TIMECED: CONTRIBUTION-BASED EVIDENCE DISCOVERY FOR MULTIVARIATE TIME SERIES CLASSIFICATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Multivariate time series classification (MTSC) models typically integrate temporal and cross-channel information into sequence-level representations, which may obscure important local information. Existing methods explicitly model or emphasize local regions, but do not directly quantify how each region contributes to the classification objective. Motivated by this, we propose TimeCED, a **C**ontribution-Based **E**vidence **D**iscovery framework that separates sequence-level representation learning from local evidence discovery. TimeCED directly quantifies the contribution of each channel-temporal region through the change in classification loss induced by candidate removal. Rather than computing this effect exactly, we use a first-order Taylor expansion to efficiently approximate it. High-contribution candidates are consolidated and reused in subsequent training to improve classification performance. We evaluate TimeCED on all 30 datasets in the UEA multivariate time series classification archive, where it achieves strong performance against nine baseline methods. Further analyses confirm the effectiveness of the defined contribution score and the reliability of the Taylor approximation.

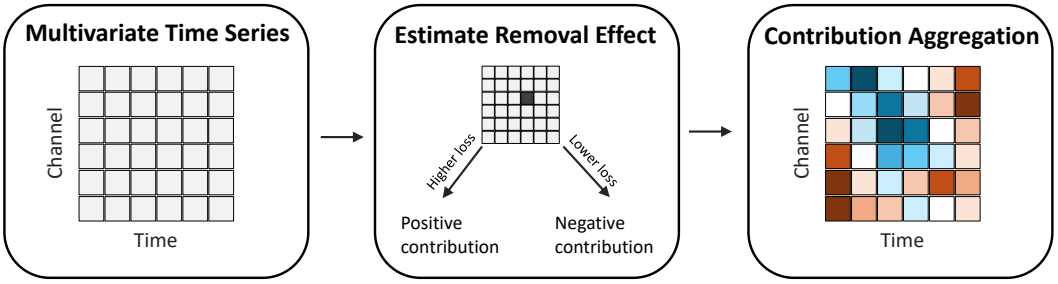

Figure 1: Overview of contribution-based evidence discovery. TimeCED estimates the contribution of each channel-temporal candidate from its removal effect and aggregates the scores across samples and discovery epochs. Candidates with the highest aggregated contributions are selected and reused as evidence for subsequent training.

1 INTRODUCTION

Multivariate time series classification (MTSC) is a fundamental task across a wide range of application domains, including human activity recognition (Xie et al., 2025), industrial monitoring (Dong et al., 2026), healthcare (Ren et al., 2026), and transportation systems (Gupta et al., 2020). A central challenge in MTSC is to model temporal dependencies within individual channels while also capturing interactions across channels. Earlier MTSC research explored a broad range of non-deep classification approaches (Bagnall et al., 2018; Ruiz et al., 2021), whereas modern deep MTSC models commonly learn sequence-level representations by jointly encoding temporal structure and cross-channel dependencies (Karim et al., 2019; Zhang et al., 2020; Foumani et al., 2024; Liu et al., 2024). Such representations are effective at capturing the structure of the full sequence for classi-

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.

fication. However, as information across time steps and channels is progressively integrated, it can obscure the distinct structure of individual local regions.

Identifying local regions is important because class-discriminative information in multivariate signals is often sparse: it may reside in only a few channels over short time intervals, while the remaining observations carry little class-specific information (Hsieh et al., 2021; Chen et al., 2024). To better exploit such localized information, related MTSC methods explicitly model or emphasize local regions during representation learning (Hsieh et al., 2021; Chen et al., 2022; 2024; Li et al., 2021). However, these approaches do not directly quantify how much each local region contributes to the classification objective. This motivates us to define the contribution of each channel-temporal region directly by its effect on the classification objective, rather than infer its importance indirectly during representation learning. These direct contribution estimates allow high-contribution regions to be identified as discriminative evidence and fed back into subsequent training to improve classification performance.

Motivated by this observation, we propose TimeCED, a **C**ontribution-Based **E**vidence **D**iscovery framework for MTSC that separates sequence-level representation learning from local evidence discovery through two complementary pathways. A sequence pathway learns a representation of the full time series by jointly encoding temporal and cross-channel dependencies. In parallel, a candidate pathway operates on explicitly indexed channel-temporal candidates, each corresponding to a local temporal region from one channel, and quantifies their contributions to the classification objective. Candidates with high contributions are then consolidated into a discriminative evidence set and reused in subsequent training. This design preserves a sequence-level representation for classification while explicitly discovering discriminative local evidence.

Training proceeds in two stages. During the discovery stage, TimeCED independently samples a subset of candidates for each training sample at every epoch and incorporates only these sampled candidates into classification together with the sequence-level representation. A coverage criterion is used to ensure sufficient exploration of the candidate space across epochs. The contribution of each candidate is defined as the change in classification loss induced by removing its representation. Since computing this effect exactly would require an additional forward pass for every candidate and is therefore computationally costly, we efficiently approximate it using a first-order Taylor expansion based on gradients already obtained from the classification backward pass. The resulting contribution estimates are accumulated by candidate identity across training samples and discovery epochs. After discovery, the candidates with the highest aggregated contribution scores are selected to form a fixed evidence set and reused for continued training within the same framework.

We evaluate TimeCED on all 30 datasets in the UEA multivariate time series classification archive, where it achieves an average accuracy of 74.4% and the lowest average rank among the compared methods. To assess whether our defined contribution score reflects how class-discriminative a candidate is, we compare high-score candidates against randomly selected and low-score candidates in the post-discovery stage. We further evaluate the reliability of the Taylor approximation against exact candidate removal, obtaining an average Spearman correlation of 0.9954 between the resulting candidate rankings. Our main contributions are summarized as follows:

- We propose **TimeCED**, an MTSC framework that separates sequence-level representation learning from contribution-based evidence discovery over channel-temporal candidates.
- We define candidate contribution as the change in classification loss induced by its removal and derive a first-order Taylor approximation that efficiently estimates this contribution using gradients obtained during training.
- We develop a stochastic evidence discovery procedure that explores candidates across training samples and epochs, aggregates their contributions by candidate identity, and consolidates high-contribution candidates into a fixed evidence set for subsequent training.
- We evaluate TimeCED on all 30 datasets in the UEA multivariate time series classification archive and further validate the effectiveness of contribution-based evidence selection and the reliability of the Taylor approximation against exact candidate removal.

2 Related Work

Multivariate Time Series Classification. MTSC has been extensively studied using both non-deep-learning and deep learning approaches (Bagnall et al., 2018; Ruiz et al., 2021). Classical methods span distance-based approaches such as DTW (Górecki & Łuczak, 2015; Shokoohi-Yekta et al., 2017), symbolic and bag-of-pattern approaches represented by SMTS and WEASEL+MUSE (Baydogan & Runger, 2015; Schäfer & Leser, 2018), and shapelet-based methods (Karlsson et al., 2016; Bostrom & Bagnall, 2017). Deep learning methods learn representations by modeling temporal structure and cross-channel interactions in multivariate time series, with representative architectures including MLSTM-FCN (Karim et al., 2019), TapNet (Zhang et al., 2020), ConvTran (Foumani et al., 2024), and TodyNet (Liu et al., 2024). These methods primarily focus on learning effective sequence-level representations by integrating temporal and cross-channel information, providing strong foundations for MTSC.

Local Evidence Modeling. Recent MTSC methods increasingly exploit localized information through different modeling strategies. Representative approaches include attention-based weighting over temporal regions or variables (Hsieh et al., 2021; Chen et al., 2022), multiple-instance learning formulations that treat local segments as instances and aggregate instance-level information for classification (Chen et al., 2024), and shapelet-based methods that explicitly learn discriminative local subsequences (Li et al., 2021; Le et al., 2024). Although these methods can learn informative local patterns, they do not directly quantify the contribution of each individual local region to the prediction. The work most related to ours is TimeSliver (Pandey et al., 2026), which learns temporal-segment contributions as part of its predictive representation. Its attribution is defined over temporal segments that span all input channels. In contrast, TimeCED preserves channel identity and defines the contribution of each region directly by its effect on the classification objective.

Taylor Approximation. Taylor approximations have been widely used for different purposes. In network pruning, Taylor-based criteria estimate the loss change caused by removing model components (Molchanov et al., 2017; 2019). For model interpretation, Deep Taylor Decomposition attributes classification decisions to input variables (Montavon et al., 2017), with subsequent extensions to class-specific attribution (Iwana et al., 2019), spatio-temporal relevance (Hiley et al., 2019), and Transformer interpretation (Chefer et al., 2021). Feature mining is more closely related to TimeCED. Representative Local Feature Mining uses a Taylor approximation of removal-induced loss change to select discriminative local features (Yan et al., 2021). It measures feature importance by the magnitude of this change. In contrast, TimeCED preserves the sign of the estimated removal effect, allowing the contribution score to capture both the direction and magnitude of a candidate's effect on classification, rather than only its absolute influence.

3 Methodology

In this section, we present TimeCED in three parts. First, we formulate the MTSC problem and introduce the notation used throughout the method in Sec. 3.1. Second, we construct the sequence-level representation together with channel-temporal evidence candidates in Sec. 3.2. Third, we introduce a contribution-based evidence discovery procedure that uses first-order Taylor estimation to identify discriminative candidates in Sec. 3.3. An overview of TimeCED is shown in Fig. 2, and the complete training procedure is summarized in Appendix A.

3.1 Problem Formulation

A multivariate time series dataset can be represented as $\mathcal{D} = \{(X^{(i)}, y^{(i)})\}_{i=1}^{N}$, where N is the number of samples, $X^{(i)} \in \mathbb{R}^{T \times C}$ is the i-th time series with T time steps and C channels, and $y^{(i)} \in \{1, \ldots, K\}$ is its class label among K classes. We denote $x^{(i),t} \in \mathbb{R}^{C}$ as the t-th observation of $X^{(i)}$. The objective of MTSC is to learn a classifier that maps an input multivariate time series to its corresponding class label.

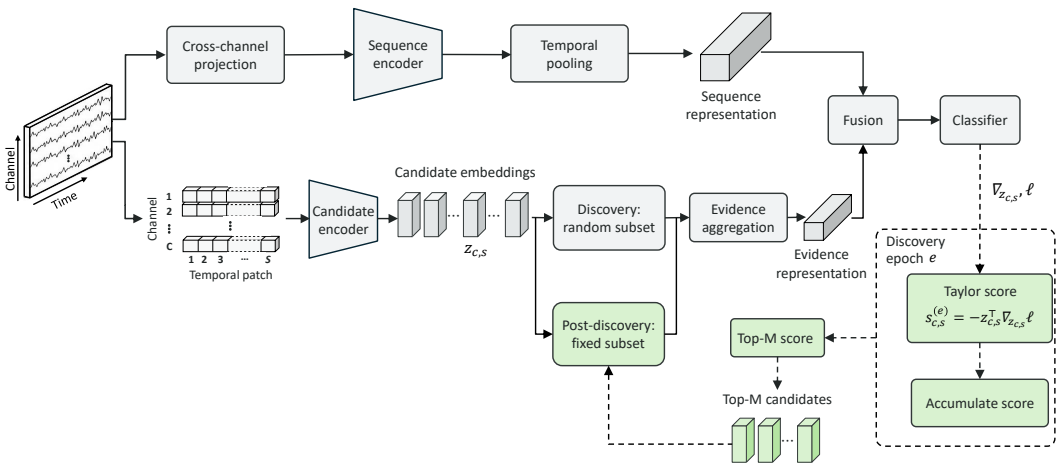

Figure 2: Overview of TimeCED. The sequence pathway encodes the full multivariate time series into a sequence-level representation, while the candidate pathway independently encodes channel-temporal patches with explicit identities. During discovery, a random subset of candidates is independently sampled for each training sample at every epoch, and TimeCED estimates their contributions using a first-order Taylor approximation of the removal-induced loss change. Contribution scores are accumulated by candidate identity to select the top-M evidence candidates. The selected set is fixed during post-discovery training, and the corresponding candidate embeddings are aggregated into an evidence representation, which is fused with the sequence representation for classification.

3.2 Representation and Evidence Candidate Construction

For notational simplicity, we omit the sample index i when describing operations on an individual time series and restore it when sample-specific quantities are required.

Sequence representation. Given a multivariate time series X, we first learn a compact representation of its multivariate temporal structure. Let $x^t \in \mathbb{R}^C$ denote the observation at time point t. A shared learnable projection maps each x^t into a latent space as $\tilde{x}^t = W_p x^t + b_p$, allowing information from different channels to interact at each time point. Positional encoding is added to retain the temporal location of each projected observation. The resulting sequence is then processed by a temporal encoder to capture dependencies across time. This produces $H = [h^1, \ldots, h^T]$, where $h^t \in \mathbb{R}^d$ denotes the encoded representation at time point t. We summarize the sequence through global mean and max pooling and project their concatenation into

$$g = P_\theta \left(\left[\frac{1}{T} \sum_{t=1}^{T} h^t \, ; \, \max_{1 \le t \le T} h^t \right] \right).$$

where P_θ denotes a projection from the pooled sequence to the final representation.

The resulting g provides a sequence-level representation for classification, in which temporal and cross-channel information has been jointly encoded. This representation does not explicitly reveal which time points and channels contribute to the prediction. TimeCED therefore retains a parallel pathway that selects discriminative evidence from candidates whose channel and temporal identities remain explicit.

Evidence candidate construction. We construct evidence candidates from each channel of X. Let $\{\mathcal{T}_s\}_{s=1}^{S}$ denote S temporal patches that evenly divide the sequence. For $c \in \{1, \ldots, C\}$ and $s \in \{1, \ldots, S\}$, the corresponding raw candidate is $e_{c,s} = \{x_c^t\}_{t \in \mathcal{T}_s}$, giving the candidate set $\mathcal{E} = \{e_{c,s}\}_{c=1,s=1}^{C,S}$.

Each candidate has a fixed channel identity c and temporal position τ_s, defined as the center of $\mathcal{T}_s$. Its raw waveform is encoded together with its identities c and τ_s as

$$z_{c,s} = E_\phi(e_{c,s}, c, \tau_s),$$

where E_ϕ is a shared event encoder applied independently to each candidate. The resulting embedding $z_{c,s}$ provides a differentiable representation of candidate $e_{c,s}$ for subsequent contribution estimation.

3.3 Contribution-Based Evidence Discovery

Stochastic candidate sampling. Training begins with an evidence discovery stage that explores the candidate set $\mathcal{E}$ across epochs. For each training sample $X^{(i)}$ at epoch e, a subset of candidate indices $\mathcal{A}^{(i,e)} \subseteq \{1,\dots,C\}\times\{1,\dots,S\}$ is sampled uniformly without replacement, with $|\mathcal{A}^{(i,e)}| = B$. The subsets are sampled independently across training samples and discovery epochs. Only the corresponding candidate embeddings $\{z_{c,s}^{(i)} \mid (c,s) \in \mathcal{A}^{(i,e)}\}$ participate in the evidence pathway for sample i at that epoch. This stochastic resampling exposes different channel-temporal candidates to the classification objective and accumulates observations for contribution estimation. The duration of this stage is determined by the per-sample candidate coverage criterion introduced below.

Given the sampled index set $\mathcal{A}^{(i,e)}$, the corresponding candidate embeddings are aggregated into an evidence representation $r^{(i,e)} = A_\psi(\{z_{c,s}^{(i)} \mid (c,s) \in \mathcal{A}^{(i,e)}\})$, where A_ψ denotes the evidence aggregation module. The sequence representation $g^{(i)}$ and evidence representation $r^{(i,e)}$ are concatenated and mapped to class logits

$$o^{(i,e)} = F_\omega\Big([g^{(i)}; r^{(i,e)}]\Big) \in \mathbb{R}^K,$$

where F_ω is the classification head. The class probabilities are $p^{(i,e)} = \mathrm{softmax}(o^{(i,e)})$, and the corresponding cross-entropy loss is $\ell^{(i,e)} = -\log p_{y^{(i)}}^{(i,e)}$. The gradient $\nabla_{z_{c,s}^{(i)}} \ell^{(i,e)}$ for each sampled candidate is obtained from the same backward pass and is used to estimate its contribution.

Contribution estimation. We quantify the contribution of candidate $e_{c,s}$ by the change in classification loss when its representation $z_{c,s}$ is suppressed to zero while the remaining inputs are unchanged. Let $\ell_{-(c,s)}$ denote the resulting loss. The removal effect is defined as $\Delta_{c,s} = \ell_{-(c,s)} - \ell$.

Inspired by Taylor-based importance estimation in neural network pruning (Molchanov et al., 2017; 2019), we derive a first-order approximation of the removal effect for each evidence candidate. Around the current candidate representation $z_{c,s}$, a first-order Taylor expansion gives

$$\ell(z_{c,s} + \delta) = \ell(z_{c,s}) + \nabla_{z_{c,s}} \ell^\top \delta + O(\|\delta\|_2^2).$$

Removing the candidate representation corresponds to setting $z_{c,s}$ to zero, giving $\delta = -z_{c,s}$. Substituting this perturbation yields

$$\Delta_{c,s} = \ell_{-(c,s)} - \ell = -z_{c,s}^\top \nabla_{z_{c,s}} \ell + O(\|z_{c,s}\|_2^2).$$

Neglecting the higher-order remainder gives the first-order approximation $\Delta_{c,s} \approx -z_{c,s}^\top \nabla_{z_{c,s}} \ell$. Restoring the sample and epoch indices, we define the contribution score as $s_{c,s}^{(i,e)} = -(z_{c,s}^{(i)})^\top \nabla_{z_{c,s}^{(i)}} \ell^{(i,e)}$. A larger score indicates a greater increase in classification loss when the candidate representation is removed, assigning higher contribution to the corresponding evidence candidate $e_{c,s}$.

Discovery duration. The discovery stage spans multiple epochs to collect repeated contribution observations across each sample's candidate space. For a fixed sample i, since B out of CS candidates are sampled uniformly at each epoch, a fixed candidate is selected with probability $q = B/(CS)$. Under independent resampling across epochs, let $n_{i,c,s}$ denote the number of epochs in which candidate (c,s) is sampled for sample i after E epochs. Then $n_{i,c,s} \sim \mathrm{Binomial}(E, q)$.

We introduce a target exposure count r. For any discovery duration satisfying $E \geq \lceil rCS/B \rceil$, the expected number of observations of each candidate satisfies $\mathbb{E}[n_{i,c,s}] = qE \geq r$.

We define the candidate coverage for sample i as the proportion of its candidates observed at least once,

$$\mathrm{Cov}_i = \frac{1}{CS} \sum_{c=1}^{C} \sum_{s=1}^{S} \mathbf{1}(n_{i,c,s} \geq 1).$$

Since each candidate is sampled with probability $q = B/(CS)$ at each epoch, the expected coverage is $\mathbb{E}[\mathrm{Cov}_i] = 1 - (1-q)^E$. Using $1 - q \leq e^{-q}$ and $qE \geq r$, we obtain

$$\mathbb{E}[\mathrm{Cov}_i] \geq 1 - e^{-r}.$$

A larger r therefore provides a higher expected coverage lower bound but incurs greater discovery cost. We set $r = 5$ by balancing coverage and efficiency, as analyzed in Appendix B. Since this choice is constrained by the computational cost of large candidate spaces, $\lceil rCS/B \rceil$ may yield an overly short discovery stage for small candidate spaces. We therefore introduce a minimum discovery duration $E_{\min}$ and set

$$E_{\mathrm{disc}} = \max\left(E_{\min}, \left\lceil \frac{rCS}{B} \right\rceil\right).$$

Evidence consolidation and reuse. For each sample $X^{(i)}$ and discovery epoch e, the sampled set $\mathcal{A}^{(i,e)}$ produces the contribution scores $\{s_{c,s}^{(i,e)} \mid (c,s) \in \mathcal{A}^{(i,e)}\}$. After completing the E_{disc} discovery epochs, contribution observations sharing the same candidate identity (c,s) are aggregated across samples and epochs. Let $\Omega_{c,s} = \{(i,e) \mid (c,s) \in \mathcal{A}^{(i,e)}\}$ denote the sample-epoch pairs in which candidate (c,s) is observed. Its aggregated contribution is

$$\bar{s}_{c,s} = \frac{1}{|\Omega_{c,s}|} \sum_{(i,e) \in \Omega_{c,s}} s_{c,s}^{(i,e)}.$$

Candidate identities are then ranked by $\bar{s}_{c,s}$, and the M candidates with the highest contributions form the selected evidence set $\mathcal{I}^* = \mathrm{TopM}_{(c,s)}\, \bar{s}_{c,s}$. After evidence consolidation, training proceeds to a post-discovery stage in which the selected evidence set $\mathcal{I}^*$ is fixed for all subsequent epochs. For each $X^{(i)}$, candidate embeddings $\{z_{c,s}^{(i)} \mid (c,s) \in \mathcal{I}^*\}$ are aggregated to form $r^{(i)}$, which is combined with $g^{(i)}$ for classification. All trainable components continue to be optimized jointly under the same cross-entropy objective.

4 Experiments

4.1 Experimental Setup

Datasets. We conduct experiments on all 30 datasets from the UEA multivariate time series classification archive. The archive includes datasets from various domains, with different numbers of channels and sequence lengths. For each dataset, we use the official training and test splits. The official test split is used only for final evaluation. Detailed dataset statistics are provided in Appendix D.

Compared Methods. We compare TimeCED with a broad set of representative MTSC baselines, including the non-deep learning methods DTW-1NN-D (Bagnall et al., 2018) and MiniROCKET (Dempster et al., 2021), as well as the deep learning methods MLSTM-FCN (Karim et al., 2019), TapNet (Zhang et al., 2020), ConvTran (Foumani et al., 2024), TodyNet (Liu et al., 2024), TimeMIL (Chen et al., 2024), MPTSNet (Mu et al., 2025), and TimeSliver (Pandey et al., 2026). For DTW-1NN-D, MLSTM-FCN, and TapNet, we use the UEA benchmark results reported in TapNet (Zhang et al., 2020). All remaining baselines are reproduced using their publicly released implementations with the default configurations. For ConvTran, TimeMIL, TodyNet, MPTSNet, and TimeSliver, we preserve their original criterion of selecting checkpoints based on accuracy by constructing a 20% split from the training data as a validation set and selecting the checkpoint with the highest validation accuracy. For each random seed, the same validation split is used across these methods. All reproduced methods are evaluated over three independent runs with random seeds 0, 1, and 42.

Implementation Details. We use a common configuration across all datasets. During the discovery stage, we set the candidate sampling budget to 100. We also require at least 20 discovery epochs to

Table 1: Comparison of classification accuracy on the 30 UEA multivariate time series classification datasets. The best result on each dataset is shown in **bold**, and the second-best is underlined.

Dataset	DTW-1NN-D	MiniROCKET KDD'21	MLSTM-FCN Neur. Netw.'19	TapNet AAAI'20	ConvTran DMKD'24	TimeMIL ICML'24	TodyNet Inf. Sci.'24	MPTSNet AAAI'25	TimeSliver ICLR'26	**TimeCED**
ArticularyWordRecognition	0.987	**0.993**	0.973	0.987	0.969	0.936	0.953	0.958	0.978	0.986
AtrialFibrillation	0.200	0.089	0.267	0.333	0.267	0.244	0.267	**0.400**	0.333	0.378
BasicMotions	0.975	**1.000**	0.950	**1.000**	**1.000**	0.983	0.900	0.875	**1.000**	**1.000**
CharacterTrajectories	0.990	0.943	0.985	**0.997**	0.987	0.984	0.984	0.968	0.968	0.992
Cricket	**1.000**	0.986	0.917	0.958	0.977	0.935	0.847	0.884	0.935	0.981
DuckDuckGeese	0.600	0.687	0.675	0.575	0.580	0.587	0.273	0.513	0.420	**0.713**
EigenWorms	0.618	**0.944**	0.504	0.489	0.794	0.723	0.664	0.466	0.733	0.891
Epilepsy	0.964	**1.000**	0.761	0.971	0.964	0.952	0.879	0.877	0.961	0.954
ERing	0.133	**0.978**	0.133	0.133	0.802	0.770	0.774	0.867	0.441	0.943
EthanolConcentration	0.323	0.449	0.373	0.323	0.265	0.283	0.286	0.376	0.340	**0.466**
FaceDetection	0.529	0.578	0.545	0.556	0.653	0.545	0.655	**0.671**	0.624	0.615
FingerMovements	0.530	0.553	**0.580**	0.530	0.497	0.527	0.503	0.497	0.507	0.560
HandMovementDirection	0.231	0.333	0.365	0.378	0.360	0.266	0.342	0.455	0.270	**0.482**
Handwriting	**0.607**	0.524	0.286	0.357	0.212	0.194	0.238	0.245	0.211	0.345
Heartbeat	0.717	0.740	0.663	0.751	0.759	0.676	0.745	0.727	0.686	**0.769**
InsectWingbeat	0.115	0.355	0.167	0.208	0.709	**0.710**	0.637	0.641	0.570	0.677
JapaneseVowels	0.949	0.959	0.976	0.965	0.985	0.950	0.959	0.925	0.852	**0.986**
Libras	0.872	**0.909**	0.856	0.850	0.835	0.719	0.717	0.741	0.437	0.819
LSST	0.551	0.641	0.373	0.568	0.627	0.639	0.626	0.569	0.621	**0.673**
MotorImagery	0.500	0.493	0.510	**0.590**	0.463	0.477	0.473	0.523	0.517	0.517
NATOPS	0.883	0.909	0.889	0.939	**0.941**	0.924	0.887	0.820	0.869	0.937
PEMS-SF	0.711	0.821	0.699	0.751	0.709	0.204	0.647	**0.838**	0.803	0.740
PenDigits	0.977	0.964	0.978	0.980	0.982	0.943	0.981	0.974	0.976	**0.983**
PhonemeSpectra	0.151	**0.313**	0.110	0.175	0.267	0.287	0.288	0.102	0.159	0.249
RacketSports	0.803	0.855	0.803	0.868	0.829	0.833	0.785	0.820	0.868	**0.914**
SelfRegulationSCP1	0.775	**0.903**	0.874	0.652	0.895	0.775	0.874	0.871	0.782	0.883
SelfRegulationSCP2	0.539	0.511	0.472	**0.550**	0.546	0.522	0.530	0.507	0.513	0.543
SpokenArabicDigits	0.963	0.805	0.990	0.983	**0.993**	0.990	0.991	0.985	0.987	0.990
StandWalkJump	0.200	0.378	0.067	0.400	0.356	0.378	0.311	0.333	**0.444**	0.422
UWaveGestureLibrary	0.903	**0.917**	0.891	0.894	0.861	0.832	0.808	0.797	0.580	0.897
Average Accuracy (↑)	0.643	0.718	0.621	0.657	0.703	0.660	0.661	0.674	0.646	**0.744**
# Best (↑)	2	**9**	1	4	3	1	0	3	2	**9**
Average Rank (↓)	6.300	4.133	6.517	4.617	4.600	6.683	6.550	6.567	6.250	**2.783**
Ours Wins	24	18	27	20	20	27	27	26	24	–
Ours Draws	0	1	1	1	1	1	0	0	2	–
Ours Losses	6	11	2	9	9	2	3	4	4	–

avoid an overly short discovery stage for small candidate spaces. Post-discovery training runs for at most 100 epochs with early stopping. The sequence encoder consists of five TCN blocks, while the shared candidate encoder is implemented as a two-layer MLP. We reserve 20% of the official training split for validation and select the best checkpoint based on validation accuracy. Additional architecture and implementation details are provided in Appendix C.

Metrics. Classification accuracy is used as the primary metric for each dataset. For reproduced baselines and our method, we report the mean accuracy over three runs. Overall benchmark performance is summarized by average accuracy, average rank, and the number of datasets on which each method achieves the best accuracy. We also report pairwise wins, draws, and losses between TimeCED and each baseline. Standard deviations for methods evaluated over three runs are provided in Appendix E.

4.2 Main Results

TimeCED achieves an average accuracy of 74.4% across all 30 UEA datasets, obtaining the best accuracy on 9 datasets and the lowest average rank of 2.78 among the ten methods. Compared with MiniROCKET, the strongest competing method in terms of average accuracy, TimeCED improves the average accuracy by 2.6 percentage points and reduces the average rank from 4.13 to 2.78. The improvement in average rank indicates that TimeCED performs consistently well across datasets rather than relying on particularly strong results on only a small subset. This is also reflected in the pairwise comparison with MiniROCKET, where TimeCED achieves 18 wins, 1 draw, and 11 losses. More broadly, against every other baseline, TimeCED achieves at least 20 wins out of 30 datasets, demonstrating that its advantage is consistently observed across a substantial majority of the benchmarks rather than being concentrated on a few datasets. Overall, these results show that TimeCED combines strong average classification performance with competitive performance across a broad range of MTSC datasets.

4.3 Ablation Study

Table 2: Ablation of key design choices.

Variant	Δ (pp)
TimeCED (Full)	–
w/o Candidate Pathway	-3.60
w/o Sequence Pathway	-5.07
w/o Candidate Channel Encoding	-0.67
w/o Candidate Position Encoding	-2.25
w/o Sequence Position Encoding	-0.79

Table 2 summarizes the ablation results on the 30 UEA datasets. The full TimeCED model achieves an average accuracy of 0.744. We first examine the two representation pathways. Removing the candidate pathway reduces the average accuracy by 3.60 percentage points, while removing the sequence pathway leads to a larger decrease of 5.07 percentage points, showing that both pathways are important for classification.

We next examine the encoding of channel and temporal information. Removing candidate channel encoding decreases accuracy by 0.67 percentage points, while removing candidate position encoding results in a larger drop of 2.25 percentage points. These results indicate that preserving the channel identity and temporal location of candidates is important for evidence discovery. Removing positional encoding from the sequence pathway reduces accuracy by 0.79 percentage points, showing that temporal position information contributes to sequence representation learning.

Table 3: Comparison of evidence selection strategies on datasets with at least 200 candidates.

Strategy	DDG	FD	FM	HMD	HB	IW	MI	NAT	PEMS	PS	SAD	Avg.
Random-100	0.667	0.612	0.507	0.374	0.745	0.673	0.497	**0.943**	0.728	0.230	0.988	0.633
Bottom-100	0.693	0.610	0.490	0.320	0.720	0.674	0.483	0.933	0.723	0.239	0.987	0.625
Top-100 (TimeCED)	**0.713**	**0.615**	**0.560**	**0.482**	**0.769**	**0.677**	**0.517**	0.937	**0.740**	**0.249**	**0.990**	**0.659**

We further evaluate whether the aggregated contribution scores provide a useful signal for evidence selection. Top-100 corresponds to the full TimeCED model and uses the 100 candidates with the highest contribution scores from the discovery stage. Random-100 replaces the selected Top-100 candidates with 100 candidates sampled uniformly at random, while Bottom-100 uses the 100 candidates with the lowest scores. For both variants, we keep the post-discovery training procedure and training budget unchanged. We compare the three strategies on datasets with at least 200 candidates, ensuring that the Top-100 and Bottom-100 subsets are disjoint.

As shown in Table 3, Top-100 achieves the highest average accuracy of 0.659, outperforming Random-100 and Bottom-100 by 2.6 and 3.4 percentage points, respectively. Top-100 achieves the best performance on 10 of the 11 datasets, outperforming Random-100 on 10 datasets and Bottom-100 on all 11 datasets. These results show that the contribution scores provide an effective ranking signal for evidence selection.

4.4 Analysis of Contribution Scores

To assess the reliability of the contribution scores, we examine their consistency across random initializations and their agreement with exact candidate removal. Figure 3 first illustrates these properties on *HandMovementDirection*.

The top row visualizes the aggregated contribution score of each channel-temporal candidate at the end of the discovery stage. Across the three seeds, high positive contributions repeatedly concentrate in the early temporal patches of lower channels and the later temporal patches of higher channels. This recurring pattern shows that TimeCED consistently identifies similar channel-temporal regions as important evidence across random initializations.

We next evaluate how faithfully the Taylor scores recover the candidate ranking induced by exact removal. For each seed, we use the checkpoint at the end of the discovery stage and perform 10 independent candidate draws on the validation set, with 100 candidates sampled per sample in each draw. For each sampled candidate, we compute its Taylor contribution and exact removal effect. The exact removal effect is obtained by setting the corresponding candidate embedding to zero and measuring the resulting change in classification loss. After all draws, both measures are aggregated by candidate identity to obtain one Taylor score and one exact removal score for each candidate.

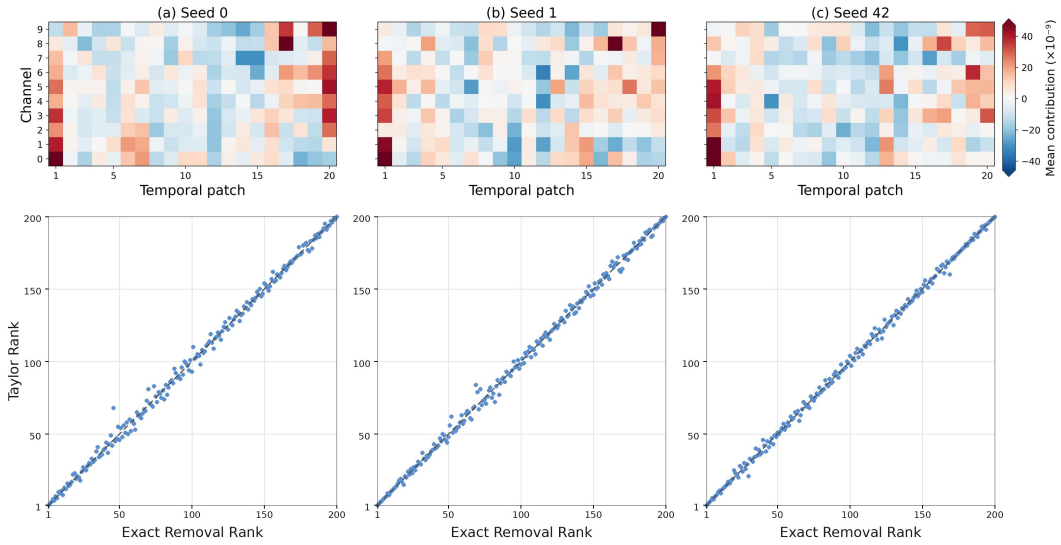

Figure 3: Contribution-score analysis on *HandMovementDirection* across three random seeds.

The bottom row of Figure 3 shows the resulting rankings on *HandMovementDirection*. The candidate ranks closely follow the diagonal across all three seeds, indicating strong agreement between Taylor estimation and exact removal. We further extend this analysis to all 11 datasets with at least 200 candidates. As shown in Table 4, the Taylor scores achieve an average Spearman correlation of 0.9954 with the exact removal scores and an average Top-100 overlap of 97.7%. The agreement remains consistently high across datasets, with all Spearman correlations above 0.97 and all Top-100 overlaps above 94%. These results show that the Taylor approximation reliably recovers the candidate importance ranking induced by exact removal and provides an effective signal for evidence selection.

Table 4: Agreement between Taylor contribution scores and exact removal effects.

Metric	DDG	FD	FM	HMD	HB	IW	MI	NAT	PEMS	PS	SAD	Avg.
Spearman ↑	0.9991	0.9998	0.9996	0.9987	0.9988	0.9864	0.9992	0.9992	0.9728	0.9957	0.9997	0.9954
Top-100 Overlap ↑	96.0%	99.0%	99.0%	99.0%	96.3%	94.7%	98.0%	100.0%	96.0%	98.3%	98.0%	97.7%

5 Conclusion

We presented TimeCED, a framework that combines sequence-level multivariate representation learning with contribution-based raw evidence discovery for MTSC. During the discovery stage, we estimate the contributions of candidates over channel-temporal regions of the raw data using a first-order Taylor approximation. After the discovery stage, we select and reuse high-contribution candidates as evidence in subsequent training to improve classification performance.

Across all 30 UEA datasets, TimeCED achieves an average accuracy of 74.4% and the lowest average rank of 2.78 among the compared methods. Ablation results demonstrate the complementary roles of the sequence and candidate pathways, while contribution-based selection consistently favors more useful evidence than random or low-contribution alternatives. Contribution-score analysis further shows consistent channel-temporal patterns across random initializations, and the Taylor approximation closely agrees with exact candidate removal in candidate ranking. These results show that TimeCED can retain strong multivariate representations while discovering and effectively leveraging discriminative evidence from the original signal.

Reproducibility Statement

The complete training procedure is provided in Appendix A. Architecture, optimization, evidence discovery, preprocessing, and evaluation details are provided in Appendix C. Dataset statistics and results across random seeds are reported in Appendices D and E, respectively.

AI Use Statement

Generative AI tools were used to assist with polishing the manuscript, retrieving relevant literature, and debugging code. All AI-assisted content was reviewed and verified by the authors. The authors take responsibility for the final content of the paper.

References

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.

Mustafa Gokce Baydogan and George C. Runger. Learning a symbolic representation for multivariate time series classification. *Data Mining and Knowledge Discovery*, 29(2):400–422, 2015. doi: 10.1007/s10618-014-0349-y.

Aaron Bostrom and Anthony Bagnall. A shapelet transform for multivariate time series classification. *arXiv preprint arXiv:1712.06428*, 2017.

Hila Chefer, Shir Gur, and Lior Wolf. Transformer interpretability beyond attention visualization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 782–791, 2021. doi: 10.1109/CVPR46437.2021.00084.

Rongjun Chen, Xuanhui Yan, Shiping Wang, and Guobao Xiao. DA-Net: Dual-attention network for multivariate time series classification. *Information Sciences*, 610:472–487, 2022. doi: 10.1016/j.ins.2022.07.178.

Xiwen Chen, Peijie Qiu, Wenhui Zhu, Huayu Li, Hao Wang, Aristeidis Sotiras, Yalin Wang, and Abolfazl Razi. TimeMIL: Advancing multivariate time series classification via a time-aware multiple instance learning. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 7190–7206. PMLR, 2024.

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*, pp. 248–257, 2021. doi: 10.1145/3447548.3467231.

Feiyan Dong, Yue Xiao, Shi Chen, Kazuyuki Demachi, Shigeru Takaya, and Masanori Yoshikawa. End-to-end discriminative representation learning for fault diagnosis in safety-critical time series. *Advanced Engineering Informatics*, 69:104094, 2026. doi: 10.1016/j.aei.2025.104094.

Navid Mohammadi Foumani, Chang Wei Tan, Geoffrey I Webb, and Mahsa Salehi. Improving position encoding of transformers for multivariate time series classification. *Data Mining and Knowledge Discovery*, 38(1):22–48, 2024. doi: 10.1007/s10618-023-00948-2.

Tomasz Górecki and Maciej Łuczak. Multivariate time series classification with parametric derivative dynamic time warping. *Expert Systems with Applications*, 42(5):2305–2312, 2015. doi: 10.1016/j.eswa.2014.11.007.

Ashish Gupta, Hari Prabhat Gupta, Bhaskar Biswas, and Tanima Dutta. An early classification approach for multivariate time series of on-vehicle sensors in transportation. *IEEE Transactions on Intelligent Transportation Systems*, 21(12):5316–5327, 2020. doi: 10.1109/TITS.2019.2957325.

Liam Hiley, Alun Preece, Yulia Hicks, A. David Marshall, and Harrison Taylor. Discriminating spatial and temporal relevance in deep Taylor decompositions for explainable activity recognition. *arXiv preprint arXiv:1908.01536*, 2019.

Tsung-Yu Hsieh, Suhang Wang, Yiwei Sun, and Vasant G. Honavar. Explainable multivariate time series classification: A deep neural network which learns to attend to important variables as well as time intervals. In *Proceedings of the 14th ACM International Conference on Web Search and Data Mining*, pp. 607–615. ACM, 2021. doi: 10.1145/3437963.3441815.

Brian Kenji Iwana, Ryohei Kuroki, and Seiichi Uchida. Explaining convolutional neural networks using softmax gradient layer-wise relevance propagation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, pp. 4176–4185, 2019. doi: 10.1109/ICCVW.2019.00513.

Fazle Karim, Somshubra Majumdar, Houshang Darabi, and Samuel Harford. Multivariate LSTM-FCNs for time series classification. *Neural Networks*, 116:237–245, 2019. doi: 10.1016/j.neunet.2019.04.014.

Isak Karlsson, Panagiotis Papapetrou, and Henrik Boström. Generalized random shapelet forests. *Data Mining and Knowledge Discovery*, 30(5):1053–1085, 2016. doi: 10.1007/s10618-016-0473-y.

Xuan-May Le, Ling Luo, Uwe Aickelin, and Minh-Tuan Tran. ShapeFormer: Shapelet transformer for multivariate time series classification. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 1484–1494, 2024.

Guozhong Li, Byron Choi, Jianliang Xu, Sourav S. Bhowmick, Kwok-Pan Chun, and Grace Lai-Hung Wong. ShapeNet: A shapelet-neural network approach for multivariate time series classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pp. 8375–8383, 2021.

Huaiyuan Liu, Donghua Yang, Xianzhang Liu, Xinglei Chen, Zhiyu Liang, Hongzhi Wang, Yong Cui, and Jun Gu. TodyNet: Temporal dynamic graph neural network for multivariate time series classification. *Information Sciences*, 677:120914, 2024.

Pavlo Molchanov, Stephen Tyree, Tero Karras, Timo Aila, and Jan Kautz. Pruning convolutional neural networks for resource efficient inference. In *International Conference on Learning Representations*, 2017.

Pavlo Molchanov, Arun Mallya, Stephen Tyree, Iuri Frosio, and Jan Kautz. Importance estimation for neural network pruning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11256–11264, 2019. doi: 10.1109/CVPR.2019.01152.

Grégoire Montavon, Sebastian Lapuschkin, Alexander Binder, Wojciech Samek, and Klaus-Robert Müller. Explaining nonlinear classification decisions with deep Taylor decomposition. *Pattern Recognition*, 65:211–222, 2017. doi: 10.1016/j.patcog.2016.11.008.

Yang Mu, Muhammad Shahzad, and Xiao Xiang Zhu. MPTSNet: Integrating multiscale periodic local patterns and global dependencies for multivariate time series classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 19572–19580, 2025.

Akash Pandey, Payal Mohapatra, Wei Chen, Qi Zhu, and Sinan Keten. TimeSliver: Symbolic-linear decomposition for explainable time series classification. In *International Conference on Learning Representations*, 2026.

Xinyue Ren, Yuxuan Xiu, Ting Chen, Zhaosheng Yao, and Wai Kin Victor Chan. TPVNet: A domain-aware graph-based framework for reliable multivariate physiological time series classification in healthcare. *Computer Methods and Programs in Biomedicine*, 276:109214, 2026. doi: 10.1016/j.cmpb.2025.109214.

Alejandro Pasos Ruiz, Michael Flynn, James Large, Matthew Middlehurst, and Anthony Bagnall. The great multivariate time series classification bake off: A review and experimental evaluation of recent algorithmic advances. *Data Mining and Knowledge Discovery*, 35:401–449, 2021. doi: 10.1007/s10618-020-00727-3.

Patrick Schäfer and Ulf Leser. Multivariate time series classification with WEASEL+MUSE. In *Proceedings of the 3rd ECML/PKDD Workshop on Advanced Analytics and Learning on Temporal Data*, 2018.

Mohammad Shokoohi-Yekta, Bing Hu, Hongxia Jin, Jun Wang, and Eamonn Keogh. Generalizing DTW to the multi-dimensional case requires an adaptive approach. *Data Mining and Knowledge Discovery*, 31(1):1–31, 2017. doi: 10.1007/s10618-016-0455-0.

Jingchao Xie, Yang Wu, Wei Li, Mingxin Yang, Rui Hou, and Yahong Li. FTC-Net: Fourier transform convolution for multivariate time series classification in human activity recognition. In *2025 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–8. IEEE, 2025. doi: 10.1109/IJCNN64981.2025.11229268.

Kun Yan, Lingbo Liu, Jun Hou, and Ping Wang. Representative local feature mining for few-shot learning. In *2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1730–1734. IEEE, 2021. doi: 10.1109/ICASSP39728.2021.9413890.

Xuchao Zhang, Yifeng Gao, Jessica Lin, and Chang-Tien Lu. TapNet: Multivariate time series classification with attentional prototypical network. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pp. 6845–6852, 2020. doi: 10.1609/aaai.v34i04.6165.

A TimeCED Algorithm

Algorithm 1 summarizes the complete training procedure of TimeCED. During the discovery stage, channel-temporal candidates are independently sampled for each training sample and their contributions are estimated from the classification gradients. The resulting scores are aggregated across samples and discovery epochs to identify candidates that consistently contribute to classification. After discovery, the top-M candidates are fixed as the evidence set and reused during subsequent training.

B Discovery Coverage and Efficiency

Table 5 shows the trade-off between the target exposure r, expected candidate coverage, and discovery cost, measured by the discovery duration E_{disc}. Increasing r improves the coverage lower bound but increases discovery cost approximately linearly. We use $r = 5$, which provides a coverage lower bound of 99.33% while limiting discovery cost for large candidate spaces.

C Additional Implementation Details

C.1 Model Architecture

The sequence pathway first projects each observation $x_i^t \in \mathbb{R}^C$ to a 128-dimensional latent representation through a shared linear layer. A fixed 128-dimensional sinusoidal positional encoding is added to each projected observation before temporal encoding. We use a five-block TCN with hidden dimensions $80, 80, 80, 80$, and 320. The dilation factors are $1, 2, 4, 8$, and 16, respectively. Each block contains two one-dimensional convolutions with a kernel size of 3, followed by GELU activation and dropout with probability 0.1. Residual connections are applied across all blocks. Padding positions are masked after each block.

The temporal outputs are normalized and summarized using masked mean and max pooling. Their concatenation is projected to a 256-dimensional sequence representation g_i. This projection consists of layer normalization, a linear layer, GELU activation, and dropout with probability 0.1.

Each evidence candidate $e_{i,j}$ contains a raw single-channel waveform together with its channel and temporal identities. The temporal identity is represented by an eight-dimensional sinusoidal encoding of the candidate center normalized by the sequence length, using frequencies $1, 2, 4$, and 8. Channel identity is represented using a C-dimensional one-hot vector. These components are concatenated and passed through a shared MLP,

$$\mathrm{LayerNorm} \to \mathrm{Linear}(256) \to \mathrm{GELU} \to \mathrm{Dropout}(0.1) \to \mathrm{Linear}(128) \to \mathrm{LayerNorm},$$

Algorithm 1 TimeCED: Contribution-Based Evidence Discovery

Require: Training data $\mathcal{D}$, number of patches S, sampling budget B, target exposure r, minimum discovery epochs $E_{\min}$, number of retained candidates M

1: Construct channel-temporal candidates $\{e_{c,s}\}_{c=1,s=1}^{C,S}$
2: Set $E_{\text{disc}} = \max(E_{\min}, \lceil rCS/B \rceil)$
3: **for** $e = 1, \ldots, E_{\text{disc}}$ **do**
4: For each sample i, independently sample $\mathcal{A}^{(i,e)} \subseteq \{1, \ldots, C\} \times \{1, \ldots, S\}$, with $|\mathcal{A}^{(i,e)}| = B$
5: Encode sampled candidates: $z_{c,s}^{(i)} = E_\phi(e_{c,s}^{(i)}, c, \tau_s)$
6: Compute sequence representation $g^{(i)}$ and evidence representation $r^{(i,e)} = A_\psi(\{z_{c,s}^{(i)} : (c,s) \in \mathcal{A}^{(i,e)}\})$
7: Compute classification loss $\ell^{(i,e)}$
8: Compute contribution scores

$$s_{c,s}^{(i,e)} = -(z_{c,s}^{(i)})^\top \nabla_{z_{c,s}^{(i)}} \ell^{(i,e)}, \quad (c,s) \in \mathcal{A}^{(i,e)}$$

9: Update model parameters using the classification loss
10: **end for**
11: Aggregate contribution scores across samples and epochs:

$$\bar{s}_{c,s} = \text{Avg}\{s_{c,s}^{(i,e)}\}$$

12: Select

$$\mathcal{I}^* = \text{TopM}_{(c,s)}\, \bar{s}_{c,s}$$

13: Continue training with the fixed evidence set $\mathcal{I}^*$
14: **return** $\mathcal{I}^*$ and learned model

Table 5: Expected candidate coverage lower bound and average exposure-controlled discovery epochs across the 30 UEA datasets for different target exposure counts.

r	Coverage Lower Bound	Avg. Exposure Epochs
1	63.21%	34.37
2	86.47%	51.33
3	95.02%	68.97
4	98.17%	86.70
5	**99.33%**	**104.57**
6	99.75%	122.60

producing the candidate embedding $z_{i,j} \in \mathbb{R}^{128}$. Each embedding is then normalized by a layer normalization.

Candidate embeddings are assigned to fixed slots according to their channel-temporal identities. Unselected or unavailable candidates are represented by zero vectors. The resulting slot representation is flattened and projected to a 256-dimensional evidence representation r_i. The sequence and evidence representations are normalized separately and concatenated. A fusion MLP then maps the resulting 512-dimensional representation to the class logits through a 256-dimensional hidden layer with GELU activation and dropout with probability 0.2.

C.2 Evidence Discovery and Training

Each channel is partitioned along the temporal dimension using `torch.chunk` with a target of 20 chunks. Because `torch.chunk` may return fewer chunks than requested, the actual number of temporal patches S can be smaller than 20. Specifically, for a sequence of length L, the chunk length is $\lceil L/20 \rceil$, yielding

$$S = \left\lceil \frac{L}{\lceil L/20 \rceil} \right\rceil.$$

For short sequences with $L < 20$, this reduces to $S = L$, with each patch containing exactly one time step.

We set the candidate sampling budget to 100. Since some datasets contain fewer than 100 candidates in total, we use all available candidates at each discovery epoch. Their contribution scores are obtained from the classification backward pass and accumulated according to candidate identity.

We optimize all classification components using AdamW. The sequence encoder and its projection use a learning rate of 2×10^{-4}, while the event encoder, evidence projection, and fusion classifier use 10^{-4}. Weight decay is set to 10^{-4} and the batch size is 32. Gradients are clipped to a global norm of 1.0. No learning-rate scheduler is used.

After discovery, we retain the top $M = 100$ candidates, or all available candidates when fewer than 100 exist. Post-discovery training runs for at most 100 epochs with an early-stopping patience of 30 epochs. Early stopping begins after evidence discovery and uses validation accuracy as the primary criterion. Validation macro-F1 and validation loss are used sequentially to break ties.

C.3 Preprocessing and Evaluation

We retain the official UEA TRAIN and TEST splits. The official TRAIN split is further divided into 80% training data and 20% validation data using stratified sampling.

Missing and non-finite values are replaced with zeros. Channel-wise means and standard deviations are computed from the official TRAIN split and used to normalize both TRAIN and TEST data. Variable-length sequences are right-padded with zeros within a batch. A temporal mask records valid positions so that padding does not contribute to sequence pooling.

Training sequences longer than 3000 time steps are divided into consecutive chunks with a maximum length of 3000. The final chunk is right-padded when necessary, and each chunk inherits the class label of its original sequence. Validation and TEST sequences retain their complete temporal length. Padding positions remain excluded through the temporal mask.

The official TEST split is then evaluated once. Experiments are conducted on NVIDIA RTX A5000 GPUs with 24 GB of memory.

D Dataset Details

Table 6 summarizes the number of channels, sequence length, number of classes, and official training and test set sizes for each dataset. Dataset abbreviations used in the paper are also provided. For the four datasets containing variable-length time series, the minimum and maximum sequence lengths are reported.

E Variance Across Random Seeds

Table 7 reports the standard deviations of classification accuracy across three seeds for all reproduced methods. TimeCED achieves a mean standard deviation of 0.0148 across the 30 datasets, indicating relatively stable performance across repeated runs. MiniROCKET exhibits the lowest mean standard deviation of 0.0086, while TimeCED shows lower average variability than the remaining deep learning baselines.

Table 6: Statistics of the 30 datasets in the UEA multivariate time series classification archive.

Dataset	Abbrev.	Channels	Length	Classes	Train Set Size	Test Set Size
ArticularyWordRecognition	AWR	9	144	25	275	300
AtrialFibrillation	AF	2	640	3	15	15
BasicMotions	BM	6	100	4	40	40
CharacterTrajectories	CT	3	60–182	20	1422	1436
Cricket	CR	6	1197	12	108	72
DuckDuckGeese	DDG	1345	270	5	50	50
EigenWorms	EW	6	17984	5	128	131
ERing	ER	4	65	6	30	270
Epilepsy	EPI	3	206	4	137	138
EthanolConcentration	EC	3	1751	4	261	263
FaceDetection	FD	144	62	2	5890	3524
FingerMovements	FM	28	50	2	316	100
HandMovementDirection	HMD	10	400	4	160	74
Handwriting	HW	3	152	26	150	850
Heartbeat	HB	61	405	2	204	205
InsectWingbeat	IW	200	2–22	10	25000	25000
JapaneseVowels	JV	12	7–29	9	270	370
LSST	LSST	6	36	14	2459	2466
Libras	LIB	2	45	15	180	180
MotorImagery	MI	64	3000	2	278	100
NATOPS	NAT	24	51	6	180	180
PEMS-SF	PEMS	963	144	7	267	173
PenDigits	PD	2	8	10	7494	3498
PhonemeSpectra	PS	11	217	39	3315	3353
RacketSports	RS	6	30	4	151	152
SelfRegulationSCP1	SCP1	6	896	2	268	293
SelfRegulationSCP2	SCP2	7	1152	2	200	180
SpokenArabicDigits	SAD	13	4–93	10	6599	2199
StandWalkJump	SWJ	4	2500	3	12	15
UWaveGestureLibrary	UW	3	315	8	120	320

Table 7: Standard deviations of classification accuracy across three seeds on the 30 UEA multivariate time series classification datasets.

Dataset	MiniROCKET	ConvTran	TimeMIL	TodyNet	MPTSNet	TimeSliver	**TimeCED**
ArticularyWordRecognition	0.0000	0.0068	0.0057	0.0047	0.0096	0.0083	0.0016
AtrialFibrillation	0.0314	0.0943	0.0629	0.0544	0.0544	0.0544	0.0831
BasicMotions	0.0000	0.0000	0.0118	0.0736	0.0816	0.0000	0.0000
CharacterTrajectories	0.0190	0.0013	0.0029	0.0012	0.0096	0.0046	0.0000
Cricket	0.0000	0.0065	0.0398	0.0818	0.0236	0.0236	0.0065
DuckDuckGeese	0.0094	0.0432	0.0680	0.0736	0.0499	0.0712	0.0094
EigenWorms	0.0036	0.0286	0.0559	0.0974	0.0187	0.0249	0.0072
Epilepsy	0.0000	0.0000	0.0090	0.0123	0.0296	0.0068	0.0034
ERing	0.0000	0.0126	0.0240	0.0579	0.0277	0.0709	0.0063
EthanolConcentration	0.0062	0.0249	0.0450	0.0260	0.0522	0.0109	0.0182
FaceDetection	0.0030	0.0102	0.0452	0.0029	0.0027	0.0068	0.0020
FingerMovements	0.0047	0.0340	0.0094	0.0094	0.0368	0.0170	0.0327
HandMovementDirection	0.0127	0.0608	0.0255	0.0989	0.0521	0.0552	0.0319
Handwriting	0.0019	0.0141	0.0080	0.0423	0.0312	0.0151	0.0040
Heartbeat	0.0023	0.0061	0.0409	0.0083	0.0105	0.0046	0.0046
InsectWingbeat	0.0429	0.0038	0.0019	0.0041	0.0006	0.0110	0.0019
JapaneseVowels	0.0034	0.0056	0.0111	0.0123	0.0270	0.0677	0.0046
Libras	0.0052	0.0094	0.0340	0.0045	0.0343	0.0228	0.0277
LSST	0.0035	0.0042	0.0143	0.0030	0.0038	0.0022	0.0048
MotorImagery	0.0047	0.0386	0.0249	0.0776	0.0262	0.0330	0.0047
NATOPS	0.0069	0.0052	0.0069	0.0146	0.0105	0.0069	0.0026
PEMS-SF	0.0082	0.0238	0.0547	0.2330	0.0216	0.0263	0.0411
PenDigits	0.0004	0.0037	0.0212	0.0011	0.0063	0.0033	0.0007
PhonemeSpectra	0.0020	0.0042	0.0084	0.0056	0.0037	0.0076	0.0041
RacketSports	0.0000	0.0142	0.0082	0.0135	0.0082	0.0054	0.0279
SelfRegulationSCP1	0.0016	0.0281	0.0174	0.0121	0.0196	0.0084	0.0113
SelfRegulationSCP2	0.0045	0.0302	0.0276	0.0131	0.0139	0.0396	0.0277
SpokenArabicDigits	0.0484	0.0017	0.0028	0.0013	0.0045	0.0014	0.0018
StandWalkJump	0.0314	0.0314	0.0831	0.0314	0.0000	0.0629	0.0629
UWaveGestureLibrary	0.0015	0.0273	0.0128	0.0273	0.0159	0.0596	0.0092
Mean Std.	0.0086	0.0192	0.0261	0.0366	0.0229	0.0244	0.0148