TimeSliver : Symbolic-Linear Decomposition for Explainable Time Series Classification

Anonymous authors
Paper under double-blind review

Abstract

Identifying the extent to which every temporal segment influences a model's predictions is essential for explaining model decisions and increasing transparency. While post-hoc explainable methods based on gradients and feature-based attributions have been popular, they suffer from reference state sensitivity and struggle to generalize across time-series datasets, as they treat time points independently and ignore sequential dependencies. Another perspective on explainable time-series classification is through interpretable components of the model, for instance, leveraging self-attention mechanisms to estimate temporal attribution; however, recent findings indicate that these attention weights often fail to provide faithful measures of temporal importance. In this work, we advance this perspective and present a novel explainability-driven deep learning framework, `TimeSliver`, which jointly utilizes raw time-series data and its symbolic abstraction to construct a representation that maintains the original temporal structure. Each element in this representation linearly encodes the contribution of each temporal segment to the final prediction, allowing us to assign a meaningful importance score to every time point. For time-series classification, `TimeSliver` outperforms other temporal attribution methods by 11% on 7 distinct synthetic and real-world multivariate time-series datasets. `TimeSliver` also achieves predictive performance within 2% of state-of-the-art baselines across 26 UEA benchmark datasets, positioning it as a strong and explainable framework for general time-series classification.

1 Introduction

Deep-learning (DL) models such as Convolutional Neural Network (CNN), Long Short-Term Memory (LSTM), and Transformer have proven to be successful as predictive models for time series classification tasks. However, while most DL models offer strong predictive performance, they are not interpretable, limiting our understanding of their decision-making process (Rudin, 2019; Doshi-Velez & Kim, 2017). Interpretable DL models are essential for trust and transparency, particularly in high-stakes domains such as healthcare, law, and finance, where explanations support informed decision-making and regulatory compliance (Rudin, 2019). They also help detect biases in training data, ensure fairer outcomes (Caruana et al., 2015; Molnar, 2020), and facilitate the extraction of new scientific knowledge (Pandey et al., 2025).

Over the past few years, several methods have been developed to explain the decisions of DL models. Popular methods like DeepLift and Integrated Gradients attribute predictions via baseline-based backpropagation or path integrals but require careful baseline selection (Shrikumar et al., 2017; Sundararajan et al., 2017). Another method called Grad-CAM, a purely gradient-based approach, attributes importance via output–feature derivatives, but is tailored for CNNs and performs poorly on temporal attribution tasks (Selvaraju et al., 2017; Saha et al., 2024). SHAP-based approaches leverage game-theory-based Shapley scores to provide consistent explanations via a unified framework, but they assume feature independence and scale poorly with dimensionality (Lundberg & Lee, 2017). All these **post-hoc interpretability methods** face shared challenges of high parametric sensitivity and explanations that often vary significantly across datasets (Turbé et al., 2023).

Another set of approaches advocates **explainability based on the model's inherent components**. For instance, some works leverage self-attention weights (Wu et al., 2020; Clark et al., 2019; Rogers et al., 2021; Vig et al., 2021) in Transformers (Vaswani et al., 2017) as a key tool for model explainability. Grad-SAM (Barkan et al., 2021) enhances this by weighting attention scores with their output gradients. However, due to the non-linearities in Transformers, attention weights often fail to align (unfaithful) with ground-truth attribution (Chefer et al., 2021; Serrano & Smith, 2019; Jain & Wallace, 2019; Wiegreffe & Pinter, 2019). Another instance is a recent Multiple Instance Learning (MIL)-based temporal attribution method (Early et al., 2024), which shows promising results in identifying the importance of each time point. However, it has not been extended to multivariate time-series settings and has limited experimental comparisons only to Grad-CAM and DeepLiftSHAP. Some more recent approaches use self-supervised model behavior consistency (Queen et al., 2023) or the modified information bottleneck (Liu et al., 2024) to compute attribution scores, but they either depend on a pretrained model (Queen et al., 2023) or are computationally complex due to multiple components and hyperparameters (Liu et al., 2024). In protein modeling, COLOR (Pandey et al., 2025) enhances explainability by segmenting protein sequences into motifs for representation learning. However, it cannot differentiate between positively and negatively attributing segments, and protein sequences are inherently univariate and composed of categorical variables, unlike multivariate continuous time series data. These limitations *(non-linearities leading to unfaithful attributions, inapplicability to multivariate time series, sensitivity to hyperparameters)* of prior methods motivate us to explore an **explainability-driven predictive modeling approach capable of handling multivariate time series** with robust attribution capabilities across domains.

In this work, we introduce `TimeSliver`, a novel deep learning model that computes Temporal attribution using Symbolic–Linear Vector Encoding for Representation. `TimeSliver` processes raw (uni- or multi-variate) time series and their symbolic counterparts (via binning) to produce localized, segment-level representations. These representations are then linearly combined into a sequence-length-independent, explainable representation that enables the computation of temporal attribution scores and facilitates insight into the model's predictions. Our main contributions are as follows:

- We propose an **explainability-driven deep learning framework**, `TimeSliver`, which learns compact representations through a novel linear composition of symbolic and latent representations of temporal segments to provide temporal importance for multivariate Time Series Classification (TSC) tasks while maintaining state-of-the-art predictive capacity (Section 2.2.3).
- `TimeSliver` provides **positive and negative temporal attribution scores** to offer a complete explanation of different time points for the model's prediction (Section 2.2.4).
- We evaluate `TimeSliver`'s explainability across **three diverse real-world applications**—audio, sleep-stage classification, and machine fault diagnosis—**as well as on four synthetic datasets, against nine baseline methods**, which place `TimeSliver` consistently as a **top-performing model for identifying positively and negatively influencing temporal segments** under various settings (Section 3.1).
- We demonstrate `TimeSliver`'s **competitive predictive performance on 26 multivariate time-series classification tasks** from the UEA benchmark (Section 3.2).

Additional Related Works. Decomposing time-series inputs into *human-understandable patterns* also contributes to explainability. Recent works explore approaches such as shapelet decomposition (Wen et al., 2025b), reinforcement learning-based subsequence selection (Gao et al., 2022a), and abstracted shape representations (Wen et al., 2024). In particular, learnable shapelet-based methods (Wen et al., 2025b; Li et al., 2021a; Qu et al., 2024a; Ma et al., 2020) for encoding subsequences are popular pattern-based explainable models. These methods are generally better suited for qualitative assessment and exhibit varied performance metrics, making them challenging to benchmark (Wen et al., 2024; 2025b). Another class of *self-explainable* methods uses neuro-symbolic approaches (Yan et al., 2022) with signal temporal logic (Mehdipour et al., 2020) to output soft-logic predicates at each time step. Architecturally, explainability can also be incorporated through concept bottleneck networks (CBMs) (Koh et al., 2020), which introduce *human-understandable concepts* as intermediate predictions. However, CBMs typically require dense concept annotations and manual editing, practices often impractical in high-stakes applications. Some recent works address this by proposing data-efficient CBMs (Koh et al., 2020) and exploring their applicability in time-series settings (van Sprang et al., 2024; Wen et al., 2025b).

2 METHODOLOGY

2.1 PRELIMINARIES AND NOTATIONS

Although DL models such as 1D CNNs, LSTMs, and Transformers have proven effective for time-series prediction, they often lack explainability, particularly in terms of *temporal attribution*.

Definition 2.1 (Temporal Attribution-Based Explainability)**.** Temporal attribution-based explainability in time series refers to a model's ability to assign importance score to each time point in an input sequence, thereby identifying which time steps most significantly influence the model's prediction.

Problem Statement. Given a dataset $\mathcal{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^{N}$ with N samples, where $\mathbf{x}_i \in \mathbb{R}^{L \times v}$ is a multivariate time-series input of length L with v features (or number of input channels), and $y_i \in \{0, \ldots, C-1\}$ is the corresponding class label, we aim to learn an explainable model f comprising two components:

$$\hat{y}_i = f_{\text{cls}}(\mathbf{x}_i) \in \{0, \ldots, C-1\}, \quad \boldsymbol{\alpha}_i = f_{\text{att}}\big(f_{cls}(\mathbf{x}_i)\big) \in \mathbb{R}^{L},$$

where $\hat{y}$ corresponds to the predictive class and $\alpha_i^{(t)}$ denotes the importance of the t-th time step in $\mathbf{x}_i$ for the model's prediction.

2.2 OUR APPROACH

In this section, we present our explainability-driven novel deep learning model, `TimeSliver`, illustrated in Figure 1 which comprises of three key modules:**(I)** conversion of the raw time-series input x_i into temporal segments and learning their representations $\boldsymbol{Q}$, **(II)** construction of a latent temporal vector $\boldsymbol{Z}$ using symbolic abstraction of x_i, and **(III)** a linear composition of $\boldsymbol{Q}$ and $\boldsymbol{Z}$ to yield a representation of x_i that maintains initial temporal structure. This combined representation is fed to a *linear layer* to predict the target label y_i.

Definition 2.2 (Temporal Segment)**.** Given a multivariate time series instance $\mathbf{x}_i \in \mathbb{R}^{L \times v}$, a *temporal segment* is defined as a contiguous sub-sequence of $\mathbf{x}_i$ of length m (with $m \leq L$). Formally, a segment is $\mathbf{x}_s = \mathbf{x}_i[t : t+m] \in \mathbb{R}^{m \times v}$, where t is its start index in x_i

2.2.1 MODULE I: LATENT REPRESENTATION OF TEMPORAL SEGMENTS

Given a multivariate time-series input $x_i \in \mathbb{R}^{L \times v}$, this module partitions x_i into $\kappa = L - m + 1$ overlapping temporal segments of size m using a 1D convolutional operator with kernel size m and stride 1. Each segment captures a localized temporal context within the time series. The 1D CNN is parameterized by learnable weights $\boldsymbol{\theta}$ and transforms each segment into a q-dimensional latent representation, resulting in a matrix $\boldsymbol{Q} \in \mathbb{R}^{\kappa \times q}$. Formally, this is defined by a learnable mapping, $g_{\boldsymbol{\theta}} : x_i \mapsto \boldsymbol{Q} = [\mathbf{q}_1; \mathbf{q}_2; \ldots; \mathbf{q}_\kappa]^T$, where $\mathbf{q}_j \in \mathbb{R}^q$ is the latent vector for the j^{th} segment, enabling end-to-end learning of localized temporal patterns.

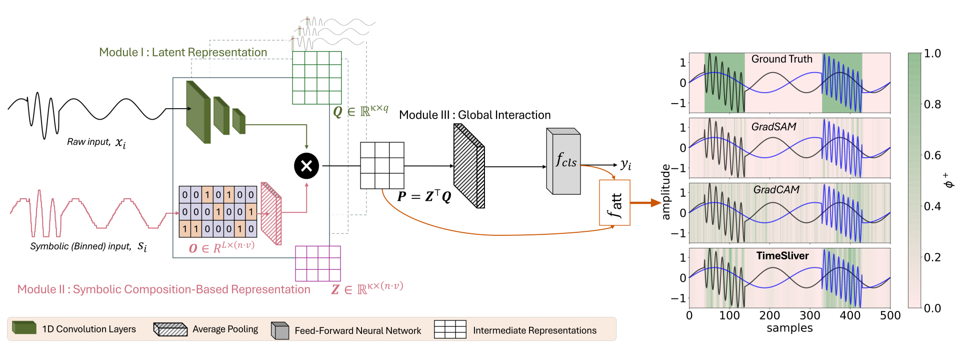

Figure 1: Overview of `TimeSliver`: (Module I) temporal segment extraction and latent representation learning; (Module II) symbolic composition of temporal segments; and (Module III) global **linear** interaction between latent and symbolic representations to generate $\boldsymbol{P}$, a representation of x_i preserving temporal structure. $\boldsymbol{P}$ is then passed through a **linear layer** to predict y_i and used to compute temporal attribution. The right column compares ground truth attribution scores with baseline methods and `TimeSliver`, where darker regions indicate positive influence.

2.2.2 Module II: Symbolic Composition-Based Representation

In this module, each variate $\mathbf{x}_i^{(j)} \in \mathbb{R}^L$, for $j \in \{0, \ldots, v-1\}$, is independently discretized into one of n categorical bins using a fixed binning strategy, as proposed by Lin et al. (2007). This yields a symbolic matrix $s_i \in \{1, \ldots, n\}^{L \times v}$, where each element $s_i^{(t,j)}$ indicates the symbolic bin index assigned to the j^{th} variate at time step t. The symbolic representation is formally defined as $s_i = h(x_i; n, w)$, where $h(\cdot)$ is a deterministic discretization function parameterized by the number of bins n and the compression window size w. In this work, we choose $w = 1$. Next, we convert s_i into a one-hot encoded matrix $\boldsymbol{\mathcal{O}} \in \mathbb{R}^{L \times (n \cdot v)}$ by independently applying one-hot encoding to each variate and concatenating the results along the feature dimension. Specifically, for each variate $j \in \{0, \ldots, v-1\}$, we construct a one-hot matrix $\boldsymbol{\mathcal{O}}^{(j)} \in \{0,1\}^{L \times n}$, where the t^{th} row $\boldsymbol{\mathcal{O}}_t^{(j)}$ corresponds to the one-hot encoding of the symbolic value $s_i^{(t,j)}$. The final matrix is formed as:

$$\boldsymbol{\mathcal{O}} = \left[\boldsymbol{\mathcal{O}}^{(1)} \parallel \boldsymbol{\mathcal{O}}^{(2)} \parallel \cdots \parallel \boldsymbol{\mathcal{O}}^{(v)} \right] \in \mathbb{R}^{L \times (n \cdot v)},$$

where $\parallel$ denotes concatenation along the column (feature) axis. In alignment with past works (Esmael et al., 2012; Combettes et al., 2024) noting that using a shared symbolic embedding space across variates can lead to semantic ambiguity and information loss, this structured symbolic encoding ensures that each variate-specific semantic identity is retained.

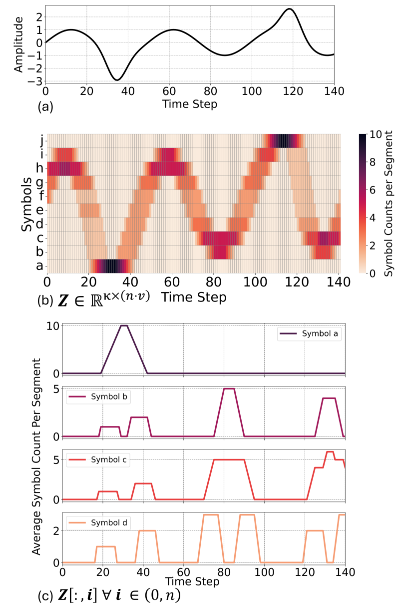

Figure 2: (a) shows a raw time series input, (b) is the symbolic composition matrix, $\boldsymbol{Z}$ and (c) shows some sample rows of $\boldsymbol{Z}$ which serve as the **Bag-of-Stencils** to modulate $\boldsymbol{P}$.

To obtain a segment-wise symbolic representation aligned with the temporal segments extracted in Section 2.2.1, we apply average pooling over the one-hot encoded matrix $\boldsymbol{\mathcal{O}} \in \mathbb{R}^{L \times (n \cdot v)}$ using a sliding window of size m and stride 1. This yields a symbolic composition matrix $\boldsymbol{Z} \in \mathbb{R}^{\kappa \times (n \cdot v)}$ as shown in Figure 2b, where each entry Z_{ij} captures the normalized frequency of the j^{th} symbolic feature within the i^{th} segment as shown in Figure 2c. Formally, this is computed as:

$$Z_{ij} = \frac{1}{m} \sum_{l=0}^{m-1} \mathcal{O}_{i+l,\, j}, \quad \text{for } i \in \{0, \ldots, \kappa - 1\},\ j \in \{0, \ldots, n \cdot v - 1\}, \tag{1}$$

where $\mathcal{O}_{t,j}$ denotes the j^{th} one-hot dimension at time step t. Thus, each row $\boldsymbol{Z}_{i:}$ represents the symbolic distribution over the i^{th} temporal segment.

Remark (Approximate Structural Analogy of Z with Spectral Representation). The Short-Time Fourier Transform (STFT) (Oppenheim et al., 1999) for a segment i (of length m) for an input x computes the energy at frequency f as,

$$S_{if} = \frac{1}{m} \left| \sum_{l=0}^{m-1} x[i+l]\, e^{-j2\pi fl/m} \right|^2$$

providing a localized decomposition of x onto the orthonormal sinusoidal basis $\{e^{-j2\pi fl/m}\}_{f=0}^{n-1}$, with S_{if} encoding the segment-level power for each frequency bin f. Analogously, from Equation 1, Z_{ij} represents the average count of symbolic pattern j within segment i, derived from the columns of $\mathcal{O}$, which are orthogonal, approximately paralleling S_{if} as a segment-level "power" measure. This structural analogy illustrates the similarity between the symbolic-linear composition matrix and spectrogram representations, both encoding the presence and intensity of discrete components—symbolic patterns and spectral frequencies, respectively—across temporal segments. While this architectural correspondence offers valuable intuition, it is important to note that the underlying mathematical principles of these methodologies are fundamentally distinct.

Generality of $\boldsymbol{Z}$ with other discretizers, $h(\cdot)$. In this case, we have chosen $h(\cdot)$ based on Lin et al. (2007). We explore other strategies such as Adaptive Brownian Bridge-based Approximation (ABBA) (Elsworth & Güttel, 2020) and Symbolic Fourier Approximation (SFA) (Schäfer & Högqvist, 2012) to construct the categorical representation and follow the same operations to obtain $\boldsymbol{Z}$. On a synthetic dataset, our explainability scores are almost similar across different discretizers (0.94 AUPRC score, approximately 10% better than the next best model; see Table 1 in Section 3.1). More datasets, evaluation metrics, and further results are given in Sections 3.1 and D.2. This highlights the generality of constructing the composition matrix Z via symbolic representation $\mathcal{O}$ for better explainability. In the next section, we show how this matrix enables the construction of a global interaction-based representation that enhances temporal attribution scores.

2.2.3 Module III: Global Interaction of Temporal Segments

Modules I and II capture local temporal patterns through segmentation, producing latent segment embeddings $\boldsymbol{Q} \in \mathbb{R}^{\kappa \times q}$ and symbolic composition vectors $\boldsymbol{Z} \in \mathbb{R}^{\kappa \times (n \cdot v)}$, respectively. Now to predict the target label y_i, it is important to consider possible interactions among different segments, and those interactions can be captured by constructing a cross-representation matrix $\boldsymbol{P} = \boldsymbol{Z}^\top \boldsymbol{Q}$, where $\boldsymbol{P} \in \mathbb{R}^{(n \cdot v) \times q}$. $\boldsymbol{P}$ aggregates the linear relationships between symbolic and latent segment features, and its size is independent of the sequence length L; thereby assisting in making models with fewer trainable parameters (model details are demonstrated in Appendix B).

Each element P_{ij} of the matrix can be expressed as:

$$\mathrm{P}_{ij} = \sum_{k=1}^{\kappa} Z_{ki} \cdot Q_{kj}, \tag{2}$$

where Z_{ki} is the i^{th} symbolic feature of the k^{th} segment and Q_{kj} is the j^{th} latent feature of the same segment. Thus, each entry in $\boldsymbol{P}$ represents a linear weighted contribution from all temporal segments, providing a global weighted summary of the time-series input.

Supporting Multiple Segment Sizes. The 2D representation $\boldsymbol{P} \in \mathbb{R}^{(n \cdot v) \times q}$, derived previously, corresponds to a fixed segment size m. However, a single segment size may not capture the diverse temporal patterns needed to accurately predict the output label y_i. To address this, we extend the computation of $\boldsymbol{P}$ to multiple segment sizes $\{m_1, m_2, \ldots, m_{|m|}\}$, and stack them along a new axis to obtain a 3D tensor:$\boldsymbol{\mathcal{R}} \in \mathbb{R}^{(n \cdot v) \times q \times |m|}$, where each slice $\boldsymbol{P}^{(m_\ell)} \in \mathbb{R}^{(n \cdot v) \times q}$ is computed as described in Section 2.2.3, and $|m|$ denotes the number of distinct segment sizes. Although $\boldsymbol{\mathcal{R}}$ is a 3D tensor, its dimensions do not reflect a spatial topology; hence, we do not apply any convolutional operations across this representation.

Intuition of constructing $\boldsymbol{P}$. Prior works such as SAX-VSM (Senin & Malinchik, 2013) demonstrate the effectiveness of representing each time-series sample as an unordered set (*Bag-of-Words*) of symbolic patterns and leveraging discriminative statistics of symbol occurrences, which enhance predictive performance. Motivated by this, our approach, in contrast, utilizes the collection of segment-wise symbolic occurrences across an input sample—i.e., the rows of $\boldsymbol{Z}$—as *a Bag-of-Stencils* (as shown in Figure 2(c)). Through the linear aggregation in Equation 2, each entry of $\boldsymbol{P}$ aggregates segments weighted by symbolic pattern occurrence (a stencil), which masks the segments where a symbol is absent and enhances the ones where it occurs more frequently, thereby modulating the corresponding latent features from $\boldsymbol{Q}$. This formulation enables $\boldsymbol{P}$ to capture relevant global discriminative interactions across the sequence linearly. This serves as the foundation for the temporal attribution scoring detailed in the following section while maintaining predictive performance.

2.2.4 Calculating Temporal Attribution

Once the model is trained to predict y, the elements of $\boldsymbol{P}$, which capture the global discriminatory features through linear operations on the learned $\boldsymbol{Q}$ (Eq. 2), enable the computation of temporal attributions. To explain the temporal attribution, we consider the case where $|m| = 1$, such that $\boldsymbol{\mathcal{R}} \equiv \boldsymbol{P}$, although it can be extended to $|m| > 1$ without loss of generality.

Let $\hat{y}^p_c$ denote the logit output corresponding to the class label of the input x_c. We first compute the influence of the element $P_{ij} \in \boldsymbol{P}$ on $\hat{y}^p_c$ as $g_{ij} = \frac{\partial \hat{y}^p_c}{\partial P_{ij}}$. We then define the gradient directionality of g_{ij} by $\sigma_{ij} = \texttt{sign}(g_{ij})$, which indicates whether perturbations in P_{ij} are expected to increase ($\sigma_{ij} = +1$) or decrease ($\sigma_{ij} = -1$) the logit. Based on Equation 2, each P_{ij} can be decomposed into κ components corresponding to κ temporal segments. Therefore, we estimate the normalized positive and negative contributions of the k^{th} ($k \in [0, \kappa - 1]$) segment for a given g_{ij} and σ_{ij} as:

$$\zeta^+_{k,ij} = |g_{ij}| \times \frac{\texttt{ReLU}\ \left(\sigma_{ij} Z_{ki} Q_{kj}\right)}{\max\limits_{l} \texttt{ReLU}\ \left(\sigma_{ij} Z_{li} Q_{lj}\right)} \quad \text{and} \quad \zeta^-_{k,ij}(g_{ij}, \sigma_{ij}) = |g_{ij}| \times \frac{\texttt{ReLU}\ \left(-\sigma_{ij} Z_{ki} Q_{kj}\right)}{\max\limits_{l} \texttt{ReLU}\ \left(-\sigma_{ij} Z_{li} Q_{lj}\right)} \tag{3}$$

While determining the contributions, it is crucial that the $Z_{ki}Q_{kj}$ terms in Equation 3 remain agnostic to the absolute scale of the terms; otherwise, this can lead to spurious attributions caused by high-magnitude but semantically irrelevant input segments. Our construction of the $\boldsymbol{Z}$ matrix, composed of the frequency of symbolic component occurrences within a segment, helps in determining a scale-invariant attribution score. A more formal representation of this property is provided in Section A of the Appendix. This design choice is motivated by our observation in experiments, where we incorporate the raw input directly, $\mathbf{x}_i \in \boldsymbol{X}$, by projecting it to the same dimension as the $\boldsymbol{Z}$ matrix, and observing an average drop of 17% across four synthetic datasets as shown in Figure 3 (details of the datasets and the AUPRC metric are provided in Section 3).

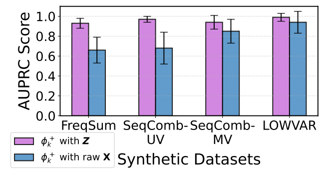

Figure 3: Explainability performance when using raw $\boldsymbol{X}$ instead of $\boldsymbol{Z}$ for computing $\boldsymbol{P}$ and thereby ϕ_k^+.

Since, $\boldsymbol{P} \in \mathbb{R}^{(n \cdot v) \times q}$ is a 2D matrix, the final positive (ϕ_k^+) and negative (ϕ_k^-) attribution score of the k^{th} temporal segment in x_c is calculated as:

$$\phi_k^+ = \sum_{i=0}^{n.v-1} \sum_{j=0}^{q-1} \zeta_{k,ij}^+ \quad \text{and} \quad \phi_k^- = \sum_{i=0}^{n.v-1} \sum_{j=0}^{q-1} \zeta_{k,ij}^- \tag{4}$$

2.2.5 Metrics to Evaluate Temporal Attribution

Evaluating on synthetic dataset. The salient time points in synthetic datasets are known (Queen et al., 2023; Liu et al., 2024) and represented by a binary vector $G \in \{0,1\}^{L \times 1}$. Temporal attribution scores (ϕ^+) are softmax-normalized into probabilities and evaluated against G using the area under the precision-recall curve (AUPRC), where higher values indicate a better method.

Evaluating on real-world datasets. To quantitatively evaluate temporal attribution scores and compare them against contemporary methods, we adapt masking-based evaluation techniques from prior work in time series (Queen et al., 2023), and protein (Pandey et al., 2025). To evaluate positive attributions, time points are first ranked based on ϕ^+. All the time points except the top $u\%$ are then masked (x_i^t=0 if masked) in both the training and test sets. The model is re-trained and evaluated using only the unmasked time points. Given that all datasets are class-balanced (see Appendix C for details), we use accuracy as the evaluation metric. Training quality with partial unmasking is sensitive to the masking method (zeroing or imputation) (Hooker et al., 2019). To ensure a fair comparison across explainable methods, re-training is performed on four different architectures, and the mean accuracies are reported. The value of u is incrementally increased, and with each step, the model is re-trained and the accuracy $e(u)$ on the test data is recorded. The area under the $e(u)$ versus u curve, $\mathcal{I}$, calculated as:

$$\mathcal{I}(\mathtt{U}) = \int_0^{\mathtt{U}} e(u)du, \tag{5}$$

is used to quantitatively compare different interpretable models. We use $\mathcal{I}(\mathtt{100})$ and $\mathcal{I}(\mathtt{20})$ for the comparison in our experiments. The former captures the entire area under the curve, reflecting overall explainability, while the latter emphasizes the model's effectiveness in identifying the most critical time points. The higher these values, the more interpretable the method. Unmasking time points from best to worst is more appropriate for time series data, as the discriminative information in time series is often distributed across many timesteps (Queen et al., 2023).

To assess negative attributions, we mask the top 2% and 5% of time points with the largest ϕ^-. The model is then re-trained and evaluated using the same protocol as for positive attributions, with accuracy normalized by $e(u = 100)$. Normalized accuracy near 1 indicates minimal impact, while values $>$1 suggest that removing noisy time points improves performance; both trends are observed in our results.

3 Experimental Results

We evaluate `TimeSliver` against nine temporal attribution methods across 7 datasets, using the explainability metrics from Section 2.2.5. We also report its accuracy on the UEA benchmark to demonstrate predictive performance.

Datasets. We use *four synthetic datasets* from Turbé et al. (2023) and Queen et al. (2023): FreqSum, SeqComb-UV, SeqComb-MV, and LowVar which capture a wide variety of temporal dynamics within univariate and multivariate settings (more details in Section B.1 of the Appendix). We leverage three

Table 1: Comparison of mean$_{\pm\text{std}}$ AUPRC on synthetic datasets. **Bold**: best, <u>underlined</u>: second-best.

Method	FreqSum	SeqComb-UV	SeqComb-MV	LOWVAR
Random	$0.35_{\pm0.06}$	$0.23_{\pm0.04}$	$0.22_{\pm0.04}$	$0.08_{\pm0.03}$
Grad-CAM	$0.64_{\pm0.09}$	$0.61_{\pm0.02}$	$0.61_{\pm0.02}$	$0.55_{\pm0.01}$
Integrated Gradient	$0.59_{\pm0.10}$	$0.36_{\pm0.16}$	$0.36_{\pm0.13}$	$0.73_{\pm0.34}$
GradientSHAP	$0.54_{\pm0.09}$	$0.57_{\pm0.09}$	$0.39_{\pm0.16}$	$0.50_{\pm0.20}$
DeepLift	$0.61_{\pm0.08}$	$0.61_{\pm0.03}$	$0.57_{\pm0.10}$	$0.54_{\pm0.06}$
DeepLiftShap	$0.61_{\pm0.08}$	$0.61_{\pm0.04}$	$0.58_{\pm0.09}$	$0.54_{\pm0.05}$
Attention Tracing	$0.35_{\pm0.06}$	$0.24_{\pm0.06}$	$0.23_{\pm0.05}$	$0.08_{\pm0.03}$
Grad-SAM	$\underline{0.67}_{\pm0.03}$	$0.61_{\pm0.02}$	$0.61_{\pm0.02}$	$0.54_{\pm0.01}$
COLOR	$0.53_{\pm0.13}$	$\underline{0.90}_{\pm0.05}$	$0.72_{\pm0.13}$	$\underline{0.96}_{\pm0.09}$
TimeX++	$0.59_{\pm0.01}$	$0.85_{\pm0.02}$	$\underline{0.76}_{\pm0.01}$	$0.95_{\pm0.01}$
TimeSliver	$\mathbf{0.94}_{\pm0.05}$	$\mathbf{0.97}_{\pm0.03}$	$\mathbf{0.94}_{\pm0.01}$	$\mathbf{0.99}_{\pm0.04}$

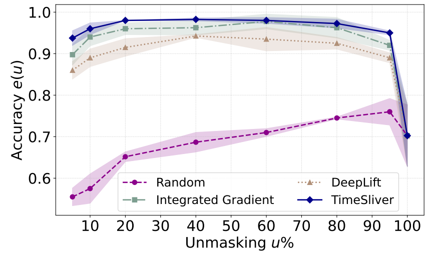

Figure 4: Positive attribution study. Accuracy curves $e(u)$ plotted against the unmasking percentage $u\%$ for EEG dataset.

real-world TSC applications: (1) single-channel electroencephalogram (EEG) data for sleep stage classification from 20 healthy individuals, with a sequence length of 3000; (2) the FordA machine fault diagnosis dataset for binary classification (Bagnall et al., 2018), with a sequence length of 500; and (3) an animal sound classification dataset from the Environmental Sound Classification corpus (ESC-50) (Piczak), consisting of 5-second audio clips. The audio data is processed using mel-frequency spectral representation, following standard practice (Piczak). More details about datasets are provided in Appendix B.

Baselines. `TimeSliver` is compared with several gradient- and sampling-based post-hoc methods originally developed for computer vision—Grad-CAM (Selvaraju et al., 2017), DeepLIFT (Shrikumar et al., 2017), Integrated Gradients (Sundararajan et al., 2017), GradientSHAP (Lundberg & Lee, 2017), and DeepLiftSHAP (Lundberg & Lee, 2017)—as well as TimeX++ (Liu et al., 2024), an information-bottleneck-based explainable approach. We also include self-attention-based explainable models—Attention Tracing (Wu et al., 2020), which estimates temporal importance from Transformer attention weights, and Grad-SAM (Barkan et al., 2021), adapted from the language domain—as well as another explainable model from the protein domain, COLOR (Pandey et al., 2025). A `Random` baseline, assigning uniform attribution scores across time points, is also included.

Explainability Study. Each dataset has three distinct splits (80% train, 10% valid and 10% test), with three trials per split. For each split, we first train the predictive model using four different backbones: 1D CNN, Transformer, COLOR, and `TimeSliver` (details in Appendix B). `TimeSliver` achieves predictive performance within 3–4% of the other backbones, indicating that all models are trained comparably well and enabling a fair comparison. Subsequently, Attention Tracing and Grad-SAM are implemented on the Transformer backbone, while all other explainable methods, except COLOR, are applied to the CNN backbone. We evaluate the temporal attribution scores computed using different explainable methods on all four backbones using the metrics discussed in Section 2.2.5. The results are then averaged across all backbones for each explainable method.

3.1 Improvement of TimeSliver over Baselines on Explainability (Temporal Attributions)

Table 1 reports the AUPRC values computed as described in Section 2.2.5 for various explainable methods for all four synthetic datasets. **`TimeSliver` achieves an average of 18% improvement**

Table 2: Positive attribution results, with the mean$_{\pm\text{std}}$ $\mathcal{I}(100)$ and $\mathcal{I}(20)$ values. **Bold**: best, <u>underlined</u>: second-best. ↑ denotes higher is better.

Method	Audio		EEG		FORD-A	
	$\mathcal{I}(100)\uparrow$	$\mathcal{I}(20)\uparrow$	$\mathcal{I}(100)\uparrow$	$\mathcal{I}(20)\uparrow$	$\mathcal{I}(100)\uparrow$	$\mathcal{I}(20)\uparrow$
Random	$67.90_{\pm0.30}$	$9.06_{\pm0.02}$	$62.66_{\pm1.85}$	$9.33_{\pm0.38}$	$73.89_{\pm1.96}$	$8.89_{\pm0.25}$
Grad-CAM	$69.79_{\pm0.38}$	$10.05_{\pm0.07}$	$67.23_{\pm0.96}$	$10.70_{\pm0.33}$	$81.43_{\pm0.04}$	$11.07_{\pm0.04}$
Integrated Gradient	$63.69_{\pm0.04}$	$8.34_{\pm0.46}$	$\underline{83.19}_{\pm0.89}$	$\underline{14.24}_{\pm0.29}$	$\underline{93.65}_{\pm0.19}$	$\underline{14.76}_{\pm0.14}$
GradSHAP	$69.53_{\pm0.31}$	$10.48_{\pm0.17}$	$63.76_{\pm2.27}$	$10.41_{\pm0.26}$	$78.54_{\pm0.64}$	$11.44_{\pm0.03}$
DeepLift	$63.35_{\pm0.39}$	$8.15_{\pm0.11}$	$80.43_{\pm0.70}$	$13.63_{\pm0.32}$	$93.11_{\pm0.08}$	$14.41_{\pm0.07}$
DeepLiftShap	$70.55_{\pm0.40}$	$10.70_{\pm0.08}$	$65.34_{\pm1.78}$	$10.66_{\pm0.34}$	$85.30_{\pm0.27}$	$13.30_{\pm0.07}$
Attention Tracing	$69.15_{\pm0.49}$	$9.75_{\pm0.42}$	$63.16_{\pm2.42}$	$9.28_{\pm0.30}$	$76.47_{\pm0.36}$	$9.60_{\pm0.42}$
Grad-SAM	$69.00_{\pm0.12}$	$9.89_{\pm0.24}$	$62.11_{\pm2.92}$	$9.38_{\pm0.34}$	$74.17_{\pm0.01}$	$9.17_{\pm0.14}$
COLOR	$71.46_{\pm0.67}$	$10.47_{\pm0.14}$	$66.48_{\pm1.47}$	$10.85_{\pm0.34}$	$83.95_{\pm0.85}$	$12.12_{\pm0.22}$
TimeX++	$\underline{73.24}_{\pm0.35}$	$\underline{11.20}_{\pm0.12}$	$74.10_{\pm0.49}$	$11.84_{\pm0.09}$	$87.85_{\pm0.53}$	$13.83_{\pm0.18}$
TimeSliver	$\mathbf{74.30}_{\pm0.68}$	$\mathbf{11.35}_{\pm0.15}$	$\mathbf{83.99}_{\pm0.61}$	$\mathbf{14.52}_{\pm0.15}$	$\mathbf{93.87}_{\pm0.01}$	$\mathbf{14.99}_{\pm0.01}$

over the leading baseline. To quantitatively evaluate different explainable methods on real-world datasets, we report $\mathcal{I}(100)$ and $\mathcal{I}(20)$ values, as defined in Section 2.2.5, computed across three splits for all three datasets (Table 2 and Appendix D). **`TimeSliver` consistently outperforms baselines by 2% in $\mathcal{I}(20)$, demonstrating superior ability to identify key positive time points.** To further demonstrate the effectiveness of `TimeSliver` in capturing positive critical time steps, we present the $e(u)$ versus $u\%$ curve for the EEG dataset in Figure 4. `TimeSliver` outperforms the strongest baselines, namely Integrated Gradients and DeepLIFT.

Table 3: Negative attribution results showing mean$_{\pm \text{std}}$ accuracy with 2% and 5% of the sequence masked by ϕ^- (normalized by $e(100)$). **Bold**: best; $\underline{\text{underlined}}$: second-best. ↑ indicates higher is better.

Methods	EEG		Audio		FordA	
	2% Masking↑	5% Masking↑	2% Masking↑	5% Masking↑	2% Masking↑	5% Masking↑
Random	$1.06_{\pm 0.03}$	$1.09_{\pm 0.05}$	$0.99_{\pm 0.04}$	$0.96_{\pm 0.01}$	$0.99_{\pm 0.01}$	$1.00_{\pm 0.01}$
Grad-CAM	$1.06_{\pm 0.03}$	$1.07_{\pm 0.05}$	$\underline{1.00}_{\pm 0.05}$	$0.97_{\pm 0.02}$	$1.00_{\pm 0.00}$	$1.01_{\pm 0.00}$
Integrated Gradient	$\underline{1.25}_{\pm 0.03}$	$\underline{1.31}_{\pm 0.02}$	$\mathbf{1.02}_{\pm 0.04}$	$\underline{1.00}_{\pm 0.05}$	$\underline{1.07}_{\pm 0.00}$	$\underline{1.08}_{\pm 0.00}$
GradSHAP	$1.08_{\pm 0.02}$	$1.07_{\pm 0.02}$	$0.98_{\pm 0.01}$	$0.95_{\pm 0.02}$	$0.97_{\pm 0.01}$	$0.97_{\pm 0.01}$
DeepLift	$1.20_{\pm 0.01}$	$1.27_{\pm 0.02}$	$\mathbf{1.02}_{\pm 0.07}$	$\mathbf{1.01}_{\pm 0.01}$	$\underline{1.07}_{\pm 0.00}$	$1.07_{\pm 0.00}$
DeepLiftShap	$1.08_{\pm 0.05}$	$1.07_{\pm 0.03}$	$0.99_{\pm 0.04}$	$0.98_{\pm 0.00}$	$0.98_{\pm 0.01}$	$0.99_{\pm 0.01}$
Attention Tracing	$1.08_{\pm 0.01}$	$1.06_{\pm 0.03}$	$0.98_{\pm 0.04}$	$0.98_{\pm 0.01}$	$0.99_{\pm 0.01}$	$0.98_{\pm 0.01}$
Grad-SAM	$1.10_{\pm 0.03}$	$1.08_{\pm 0.01}$	$0.99_{\pm 0.00}$	$0.97_{\pm 0.01}$	$0.98_{\pm 0.01}$	$0.98_{\pm 0.00}$
COLOR	$1.08_{\pm 0.06}$	$1.05_{\pm 0.07}$	$0.99_{\pm 0.02}$	$0.99_{\pm 0.05}$	$1.03_{\pm 0.01}$	$1.02_{\pm 0.01}$
TimeX++	$1.12_{\pm 0.02}$	$1.11_{\pm 0.01}$	$0.99_{\pm 0.04}$	$0.98_{\pm 0.04}$	$1.01_{\pm 0.01}$	$1.00_{\pm 0.02}$
`TimeSliver`	$\mathbf{1.31}_{\pm 0.00}$	$\mathbf{1.36}_{\pm 0.01}$	$\mathbf{1.02}_{\pm 0.04}$	$\mathbf{1.01}_{\pm 0.01}$	$\mathbf{1.08}_{\pm 0.00}$	$\mathbf{1.09}_{\pm 0.00}$

Interestingly, Figure 4 shows a sharp accuracy drop when all time steps are unmasked ($e(100)$), revealing negatively contributing segments. Table 3 compares `TimeSliver` with baselines on computing negative temporal attributions (Section 2.2.5), with masked accuracies normalized to full-input performance ($e(100)$). On EEG, `TimeSliver` achieves about 33% higher accuracy than the full input, confirming strongly detrimental features and its ability to detect them. The results in Table 3 highlight **`TimeSliver`'s strength in enhancing performance by removing detrimental time points—achieving notable gains on EEG (4% over baselines) and FordA (1% over baselines)**—while maintaining stable accuracy on the audio dataset, suggesting a lower presence of negatively contributing segments.

Table 4: `TimeSliver` vs. 16 baselines on 26 UEA datasets. **Bold**: best, $\underline{\text{Underlined}}$: second-best.

Type	Method	Bio.	Motion	Audio	Coord.	Misc.	All
Distance-Based	DTW_D	46.9	87.5	44.5	95.4	63.9	66.8
	DTW_I	47.9	77.1	34.4	90.7	59.2	61.3
	DTW_A	45.4	88.1	**63.3**	95.4	71.3	69.7
Dictionary-Based	MUSE	55.7	88.3	51.8	96.0	$\underline{81.7}$	72.3
	gRSF	49.0	84.3	46.1	88.3	71.7	63.8
	CIF	54.0	85.4	55.1	96.2	**83.0**	71.5
Feature-Based ML	MrSEQL	52.3	87.8	47.6	94.2	76.6	69.2
	ROCKET	53.8	90.1	48.7	96.6	75.6	70.2
Deep-Learning	TapNet	49.9	84.5	51.5	91.5	65.4	64.8
	ResNet	48.3	90.6	47.5	97.3	57.2	63.3
	IncTime	60.3	**96.4**	$\underline{61.6}$	95.0	69.1	$\underline{74.3}$
	FCN	56.76	90.8	58.4	$\underline{97.8}$	63.4	72.2
	TS2Vec	48.57	87.4	53.2	94.7	65.0	68.0
	TimesNet	61.06	77.9	42.1	91.3	61.9	67.3
	ShapeNet	55.28	86.3	59.3	94.0	53.3	68.2
	RLPAM	66.75	89.5	55.1	90.0	67.4	74.3
	ShapeConv	$\underline{61.24}$	89.0	54.1	95.0	64.8	72.4
	SBM	59.8	86.0	45.8	94.1	66.7	70.5
	InterpGN	58.73	$\underline{91.4}$	52.6	**98.3**	70.7	73.7
Ours	`TimeSliver`	**66.9**	90.9	55.2	93.6	76.1	**75.6**

3.2 Competitive Performance of `TimeSliver` on Multivariate Time Series Classification

We evaluate the predictive performance of `TimeSliver` on 26 datasets from the UEA multivariate time-series classification archive (Ruiz et al., 2021), which span 8 electrical biosignal (Bio.) datasets, 3 audio datasets, 7 accelerometer-based motion datasets, 3 gesture and digit recognition datasets in Cartesian coordinates (Coord.), and other miscellaneous datasets and evaluate `TimeSliver` against five methodological categories: *Distance-based* methods (Bagnall et al., 2016); *Dictionary/interval-based* methods (Schäfer & Leser, 2017); *Feature-based ML* models; *Deep learning* models including ResNet (Wang et al., 2017), InceptionTime (Fawaz et al., 2020), FCN (Karim et al., 2017), TS2vec (Yue et al., 2022), TimesNet (Wu et al., 2022), ShapeNet (Li et al., 2021b), RLPAM (Gao et al., 2022b), ShapeConv (Qu et al., 2024b), SBM (Wen et al., 2025a), InterpGN (Wen et al., 2025a); and *Ensemble-based* methods. More details are provided in Section E of the Appendix. We observe that: (1) `TimeSliver` demonstrates **superior performance on datasets with long sequences** (length > 1000), achieving **an average absolute improvement of 4.4%** over the best-performing baselines (details in Appendix E); (2) it **improves performance on electrical biosignals by 6.3% on average**; and (3) overall, `TimeSliver` delivers **competitive predictive performance, staying within 2% of the best baselines** across diverse applications. To further showcase `TimeSliver`'s strong performance, we report its average rank, top-1, and top-3 counts compared to all baselines in Appendix E.

To empirically demonstrate `TimeSliver`'s ability to capture temporally disjoint yet jointly informative patterns, we construct a synthetic dataset with multiplicative interactions in disjoint segments (Appendix D.3) and evaluate its predictive performance against multiple architectures. As shown in Appendix D.3, `TimeSliver` achieves performance within 1% of the baselines, confirming its effectiveness in modeling temporally disjoint interactions.

3.3 Understanding the components of TimeSliver

We assess the impact of `TimeSliver`'s core design choices through ablation and sensitivity analysis to gain deeper insights.

Ablation Study (Figure 5). The framework of Boureau et al. (2010) shows that average pooling lowers feature map resolution, reducing sensitivity to local perturbations and acting as a regularizer. We verify this by removing average pooling after $\mathcal{R}$ or replacing it with max pooling, which causes a $\sim$5% drop in accuracy with negligible impact on explainability. To assess the role of `ReLU` in Equations 4 in identifying positive and negative attributing time points, we replace `ReLU` with `abs`. This assigns equal importance to time points with equal $\texttt{abs}(Z_{ki}Q_{kj})$ but opposite signs. This modification leads to a 13% drop in explainability, underscoring the importance of `ReLU` in correctly distinguishing positive and negative contributions.

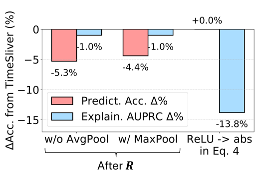

Figure 5: Impact of model variants on prediction (Predict.) and explainability (Explain.).

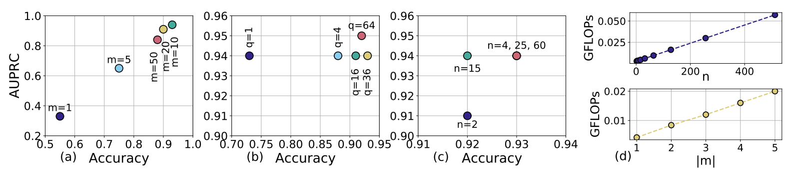

Figure 6: Effect of (a) segment size m, (b) latent dimension q, and (c) number of bins n on predictability (Accuracy) and explainability (AUPRC); (d) GFLOPs variation with n and $|m|$.

Sensitivity Analysis (Figure 6). The final architecture of `TimeSliver` is determined by three key hyperparameters: the segment size m, the latent representation dimension q (both defined in Section 2.2.1), and the number of bins n used to discretize raw inputs x_i into symbolic representations s_i (described in Section 2.2.2). On the FreqSum dataset, Figure 6a shows that explainability declines for $m > 10$, as larger segments can lead the model to over-attribute importance to regions where only a small part is relevant. Conversely, setting $m = 1$ results in poor predictability and explainability, as the segment is too short to capture meaningful temporal patterns. The effect of the latent dimension q and the bin count n is minimal beyond values of 4, with both explainability and predictive accuracy remaining stable (see Figures 6b and 6c). Although this analysis is based on FreqSum data, the relative sensitivity trends are expected to generalize across a wide range of real-world datasets. Figure 6d shows that `TimeSliver`'s GFLOPs only scale linearly with $n \in [2, 500]$ and $|m| \in [1, 5]$, remaining 5–10 times lower than those of Transformers (GFLOP = 0.2), highlighting its efficient scalability.

4 Conclusion

In this work, we presented `TimeSliver`—a novel deep learning framework that linearly combines raw time series with their symbolic counterparts to construct a global representation facilitating temporal attribution calculation. Our importance scores offer insights into positively and negatively influencing time segments. The effectiveness of `TimeSliver` is demonstrated by its average improvement of 11% over the best baselines across seven diverse datasets, spanning real-world and synthetic, univariate and multivariate time series with varied temporal dynamics, while maintaining high predictive performance. In the future, it will be interesting to consider human-in-the-loop expert validation (for tasks like sleep-stage classification using EEG) to harness `TimeSliver`'s explainability for practical applications. Additionally, `TimeSliver`'s principles can be extended to provide feature attribution, identifying which input features are most influential at each time segment, especially by considering a time-frequency representation of time-series data.

Ethics Statement.

This work does not involve human subjects, sensitive data, or issues related to fairness, discrimination, or legal compliance. `TimeSliver` is designed to identify influential temporal segments in time series, providing more transparent and interpretable model predictions. By improving explainability, particularly for applications such as healthcare time-series classification, `TimeSliver` supports responsible and trustworthy deployment of machine learning models.

Reproducibility Statement

All source code to reproduce experimental results (with instructions for running the code) is provided in the Supplementary Materials. We use public datasets and include implementation details in the Appendix. All baselines either adopt published hyperparameters or are tuned when unspecified.

LLM Usage Statement

The usage of LLMs in this work is limited to paper writing support, language refinement, and experimental data processing. Specifically, LLMs assisted in improving the clarity and coherence of the manuscript, generating LaTeX tables, and formatting results for presentation. Importantly, LLMs were not involved in the design of algorithms, the development of theoretical results, or the execution of experiments, ensuring that all core scientific contributions remain entirely the work of the authors.

References

Anthony Bagnall, Jason Lines, Aaron Bostrom, James Large, and Eamonn Keogh. The great time series classification bake off: A review and experimental evaluation of recent algorithmic advances. *Data Mining and Knowledge Discovery*, 31:606–660, 2016.

Anthony Bagnall, Jason Lines, William Vickers, and Eamonn Keogh. Time series classification with ensembles of elastic distance measures. *Data Mining and Knowledge Discovery*, 31:565–592, 2017.

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.

Oren Barkan, Edan Hauon, Avi Caciularu, Ori Katz, Itzik Malkiel, Omri Armstrong, and Noam Koenigstein. Grad-sam: Explaining transformers via gradient self-attention maps. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*, pp. 2882–2887, 2021.

Y-Lan Boureau, Jean Ponce, and Yann LeCun. A theoretical analysis of feature pooling in visual recognition. In *Proceedings of the 27th International Conference on International Conference on Machine Learning*, ICML'10, pp. 111–118, 2010.

Rich Caruana, Yin Lou, Johannes Gehrke, Paul Koch, Marc Sturm, and Noemie Elhadad. Intelligible models for healthcare: Predicting pneumonia risk and hospital 30-day readmission. In *Proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining*, pp. 1721–1730, 2015.

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.

Kevin Clark, Urvashi Khandelwal, Omer Levy, and Christopher D. Manning. What does BERT look at? an analysis of BERT's attention. In Tal Linzen, Grzegorz Chrupała, Yonatan Belinkov, and Dieuwke Hupkes (eds.), *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pp. 276–286, Florence, Italy, August 2019. Association for Computational Linguistics. doi: 10.18653/v1/W19-4828. URL https://aclanthology.org/W19-4828/.

Sylvain W Combettes, Charles Truong, and Laurent Oudre. Symbolic representation for time series. In *2024 32nd European Signal Processing Conference (EUSIPCO)*, pp. 1962–1966. IEEE, 2024.

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*, 34:1454–1495, 2020.

Finale Doshi-Velez and Been Kim. Towards a rigorous science of interpretable machine learning. *arXiv preprint arXiv:1702.08608*, 2017.

Joseph Early, Gavin Cheung, Kurt Cutajar, Hanting Xie, Jas Kandola, and Niall Twomey. Inherently interpretable time series classification via multiple instance learning. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=xriGRsoAza.

Steven Elsworth and Stefan Güttel. Abba: adaptive brownian bridge-based symbolic aggregation of time series. *Data Mining and Knowledge Discovery*, 34(4):1175–1200, 2020.

Bilal Esmael, Arghad Arnaout, Rudolf K Fruhwirth, and Gerhard Thonhauser. Multivariate time series classification by combining trend-based and value-based approximations. In *Computational Science and Its Applications–ICCSA 2012: 12th International Conference, Salvador de Bahia, Brazil, June 18-21, 2012, Proceedings, Part IV 12*, pp. 392–403. Springer, 2012.

Hassan Ismail Fawaz, Benjamin Lucas, Germain Forestier, et al. Inceptiontime: Finding alexnet for time series classification. *Data Mining and Knowledge Discovery*, 34:1936–1962, 2020.

Ge Gao, Qitong Gao, Xi Yang, Miroslav Pajic, and Min Chi. A reinforcement learning-informed pattern mining framework for multivariate time series classification. In *31st International Joint Conference on Artificial Intelligence (IJCAI)*, 2022a.

Ge Gao, Qitong Gao, Xi Yang, Miroslav Pajic, and Min Chi. A reinforcement learning-informed pattern mining framework for multivariate time series classification. In *31st International Joint Conference on Artificial Intelligence (IJCAI)*, 2022b.

Sara Hooker, Dumitru Erhan, Pieter-Jan Kindermans, and Been Kim. A benchmark for interpretability methods in deep neural networks. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper_files/paper/2019/file/fe4b8556000d0f0cae99daa5c5c5a410-Paper.pdf.

Sarthak Jain and Byron C. Wallace. Attention is not Explanation. In Jill Burstein, Christy Doran, and Thamar Solorio (eds.), *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 3543–3556, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1357. URL https://aclanthology.org/N19-1357/.

Fazle Karim, Somshubra Majumdar, Houshang Darabi, and Shun Chen. Lstm fully convolutional networks for time series classification. *IEEE access*, 6:1662–1669, 2017.

Isak Karlsson, Panagiotis Papapetrou, and Henrik Boström. Mr-seql: Multiple representation sequence learner. In *2016 IEEE 16th International Conference on Data Mining Workshops (ICDMW)*, pp. 1–8, 2016.

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*, pp. 5338–5348. PMLR, 2020.

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, 2021a.

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, 2021b.

Jessica Lin, Eamonn Keogh, Li Wei, and Stefano Lonardi. Experiencing sax: a novel symbolic representation of time series. *Data Mining and knowledge discovery*, 15:107–144, 2007.

Jason Lines, Sarah Taylor, and Anthony Bagnall. Time series classification with hive-cote: The hierarchical vote collective of transformation-based ensembles. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 12(5):1–35, 2018.

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. ICML'24. JMLR.org, 2024.

Scott M Lundberg and Su-In Lee. A unified approach to interpreting model predictions. *Advances in neural information processing systems*, 30, 2017.

Qianli Ma, Wanqing Zhuang, Sen Li, Desen Huang, and Garrison Cottrell. Adversarial dynamic shapelet networks. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pp. 5069–5076, 2020.

Noushin Mehdipour, Cristian-Ioan Vasile, and Calin Belta. Specifying user preferences using weighted signal temporal logic. *IEEE Control Systems Letters*, 5(6):2006–2011, 2020.

Matthew Middlehurst, James Large, and Anthony Bagnall. The canonical interval forest (cif) classifier for time series classification. In *2020 IEEE International Conference on Data Mining (ICDM)*, pp. 948–953, 2020.

Christoph Molnar. *Interpretable machine learning*. Lulu. com, 2020.

Alan V. Oppenheim, Ronald W. Schafer, and John R. Buck. *Discrete-Time Signal Processing*. Prentice-hall Englewood Cliffs, 1999.

Akash Pandey, Wei Chen, and Sinan Keten. Color: A compositional linear operation-based representation of protein sequences for identification of monomer contributions to properties. *Journal of Chemical Information and Modeling*, 2025.

Karol J. Piczak. ESC: Dataset for Environmental Sound Classification. In *Proceedings of the 23rd Annual ACM Conference on Multimedia*, pp. 1015–1018. ACM Press. ISBN 978-1-4503-3459-4. doi: 10.1145/2733373.2806390. URL http://dl.acm.org/citation.cfm?doid=2733373.2806390.

Eric Qu, Yansen Wang, Xufang Luo, Wenqiang He, Kan Ren, and Dongsheng Li. Cnn kernels can be the best shapelets. In *The Twelfth International Conference on Learning Representations*, 2024a.

Eric Qu, Yansen Wang, Xufang Luo, Wenqiang He, Kan Ren, and Dongsheng Li. Cnn kernels can be the best shapelets. In *The Twelfth International Conference on Learning Representations*, 2024b.

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.

Anna Rogers, Olga Kovaleva, and Anna Rumshisky. A primer in bertology: What we know about how bert works. *Transactions of the association for computational linguistics*, 8:842–866, 2021.

Cynthia Rudin. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature machine intelligence*, 1(5):206–215, 2019.

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(2):401–449, 2021.

Avinab Saha, Shashank Gupta, Sravan Kumar Ankireddy, Karl Chahine, and Joydeep Ghosh. Exploring explainability in video action recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8176–8181, 2024.

Patrick Schäfer and Mikael Högqvist. Sfa: a symbolic fourier approximation and index for similarity search in high dimensional datasets. In *Proceedings of the 15th international conference on extending database technology*, pp. 516–527, 2012.

Patrick Schäfer and Ulf Leser. Multivariate time series classification with weasel+muse. In *2017 IEEE International Conference on Data Mining (ICDM)*, pp. 447–456, 2017.

Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*, pp. 618–626, 2017.

Pavel Senin and Sergey Malinchik. Sax-vsm: Interpretable time series classification using sax and vector space model. In *2013 IEEE 13th international conference on data mining*, pp. 1175–1180. IEEE, 2013.

Sofia Serrano and Noah A Smith. Is attention interpretable? *arXiv preprint arXiv:1906.03731*, 2019.

Avanti Shrikumar, Peyton Greenside, and Anshul Kundaje. Learning important features through propagating activation differences. In *International conference on machine learning*, pp. 3145–3153. PMlR, 2017.

Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In *International conference on machine learning*, pp. 3319–3328. PMLR, 2017.

Hugues Turbé, Mina Bjelogrlic, Christian Lovis, and Gianmarco Mengaldo. Evaluation of post-hoc interpretability methods in time-series classification. *Nature Machine Intelligence*, 5(3):250–260, 2023.

Angela van Sprang, Erman Acar, and Willem Zuidema. Enforcing interpretability in time series transformers: A concept bottleneck framework. *arXiv preprint arXiv:2410.06070*, 2024.

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

Jesse Vig, Ali Madani, Lav R. Varshney, Caiming Xiong, richard socher, and Nazneen Rajani. {BERT}ology meets biology: Interpreting attention in protein language models. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=YWtLZvLmud7.

Zhiguang Wang, Weizhong Yan, and Tim Oates. Time series classification from scratch with deep neural networks: A strong baseline. In *2017 International Joint Conference on Neural Networks (IJCNN)*, pp. 1578–1585, 2017.

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

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*, 2025a.

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*, 2025b.

Sarah Wiegreffe and Yuval Pinter. Attention is not not explanation. In Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan (eds.), *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pp. 11–20, Hong Kong, China, November 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1002. URL https://aclanthology.org/D19-1002/.

Haixu Wu, Tengge Hu, Yong Liu, Hang Zhou, Jianmin Wang, and Mingsheng Long. Timesnet: Temporal 2d-variation modeling for general time series analysis. *arXiv preprint arXiv:2210.02186*, 2022.

Zhengxuan Wu, Thanh-Son Nguyen, and Desmond C Ong. Structured self-attention weights encode semantics in sentiment analysis. *arXiv preprint arXiv:2010.04922*, 2020.

Ruixuan Yan, Tengfei Ma, Achille Fokoue, Maria Chang, and Agung Julius. Neuro-symbolic models for interpretable time series classification using temporal logic description. In *2022 IEEE International Conference on Data Mining (ICDM)*, pp. 618–627. IEEE, 2022.

Zhihan Yue, Yujing Wang, Juanyong Duan, Tianmeng Yang, Congrui Huang, Yunhai Tong, and Bixiong Xu. Ts2vec: Towards universal representation of time series. In *Proceedings of the AAAI conference on artificial intelligence*, volume 36, pp. 8980–8987, 2022.

Appendix

This appendix provides additional details for "`TimeSliver`: Symbolic-Linear Decomposition for Explainable Time Series Classification". Additional implementation details for `TimeSliver` and the backbone models are presented in Section B. Class distribution for the four datasets used in the interpretability study are provided in Section C. Detailed results on interpretability and predictive performance are given in Sections D and E, respectively.

A Scale-Invariance Properties of TimeSliver

Remark (Determining Scale-Invariant Attributions). Let $\mathbf{P}_{\text{raw}} = \mathbf{X}\mathbf{Q}$ and $\mathbf{P}_{\text{sym}} = \boldsymbol{Z}\mathbf{Q}$, where $\mathbf{Q}$ are learned weights. Suppose $\mathbf{X} = \boldsymbol{Z}\mathbf{D}$ for a diagonal scaling matrix $\mathbf{D}$. Then, for any position t,

$$\|\mathbf{P}_{\text{raw}}^{(t)}\|_2 = d_t \|\mathbf{P}_{\text{sym}}^{(t)}\|_2,$$

where d_t is the t-th diagonal entry of $\mathbf{D}$. Thus, only $\mathbf{P}_{\text{sym}}$ yields attributions invariant to input scaling, and explanations depend solely on the symbolic pattern, not on the magnitude of the input.

Implication. This property prevents spurious attributions caused by high-magnitude but semantically irrelevant input segments, and is essential for robust and interpretable explanations. The effectiveness of this property is further validated by our ablation study, where replacing the one-hot encoding representation $\mathcal{O}$ with raw data x in Equation 1 results in an average 17% decrease in explainability as shown in Figure 3.

B Additional Implementation Details

B.1 Dataset Description

FreqSum is a multivariate time series with randomly embedded sine-wave segments; classes indicate whether the sum of their frequencies exceeds a threshold. As described in (Turbé et al., 2023), each sample in the dataset consists of 6 features and 500 time steps. To simulate realistic temporal dependencies, each feature includes a baseline sine wave with a frequency uniformly sampled from the range $[2, 5]$. Two randomly selected features per sample are injected with discriminative sine waves, each supported over 100 time steps, with frequencies drawn from a discrete uniform distribution in the range $[10, 50]$. In the remaining four features, a square wave is optionally added with 50% probability, also using frequencies sampled from the same range. The classification task is binary: the model must predict whether the sum of the two discriminative frequencies exceeds a predefined threshold, set to $\tau = 60$.

SeqComb-UV, SeqComb-MV, and **LowVar** are generated using the exact technique dsicussed in Queen et al. (2023). **SeqComb-UV** is a univariate series with two non-overlapping increasing or decreasing subsequences, with four classes defined by their trend combinations. **SeqComb-MV** is the multivariate extension of SeqComb-UV. **LowVar** is a multivariate series with four classes determined by the presence of a low-variance subsequence in a specific channel.

Audio Dataset. We use a manually curated subset of the ESC-50 audio dataset, focusing exclusively on animal sounds. This subset was selected to leverage the temporal localization of animal sounds, which typically occur within short bursts in the observation window, as opposed to environmental sounds that span the entire duration and yield robust results even with randomly sampled segments. This temporal sparsity makes animal sounds particularly useful for evaluating interpretability methods that rely on temporal attribution. For preprocessing, we extract Mel-frequency cepstral coefficients (MFCCs) from the audio using a Mel spectrogram with 40 Mel bands, employing standard settings such as centered windowing and normalization.

EEG Dataset. This dataset comprises single-channel EEG recordings collected from 20 subjects, with the objective of classifying five sleep stages: wake, N1, N2, N3 (non-REM stages), and REM (rapid eye movement). The temporal structure of EEG signals makes this dataset well-suited for tasks requiring time-series modeling and interpretation. We balance all the classes in the dataset before using it for the study.

FordA Dataset. We adopt the data preprocessing and train-test splits for the FordA dataset as defined in the MTS-Bakeoff benchmark Ruiz et al. (2021).

B.2 Dataset Details

Information such as the number of variates (v), maximum sequence length, and dataset splits is provided in Table 5.

Table 5: Summary of the four datasets used in the interpretability study.

Dataset	Num. of Variates, v	Max Seq. Length	Train	Valid	Test
FreqSum	6	500	5000	500	500
SeqComb-UV	1	200	5000	1000	1000
SeqComb-MV	4	200	5000	1000	1000
LowVar	2	200	5000	1000	1000
Audio	40	501	280	60	60
EEG	1	3000	5005	1295	3515
Ford-A	1	500	853	106	119

B.3 Model Details

The complete details of `TimeSliver` for all four datasets are given in Table 6. Additionally, the details for the other three backbones used in the interpretability study are given in Table 7.

Table 6: Architecture details of `TimeSliver` used for different datasets.

Dataset	Num. of categorical bins, n	Num. of columns in $\mathcal{O}$, $n \times v$	Latent vector size, q	Segment size, m	Trainable parameters
FreqSum	15	90	36	7	5,858
SeqComb-UV	20	20	36	[4,7]	14,518
SeqComb-MV	10	40	36	[4,7]	20,576
LowVar	20	40	36	4	5,078
Audio	10	400	12	1	20,110
EEG	25	25	12	10	6,441
Ford-A	70	70	36	10	8,280

Table 7: Number of trainable parameters for different model architectures across datasets.

Dataset	CNN	COLOR	Transformer
FreqSum	42,378	2,660	46,714
SeqComb-UV	42,076	16,844	361,156
SeqComb-MV	42,268	21,452	361,540
LowVar	42,140	16,880	361,284
Audio	224,938	8,206	370,498
EEG	74,981	43,309	230,805
Ford-A	42,058	26,536	361,090

B.4 Training and Optimization Details

All experiments are conducted on a server running Ubuntu OS, equipped with NVIDIA RTX A6000 GPUs, using the PyTorch framework. During model training, we employ the Adam optimizer with a learning rate ranging from 3×10^{-4} to 1×10^{-3}. Validation accuracy is used for early stopping and to save the best model checkpoint.

B.5 Predictive Results on different backbone

Table 4 presents the predictive performance of the four deep learning models used as backbones in the interpretability study. The CNN backbone is used for all post-hoc interpretability methods, while the Transformer is employed for attention tracing and the Grad-SAM method. COLOR, originally developed for protein sequence design, is inherently interpretable. The predictive performance of `TimeSliver` on the four datasets used in the interpretability study is within 3–4% of the best-performing model. All the post-hoc methods are implemented using the Captum library [1] in PyTorch.

Table 8: Accuracy (mean$_{\pm\text{std}}$) over 3 runs for different predictive backbone and dataset (supporting results for Section 3 in the main paper).

Dataset	CNN	COLOR	Transformer	`TimeSliver`
FreqSum	$0.93_{\pm0.028}$	$0.93_{\pm0.014}$	$0.95_{\pm0.0071}$	$0.93_{\pm0.014}$
SeqComb-UV	$1.00_{\pm0.00}$	$1.00_{\pm0.00}$	$1.00_{\pm0.00}$	$1.00_{\pm0.00}$
SeqComb-MV	$1.00_{\pm0.00}$	$1.00_{\pm0.00}$	$1.00_{\pm0.00}$	$1.00_{\pm0.00}$
LowVar	$1.00_{\pm0.00}$	$1.00_{\pm0.00}$	$1.00_{\pm0.00}$	$1.00_{\pm0.00}$
Audio	$0.78_{\pm0.0071}$	$0.80_{\pm0.021}$	$0.80_{\pm0.000}$	$0.81_{\pm0.0071}$
EEG	$0.78_{\pm0.019}$	$0.73_{\pm0.039}$	$0.68_{\pm0.038}$	$0.72_{\pm0.042}$
FordA	$0.92_{\pm0.0071}$	$0.93_{\pm0.000}$	$0.83_{\pm0.0071}$	$0.88_{\pm0.000}$

C Class Distribution

The class distribution for all four datasets is shown in Figure 7, indicating that there is no class imbalance in any of the datasets used in the explainability study.

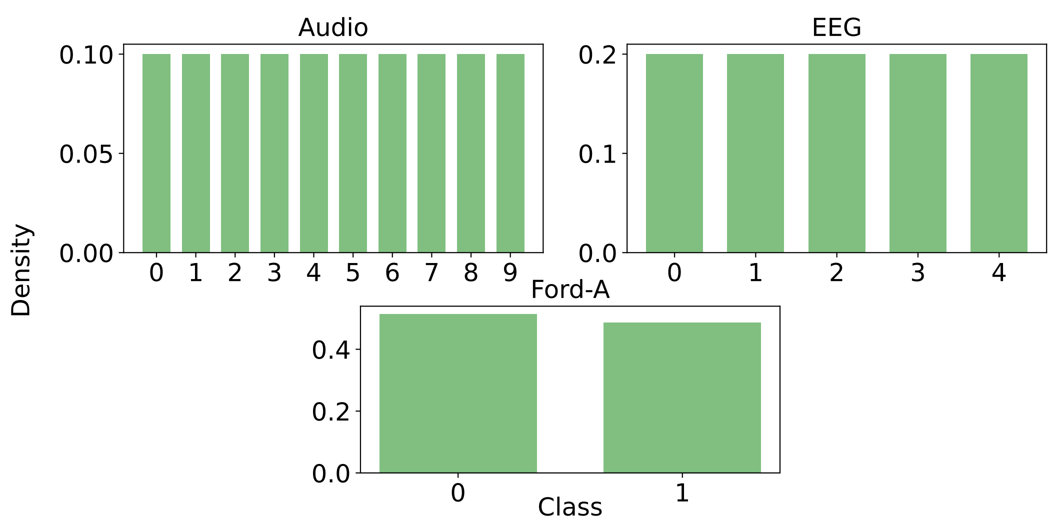

Figure 7: Class distribution among different datasets.

D Detailed Explainability Results

D.1 Evaluating Positive Temporal Attribution

Figure 8 shows the mean $e(u)$ versus unmasking percentage ($u\%$) curves obtained using different interpretability methods, along with their standard deviations. The trend of the curves clearly demonstrates that `TimeSliver` outperforms the baseline methods in the lower unmasking range (5–20%), highlighting its effectiveness in identifying the most critical time points.

[1] https://github.com/pytorch/captum

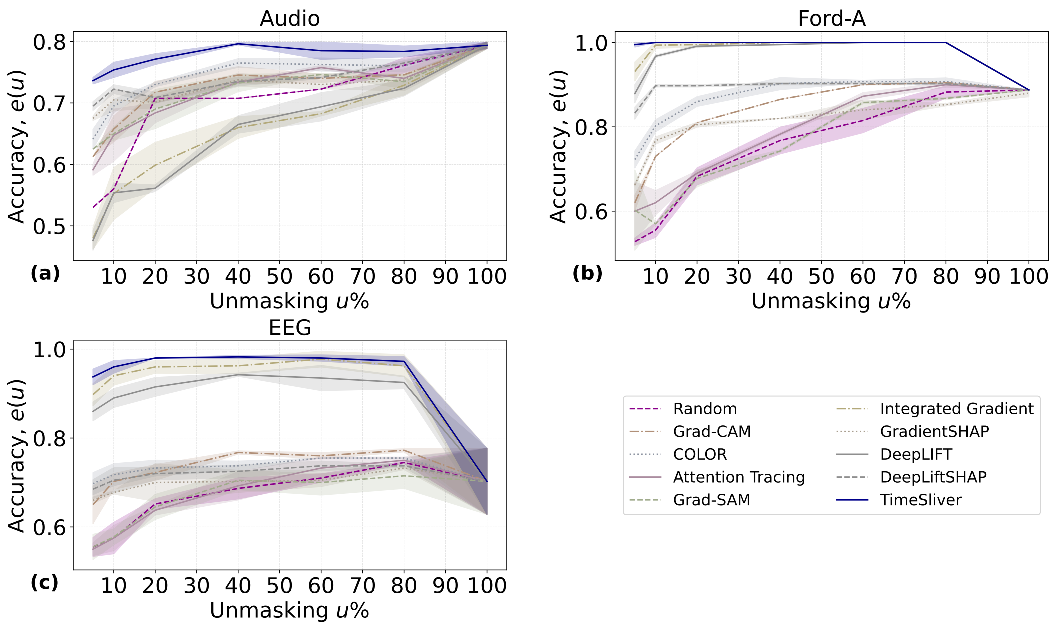

Figure 8: Positive attribution study. Accuracy curves $e(u)$ plotted against the unmasking percentage $u\%$ for various methods on three datasets: a) Audio, b) Ford-A, and c) EEG SSC. Each curve represents the mean accuracy over three runs (supporting results for Section 3.1 in the main paper).

The areas under the curves shown in Figure 8, $\mathcal{I}(100)$ and $\mathcal{I}(20)$, are used to quantitatively compare the different interpretability methods.

D.2 Impact of other symbolic representations

In Table 9, we show the impact of choosing different discretization functions, $h(\cdot)$, defined in Section 2.2.1, to convert the continuous time series data into symbolic representation as $s_i = h(x_i; n, w)$. Both ABBA and SFA preserve explainability (AUPRC remains unchanged). However, if we do not convert x into symbolic representation and utilize the higher dimensional projection of x, x_{proj}), to calculate Z, the explainability drops by 38%. This highlights that discretizing the continuous time series x into a symbolic representation (s) yields an input value-agnostic encoding, ensuring that all time points are treated uniformly without dependence on input scaling.

Symbolic Representation	**Explainability** AUPRC($\Delta\%$)
`TimeSliver` with binning	$0.94_{\pm 0.045}$
ABBA	$0.94_{\pm 0.048}$ (0%)
SFA	$0.93_{\pm 0.068}$ (-1.0%)
$\mathcal{O} \rightarrow x_{proj}$	$0.66_{\pm 0.13}$ (-38.7%)

Table 9: Impact of symbolic representations on explainability.

D.3 TimeSliver's effectiveness in capturing far-field interaction

Although P in Equation 2 is formulated as a linear aggregation of temporal segments, it does not significantly affect the ability of `TimeSliver` to capture far-field multiplicative interactions. To demonstrate this, we construct a synthetic dataset designed specifically to exhibit strong far-field dependencies.

Input Construction. We generate $N = 1000$ samples, each of length $L = 100$. Each sample x_i is defined as:

$$x_i(t) = \sin(f_i t + \phi_i) + \eta_i(t) \cdot \left(t - \frac{L}{2}\right), \quad t = 1, \ldots, L, \tag{6}$$

where:

- $f_i \sim \mathcal{U}(1, 10)$ is a randomly sampled frequency,
- $\phi_i \sim \mathcal{U}(0, 2\pi)$ is a random phase shift, and
- $\eta_i(t)$ is Gaussian noise scaled by the time component $\left(t - \frac{L}{2}\right)$ to amplify far-field interactions.

Output Property (Far-Field Interaction). For each sample x_i, we define the far-field interaction property:

$$p_i = \sum_{j=1}^{L/2} x_i[j] \cdot x_i[L - j + 1], \tag{7}$$

where $x_i[j]$ is the j^{th} element and $x_i[L - j + 1]$ is its *far-field pair* from the opposite end of the sequence.

A binary label y_i is then assigned:

$$y_i = \begin{cases} 1, & \text{if } p_i > 0, \\ 0, & \text{if } p_i \leq 0, \end{cases} \tag{8}$$

resulting in a balanced class distribution with a 0:1 ratio of 1:1.

Results. Table 10 compares the predictive performance of `TimeSliver` with Transformer, LSTM, and FCN baselines. The results empirically confirm that `TimeSliver` effectively captures multiplicative far-field interactions, which we attribute to the fully connected neural network layer present after P in the architecture.

Metric	Transformer	LSTM	`TimeSliver`	FCN
Balanced Accuracy	0.68 (0.014)	0.73 (0.04)	0.76 (0.02)	0.77 (0.05)

Table 10: Predictive performance on the synthetic far-field interaction dataset. Results are reported as mean (std) accuracy over three runs.

E Detailed Predictability Results

We evaluate `TimeSliver` against five methodological categories: (1) *Distance-based* methods, including DTW variants (Bagnall et al., 2016); (2) *Dictionary/interval-based* methods, such as MUSE (Schäfer & Leser, 2017) and gRFS/CIF (Middlehurst et al., 2020); (3) *Feature-based ML* models like ROCKET (Dempster et al., 2020) and MrSEQL (Karlsson et al., 2016); (4) *Deep learning* models including ResNet (Wang et al., 2017), InceptionTime (Fawaz et al., 2020), FCN (Karim et al., 2017), TS2vec (Yue et al., 2022), TimesNet (Wu et al., 2022), ShapeNet (Li et al., 2021b), RLPAM (Gao et al., 2022b), ShapeConv (Qu et al., 2024b), SBM (Wen et al., 2025a), InterpGN (Wen et al., 2025a); and (5) *Ensemble-based* approaches such as CBOSS, STC (Bagnall et al., 2016; 2017), RISE, TSF, and HC (Lines et al., 2018). The predictive performance of `TimeSliver` on all 26 UEA datasets, along with the results of the baseline methods, is presented in Table 11. The comparison of `TimeSliver` with baseline methods in terms of average rank, top-1, and top-3 counts is presented in Table 12.

Problem	DTW_D	DTW_I	DTW_A	MUSE	gRSF	CIF	MrSEQL	ROCKET	CBOSS	STC	RISE	TSF	HC	TapNet	ResNet	Inception Time	TimeSliver
ArticularyWordRecognition	98.87	94.31	98.94	98.87	98.21	97.89	98.98	**99.56**	97.56	97.51	95.73	94.82	97.99	97.13	98.26	99.10	99.33
AtrialFibrillation	23.56	34.67	22.44	**74.00**	27.56	25.11	36.89	24.89	30.44	31.78	24.44	29.78	29.33	30.22	36.22	22.00	73.00
BasicMotions	95.25	97.17	99.92	**100.00**	99.83	99.75	94.83	99.00	98.75	97.92	**100.00**	98.78	**100.00**	99.08	**100.00**	**100.00**	**100.00**
Cricket	**100.00**	95.74	**100.00**	99.77	97.41	98.38	99.21	**100.00**	97.55	98.94	97.78	93.15	99.26	97.50	99.40	99.44	98.61
DuckDuckGeese	49.20	29.27	56.67	56.00	44.47	56.00	39.27	46.13	43.07	43.47	50.80	38.87	47.60	58.27	63.20	**63.47**	56.10
EigenWorms	64.58	44.20	97.85	99.33	83.00	90.33	72.16	86.28	62.80	74.68	81.93	76.62	78.17	83.00	45.45	**98.68**	89.31
Epilepsy	96.30	67.03	97.37	99.64	97.34	98.38	99.93	99.08	99.83	98.74	99.86	99.83	**100.00**	96.09	98.16	98.65	98.55
EthanolConcentration	30.15	30.68	29.87	48.64	34.06	72.89	60.18	44.68	39.62	**82.36**	49.16	45.42	80.68	28.99	28.62	27.92	43.73
ERing	92.91	91.42	92.89	**96.89**	91.98	95.65	93.19	98.05	84.48	84.28	82.44	89.84	94.26	89.46	87.19	92.10	84.10
FaceDetection	53.28	51.53	–	68.89	55.06	69.17	62.97	69.42	52.32	69.76	51.17	68.95	69.17	52.87	53.13	**77.24**	69.97
FingerMovements	54.17	55.50	54.93	54.77	54.43	53.90	55.53	55.27	51.03	53.40	52.10	53.17	53.77	51.33	54.70	56.13	**65.00**
HandMovementDirection	30.32	26.67	30.72	38.02	32.07	**52.21**	35.23	44.59	28.87	34.95	28.24	48.51	37.79	32.34	35.32	42.39	46.00
Handwriting	61.21	34.33	60.55	51.85	36.06	35.13	54.04	56.67	49.09	28.77	18.27	36.42	50.41	32.95	59.78	**65.74**	57.10
Heartbeat	68.88	63.80	69.87	73.59	78.49	76.52	72.52	71.76	72.15	72.15	73.22	72.28	72.18	73.97	63.89	78.62	**80.49**
Libras	88.04	78.63	87.85	90.30	75.56	**91.67**	86.57	90.61	85.26	84.46	81.67	79.72	90.28	83.63	**94.11**	88.72	83.33
LSST	54.76	49.57	56.96	63.62	58.05	56.17	60.28	63.15	43.62	57.82	50.58	34.31	53.84	46.33	42.94	33.97	**68.53**
MotorImagery	56.10	49.63	50.37	52.17	53.80	51.80	53.00	53.13	52.37	50.83	49.83	53.80	52.17	45.37	49.77	51.17	**61.90**
NATOPS	82.04	76.07	81.48	87.13	82.37	84.41	86.43	88.54	82.48	84.35	80.59	77.72	82.85	90.30	97.11	96.63	**98.89**
PenDigits	99.28	99.22	99.27	98.68	91.27	98.97	97.14	99.56	95.61	97.70	87.47	94.11	97.19	93.65	99.64	**99.68**	98.10
PEMS-SF	77.05	80.23	78.73	99.85	91.27	**99.86**	97.15	85.63	96.57	98.40	98.98	96.76	97.98	79.21	81.95	82.83	94.80
PhonemeSpectra	15.39	10.18	–	25.86	15.27	32.87	30.86	28.35	19.43	30.62	26.78	14.52	32.87	22.17	15.39	**36.74**	29.00
RacketSports	85.64	81.69	85.79	89.56	87.79	89.30	88.73	**92.79**	88.90	88.09	84.17	88.29	90.64	85.81	91.23	91.69	92.10
SelfRegulationSCP1	81.81	80.63	81.34	73.58	79.74	85.94	82.86	86.55	81.33	84.73	73.17	84.73	86.02	**95.68**	76.11	84.69	90.00
SelfRegulationSCP2	54.09	48.48	52.43	49.52	50.62	48.87	49.61	51.35	50.62	51.63	50.28	50.62	51.67	56.05	50.24	52.04	**62.00**
StandWalkJump	22.00	35.78	25.56	34.67	38.44	45.11	42.00	45.56	36.89	44.00	34.00	33.33	40.67	35.11	30.89	42.00	**67.00**
UWaveGestureLibrary	92.28	87.58	91.51	90.39	89.59	92.42	91.32	**94.43**	86.13	87.03	71.11	85.05	91.31	89.59	88.35	91.23	91.00

Table 11: Detailed predictive performance comparison across 26 datasets in UEA (supporting results for Section 3.2 in the main paper).

Table 12: `TimeSliver` vs. 16 baselines on 26 UEA datasets, evaluated using Average Rank, Top-1 Count, and Top-3 Count. **Bold** indicates the best result, underlined indicates the second-best. ↑ and ↓ denote that higher and lower values are better, respectively.

Type	Method	Average Rank ↓	Top-1 Count ↑	Top-3 Count ↑
Distance-Based	DTW_D	16.88	0	0
	DTW_I	18.65	0	0
	DTW_A	14.67	1	1
Dictionary-Based	MUSE	10.50	3	6
	gRSF	16.88	0	0
	CIF	11.31	1	3
Feature-Based ML	MrSEQL	12.88	0	0
	ROCKET	9.31	<u>4</u>	7
Deep Learning	TapNet	17.77	1	1
	ResNet	15.38	1	1
	IncTime	7.69	<u>4</u>	7
	FCN	9.92	1	5
	TS2Vec	16.88	0	0
	TimesNet	15.92	0	0
	ShapeNet	12.58	1	2
	RLPAM	10.23	<u>4</u>	<u>10</u>
	ShapeConv	11.19	1	4
	SBM	11.15	1	3
	InterpGN	<u>7.08</u>	3	7
Ours	TimeSliver	**7.00**	**6**	**11**