closes the loop through a single self-supervised signal. A frozen probabilistic forecaster predicts a distribution over the next observation at every timestep of the sequence. We call the mismatch between its prediction and the realized signal, the *predictive surprise*. This signal both locates segment boundaries and characterizes each resulting segment through the forecaster's own predictive statistics. Segments are thus grounded by *how* they behaved relative to expectation, not by what they look like. These summaries cluster into a vocabulary of dynamical regimes that we call *concepts*. The vocabulary distinguishes calm-after-spike from calm-after-calm, and unifies visually distinct segments that violate expectation in the same way. A lightweight head classifies from the concept sequence, and the distance to the nearest concept yields per-input reliability for free. Across seventeen UEA, human-activity, biomedical, and fault-diagnosis benchmarks, CONCEPTTIME matches state-of-the-art black-box accuracy, beats every interpretable baseline, and retains near 98% accuracy on Epilepsy with 1% of labels. It equals or exceeds softmax, prototype-distance, and SHAP/TimeX/TimeX++ on OOD detection and deletion-faithfulness, without a single annotation, language-model call, or domain expert. Code is available at: `https://github.com/Sachithx/ConceptTime-1.0.git`

1 Introduction

Time-series classifiers deployed in healthcare [23, 32], wearable sensing [21], environmental monitoring [40], and industrial telemetry [46, 60] face three simultaneous demands. Predictions must be accurate enough to compete with end-to-end deep models. They must be interpretable, so that users can inspect which temporal patterns drove a decision. And they must support calibrated abstention, so that the system defers when its evidence is weak. No existing approach delivers all three(Fig. 1).

Black-box classifiers cover only accuracy. Post-hoc explainers describe a model output without constraining how that output was produced. Interpretable-by-design methods come closest, but they

Preprint.

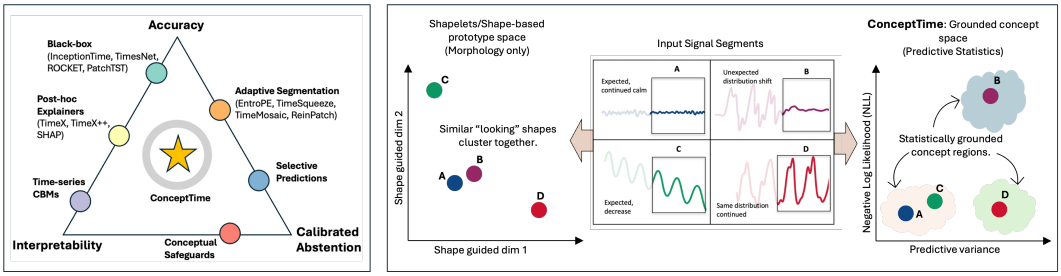

Figure 1: **ConceptTime targets accuracy, interpretability, and calibrated abstention through a single self-supervised primitive.** *Left*, existing methods approach individual corners of the accuracy-interpretability-abstention. *Right*, four segments (A-D) and their preceding context. Raw-shape space conflates A and B (both end calm) and splits C from D. Predictive-geometry space, spanned by predictive variance and NLL, collapses A, C, and D as expectation satisfied and isolates B as a *surprise*. Concepts grounded in deviation from prediction, therefore, distinguish dynamical regimes that shape-based vocabularies cannot.

stumble on the cost of concept annotation, which is rare for sequential data and expensive to obtain at scale. The deeper obstacle is structural and unique to sequences. Concept bottleneck models for time series face a chicken-and-egg problem. Meaningful temporal concepts cannot be defined without knowing where one segment ends and another begins. Yet meaningful segment boundaries cannot be placed without knowing what concepts to look for. Every prior method sidesteps this circularity by decoupling the two stages. They segment first with fixed windows or threshold-based change detection, then learn concepts on the resulting chunks. Each gives up one of the three properties as a result. Fixed-window CBMs [61, 56, 15] achieve interpretability but require predefined vocabularies and lose calibrated abstention. Adaptive-patching methods [1, 45] segment well but feed patches into standard encoder-classifier pipelines, exposing no concepts at all. Vector-quantized models [54, 58, 5] produce discrete tokens, but the tokens are latent codes without inherent semantic structure.

The same self-supervised signal that locates segment boundaries can also describe what each resulting segment looks like. A frozen probabilistic forecaster predicts a distribution over the next observation at every timestep. The mismatch between this prediction and the realized signal, which we call *predictive surprise*, is the natural place to draw a boundary, since high mismatch indicates a regime change. The same forecaster also produces predictive statistics that describe how each segment behaved relative to expectation. A single frozen model can therefore close the segmentation-concept loop, and no prior method exploits this dual use.

We introduce CONCEPTTIME, which operationalizes this opportunity. A frozen autoregressive Gaussian density model is trained once on raw multivariate signals via next-step density estimation. At every timestep, it predicts a Gaussian distribution over the next observation conditioned on history. The negative log-likelihood of the realized signal under that prediction is our predictive surprise. This single quantity locates segment boundaries via length-constrained dynamic programming, placing exactly K patches at moments where the signal most strongly violates prior expectation. The same model characterizes each resulting patch through its predictive statistics, yielding contextual fingerprints that describe how a segment behaved relative to expectation rather than what it looks like. Patches that violate prior expectation in similar ways receive similar fingerprints, even when their raw waveforms differ visibly. The path from raw signal to concept activation is fully deterministic. Signatures are statistics of the frozen model, prototypes are cluster centroids in signature space, and patch-to-concept assignment is nearest-prototype Mahalanobis lookup. A lightweight classifier is the only label-trained component. The same prototype distances yield a built-in fidelity score that asks whether the input is even *describable* in the concept space rather than whether the classifier is *confident*. Our contributions are threefold,

1. We introduce the first concept-bottleneck framework to our knowledge that resolves the segmentation-concept circularity through a single self-supervised primitive, eliminating fixed windows, expert annotations, and language-model supervision.

2. We provide a fully non-parametric concept-grounding pipeline whose path from raw signal to concept activation is deterministic and inspectable, with all task-specific learning confined to a lightweight classifier on the concept tensor.
3. We identify prototype distance as a built-in fidelity signal and propose a prototype-distance abstention mechanism for time-series concept bottlenecks.

Across seventeen benchmarks spanning UEA, human-activity, biomedical, and fault-diagnosis tasks, CONCEPTTIME matches black-box accuracy, beats every interpretable baseline, and retains 98% accuracy on Epilepsy at 1% labels. Its fidelity score outperforms softmax and competing prototype methods on OOD detection, and beats SHAP, TimeX, and TimeX++ on deletion faithfulness, all without annotations, language-model calls, or domain experts.

2 Related Work

Concept bottleneck models. CBMs [28] route predictions through interpretable intermediate representations, enabling inspection and intervention. The original formulation requires dense annotations; label-free variants source concepts from vision-language models [44, 65, 52, 49], with parallel work on robustness, counterfactuals, and locality [41, 50, 11, 25]. The few existing time-series CBMs each commit to predefined vocabularies on fixed-length windows: predefined statistical summaries [61], CKA-aligned concepts for forecasting [56], or Signal Temporal Logic predicates [15]. Two open problems remain: every variant requires *external semantic supervision* (annotations or vision-language priors), which is especially constraining for sequential data where temporal concepts are local and rarely linguistically nameable; and none addresses the *segmentation problem*, the question of where one concept ends and another begins.

Adaptive patching and discrete tokenization. Two threads partially address segmentation. The Byte Latent Transformer [45] places boundaries at high next-token entropy points, an idea adapted to time series by EntroPE [1] via greedy thresholding on categorical entropy; concurrent work explores reinforcement-learned boundaries [63] and content-aware compression [4]. Vector-quantized models [54, 58, 5] build discrete vocabularies through learned tokenization. Both stop short of grounded concepts: adaptive patches feed standard encoder-classifier pipelines with no exposed semantic representation, and VQ tokens are latent codes without claim to interpretability. The open problem is that the same frozen model that segments could, in principle, also describe each segment, but no prior method exploits this dual use.

Prototype, shape-based, and post-hoc methods. Prototype and shapelet classifiers [66, 42, 20, 14, 59, 33] provide local, case-based interpretability but anchor prototypes in raw waveform shape on fixed windows, with no probabilistic grounding. Post-hoc explainers [47, 34, 8, 22] attribute black-box predictions through saliency or perturbation but do not constrain the classifier to use the explanation. The open problem is that prototypes anchored in raw shape miss the dynamical structure, i.e., a flat patch following a spike and a flat patch following calm look identical, while post-hoc explanations describe model behavior without making the classifier interpretable by construction.

Selective prediction and concept-level uncertainty. Selective prediction has relied on softmax confidence [17], ensemble disagreement [29], or distance-based OOD scoring [30, 53, 36, 37]. In vision, concept-based selective prediction has emerged through additional uncertainty heads, calibration networks, or stochastic concept layers [26, 35, 57, 27, 51]. No analogous mechanism exists for time series, and existing approaches either operate in label space (softmax, divorced from concept structure) or require an additional trained component (uncertainty heads, calibration).

CONCEPTTIME addresses these open problems jointly through a single self-supervised primitive. A frozen Gaussian density model is reused for both segmentation and concept grounding (closing the segmentation-concept loop and resolving the dual-use question). Concepts are geometric centroids in the space spanned by the model's predictive statistics, removing the need for annotations or linguistic priors and capturing dynamical structure that raw-shape prototypes miss. Patch-to-concept assignment is a non-parametric Mahalanobis lookup, eliminating the learned-encoder pathway through which concept leakage typically occurs [41]. The same prototype distances yield a built-in fidelity score for selective prediction with no additional component. On matched dynamic patches and matched concept count, replacing predictive-geometry signatures with classical raw-shape descriptors collapses HAR accuracy by 46 points (Section 4.4), demonstrating that predictive grounding, not the prototype

Table 1: Comparison of major directions in interpretable time-series classification. ✓ = direct support; ✗ = absent; ∼ = partial or with caveats.

Property	Post-hoc	CBMs	Adaptive Patch	Prototype / Shapelet	IB / Repr. Learning	ConceptTime (Ours)
Decision routes through concepts	✗	✓	✗	∼	✗	✓
No external semantic supervision required	✓	∼	✓	✓	✓	✓
Concepts discovered from data, not predefined	∼	∼	–	∼	✓	✓
Adaptive segmentation	✗	✗	✓	✗	✗	✓
Local, segment-level explanations	∼	∼	✗	✓	✗	✓
Built-in abstention without auxiliary head	✗	∼	✗	✗	∼	✓
Competitive supervision accuracy	✓	∼	✓	∼	✓	✓
Strong few-shot performance	✗	✗	✗	✗	∼	✓

Notes. Post-hoc: TimeX, TimeX++, ShapeX [47, 34, 22]. CBMs: vision-language CBMs and time-series CBMs [28, 44, 61, 56]; Adaptive Patch: BLT, EntroPE [45, 1]; the dash indicates the property is outside scope (these methods are not designed to expose concepts). Prototype/Shapelet: ProSeNet, ECATS, shapelets [42, 14, 66]. IB/Repr.: information-bottleneck and contrastive representation methods [39, 7].

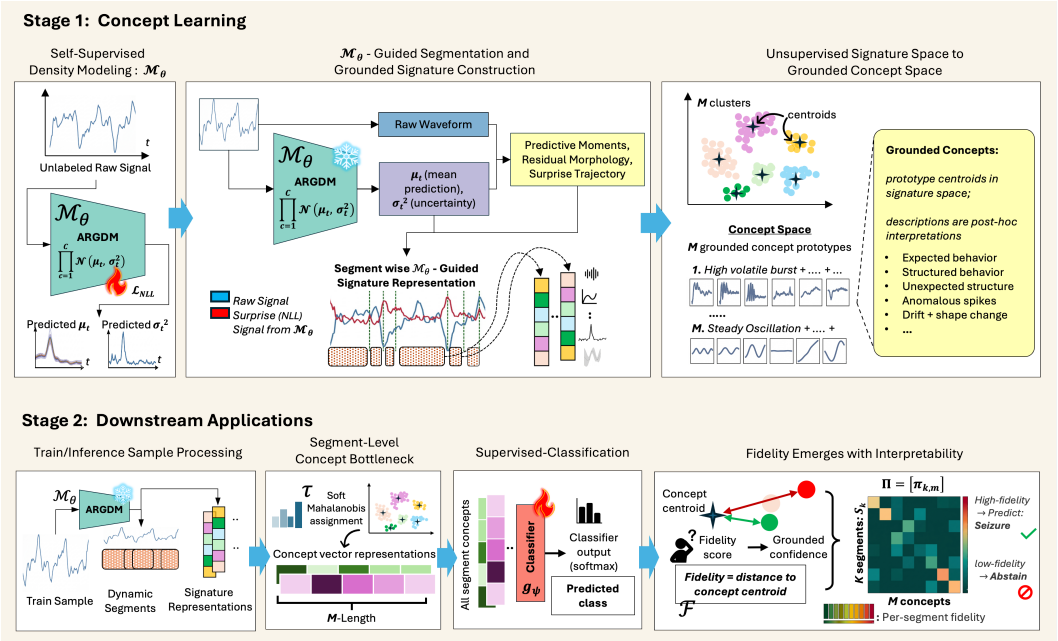

Figure 2: **Overview of ConceptTime. Stage 1:** a frozen Gaussian density model $\mathcal{M}_\theta$, trained once on raw signals, produces predictive surprise s_t that drives length-constrained DP segmentation. Each segment is summarised by a signature vector combining *predictive moments*, *residual morphology*, and *surprise trajectory* 3.4. Then signatures are clustered into M concept prototypes; textual labels are post-hoc, not training signals. **Stage 2:** new inputs are segmented by $\mathcal{M}_\theta$, mapped to concepts by Mahalanobis nearest-prototype assignment, and classified through the $K \times M$ concept tensor. Each segment's distance to its nearest prototype yields the fidelity score (Eq. 5) used for abstention.

paradigm itself, drives the accuracy. To our knowledge, no prior method combines annotation-free concept discovery, predictive-distribution grounding, adaptive segmentation, and prototype-based abstention in a single coherent framework for multivariate time series. Table 6 summarizes how ConceptTime relates to the major directions surveyed above.

3 ConceptTime

ConceptTime factors classification into stages with disjoint training signals: a self-supervised density model produces every representation, clustering induces a non-parametric concept vocabulary, and a single linear or one-layer transformer head consumes labels. Every intermediate is human-inspectable, and prototype distances yield a built-in fidelity score for selective prediction at no additional cost. The pipeline is illustrated in Figure 2.

3.1 Problem Formulation

A multivariate input is $x \in \mathbb{R}^{T \times C}$ with T timesteps and C channels. The classifier $f : \mathbb{R}^{T \times C} \to \{1, \ldots, Y\}$ is trained on $\{(x^{(i)}, y^{(i)})\}_{i=1}^{N}$. The density model and concept vocabulary are fit on an unlabeled corpus $\mathcal{D}_{\text{u}}$; in our experiments, we use the training inputs without labels, but the framework supports any unlabeled corpus over the same signal space.

3.2 Frozen Gaussian Density Model

We train one autoregressive model $\mathcal{M}_\theta$ to predict the next-step diagonal Gaussian

$$p_\theta(x_{t+1} \mid x_{\le t}) = \prod_{c=1}^{C} \mathcal{N}\big(x_{t+1,c};\, \mu_{t,c},\, \sigma_{t,c}^2\big), \qquad (\mu_t,\, \log \sigma_t^2) = \mathcal{M}_\theta(x_{\le t}), \tag{1}$$

with a 4-layer causal Transformer. Training minimises per-step Gaussian NLL augmented with a calibration regularizer $\big(\mathbb{E}[(x-\mu)^2/\sigma^2] - 1\big)^2$ that forces predicted variances to match realised residual energy. Calibration is not optional: without it, σ_t^2 collapses to a free fudge factor, and every downstream signature derived from it becomes uninterpretable. After this single self-supervised stage $\mathcal{M}_\theta$ is frozen and reused for both segmentation and signature construction; classification supervision never updates it.

Per-timestep surprise. The realised negative log-likelihood under $\mathcal{M}_\theta$,

$$s_t = \frac{1}{C} \sum_{c=1}^{C} \tfrac{1}{2} \left(\log \sigma_{t,c}^2 + \frac{(x_{t,c} - \mu_{t,c})^2}{\sigma_{t,c}^2} \right), \tag{2}$$

drives segmentation. Unlike differential entropy $\frac{1}{2}\big(1 + \log 2\pi\sigma_t^2\big)$, which depends only on the predicted variance, s_t additionally weights the realised residual $(x_t - \mu_t)^2$. It is therefore high precisely when the model was *wrong despite being confident*, which is the regime transitions a segmenter should mark.

3.3 Length-Constrained Adaptive Segmentation

Given $\{s_t\}_{t=1}^{T}$ and a target patch count K, we partition x into K non-overlapping segments by placing $K-1$ internal boundaries that maximise cumulative surprise subject to per-patch length constraints $[L_{\min}, L_{\max}]$. The dynamic program is

$$B[k,t] = \max_{t':\, L_{\min} \le t - t' \le L_{\max}} \big(B[k-1,\, t'] + s_t \big), \qquad B[0,0] = 0, \tag{3}$$

with the same length constraint enforced on the trailing segment $T - t_{K-1}$. A monotone deque resolves the inner maximum in amortized constant time, giving $O(TK)$ overall.

Length constraints prevent two failure modes of greedy threshold-based segmenters, such as EntroPE [1]: degenerate single-patch solutions on flat-surprise recordings, and tail-collapse when boundaries cluster at peaks.

3.4 Predictive-Geometry Signatures

Each patch $p_j = x_{t_j : t_{j+1}}$ is reduced to a signature $\phi_j \in \mathbb{R}^{2C+13}$ assembled from three families. All three are functions of $\mathcal{M}_\theta$'s output; no neural component sits between the signal and ϕ_j.

Predictive moments ($2C + 1$ dims): patch-mean of μ_t per channel, patch-mean of σ_t^2 per channel, and log patch length. These describe what the model expected, on average, and how confident it was.

Residual morphology (8 dims): FFT (Fast Fourier Transform) peak frequency [38], autocorrelations at lags 1–3, skewness, excess kurtosis, zero-crossing rate, and peak-to-peak range, all computed on the standardised residual, $r_t = (x_t - \mu_t)/\sigma_t$.

Subtracting μ_t removes the predictable background dynamics that $\mathcal{M}_\theta$ already explains; dividing by σ_t amplifies deviations in confident regions and damps them in uncertain ones. The resulting shape descriptors therefore reflect class-discriminative deviations from prediction rather than absolute amplitude or trend, which dominate the same statistics computed on raw x_t.

Surprise trajectory (4 dims): the within-patch position of $\arg\max_t s_t$ normalised to $[0, 1]$; the standard deviation of $\{s_t\}$; confidence-weighted surprise $\frac{1}{L}\sum_t s_t\big(1 - H_t/H_{\max}\big)$, which up-weights errors at instants where the model was most certain; and mean Mahalanobis residual energy $\frac{1}{L}\sum_t \|r_t\|^2$. Together, these specify *when within the patch* and *how severely* the model was surprised. Two patches with identical mean surprise but different temporal profiles (uniform versus a single late spike) produce distinct surprise-trajectory features and are routed to different concepts.

The three families capture mutually orthogonal views, namely, what was predicted (moments), what the residual looked like (morphology), and how surprise unfolded over time (trajectory). Signatures are standardised using 99th-percentile-clipped statistics estimated on $\mathcal{D}_{\text{u}}$, which prevents single-patch outliers from collapsing other dimensions to near-zero variance.

3.5 Non-Parametric Concept Vocabulary

We fit a Gaussian Mixture Model (GMM) with diagonal covariance and M components on the standardised signatures from $\mathcal{D}_{\text{u}}$, with $M \in \{16, 32\}$ throughout. This range is small enough that the resulting concept-transition matrix remains human-readable and large enough for class separability. Diagonal covariance handles the heterogeneous scales of the three signature families, and component-specific Mahalanobis distance corrects for them at assignment time.

A patch with signature ϕ_j assigns to prototype c_m via temperature-scaled soft assignment:

$$\pi_{j,m} = \frac{\exp(-d(\phi_j, c_m)/\tau)}{\sum_{m'} \exp(-d(\phi_j, c_{m'})/\tau)}, \qquad d(\phi_j, c_m) = \sqrt{(\phi_j - c_m)^\top \Sigma_m^{-1} (\phi_j - c_m)}. \tag{4}$$

The concept tensor $\Pi \in \mathbb{R}^{K \times M}$ stacks $\{\pi_j\}_{j=1}^K$, and the nearest-prototype distance $d_j = \min_m d(\phi_j, c_m)$ is retained for the fidelity score below.

Because the entire path from signal to Π is the composition of $\mathcal{M}_\theta$, the signature standardiser, and the GMM (none of which sees labels), the resulting concepts are immune to the concept-leakage failure mode in which a learned concept extractor smuggles label information into nominally interpretable activations [41]. Each prototype is a geometric centroid in signature space whose nearest training patches act as canonical exemplars: this is sufficient for human inspection without any post-hoc interpretation tooling.

3.6 Classifier and Fidelity Score

A lightweight classifier g_ψ maps the concept tensor Π to class logits and is trained by cross-entropy. With $M \in \{16, 32\}$, g_ψ contains $\sim 10^3$–10^4 parameters; it is the only component trained on labels. Frozen $\mathcal{M}_\theta$, prototypes, and standardiser are shared across classifiers and tasks.

Fidelity score. The same prototype distances $\{d_j\}_{j=1}^K$ already computed during concept assignment yield a built-in input-level fidelity measure:

$$F(x) = \exp\left(-\frac{1}{K\tau}\sum_{j=1}^{K} d_j\right) \in (0, 1], \tag{5}$$

where τ is set to the median patch-prototype distance on $\mathcal{D}_{\text{u}}$. High $F(x)$ means every patch lies close to some concept centroid: the input is well-described by the learned vocabulary. Low $F(x)$ means at least one segment falls in a region of signature space that no prototype covers, the input contains regimes the vocabulary cannot describe.

This is structurally different from softmax confidence. Softmax answers *is the classifier confident in its prediction?*; $F(x)$ answers *is the input describable in the concept space at all?* The latter is a representation-level question that the classifier itself cannot pose, because by the time g_ψ sees Π all absolute distance information from ϕ_j to its nearest prototype has been normalised away by Equation 4. Empirically (Section 4.3), $F(x)$ achieves AUROC up to 0.99 for OOD detection where softmax remains near chance.

Table 2: Classification accuracy (%). Best interpretable method in **bold**; best overall underlined.

		UCI-HAR			Epilepsy			Sleep-EDF			UEA (10 avg)		
Group	**Method**	Full	1%	5%	Full	1%	5%	Full	1%	5%	Full	1%	5%
Black-box	ROCKET	93.91	82.21	89.65	97.15	93.68	92.25	76.38	70.21	73.67	72.30	55.69	59.87
	MiniRocket	97.31	86.55	95.63	98.26	94.71	96.86	85.91	79.53	82.15	73.80	51.90	54.89
	InceptionTime	95.32	86.38	91.03	98.21	95.14	97.28	84.62	77.64	81.11	67.42	48.37	50.80
	TS-TCC	93.72	70.23	75.82	97.12	91.20	93.20	85.67	75.26	78.98	72.71	62.37	63.87
	TimesNet	91.82	85.33	90.05	97.57	93.74	96.33	75.88	52.36	55.62	68.76	60.01	62.28
	PatchTST	85.71	28.27	73.46	93.65	92.57	96.67	69.93	65.19	67.72	70.52	49.43	50.90
	GPT4TS	89.37	84.58	87.36	85.51	82.71	83.93	76.22	69.36	71.22	68.31	59.31	65.47
	CrossFormer	77.35	69.89	74.12	76.18	71.55	75.28	53.98	51.28	55.68	70.58	59.63	67.54
	TSLANet	95.41	86.31	90.99	97.26	95.41	96.28	81.51	74.39	78.88	75.35	52.18	59.78
	EntroPE	84.41	71.29	74.14	89.94	85.62	87.61	75.12	70.95	71.84	61.06	56.37	58.27
	MLP (1 layer)	63.88	50.12	56.32	68.14	51.28	59.63	49.57	32.18	33.97	69.39	53.98	57.88
Interpret.	TOTEM	83.83	73.86	78.27	94.56	91.65	93.57	67.75	48.29	52.73	52.99	34.96	38.60
	ProSeNet	84.12	47.63	76.38	96.91	65.51	95.06	61.25	36.67	45.40	52.61	49.67	50.32
	VQShape	72.24	59.00	66.00	90.43	81.00	87.00	59.95	51.36	55.96	59.82	54.67	54.81
	catch22	58.87	44.37	53.92	88.34	86.03	87.12	71.36	68.59	69.71	43.29	32.51	35.11
	CONCEPTTIME-Static (Ours)	81.55	69.00	71.00	95.47	94.00	94.00	73.36	68.26	69.92	67.82	55.98	51.29
	CONCEPTTIME-Dynamic (Ours)	**88.91**	**74.60**	**79.60**	**98.39**	**97.91**	**98.01**	**78.15**	**72.00**	**74.00**	**71.82**	**58.50**	**60.10**

4 Experiments

We evaluate CONCEPTTIME along three axes corresponding to the properties motivating the method, namely, *accuracy*, *interpretability*, and *calibrated abstention*.

4.1 Experimental Setup

Datasets. We evaluate on benchmarks spanning four domains, namely, (i) wearable sensing (UCI HAR [3]), (ii) EEG-based biomedical monitoring (UCI Epilepsy [2], merged to seizure/non-seizure), sleep staging (Sleep-EDF [18, 12], Fpz-Cz channel), (iii) 10 UEA archive datasets [6, 62], and (iv) Fault Diagnosis (FD-A through FD-D [31, 48]) for cross-domain transfer. Full dataset details: App. E.

Configuration. The frozen density model is a 4-layer causal transformer ($d = 128$, $\lambda_{\text{cal}} = 0.25$, 50 epochs). We use $(L_{\min}, L_{\max}) = (8, 40)$, and $K = 8$ patches with small datasets, $K = 32$ on Others. The classifier is a two-layer transformer with $d_{\text{cls}} = 64$. Results are the mean over three seeds (App. Tab. 9); full hyperparameters in Appendix D. We used NVIDIA RTX A5000 single-GPU hardware for all the experiments.

Baselines. We compare against four categories. *Adaptive patching:* CONCEPTTIME-Static (equal-length patches), PatchTST [43], EntroPE [1]. *Black-box accuracy ceiling:* ROCKET [9], MiniRocket [10], InceptionTime [24], TS-TCC [12], TimesNet [62], CrossFormer [67], GPT4TS [68], TSLANet [13]; a one-layer MLP as a lower bound. *Interpretable methods:* catch22 [38], ProSeNet [42], VQShape [58], TOTEM [54]. CONCEPTTIME-Dynamic combines adaptive patching with predictive-geometry concepts and is our primary method.

4.2 Main Classification Results

Across all four benchmarks and all label fractions, CONCEPTTIME-Dynamic is the strongest interpretable method, with margins ranging from 1.5 points (Epilepsy full-shot) to over 30 points (HAR vs. catch22), while every other interpretable method degrades sharply at 1% and 5% labels. At full supervision, CONCEPTTIME matches or exceeds the strongest black-box baseline on Epilepsy and trails by 2-8 points elsewhere, a controlled cost for enforcing a concept bottleneck. Under label scarcity this gap inverts: at 1% labels, CONCEPTTIME beats every black-box baseline on Epilepsy (97.91% vs. TSLANet 95.41%) and remains competitive on UEA. The classifier learns a mapping from concept activations to labels, not from raw signals, which makes it intrinsically more sample-efficient (Table 2).

Predictive geometry, not adaptive patching alone, drives the gain. Three ablations isolate the source of accuracy. Replacing dynamic patches with equal-length windows (CONCEPTTIME-Static) loses 7.4 points on HAR and 2.9 on Epilepsy. Replacing continuous Gaussian segmentation with discrete categorical entropy (EntroPE) loses 4.5–8.5 points across benchmarks. And replacing predictive-geometry signatures with classical morphological descriptors on identical patches and

Table 3: Cross-domain few-shot transfer on Fault Diagnosis (target test accuracy, %).

Setting	A→B	A→C	A→D	B→A	B→C	B→D	C→A	C→B	C→D	D→A	D→B	D→C	#1st places
Direct, 0%	36.25	44.46	29.58	50.62	81.82	98.57	52.05	68.84	73.20	25.99	46.37	48.79	–
Source-$\mathcal{M}_\theta$, 5%	90.14	75.84	91.06	90.40	84.02	98.53	**96.66**	97.43	97.62	77.93	45.56	47.80	1
Target-$\mathcal{M}_\theta$, 5%	**95.42**	**91.20**	**97.29**	**93.73**	**90.54**	98.72	96.08	**98.97**	**98.28**	77.38	56.63	45.45	7
TSLANet, 5%	85.89	82.04	86.36	86.73	87.87	**99.56**	89.11	96.30	96.00	**90.65**	**99.85**	**88.78**	4

clusters (catch22) collapses HAR by 30 points and UEA by 28. The discriminative signal is deviation from prediction, not raw waveform shape; adaptive patching is necessary but not sufficient.

Cross-domain transfer. We evaluate transfer across operating conditions on FD-A through FD-D in three settings (Table 3): *Direct* (source pipeline applied to target with no retraining), *Source-*$\mathcal{M}_\theta$ (5% target labels, fixed source density model and vocabulary, retrain only the classifier), and *Target-*$\mathcal{M}_\theta$ (5% target labels, target-trained density model, source vocabulary fixed). Both transfer settings are classified in the source concept coordinate system via soft assignment, and we compare against TSLANet's masked-reconstruction pretraining plus full-model fine-tuning.

Direct transfer fails (54.23% average), confirming that raw signal statistics are not invariant across operating regimes. The source concept space transfers cleanly: training only a target classifier raises accuracy to 82.72%, and additionally retraining the density model lifts it to 86.99% with eight of twelve direction wins. This isolates local signal statistics, not the concept vocabulary, as the binding constraint on transfer. TSLANet achieves a higher overall average (90.76%) but its advantage is concentrated in transfers from FD-D; on the remaining nine directions CONCEPTTIME matches or exceeds it while providing an inspectable concept vocabulary and fidelity-based abstention.

4.3 Selective Prediction via Concept Fidelity

The same prototype distances $\{d_j\}$ used for concept assignment yield a per-input fidelity score $F(x)$ (Eq. 5) at no additional cost. We test whether this score is informative at the sample level (does it flag inputs the vocabulary cannot describe?) and at the patch level (does it rank the patches the classifier actually relies on?).

Sample-level OOD detection. Figure 3 compares fidelity distance against max-softmax confidence on the same classifier and against prototype-distance scores from TOTEM and VQShape, under three Gaussian-noise intensities and three shuffle perturbations on HAR. Fidelity dominates across all six conditions, scaling smoothly with severity (AUROC $0.864 \rightarrow 0.992$ as noise factor grows from 0.5 to 2.0) while every other score remains near chance. The TOTEM and VQShape comparisons matter: prototype-based OOD detection is not a free property of codebook methods. It requires prototypes anchored in predictive-distribution geometry, not reconstruction-based codes.

Patch-level deletion faithfulness. We rank patches by fidelity, progressively delete the highest-ranked ones, and measure accuracy decay (Algorithm 3); lower AUC means the deleted patches were more important. We compare against three post-hoc explainers (GradientSHAP, TimeX, TimeX++) applied to the same CONCEPTTIME classifier, plus random and reverse-fidelity baselines.

Figure 4 shows that fidelity-guided deletion achieves the lowest or near-lowest AUC on both Epilepsy and HAR, and the reverse baseline is substantially weaker, confirming the ranking is non-arbitrary. Mask-based methods (TimeX, TimeX++) cannot improve on random when explaining a concept-bottleneck classifier: reverse-engineering attributions through input perturbation is unreliable when the model's decision pathway is already structured.

Why fidelity beats softmax structurally. Softmax asks *is the classifier confident in its prediction?*; fidelity asks *is the input describable in the concept space at all?*. By the time logits are computed, absolute distance from each signature to its nearest prototype has been normalised away by Eq. 4; fidelity recovers exactly that information, at no additional training or inference cost.

4.4 Ablation Study

Three patterns emerge (Table 4). **Predictive geometry dominates morphology:** across all three datasets, every density-derived single family (moments, entropy, distributional) outperforms raw-

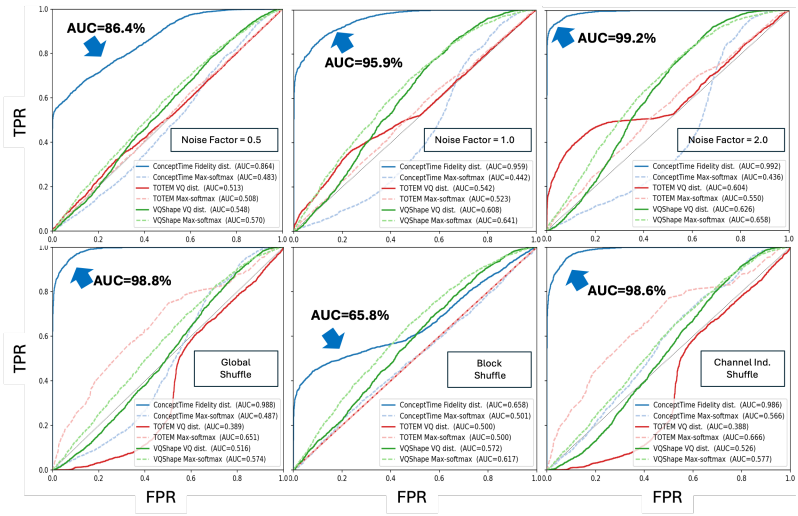

Figure 3: OOD detection on HAR under Gaussian-noise and temporal-shuffle perturbations.

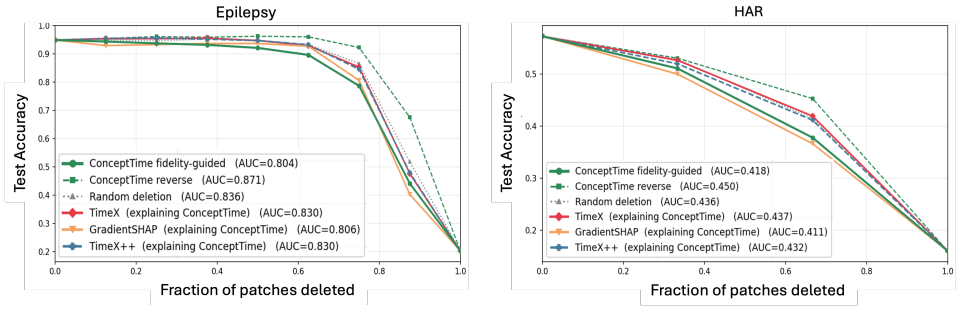

Figure 4: Deletion-based faithfulness on Epilepsy and HAR.

Table 4: Unified ablation on HAR, Epilepsy, and Sleep-EDF (full-supervision accuracy, %). Best value per dataset and ablation group in **bold**.

Dataset	Signature components						Vocabulary size (M)					Boundary signal		
	Dist.	Ent.	Morph.	Mom.	Full	Surp.f.	8	16	32	64	128	Ent.	KL	Surp.
UCI-HAR	72.8	59.9	39.4	**87.3**	82.8	85.1	78.8	82.5	85.1	84.7	**87.9**	**83.4**	81.6	82.5
Epilepsy	97.2	96.1	94.0	96.4	96.7	**98.0**	98.0	98.1	98.0	98.0	**98.4**	97.4	97.6	**98.1**
Sleep-EDF	71.4	64.2	59.0	74.4	**77.5**	74.8	74.3	75.2	74.8	73.4	**76.7**	73.8	**75.0**	74.5

Notes. Dist. = distributional; Ent. = entropy only (signature) / entropy boundary; Morph. = raw-signal morphology; Mom. = predictive moments; Full = all four families ($17{+}2C$ dims); Surp.f. = residual-aware three-family reformulation ($2C{+}13$ dims); KL = KL-shift between consecutive predictive Gaussians; Surp. = per-timestep surprise (Eq. 2).

signal morphology by margins from a few points on Epilepsy to nearly fifty on HAR. The morphological-only column is a catch22-style baseline on our adaptive patches; its gap to the full signature isolates the contribution of predictive grounding from the contribution of adaptive segmentation, and the sign is unambiguous. **No single family dominates:** predictive moments lead on HAR (amplitude-driven activities), distributional features lead on Epilepsy and Sleep-EDF (deviation-driven regimes), which is why the default concatenates all four; the residual-aware reformulation wins on HAR and Epilepsy and is the strongest alternative when class differences manifest as predictive deviations. **Vocabulary and boundary signal are robust:** M varies by at most a few points outside HAR (where larger M helps on a six-class task), and entropy, KL-shift, and surprise differ by at most 1.8 points within any dataset. We default to $M = 32$ for inspectability and to surprise for its alignment with the signature features.

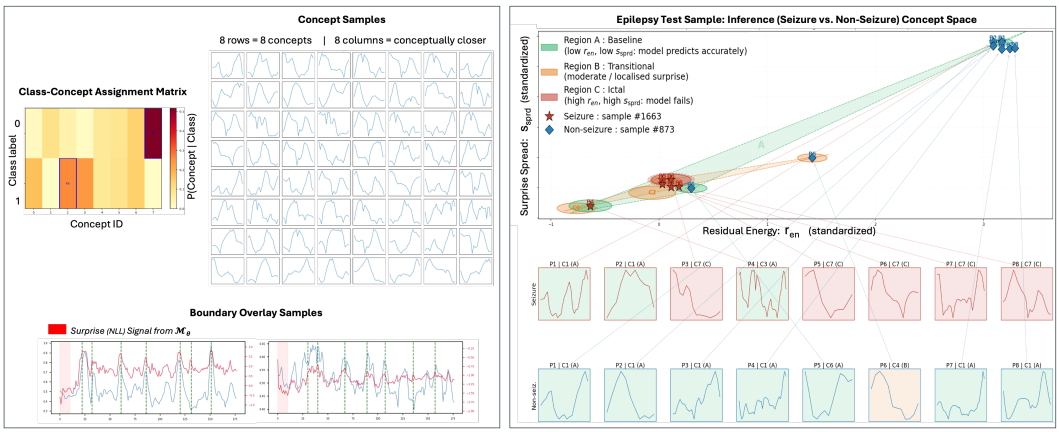

Figure 5: **Qualitative analysis on Epilepsy.** *Top left*, class-conditional concept distribution $P(\text{concept} \mid \text{class})$. Class 0 (non-seizure) concentrates in C7, class 1 (seizure) in C2 (highlighted). *Top middle*, eight nearest-prototype patches per concept (rows C0-C7). Within-row morphological consistency confirms concepts capture coherent regimes. *Bottom left*, predictive surprise s_t (red) on two test samples with DP boundaries. Boundaries align with surprise peaks. *Right*, concept space spanned by residual energy r_{en} and surprise spread s_{sprd}. Two opposite samples are shown. Seizure ($\star$) segments concentrate in Region C where the model fails, non-seizure ($\diamond$) segments in Region A where the model predicts accurately. Both samples contain shared and discriminative segments, and the concept space separates them despite raw-shape similarity. Patches drawn from each region (bottom right) confirm that predictive geometry, not raw shape, drives concept assignment.

4.5 Qualitative Analysis

Figure 5 validates the central claim along four axes. *Concepts are class-discriminative without label supervision* (top left). Seizure and non-seizure samples concentrate in disjoint concepts (C2 and C7) despite the GMM never seeing labels. *Concepts capture coherent dynamical regimes* (top middle). Nearest-prototype patches within each row share predictive behavior, not raw appearance, illustrating that concept membership is driven by deviation from prediction rather than visual similarity. *Boundaries align with predictive surprise* (bottom left). DP segmentation places cuts at s_t peaks, so transitions are localized at segment edges. *Predictive geometry, not raw shape, drives concept assignment* (right). Seizure patches concentrate in Region C where the model fails, non-seizure patches in Region A where the model predicts accurately, and patches drawn from each region are visually heterogeneous within a region while sharing predictive behavior. This is the empirical realization of the central claim from Figure 1. Additional visualizations appear in Appendix A.

5 Conclusion

We introduced ConceptTime, a concept-bottleneck framework that resolves the segmentation-concept circularity for time series through a single self-supervised primitive, *predictive surprise*. The same frozen model places adaptive segment boundaries, grounds a non-parametric concept vocabulary, and supplies prototype distances that double as a built-in fidelity score. Across diverse classification and cross-domain transfer benchmarks, ConceptTime achieves competitive accuracy with black-box methods, dominates every interpretable baseline including in few-shot settings, and beats softmax, prototype-distance, and post-hoc explainers on OOD detection and deletion faithfulness. Interpretability, label efficiency, and reliable abstention emerge jointly when concept grounding is anchored in predictive surprise rather than human annotation.

Limitations and future directions. Concept granularity is fixed rather than learned, and prototype labels remain post-hoc human interpretations. Extensions include hierarchical vocabularies, online updates under distribution shift, fidelity-triggered active relabeling, and applications beyond classification in regime discovery, anomaly localization, and trustworthy decision-support.

References

[1] Sachith Abeywickrama, Emadeldeen Eldele, Min Wu, Xiaoli Li, and Chau Yuen. Entrope: Entropy-guided dynamic patch encoder for time series forecasting. *arXiv preprint arXiv:2509.26157*, 2025.

[2] Ralph G Andrzejak, Klaus Lehnertz, Florian Mormann, Christoph Rieke, Peter David, and Christian E Elger. Indications of nonlinear deterministic and finite-dimensional structures in time series of brain electrical activity: Dependence on recording region and brain state. *Physical Review E*, 64(6):061907, 2001.

[3] Davide Anguita, Alessandro Ghio, Luca Oneto, Xavier Parra, Jorge Luis Reyes-Ortiz, et al. A public domain dataset for human activity recognition using smartphones. In *Esann*, volume 3, pages 3–4, 2013.

[4] Sravan Kumar Ankireddy, Nikita Seleznev, Nam H Nguyen, Yulun Wu, Senthil Kumar, Furong Huang, and C Bayan Bruss. Timesqueeze: Dynamic patching for efficient time series forecasting. *arXiv preprint arXiv:2603.11352*, 2026.

[5] Abdul Fatir Ansari, Lorenzo Stella, Ali Caner Turkmen, Xiyuan Zhang, Pedro Mercado, Huibin Shen, Oleksandr Shchur, Syama Sundar Rangapuram, Sebastian Pineda Arango, Shubham Kapoor, Jasper Zschiegner, Danielle C. Maddix, Hao Wang, Michael W. Mahoney, Kari Torkkola, Andrew Gordon Wilson, Michael Bohlke-Schneider, and Bernie Wang. Chronos: Learning the language of time series. *Transactions on Machine Learning Research*, 2024. ISSN 2835-8856. URL `https://openreview.net/forum?id=gerNCVqqtR`. Expert Certification.

[6] Anthony Bagnall, Hoang Anh Dau, Jason Lines, Michael Flynn, James Large, Aaron Bostrom, Paul Southam, and Eamonn Keogh. The uea multivariate time series classification archive, 2018. *arXiv preprint arXiv:1811.00075*, 2018.

[7] MinGyu Choi and Changhee Lee. Conditional information bottleneck approach for time series imputation. In *The Twelfth International Conference on Learning Representations*, 2023.

[8] Jonathan Crabbé and Mihaela Van Der Schaar. Explaining time series predictions with dynamic masks. In *International conference on machine learning*, pages 2166–2177. PMLR, 2021.

[9] Angus Dempster, François Petitjean, and Geoffrey I Webb. Rocket: exceptionally fast and accurate time series classification using random convolutional kernels. *Data Mining and Knowledge Discovery*, 34(5):1454–1495, 2020.

[10] Angus Dempster, Daniel F Schmidt, and Geoffrey I Webb. Minirocket: A very fast (almost) deterministic transform for time series classification. In *Proceedings of the 27th ACM SIGKDD conference on knowledge discovery & data mining*, pages 248–257, 2021.

[11] Gabriele Dominici, Pietro Barbiero, Francesco Giannini, Martin Gjoreski, Giuseppe Marra, and Marc Langheinrich. Counterfactual concept bottleneck models. In *The Thirteenth International Conference on Learning Representations*, 2025. URL `https://openreview.net/forum?id=w7pMjyjsKN`.

[12] Emadeldeen Eldele, Mohamed Ragab, Zhenghua Chen, Min Wu, Chee Keong Kwoh, Xiaoli Li, and Cuntai Guan. Time-series representation learning via temporal and contextual contrasting. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI-21*, pages 2352–2359, 2021.

[13] Emadeldeen Eldele, Mohamed Ragab, Zhenghua Chen, Min Wu, and Xiaoli Li. Tslanet: Rethinking transformers for time series representation learning. In *International Conference on Machine Learning*, pages 12409–12428. PMLR, 2024.

[14] Irene Ferfoglia, Gaia Saveri, Laura Nenzi, and Luca Bortolussi. Ecats: Explainable-by-design concept-based anomaly detection for time series. In *International Conference on Neural-Symbolic Learning and Reasoning*, pages 175–191. Springer, 2024.

[15] Irene Ferfoglia, Simone Silvetti, Gaia Saveri, Laura Nenzi, and Luca Bortolussi. Towards interpretable concept learning over time series via temporal logic semantics. *arXiv preprint arXiv:2508.03269*, 2025.

[16] Alan H Gee, Diego Garcia-Olano, Joydeep Ghosh, and David Paydarfar. Explaining deep classification of time-series data with learned prototypes. In *CEUR workshop proceedings*, volume 2429, page 15, 2019.

[17] Yonatan Geifman and Ran El-Yaniv. Selectivenet: A deep neural network with an integrated reject option. In *International conference on machine learning*, pages 2151–2159. PMLR, 2019.

[18] 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.

[19] Mononito Goswami, Konrad Szafer, Arjun Choudhry, Yifu Cai, Shuo Li, and Artur Dubrawski. MOMENT: A family of open time-series foundation models. In *Forty-first International Conference on Machine Learning*, 2024. URL `https://openreview.net/forum?id=FVvf69a5rx`.

[20] Josif Grabocka, Nicolas Schilling, Martin Wistuba, and Lars Schmidt-Thieme. Learning time-series shapelets. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 392–401, 2014.

[21] Jajack Heikenfeld, Andrew Jajack, Jim Rogers, Philipp Gutruf, Lei Tian, Tingrui Pan, Ruya Li, Michelle Khine, Jintae Kim, and Juanhong Wang. Wearable sensors: modalities, challenges, and prospects. *Lab on a Chip*, 18(2):217–248, 2018.

[22] Bosong Huang, Ming Jin, Yuxuan Liang, Johan Barthelemy, Debo Cheng, Qingsong Wen, Chenghao Liu, and Shirui Pan. Shapex: Shapelet-driven post hoc explanations for time series classification models. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2026. URL `https://openreview.net/forum?id=EAtSPyQ09Z`.

[23] Md Milon Islam, Ashikur Rahaman, and Md Rashedul Islam. Development of smart healthcare monitoring system in iot environment. *SN computer science*, 1(3):185, 2020.

[24] Hassan Ismail Fawaz, Benjamin Lucas, Germain Forestier, Charlotte Pelletier, Daniel F Schmidt, Jonathan Weber, Geoffrey I Webb, Lhassane Idoumghar, Pierre-Alain Muller, and François Petitjean. Inceptiontime: Finding alexnet for time series classification. *Data mining and knowledge discovery*, 34(6):1936–1962, 2020.

[25] Sujin Jeon, Hyundo Lee, Eungseo Kim, Sanghack Lee, Byoung-Tak Zhang, and Inwoo Hwang. Locality-aware concept bottleneck model. *arXiv preprint arXiv:2508.14562*, 2025.

[26] Hailey Joren, Charles Thomas Marx, and Berk Ustun. Classification with conceptual safeguards. In *The Twelfth International Conference on Learning Representations*, 2024. URL `https://openreview.net/forum?id=t8cBsT9mcg`.

[27] Eunji Kim, Dahuin Jung, Sangha Park, Siwon Kim, and Sungroh Yoon. Probabilistic concept bottleneck models. In *International Conference on Machine Learning*, pages 16521–16540. PMLR, 2023.

[28] Pang Wei Koh, Thao Nguyen, Yew Siang Tang, Stephen Mussmann, Emma Pierson, Been Kim, and Percy Liang. Concept bottleneck models. In *International conference on machine learning*, pages 5338–5348. PMLR, 2020.

[29] Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. *Advances in neural information processing systems*, 30, 2017.

[30] Kimin Lee, Kibok Lee, Honglak Lee, and Jinwoo Shin. A simple unified framework for detecting out-of-distribution samples and adversarial attacks. *Advances in neural information processing systems*, 31, 2018.

[31] Christian Lessmeier, James Kuria Kimotho, Detmar Zimmer, and Walter Sextro. Condition monitoring of bearing damage in electromechanical drive systems by using motor current signals of electric motors: A benchmark data set for data-driven classification. In *PHM society European conference*, volume 3, 2016.

[32] Xiaofeng Liu, Zhihong Liu, Jie Li, and Xiang Zhang. Semi-supervised contrastive learning for time series classification in healthcare. *IEEE Transactions on Emerging Topics in Computational Intelligence*, 9(1):318–331, 2024.

[33] Zhen Liu, Yucheng Wang, Boyuan Li, Junhao Zheng, Emadeldeen Eldele, Min Wu, and Qianli Ma. A unified shape-aware foundation model for time series classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2026.

[34] Zichuan Liu, Tianchun Wang, Jimeng Shi, Xu Zheng, Zhuomin Chen, Lei Song, Wenqian Dong, Jayantha Obeysekera, Farhad Shirani, and Dongsheng Luo. Timex++: Learning time-series explanations with information bottleneck. In *Forty-first International Conference on Machine Learning*, 2024.

[35] Joshua Lockhart, Daniele Magazzeni, and Manuela Veloso. Learn to explain yourself, when you can: Equipping concept bottleneck models with the ability to abstain on their concept predictions. *arXiv preprint arXiv:2211.11690*, 2022.

[36] Haodong Lu, Dong Gong, Shuo Wang, Jason Xue, Lina Yao, and Kristen Moore. Learning with mixture of prototypes for out-of-distribution detection. In *The Twelfth International Conference on Learning Representations*, 2024. URL `https://openreview.net/forum?id=uNkKaD3MCs`.

[37] Wang Lu, Jindong Wang, Xinwei Sun, Yiqiang Chen, Xiangyang Ji, Qiang Yang, and Xing Xie. Diversify: A general framework for time series out-of-distribution detection and generalization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(6):4534–4550, 2024.

[38] Carl H Lubba, Sarab S Sethi, Philip Knaute, Simon R Schultz, Ben D Fulcher, and Nick S Jones. catch22: Canonical time-series characteristics. *arXiv preprint arXiv:1901.10200*, 2019.

[39] Dongsheng Luo, Wei Cheng, Yingheng Wang, Dongkuan Xu, Jingchao Ni, Wenchao Yu, Xuchao Zhang, Yanchi Liu, Yuncong Chen, Haifeng Chen, et al. Time series contrastive learning with information-aware augmentations. In *Proceedings of the AAAI conference on artificial intelligence*, volume 37, pages 4534–4542, 2023.

[40] Panu Maijala, Zhao Shuyang, Toni Heittola, and Tuomas Virtanen. Environmental noise monitoring using source classification in sensors. *Applied Acoustics*, 129:258–267, 2018.

[41] Andrei Margeloiu, Matthew Ashman, Umang Bhatt, Yanzhi Chen, Mateja Jamnik, and Adrian Weller. Do concept bottleneck models learn as intended? *arXiv preprint arXiv:2105.04289*, 2021.

[42] Yao Ming, Panpan Xu, Huamin Qu, and Liu Ren. Interpretable and steerable sequence learning via prototypes. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 903–913, 2019.

[43] Yuqi Nie, Nam H Nguyen, Phanwadee Sinthong, and Jayant Kalagnanam. A time series is worth 64 words: Long-term forecasting with transformers. In *The Eleventh International Conference on Learning Representations*, 2023. URL `https://openreview.net/forum?id=Jbdc0vTOcol`.

[44] Tuomas Oikarinen, Subhro Das, Lam M. Nguyen, and Tsui-Wei Weng. Label-free concept bottleneck models. In *The Eleventh International Conference on Learning Representations*, 2023. URL `https://openreview.net/forum?id=FlCg47MNvBA`.

[45] Artidoro Pagnoni, Ramakanth Pasunuru, Pedro Rodriguez, John Nguyen, Benjamin Muller, Margaret Li, Chunting Zhou, Lili Yu, Jason E Weston, Luke Zettlemoyer, et al. Byte latent transformer: Patches scale better than tokens. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 9238–9258, 2025.

[46] Stanislav Ponomarev and Travis Atkison. Industrial control system network intrusion detection by telemetry analysis. *IEEE Transactions on Dependable and Secure Computing*, 13(2):252–260, 2015.

[47] Owen Queen, Tom Hartvigsen, Teddy Koker, Huan He, Theodoros Tsiligkaridis, and Marinka Zitnik. Encoding time-series explanations through self-supervised model behavior consistency. *Advances in Neural Information Processing Systems*, 36:32129–32159, 2023.

[48] Mohamed Ragab, Zhenghua Chen, Min Wu, Haoliang Li, Chee-Keong Kwoh, Ruqiang Yan, and Xiaoli Li. Adversarial multiple-target domain adaptation for fault classification. *IEEE Transactions on Instrumentation and Measurement*, 70:1–11, 2020.

[49] Fawaz Sammani, Jonas Fischer, and Nikos Deligiannis. CLIP-free, label-free, zero-shot concept bottleneck models. 2025. URL https://openreview.net/forum?id=9YpbmkPmuT.

[50] Yoshihide Sawada and Keigo Nakamura. Concept bottleneck model with additional unsupervised concepts. *IEEE Access*, 10:41758–41765, 2022.

[51] Simon Schrodi, Julian Schur, Max Argus, and Thomas Brox. Selective concept bottleneck models without predefined concepts. *Transactions on Machine Learning Research*, 2025. ISSN 2835-8856. URL https://openreview.net/forum?id=PMO30TLI4l.

[52] Divyansh Srivastava, Ge Yan, and Tsui-Wei Weng. Vlg-cbm: Training concept bottleneck models with vision-language guidance. *Advances in Neural Information Processing Systems*, 37:79057–79094, 2024.

[53] Yiyou Sun, Yifei Ming, Xiaojin Zhu, and Yixuan Li. Out-of-distribution detection with deep nearest neighbors. In *International conference on machine learning*, pages 20827–20840. PMLR, 2022.

[54] Sabera J Talukder, Yisong Yue, and Georgia Gkioxari. TOTEM: TOkenized time series EMbeddings for general time series analysis. *Transactions on Machine Learning Research*, 2024. ISSN 2835-8856. URL https://openreview.net/forum?id=QlTLkH6xRC.

[55] Denis Ullmann, Olga Taran, and Slava Voloshynovskiy. Multivariate time series information bottleneck. *Entropy*, 25(5):831, 2023.

[56] Angela van Sprang, Erman Acar, and Willem Zuidema. Interpretability for time series transformers using a concept bottleneck framework. *arXiv preprint arXiv:2410.06070*, 2024.

[57] Moritz Vandenhirtz, Sonia Laguna, Ričards Marcinkevičs, and Julia E Vogt. Stochastic concept bottleneck models. *Advances in Neural Information Processing Systems*, 37:51787–51810, 2024.

[58] Yunshi Wen, Tengfei Ma, Tsui-Wei Weng, Lam M Nguyen, and Anak A Julius. Abstracted shapes as tokens-a generalizable and interpretable model for time-series classification. *Advances in Neural Information Processing Systems*, 37:92246–92272, 2024.

[59] Yunshi Wen, Tengfei Ma, Ronny Luss, Debarun Bhattacharjya, Achille Fokoue, and Anak Agung Julius. Shedding light on time series classification using interpretability gated networks. In *The Thirteenth International Conference on Learning Representations*, 2025.

[60] Sithumi Wickramasinghe, Bikramjit Das, and Dorien Herremans. Smart timing for mining: A deep learning framework for bitcoin hardware roi prediction. *arXiv preprint arXiv:2512.05402*, 2025.

[61] Carissa Wu, Sonali Parbhoo, Marton Havasi, and Finale Doshi-Velez. Learning optimal summaries of clinical time-series with concept bottleneck models. In *Machine Learning for Healthcare Conference*, pages 648–672. PMLR, 2022.

[62] Haixu Wu, Tengge Hu, Yong Liu, Hang Zhou, Jianmin Wang, and Mingsheng Long. Timesnet: Temporal 2d-variation modeling for general time series analysis. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=ju_Uqw384Oq.

[63] Yulun Wu, Sravan Kumar Ankireddy, Samuel Sharpe, Nikita Seleznev, Dehao Yuan, Hyeji Kim, and Nam H Nguyen. Dynamic tokenization via reinforcement patching: End-to-end training and zero-shot transfer. *arXiv preprint arXiv:2603.26097*, 2026.

[64] Jia Xie, Zhu Wang, Zhiwen Yu, Yasan Ding, and Bin Guo. Prototype learning for medical time series classification via human–machine collaboration. *Sensors*, 24(8):2655, 2024.

[65] Yue Yang, Artemis Panagopoulou, Shenghao Zhou, Daniel Jin, Chris Callison-Burch, and Mark Yatskar. Language in a bottle: Language model guided concept bottlenecks for interpretable image classification. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 19187–19197, 2023.

[66] Lexiang Ye and Eamonn Keogh. Time series shapelets: a new primitive for data mining. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 947–956, 2009.

[67] Yunhao Zhang and Junchi Yan. Crossformer: Transformer utilizing cross-dimension dependency for multivariate time series forecasting. In *The Eleventh International Conference on Learning Representations*, 2023. URL `https://openreview.net/forum?id=vSVLM2j9eie`.

[68] Tian Zhou, Peisong Niu, Liang Sun, Rong Jin, et al. One fits all: Power general time series analysis by pretrained lm. *Advances in neural information processing systems*, 36:43322–43355, 2023.

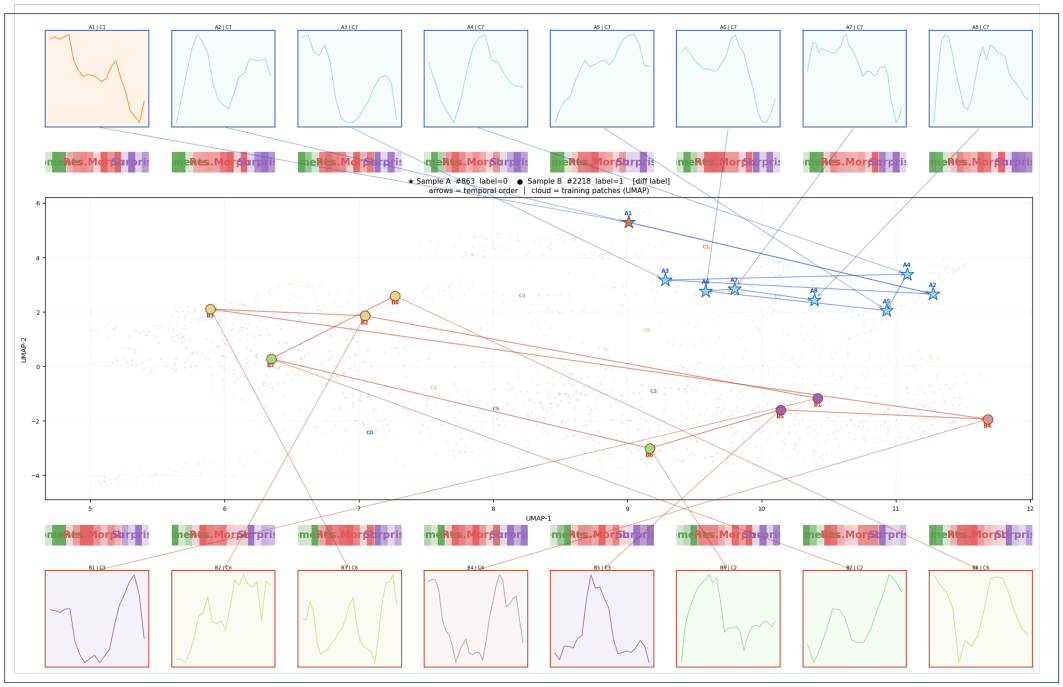

Figure 6: Seizure and Non-Seizure in Concept Space.

A Showcase

B Few-Shot Transfer Learning Results

Table 5: Cross-domain few-shot transfer on Fault Diagnosis (target test accuracy, %).

Setting	A→B	A→C	A→D	B→A	B→C	B→D	C→A	C→B	C→D	D→A	D→B	D→C	#1st places
Direct, 0%	36.25	44.46	29.58	50.62	81.82	98.57	52.05	68.84	73.20	25.99	46.37	48.79	–
Source-$\mathcal{M}_\theta$, 5%	90.14	75.84	91.06	90.40	84.02	98.53	**96.66**	97.43	97.62	77.93	45.56	47.80	1
Target-$\mathcal{M}_\theta$, 5%	**95.42**	**91.20**	**97.29**	**93.73**	**90.54**	98.72	96.08	**98.97**	**98.28**	77.38	56.63	45.45	7
TSLANet, 5%	85.89	82.04	86.36	86.73	87.87	**99.56**	89.11	96.30	96.00	**90.65**	**99.85**	**88.78**	4

C Extended Related Work

CONCEPTTIME sits at the intersection of four research threads: concept bottleneck models, adaptive patching for sequences, prototype- and shape-based interpretable time-series classifiers, and selective prediction. We position our contribution against each thread and then summarize what is, to our knowledge, the gap that no prior work fills.

Concept bottleneck models. Concept bottleneck models (CBMs) [28] route predictions through human-interpretable intermediate concepts, enabling inspection and intervention. Classical CBMs require dense concept annotations, motivating a line of work on *label-free* variants that source concepts from large pretrained models. Label-free CBM [44] and Language-in-a-Bottle [65] use CLIP and LLMs to mine concept names; VLG-CBM [52] grounds concepts via open-vocabulary detectors; recent work eliminates the CLIP dependency through unsupervised classifier-distribution alignment [49]. Other extensions improve concept robustness [41, 50], support counterfactual reasoning [11], and add concept locality [25]. All of these methods, however, rely on *external semantic supervision*, either human annotations or vision-language priors, to populate the concept vocabulary. CONCEPTTIME departs from this paradigm. Its concepts are grounded purely in the geometry of a frozen density model's predictive distribution, requiring neither language models nor human labels. Because patch-to-concept assignment is a non-parametric nearest-centroid operation,

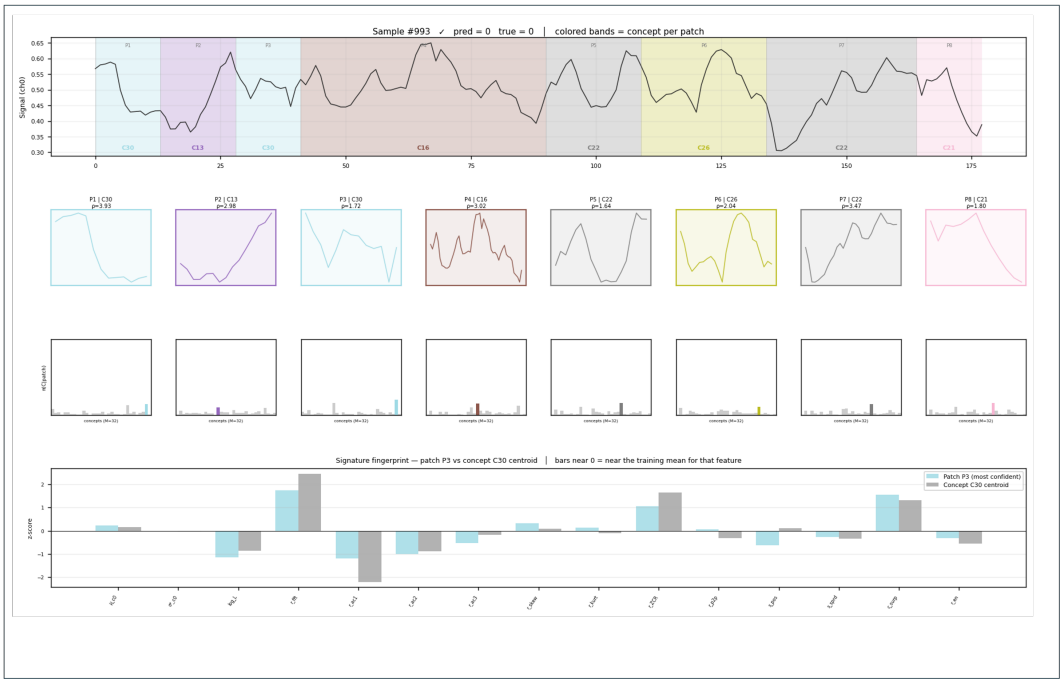

Figure 7: Concept Assignment Confidence of Epilepsy Signal

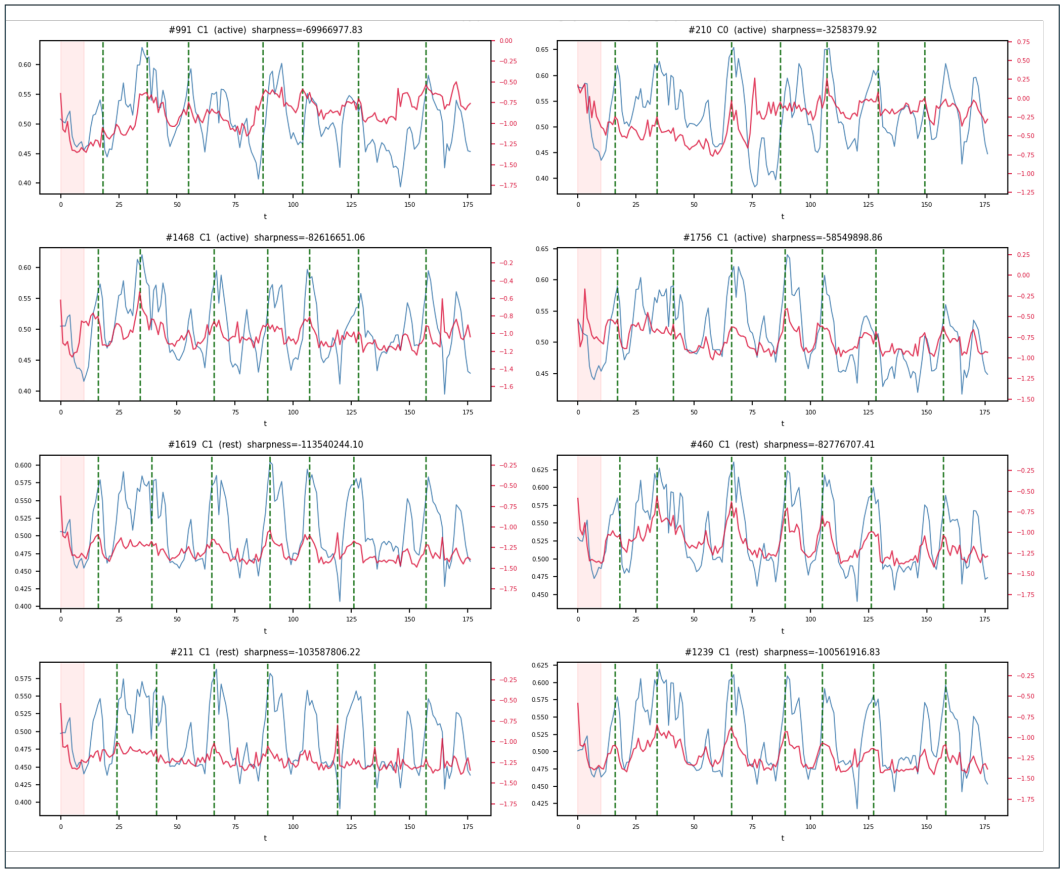

Figure 8: Epilepsy Boundary Overlay

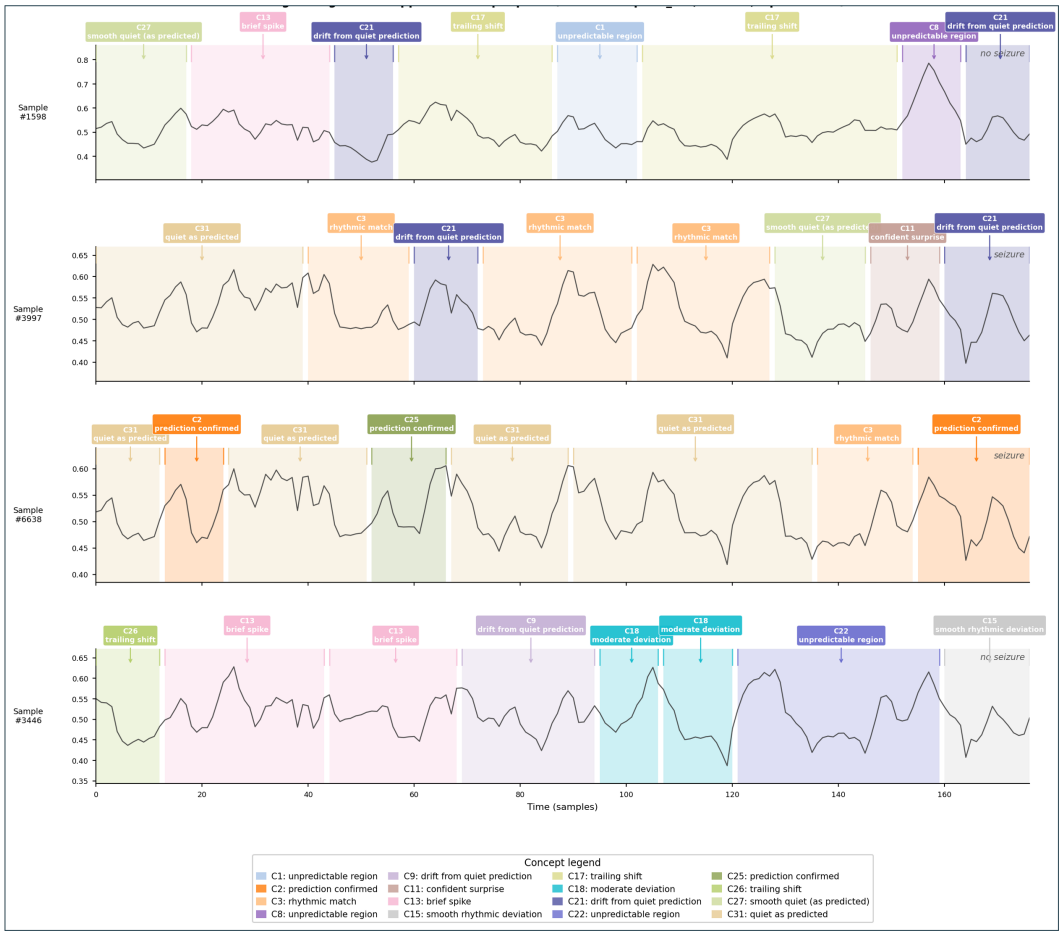

Figure 9: Epilepsy Segments

CONCEPTTIME also avoids the trainable concept-prediction networks that CBMs typically rely on, removing a common source of concept leakage [41].

Concept bottlenecks for time series. CBMs for sequential data remain underexplored. Wu et al. [61] use predefined statistical summaries as concepts for clinical signals; van Sprang et al. [56] apply CKA-aligned concepts to forecasting transformers; Ferfoglia et al. [15] use Signal Temporal Logic formulae as interpretable predicates. Each occupies a narrow niche and depends on a pre-specified concept vocabulary together with fixed-length windowing. None resolves the fundamental coupling between segmentation and concept identity. To our knowledge CONCEPTTIME is the first time-series CBM in which both segmentation and concept construction are driven by the same self-supervised primitive, predictive surprise, removing the need for pre-specified concepts entirely.

Adaptive patching for time series. Patch-based transformers such as PatchTST have demonstrated the value of segmenting time series into patches. The Byte Latent Transformer (BLT) [45] extends this idea by placing boundaries at points of high next-token entropy in language modelling. EntroPE [1] adapts the idea to time series, training a discrete autoregressive model and segmenting via a greedy dual-threshold rule on categorical entropy from quantized tokens. While EntroPE demonstrates that entropy-guided segmentation improves forecasting, it produces no interpretable concept representation from the segmentation backbone. Adaptive patches are simply fed into a standard encoder-classifier pipeline. Concurrent work explores reinforcement-learned patch boundaries (ReinPatch [63]) and content-aware compression for forecasting efficiency (TimeSqueeze [4]), confirming the broader move toward data-adaptive segmentation. These methods optimize patching jointly with a downstream forecasting objective and produce no interpretable concept representation; CONCEPTTIME differs by separating the frozen segmentation-driving density model from the classifier and reusing the same model to ground concepts. CONCEPTTIME differs in three ways. First,

Figure 10: HAR Segments

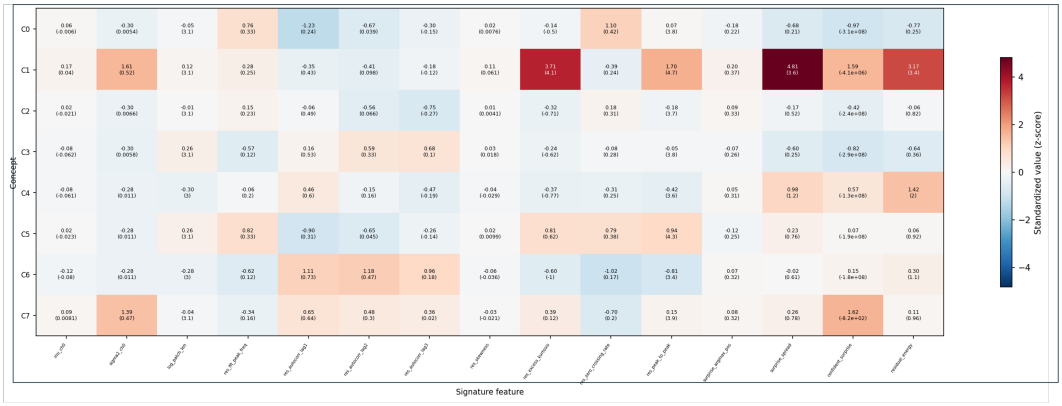

Figure 11: Epilepsy Concept Space with Signature Fingerprints

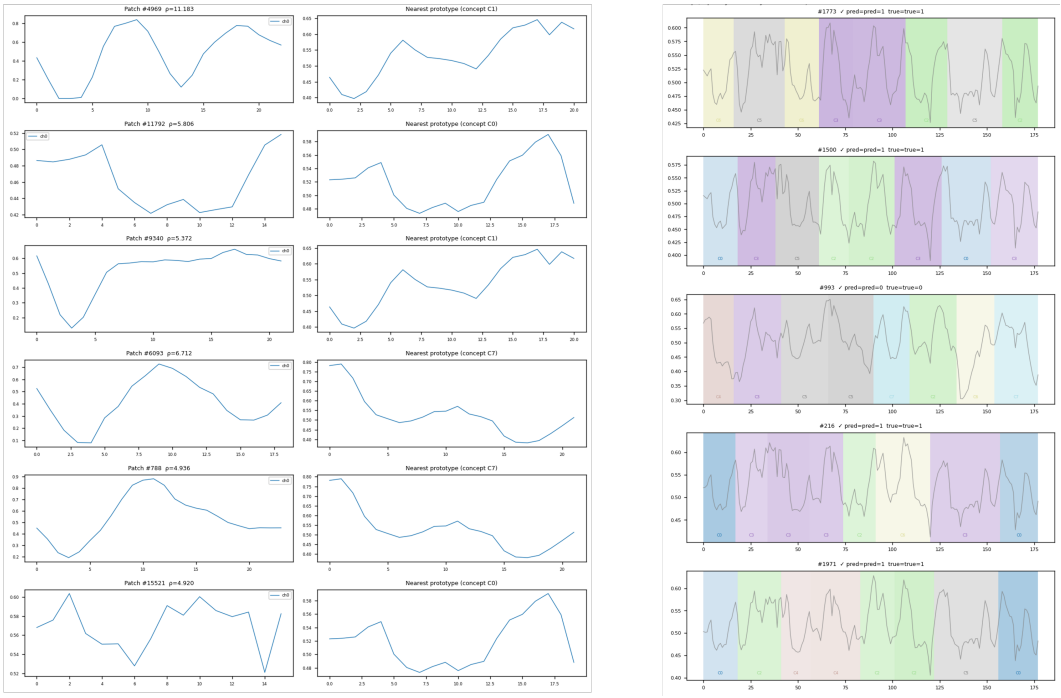

Figure 12: (left): How abstention-based decisions are made, (right): Fidelity weighted segment-concept allocation

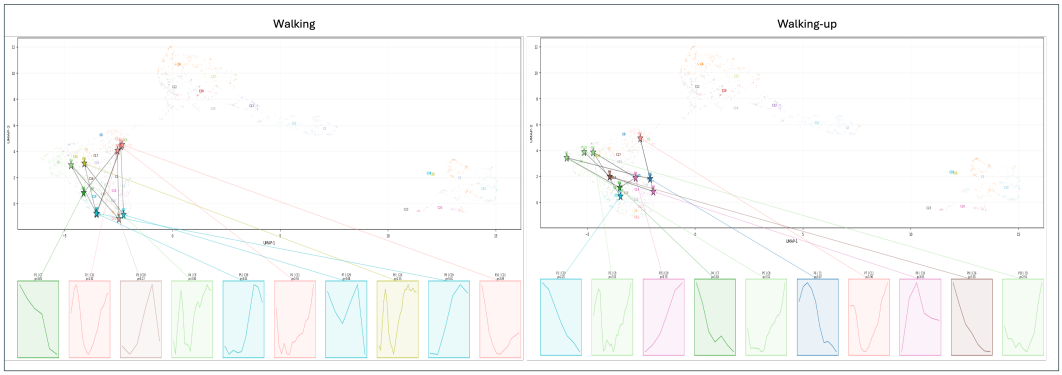

Figure 13: Two close HAR Activities, differentiating in some segments to help discriminate the class via concept space.

it uses continuous Gaussian differential entropy rather than categorical entropy, avoiding binning artefacts and producing rich predictive statistics (μ, σ^2, NLL, KL) that ground the concept vocabulary. Second, it formulates segmentation as length-constrained dynamic programming rather than greedy thresholding, guaranteeing a fixed-cardinality partition with explicit minimum and maximum patch lengths. Third, and most importantly, the same density model that drives segmentation also produces the per-patch features that define concepts, a dual use that no prior adaptive-patching method exploits.

Discrete tokenization and codebook representations. Vector-quantized models offer a related but distinct path to discrete time-series representations. TOTEM [54] learns a VQ-VAE codebook for time-series patches; VQShape [58] learns a shape-level codebook with subsequence reconstruction; Chronos [5] tokenizes scaled values into a 4096-symbol vocabulary for autoregressive forecasting. These methods produce discrete tokens, but the tokens themselves are latent codes without inherent semantic structure. CONCEPTTIME differs by constructing concepts as *prototypes in a meaningful predictive-statistics space*, where the axes (predictability, deviation, activity, morphology) carry direct interpretive meaning, and where prototype distance yields a built-in fidelity score. Where TOTEM tokens require external probing to associate with semantic concepts, CONCEPTTIME's concepts are interpretable by construction.

Prototype- and shape-based interpretable classifiers. A long line of work uses prototypes or shapelets as interpretable units for time-series classification [66, 42, 16, 64]. ProSeNet [42] learns a small library of prototype sequences with case-based reasoning. Shapelet methods [66, 20] discover discriminative subsequences and classify by distance. ECATS [14] and related work [59, 58] use symbolic predicates and abstracted shapes. Recent foundation-model work in this space includes UniShape [33], which uses prototype-based supervised pretraining over multi-scale windowed shapes for transferable classification. Unlike ConceptTime, UniShape requires labeled pretraining and operates on fixed multi-scale windows; its prototypes are anchored in shape space rather than predictive-distribution geometry, and consequently provide no fidelity score for selective prediction. These methods provide local, case-based interpretability but operate on fixed-length windows or learned prototypes that lack a probabilistic grounding. CONCEPTTIME retains the local, prototype-based interpretation paradigm but anchors prototypes in the predictive geometry of a probabilistic world model rather than in raw shape templates alone. This grounding is what enables the fidelity score, an out-of-distribution signal that has no analog in shape-template methods.

Post-hoc explanations for time series. A complementary line of work explains pretrained black-box classifiers post hoc through saliency, attribution, or perturbation [47, 34, 8, 22]. TimeX [47] and TimeX++ [34] apply information bottleneck principles to extract sub-instance explanations; ShapeX [22] aligns Shapley attribution with shapelet segments. These methods are valuable for auditing existing models but do not constrain the classifier to use the explanation during prediction. CONCEPTTIME is interpretable *by construction*: the prediction is routed through the discovered concept vocabulary, so the explanation is the decision pathway rather than a post-hoc reconstruction.

Selective prediction and concept-level uncertainty. Selective prediction has been studied through softmax confidence [17], ensemble disagreement [29], and distance-based OOD scoring [30, 53]; recent work explores prototype-based OOD detection with mixtures [36], and distance to in-distribution centroids. In vision, *concept-based* selective prediction has emerged through Conceptual Safeguards [26] and Sidecar-CBM [35], which propagate concept-level uncertainty into the abstention decision, and through stochastic concept representations [57, 27] and selective concept models [51]. For time series, OOD generalization has been studied independently [37], but no prior work couples a TS concept bottleneck with prototype-distance-based abstention. CONCEPTTIME fills this gap. The fidelity score arises directly from the geometry of the concept space: it is a single-pass, representation-level uncertainty signal that asks not whether the classifier is confident, but whether the input is even *describable* in the learned concept vocabulary.

Predictive-state and information-theoretic representations. Our use of a probabilistic backbone for representation learning connects to predictive state representations, regime discovery, and information-bottleneck approaches for sequential data [39, 7, 55]. Time-series foundation models such as MOMENT [19] and Chronos [5] demonstrate that self-supervised pretraining on raw signals yields useful general-purpose representations. These approaches confirm that predictive behavior defines a meaningful geometry over temporal data, but their primary goal is forecasting, imputation, or general-purpose representation learning, not concept-bottleneck classification. CONCEPTTIME repurposes a frozen density model for a fundamentally different end: to drive both segmentation and

Table 6: Comparison of major directions in interpretable time-series classification. ✓ = direct support; ✗ = absent; ∼ = partial or with caveats.

Property	Post-hoc	CBMs	Adaptive Patch	Prototype / Shapelet	IB / Repr. Learning	CONCEPTTIME (Ours)
Decision routes through concepts	✗	✓	✗	∼	✗	✓
No external semantic supervision required	✓	∼	✓	✓	✓	✓
Concepts discovered from data, not predefined	∼	∼	–	∼	✓	✓
Adaptive segmentation	✗	✗	✓	✗	✗	✓
Local, segment-level explanations	∼	∼	✗	✓	✗	✓
Built-in abstention without auxiliary head	✗	∼	✗	✗	∼	✓
Competitive supervision accuracy	✓	∼	✓	∼	✓	✓
Strong few-shot performance	✗	✗	✗	✗	∼	✓

Notes. Post-hoc: TimeX, TimeX++, ShapeX [47, 34, 22]. CBMs: vision-language CBMs and time-series CBMs [28, 44, 61, 56]; Adaptive Patch: BLT, EntroPE [45, 1]; the dash indicates the property is outside scope (these methods are not designed to expose concepts). Prototype/Shapelet: ProSeNet, ECATS, shapelets [42, 14, 66]. IB/Repr.: information-bottleneck and contrastive representation methods [39, 7].

the construction of an annotation-free concept vocabulary, with abstention emerging naturally from the geometry of that vocabulary.

Positioning. Table 6 summarizes how CONCEPTTIME relates to the major directions surveyed above. The contribution is not the introduction of any single component but their synthesis. A frozen probabilistic model whose predictive uncertainty drives length-constrained adaptive segmentation, whose predictive statistics ground a non-parametric concept vocabulary, and whose prototype geometry yields a built-in fidelity score for selective prediction. To our knowledge, no prior method combines annotation-free concept discovery, predictive-distribution grounding, adaptive segmentation aligned with concepts, and prototype-based abstention within a single coherent framework for multivariate time series.

D Training Hyperparameter

For all training, all datasets we used the same transformer classifier head with M=8, 16, and 32.
Boundary mode = surprise
Patcher = Dynamic
Signature mode = Surprise Full
Classifier lr = 3e-3
Classifier batch size = 64
Classifier epochs = 200
Optimizer = AdamW (wd=0.01)
Scheduler = CosineAnnealing

E Dataset Details

Table 7 summarizes the main benchmark groups used in our experiments. For UEA datasets, we use the official train/test splits and reserve a validation subset from the training split when needed. For the fault-diagnosis transfer study, each working condition is treated as a separate domain with the same label space.

Dataset descriptions. We use datasets spanning wearable sensing, biomedical monitoring, sleep staging, heterogeneous multivariate archives, and cross-domain fault diagnosis.

- **UCI HAR.** The UCI Human Activity Recognition dataset contains smartphone inertial recordings from 30 subjects performing six daily activities: walking, walking upstairs, walking downstairs, sitting, standing, and lying down. Signals were collected using a Samsung Galaxy S2 mounted on the waist, with embedded accelerometer and gyroscope sensors capturing 3-axis linear acceleration and 3-axis angular velocity. Each sample is represented as a 9-channel window of length $T = 128$, and the task is 6-way activity classification.

Table 7: Summary of datasets used in the experiments.

Dataset	Domain / Modality	#Train	#Test	#Ch.	T	Task / Classes
UCI HAR [3]	Wearable inertial sensing	7352	2947	9	128	6 activities
UCI Epilepsy [2]	EEG	9200	2300	1	178	Seizure / non-seizure
Sleep-EDF [18, 12]	Sleep EEG	25612	8910	1	3000	5 sleep stages
UEA subset [6, 62]	Mixed multivariate time series	Varies	Varies	Varies	Varies	10 classification datasets
Fault Diagnosis [31, 48]	Bearing fault diagnosis	8184	2728	1	5120	3 classes; 4 transfer domains

Table 8: UEA multivariate archive subset used in our evaluation. The split column follows the format #train/#val/#test before any validation split is carved from the training set.

Dataset	#Ch.	T	Split	Domain
EthanolConcentration	3	1751	261 / 0 / 263	Alcohol industry
FaceDetection	144	62	5890 / 0 / 3524	Face / sensor signals
Handwriting	3	152	150 / 0 / 850	Handwriting
Heartbeat	61	405	204 / 0 / 205	Heartbeat
JapaneseVowels	12	29	270 / 0 / 370	Voice
PEMS-SF	963	144	267 / 0 / 173	Transportation
SelfRegulationSCP1	6	896	268 / 0 / 293	Health / EEG
SelfRegulationSCP2	7	1152	200 / 0 / 180	Health / EEG
SpokenArabicDigits	13	93	6599 / 0 / 2199	Voice
UWaveGestureLibrary	3	315	120 / 0 / 320	Gesture recognition

- **UCI Epilepsy.** The Epileptic Seizure Recognition dataset consists of single-channel EEG recordings. Each recording corresponds to 23.6 seconds of brain activity. The original dataset contains five classes, but only one class corresponds to epileptic seizure activity. Following common practice, we merge the remaining four non-seizure classes into a single class and formulate the task as binary seizure detection. Each sample has length $T = 178$.
- **Sleep-EDF.** Sleep-EDF contains whole-night polysomnography recordings from the PhysioBank database. The task is sleep-stage classification into five classes: Wake, N1, N2, N3, and REM. Following prior work, we use the single EEG channel Fpz-Cz sampled at 100Hz. Each input window has length $T = 3000$, corresponding to a 30-second EEG epoch.
- **UEA multivariate archive subset.** We evaluate on a subset of 10 datasets from the UEA multivariate time-series classification archive. These datasets cover diverse domains, including alcohol industry measurements, face detection, handwriting, heartbeat monitoring, speech, transportation, EEG self-regulation, spoken digit recognition, and gesture recognition. They vary substantially in dimensionality, sequence length, and number of classes, making them useful for testing whether the learned concept representation generalizes beyond a single sensing modality.
- **Fault Diagnosis.** The Fault Diagnosis dataset is a real-world bearing fault classification benchmark collected under four working conditions, denoted FD-A, FD-B, FD-C, and FD-D. Each working condition is treated as a separate domain because the signal characteristics differ across operating regimes. All domains share the same label space with three classes: healthy, inner fault, and outer fault. We use this dataset for cross-domain transfer experiments, where a model is trained on one operating condition and adapted or evaluated on another.

F Extended Results Section

The Table 9 has the bars with 3 random seeds. We chose TSLANet as the best blackbox baseline for comparison.

Table 9: Accuracy (%) with ± ranges for TSLANet and CONCEPTTIME-Dynamic. The ± values denote standard deviation over three random seeds.

Dataset	Labels	TSLANet	CONCEPTTIME-Dynamic
UCI-HAR	Full	95.41 ± 0.63	88.91 ± 0.84
	1%	86.31 ± 1.42	74.60 ± 1.76
	5%	90.99 ± 1.08	79.60 ± 1.31
Epilepsy	Full	97.26 ± 0.51	98.39 ± 0.42
	1%	95.41 ± 1.19	97.91 ± 0.96
	5%	96.28 ± 0.87	98.01 ± 0.71
Sleep-EDF	Full	81.51 ± 1.24	78.15 ± 1.53
	1%	74.39 ± 2.31	72.00 ± 1.84
	5%	78.88 ± 1.39	74.00 ± 1.67
UEA avg.	Full	75.35 ± 0.82	71.82 ± 1.13
	1%	52.18 ± 2.06	58.50 ± 2.24
	5%	59.78 ± 1.58	60.10 ± 2.11

G Experimental Details

Transfer Learning.

OOD detection protocol. We evaluate whether concept fidelity detects distribution shift by comparing clean test samples against corrupted versions of the same samples. For CONCEPTTIME, the OOD score is the mean distance from each patch signature to its nearest concept prototype; larger values indicate that the input is less describable by the learned concept vocabulary. We compare this representation-level score against max-softmax confidence from the same classifier and against analogous prototype-distance scores from TOTEM and VQShape. OOD detection is evaluated by ROC-AUC, treating clean test samples as in-distribution and corrupted samples as out-of-distribution.

Deletion-based faithfulness protocol. We evaluate whether a patch-importance score identifies temporal regions that are causally relevant to the classifier. Given a trained CONCEPTTIME model, we rank the K patches of each test sample by an attribution method and progressively delete the highest-ranked patches. After each deletion step, the frozen classifier is evaluated on the modified concept sequence. A faithful attribution should remove decision-critical evidence early, causing accuracy to drop rapidly; therefore, lower deletion AUC indicates stronger faithfulness. All methods are evaluated on the same frozen CONCEPTTIME classifier, ensuring that differences reflect the attribution ranking rather than differences in the underlying predictive model.

TimeX and TimeX++ adaptation. TimeX and TimeX++ produce timestep-level masks over the raw input, whereas CONCEPTTIME operates on adaptive patches. To compare them under the same deletion protocol, we first train the TimeX/TimeX++ mask generators to explain the frozen CONCEPTTIME predictions. We then convert their timestep masks into patch-level scores by averaging the mask values over the timesteps belonging to each CONCEPTTIME patch. This yields one importance score per patch, allowing all attribution methods to be evaluated using the same patch-deletion procedure.

Use of Large Language Models

Parts of the text in this paper were refined with the assistance of a large language model (ChatGPT, GPT-5), used exclusively for language polishing and improving clarity of exposition. All ideas, methodology, experiments, and analyses were conceived and executed solely by the authors.

Algorithm 1 Cross-Domain Transfer with Source Concept Vocabulary

Require: Source domain $\mathcal{D}_s$, target domain $\mathcal{D}_t$, trained source pipeline $(\mathcal{M}_{\theta_s}, \mathcal{S}_s, \mathcal{C}_s, g_s)$.
Require: Source density model $\mathcal{M}_{\theta_s}$, source standardizer $\mathcal{S}_s$, source prototypes $\mathcal{C}_s = \{c_m^s\}_{m=1}^M$, source classifier g_s.
Require: Target splits $\mathcal{D}_t^{\mathrm{tr}}, \mathcal{D}_t^{\mathrm{val}}, \mathcal{D}_t^{\mathrm{te}}$ and label fractions $\mathcal{R} = \{0.05, 1.00\}$.
Ensure: Direct source accuracy and transfer accuracies for each target-label fraction.

Direct source deployment.

for each target test sample $x \in \mathcal{D}_t^{\mathrm{te}}$ **do**
  Use $\mathcal{M}_{\theta_s}$ to segment x and extract patch signatures $\Phi_s(x) = \{\phi_j^s(x)\}_{j=1}^K$.
  Standardize signatures with the source standardizer:

$$\tilde{\phi}_j^s(x) = \mathcal{S}_s(\phi_j^s(x)).$$

  Assign each patch to the source concept vocabulary:

$$\pi_j^s(x) = \mathrm{softassign}\big(\tilde{\phi}_j^s(x), \mathcal{C}_s\big).$$

  Form the source-coordinate concept sequence:

$$Z_s(x) = [\pi_1^s(x), \ldots, \pi_K^s(x)].$$

  Predict with the frozen source classifier:

$$\hat{y} = g_s(Z_s(x)).$$

end for
Compute direct source deployment accuracy on $\mathcal{D}_t^{\mathrm{te}}$.

Source-$\mathcal{M}_\theta$ transfer.

for each target split $q \in \{\mathrm{tr}, \mathrm{val}, \mathrm{te}\}$ **do**
  for each sample $x \in \mathcal{D}_t^q$ **do**
    Use $\mathcal{M}_{\theta_s}$ to extract source-GEM signatures $\Phi_s(x)$.
    Standardize with $\mathcal{S}_s$ and assign to $\mathcal{C}_s$ by soft assignment.
    Obtain source-coordinate concept sequence $Z_s(x)$.
  end for
end for
for each label fraction $r \in \mathcal{R}$ **do**
  Select a stratified r-fraction subset of $\mathcal{D}_t^{\mathrm{tr}}$.
  Train a new target classifier $g_{t|s}^{(r)}$ on source-coordinate concept sequences $Z_s(x)$.
  Evaluate $g_{t|s}^{(r)}$ on target test concept sequences.
end for

Target-$\mathcal{M}_\theta$ transfer.

Train or load a target density model $\mathcal{M}_{\theta_t}$.
for each target split $q \in \{\mathrm{tr}, \mathrm{val}, \mathrm{te}\}$ **do**
  for each sample $x \in \mathcal{D}_t^q$ **do**
    Use $\mathcal{M}_{\theta_t}$ to extract target-GEM signatures $\Phi_t(x)$.
    Standardize with the source standardizer:

$$\tilde{\phi}_j^t(x) = \mathcal{S}_s(\phi_j^t(x)).$$

    Assign to the fixed source prototypes:

$$\pi_j^{t\to s}(x) = \mathrm{softassign}\big(\tilde{\phi}_j^t(x), \mathcal{C}_s\big).$$

    Obtain source-coordinate concept sequence:

$$Z_{t\to s}(x) = [\pi_1^{t\to s}(x), \ldots, \pi_K^{t\to s}(x)].$$

  end for
end for
for each label fraction $r \in \mathcal{R}$ **do**
  Select a stratified r-fraction subset of $\mathcal{D}_t^{\mathrm{tr}}$.
  Train a new target classifier $g_{t|t\to s}^{(r)}$ on $Z_{t\to s}(x)$.
  Evaluate $g_{t|t\to s}^{(r)}$ on the target test set.
end for

Repeat.

Repeat for all ordered source–target pairs (s, t) with $s \neq t$.

Algorithm 2 OOD Detection via Concept Fidelity

Require: Trained CONCEPTTIME pipeline with density model M_θ, concept prototypes $\{c_m\}_{m=1}^{M}$, classifier g_ψ, ID test set $\mathcal{D}_{\text{id}}$, corruption operator $\mathcal{C}$.

Ensure: OOD detection AUROC for fidelity distance and softmax confidence.

Step 1: Compute ID scores.

for each test sample $x \in \mathcal{D}_{\text{id}}$ **do**

Use M_θ to compute predictive statistics and surprise sequence.

Segment x into K patches using the trained patcher.

Extract standardized patch signatures $\{\phi_j\}_{j=1}^{K}$.

Assign each patch to the concept vocabulary:

$$\pi_j = \text{softassign}(\phi_j, \{c_m\}_{m=1}^{M}).$$

Form the concept sequence $Z = [\pi_1, \ldots, \pi_K]$.

Compute classifier logits $\ell = g_\psi(Z)$.

Compute prototype distances:

$$d_j = \min_m \|\phi_j - c_m\|_2.$$

Define ID OOD score using fidelity distance:

$$s_{\text{fid}}(x) = \frac{1}{K} \sum_{j=1}^{K} d_j.$$

Define softmax OOD score:

$$s_{\text{sm}}(x) = -\max_y \text{softmax}(\ell)_y.$$

end for

Step 2: Compute OOD scores.

Construct corrupted test set:

$$\mathcal{D}_{\text{ood}} = \{\mathcal{C}(x) : x \in \mathcal{D}_{\text{id}}\}.$$

for each corrupted sample $\tilde{x} \in \mathcal{D}_{\text{ood}}$ **do**

Repeat Steps 2–10 above to obtain $s_{\text{fid}}(\tilde{x})$ and $s_{\text{sm}}(\tilde{x})$.

end for

Step 3: Evaluate OOD detection.

Assign binary labels: ID samples have label 0, corrupted samples have label 1.

Compute ROC-AUC using s_{fid} as the OOD score.

Compute ROC-AUC using s_{sm} as the OOD score.

Repeat for each corruption type and severity.

Baselines.

For TOTEM and VQShape, replace s_{fid} with the corresponding VQ/prototype distance score, and compute the same softmax-confidence baseline from their classifier outputs.

Algorithm 3 Deletion-Based Faithfulness Evaluation

Require: Trained CONCEPTTIME classifier g_ψ, test concept sequences $\{Z_i\}_{i=1}^N$, labels $\{y_i\}_{i=1}^N$, patch-level importance method A.

Ensure: Deletion curve and deletion AUC.

Step 1: Obtain patch-level importance scores.

for each test sample x_i **do**

Obtain its concept sequence:

$$Z_i = [\pi_{i,1}, \ldots, \pi_{i,K}].$$

Compute patch importance scores:

$$a_{i,1}, \ldots, a_{i,K} = A(x_i).$$

Rank patches from most important to least important according to $a_{i,j}$.

end for

Step 2: Progressive patch deletion.

for each deletion fraction $\rho \in \{0, 1/K, 2/K, \ldots, 1\}$ **do**

for each test sample x_i **do**

Let $r = \lfloor \rho K \rfloor$.

Select the top-r most important patches according to A.

Delete selected patches by zeroing their concept activations in Z_i:

$$Z_{i,j}^{(\rho)} = \begin{cases} 0, & \text{if patch } j \text{ is selected for deletion,} \\ Z_{i,j}, & \text{otherwise.} \end{cases}$$

end for

Predict with the frozen classifier:

$$\hat{y}_i^{(\rho)} = \arg\max g_\psi(Z_i^{(\rho)}).$$

Compute test accuracy:

$$\mathrm{Acc}(\rho) = \frac{1}{N} \sum_{i=1}^{N} \mathbb{I}[\hat{y}_i^{(\rho)} = y_i].$$

end for

Step 3: Compute faithfulness score.

Compute deletion AUC:

$$\mathrm{AUC}_{\mathrm{del}} = \int_0^1 \mathrm{Acc}(\rho)\, d\rho.$$

Lower deletion AUC indicates a more faithful attribution method, because accuracy drops faster when important patches are removed.

Compared attribution methods.

CONCEPTTIME: rank patches by prototype distance / fidelity score.

GradientSHAP: compute gradients on the frozen CONCEPTTIME classifier and aggregate attribution over concepts per patch.

TimeX and TimeX++: train mask generators to explain the frozen CONCEPTTIME predictions, then average timestep masks within each CONCEPTTIME patch.

Random deletion: delete patches in random order.

Reverse deletion: delete patches in the reverse of the CONCEPTTIME fidelity ranking.