machine learning & knowledge extraction

MDPI

Article

What about the Latent Space? The Need for Latent Feature Saliency Detection in Deep Time Series Classification

Maresa Schröder [2,3,*,†], Alireza Zamanian [1,2,†] and Narges Ahmidi [2,*]

[1] Department of Computer Science, TUM School of Computation, Information and Technology, Technical University of Munich, 80333 Munich, Germany; alireza.zamanian@iks.fraunhofer.de
[2] Fraunhofer Institute for Cognitive Systems IKS, 80686 Munich, Germany
[3] Department of Mathematics, TUM School of Computation, Information and Technology, Technical University of Munich, 80333 Munich, Germany
* Correspondence: maresa.schroeder@tum.de (M.S.); narges.ahmidi@iks.fraunhofer.de (N.A.)
† These authors contributed equally to this work.

Abstract: Saliency methods are designed to provide explainability for deep image processing models by assigning feature-wise importance scores and thus detecting informative regions in the input images. Recently, these methods have been widely adapted to the time series domain, aiming to identify important temporal regions in a time series. This paper extends our former work on identifying the systematic failure of such methods in the time series domain to produce relevant results when informative patterns are based on underlying latent information rather than temporal regions. First, we both visually and quantitatively assess the quality of explanations provided by multiple state-of-the-art saliency methods, including Integrated Gradients, Deep-Lift, Kernel SHAP, and Lime using univariate simulated time series data with temporal or latent patterns. In addition, to emphasize the severity of the latent feature saliency detection problem, we also run experiments on a real-world predictive maintenance dataset with known latent patterns. We identify Integrated Gradients, Deep-Lift, and the input-cell attention mechanism as potential candidates for refinement to yield latent saliency scores. Finally, we provide recommendations on using saliency methods for time series classification and suggest a guideline for developing latent saliency methods for time series.

Keywords: explainability (XAI); time series classification; saliency methods; latent feature importance; deep learning

Citation: Schröder, M.; Zamanian, A.; Ahmidi, N. What about the Latent Space? The Need for Latent Feature Saliency Detection in Deep Time Series Classification. *Mach. Learn. Knowl. Extr.* **2023**, *5*, 539–559. https://doi.org/10.3390/make5020032

Academic Editor: Luca Longo

Received: 4 April 2023
Revised: 30 April 2023
Accepted: 7 May 2023
Published: 18 May 2023

1. Introduction

Saliency methods are designed to explain the decision-making process of deep learning models by estimating the effect of each input feature on the prediction. Initially developed for image data, these methods assign importance scores to individual input features, namely, the pixels [1,2]. As a result, class-distinctive patterns and regions of the input feature space are identified based on the positional information. This strategy suits the image domain, where the class label is often associated with specific input regions (e.g., the presence of an informative object in the scene). As a result, the generated saliency maps are immediately interpretable. The image saliency methods are directly adopted to time series data [3,4], where they assign importance scores to the time points. In the case of class-distinctive temporal patterns, these methods suit time series problems. On many occasions, however, the labels depend on one or more underlying latent features, such as a dominant frequency, the state-space model parameters, or the overall trend of a nonstationary time series. In this case, even though the classifier might achieve a high prediction performance, the calculated positional saliency scores will not directly explain the importance of the underlying latent features. Hence, the generated saliency maps do not provide easy interpretability and thus fail to fulfill their original purpose.

This paper extends our preliminary work in [5], which originally introduced and formulated the explainability dilemma of adopted time series saliency methods. In this work, we extend our analyses in multiple directions by including further time series classifiers and saliency methods, validating our findings using a real-world dataset, and integrating quantitative metrics to evaluate saliency methods. In summary, our main contributions are as follows:

1. We provide an extension of our study of prominent time series saliency methods on top of six classification methods by incorporating two additional classifiers (temporal convolutional network, input-cell attention LSTM). As a byproduct of this extension, we contribute to the discussion of "attention as explanation" by identifying the input-cell attention mechanism [6] as a suitable saliency method for time series classification. We provide evidence for the vanishing saliency problem in recurrent neural networks and compare the functionality of two solutions proposed in the literature.
2. We empirically investigate the problem of latent feature saliency using an experimental framework of simulated and real-world datasets. This framework includes an architecture for simulation studies, visual investigation of saliency maps, and quantitative validation of the methods. To simulate natural and realistic time series, we employ the popular Fourier series latent model. According to the Fourier theorem, any given signal can be expressed as the sum of Fourier sinusoidal components. Hence, this framework can also be applied to other latent models which are not covered in this paper.
3. Based on the results of the experiments, we compile a list of recommendations for more effective utilization of the investigated time series saliency methods. Furthermore, we identify effective candidate methods for tackling the problem of latent feature saliency detection.

The remainder of the paper has the following structure:

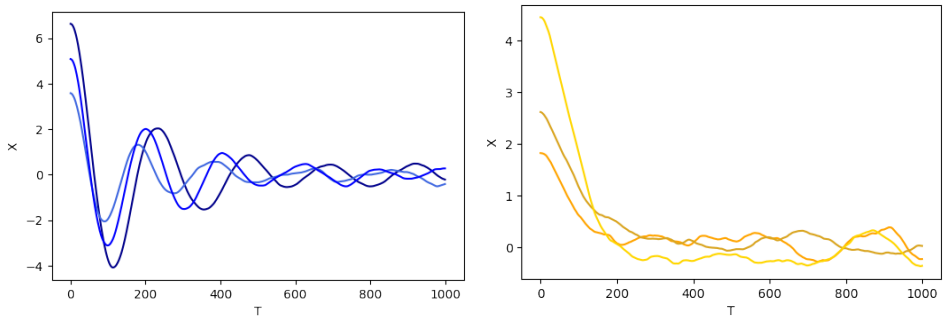
- Problem definition: First, we discuss the distinction between shapelet- and latent space-related classification problems. We posit the argument of the ideal saliency methods for latent features and suggest a framework for extending the current methods (Section 1.1).
- Empirical evidence: To evaluate multiple state-of-the-art post hoc saliency methods, we extend the experiments in [5] on simulated time series data with patterns in both temporal and latent spaces. Through visual and quantitative evaluations, we demonstrate their lack of interpretable outputs when the informative pattern originates from the latent space rather than the temporal domain. Additionally, we run similar experiments using a real-world predictive maintenance dataset (Sections 3 and 4).
- Recommendations for use of saliency methods: Finally, we provide a list of recommendations for utilizing common saliency methods in time series problems and identify potential candidate methods for the development of a latent feature saliency detection method (Section 5).

1.1. Problem Formulation

Let $D = (X, Y)$ with a univariate time series $X \in \mathcal{X}$ and a binary label $Y \in \{0, 1\}$ formulate time series classification data. We assume a latent representation $Z \in \mathcal{Z}$ such that the direct mapping $f_{XY} : \mathcal{X} \mapsto \mathcal{Y}$ in the learning procedure is replaced by two feature-to-latent and latent-to-label mappings, i.e., $f_{XY} \equiv f_{XZ} \circ f_{ZY}$.

One way to categorize the time series classification problems is with respect to the nature of the class-distinctive patterns in the signals, namely, into two shapelet-based and latent-based categories [5]. This view highlights the post hoc saliency detection problem present in time series classification tasks. In shapelet-based problems, the informative class-distinctive patterns appear in the time domain as shapelets [7]. This generally includes a sudden rise or fall or a specific pattern of fluctuations. On the other hand, in latent-based problems, the informative patterns are found to be related to the parameters of the latent space, e.g., the time series frequency. As emphasized in [5], current saliency

methods for time series, by design, are only able to discover the informative patterns in the time domain. As a result, they systematically fail to highlight the important patterns in latent-based problems. Note that temporal patterns may still appear as a proxy for the latent information (Figure 1). As a result, shapelet-based methods may potentially solve latent-based classification problems, yet the adopted saliency methods fall short of explainability.



(**a**) (**b**)

Figure 1. Example of classification of vibration systems where shapelets (oscillation patterns) are proxies for latent parameters (damping ratio). Each plot contains the responses from three example systems of the same system type (illustrated with different shades of the same color). For these systems, a label, correlated with the damping ratio ζ, can be potentially predicted by shapelet-based classifiers; however, a conventional saliency method applied to this problem will only highlight a proxy of the informative latent feature, namely, the existing fluctuations and oscillations of the time series. (**a**) Underdamped systems: $\zeta < 1$; (**b**) overdamped systems: $\zeta > 1$.

We argue that despite the systematic failure of the saliency methods, the output of the failed methods might still carry enough information to identify the associated latent space and its important class-distinctive pattern. To formulate the discussion, consider a classification problem with an effective latent representation associated with the latent space $\mathcal{Z}^{(*)}$ out of K candidate counterparts $\mathcal{Z}^{(k)}, k = 1, \ldots, K$. Furthermore, assume that each latent space has a corresponding saliency map $M^{(i)} \in \mathbb{R}_+^{|Z^{(i)}|}$, and that these maps are "reliable", namely, we have $\forall i \neq * : \max M^{(*)} > \max M^{(i)}$. Let $m_T : \mathcal{X} \mapsto \mathbb{R}_+^{|X|}$ represent a saliency method which yields the saliency map $M_T \in \mathbb{R}_+^T$ in the time domain. We say that m_T is a promising method if the M_T bears enough information to infer $M^{(*)}$, possibly via a simple mapping (e.g., a shallow network) [5]. Figure 2 schematically illustrates the saliency map of a good failing method when the label is associated with either the frequency or amplitude of a Fourier model, the trend of an additive model, or a shapelet. The highlighted regions are sufficient to deduce the informative parameter.

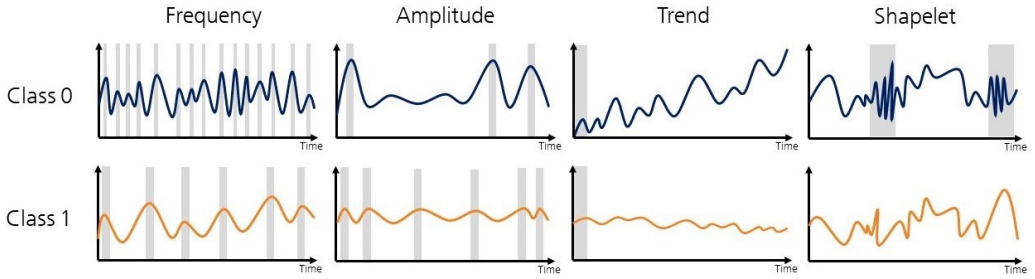


Figure 2. Explainability toy examples of multiple label-making scenarios in the time series domain. Per each column, example time series are colored blue for class 0, and gold for class 1. Influential time steps (regions with high saliency scores) are shaded in gray for frequency (the peaks), amplitude (highest peaks), trend (a window enough for inferring about the trend), and shapelet (presence of the informative pattern) [5].

Subsequently, we continue this paper by evaluating state-of-the-art saliency methods in such experiments to underline our arguments empirically and to identify the best candidates for developing latent feature methods.

2. Literature Review on Post Hoc Saliency Methods

Throughout our studies, the focus is placed on the post hoc saliency methods which are applied to classifiers after they are trained, thus being decoupled from the original classification model. Categorizing by the approach, these methods fall into three groups: (1) gradient-based feature attribution (FA) methods, (2) model-agnostic FA methods, and (3) counterfactual explanations (CF).

(1) *Gradient-based feature attribution (FA)* assigns importance scores to each input feature based on the magnitude of the gradient of the output with respect to the separate input features. These methods are model-specific since direct access to the model parameters is needed. The fundamental attribution method Saliency [8] generates saliency scores based on the raw gradients directly. Gradient $\times$ Input [9] adjusts the previous work by multiplying the gradients by the input. DeCovNet [10], Guided Backpropagation [11], and SmoothGrad [12] introduce adaptions of backpropagation rules to reduce noise in the gradients [13]. Deep-Lift [14] utilizes a neuron attribution-based difference-from-reverence approach to assign the scores. The ubiquitous problem of gradient saturation is tackled by the method Integrated Gradients (IG) [15] through calculating the path integrals from a noninformative baseline input to the respective input feature [16]. Relevance-based methods, such as Layer-wise Relevance Propagation (LRP) [17] and Deep Taylor Decomposition [18], retrieve attribution scores based on the backpropagation of relevance scores from the output back through the network via various designed propagation rules.

(2) *Model-agnostic* FA methods are another class of post hoc methods that can be applied to any black-box classifier without access to the model's internals [19,20]. Methods such as Occlusion [10], Meaningful Perturbations [21], and RISE [20] assign saliency scores to each input feature based on the change in the output when the respective feature is perturbed. LIME [22] first fits local interpretable surrogate models to the classifier in the neighborhood of the target sample. The saliency scores are then calculated based on the parameters of the surrogate models. Multiple other methods are inspired by game-theoretic theorems and concepts [23–25], such as the Shapley value [26], in particular the SHAP method [27], measuring feature importance as the Shapley value of a conditional expectation function of the to-be-explained prediction model, and its extensions have gained high popularity in the literature.

(3) Another class of post hoc methods aims at generating *counterfactual explanations*. These methods estimate how much variation in individual input features is required to change the classification outcome and associate small changes with higher saliency scores. Examples are LASTS [28], time series tweaking [29], LatentCF++ [30], and CoMTE [31]. Unlike FA methods, counterfactual explanation methods do not provide saliency scores. They, rather, identify countersamples from the other class to provide explainability for a sample. To be effective, the found counterfactuals must provide sparse explanations so that differences (perturbations) are found in as few features as possible. Delaney et al. [32] proposed their CF method Native Guide, fulfilling these properties. This method determines a train data sample closest to the to-be-explained instance under the dynamic time-warping distance with a different class label. Afterward, this sample is perturbed towards the decision boundary. Since we analyze the saliency maps and quantitative scores in our experiments, we will not include counterfactual explanation methods in the comparisons. However, in Section 5, we will explain how these methods can provide explainability when FA methods fail.

3. Materials and Methods

In this section, we state the selected time series saliency methods for the experiment and introduce the simulation setup, the real-world dataset, and the utilized quantitative metrics. Implementation details are stated in Appendix A.1.

3.1. Classification Models and Saliency Methods

The experiments' results depend on the methods' approach to identifying salient features. To cover the three existing approaches described in Section 3.1 and diversify the findings, we selected five candidate methods from different classes of post hoc methods: the Integrated Gradients (IG) and Deep-Lift (DL) methods from the first, LIME and Kernel SHAP (SHAP) methods from the second, and the counterfactual method Native Guide (NG) method from the third family.

In our previous work, we utilized the long short-term memory networks (LSTM) [33] and convolutional neural networks (CNN) [34], each trained both in a standard fashion and via the saliency-guided training procedure (SGT) [35]. This allowed the networks to produce more consistent saliency scores, as the saliency feedback is used for training the network weights. This work extends the experiments by incorporating the temporal convolutional networks (TCN) [36] into the classifiers pool.

Finally, we draw attention to an analysis of the vanishing saliency problem in recurrent neural networks by Ismail et al. [35] and a developed antidote to this issue, namely, the input-cell attention mechanism for LSTMs [6]. The adjustment helps the network with better training in addition to providing saliency scores. Therefore, we also consider the mechanism as an ante hoc explainability method in our experiments. Table 1 presents a list of employed classifiers and saliency methods. Implementation details are stated in Appendix A.1.

Table 1. Classifiers and saliency methods employed in the experiments.

Model	Remarks
Time series classifiers	
LSTM	
LSTM + SGT	Trained via the saliency-guided training procedure
AttentionLSTM	Input-cell attention mechanism combined with LSTM
CNN	
CNN + SGT	Trained via the saliency-guided training procedure
TCN	
Saliency methods	
Integrated Gradients	Gradient-based
Deep-Lift	Gradient-based
LIME	Model-agnostic
Kernel SHAP	Model-agnostic
Input-cell attention	Ante hoc method (with LSTM classifier)

3.2. Synthetic Data Generation

We generated a total of ten balanced datasets containing 2560 split in a ratio of 80%, 10%, and 10% for training, validation, and testing, respectively, to understand the response of different saliency methods to different patterns. The experiments included four experiments with temporal shapelet patterns, two with latent amplitude patterns, two with latent frequency patterns, and two with latent phase shift patterns (Table 2). The data generation mechanism, introduced in [5], is described in detail in Appendix A.2. Synthetic data generation allows us to study the latent features individually and in a controlled manner. Potential poor results can therefore be confidently attributed to the intrinsic weakness of the saliency methods rather than inappropriate classifiers.

Table 2. Description of synthetic datasets.

Experiment	Label Associated with	Remarks
1–4	Shapelet	The label depends on existence of a shapelet, appearing at the beginning, middle, or end of the time series, or a random position (4 scenarios).
5, 6	Frequency	The labels depend on the frequency of the dominant sine wave in the Fourier series. The experiments differ in the number of sines waves, thus rendered as simple or complex (2 scenarios).
7, 8	Phase shift	Similar to the experiments 5, 6, except for the deciding latent parameter which is the phase shift of the Fourier series.
9, 10	Amplitude	Similar to the experiments 5, 6, except for the deciding latent parameter which is the maximum amplitude of the Fourier series.

3.3. Real-World Dataset

To supplement the simulation studies, we tested the saliency methods on a real-world dataset in which the label depends on the parameters of a latent model (known by prior knowledge). The CWRU Bearing Dataset [37] is a predictive maintenance dataset containing vibration data from 2300 bearings for bearing failure prediction. Defects of different sizes and locations were exerted on a group of bearings, resulting in highly diverse vibration signals at operation. The dataset is unbalanced, with a 90% failure percentage. In addition to the raw time series data, the dataset provides nine handcrafted features. Figure 3 illustrates the histogram of six of the nine features per class. As depicted, the maximum, minimum, standard deviation, and root mean square features significantly distinguish between the two classes. These features are highly associated with the amplitude and frequency of a signal, implying that assuming an association between latent parameters and the label is a valid assumption for vibration signals. Other noninformative features not depicted are kurtosis, crest, and form. Analysis of the feature-based classification revealed the "maximum" feature as the most important among the provided features, signifying the label–Fourier series amplitude association.

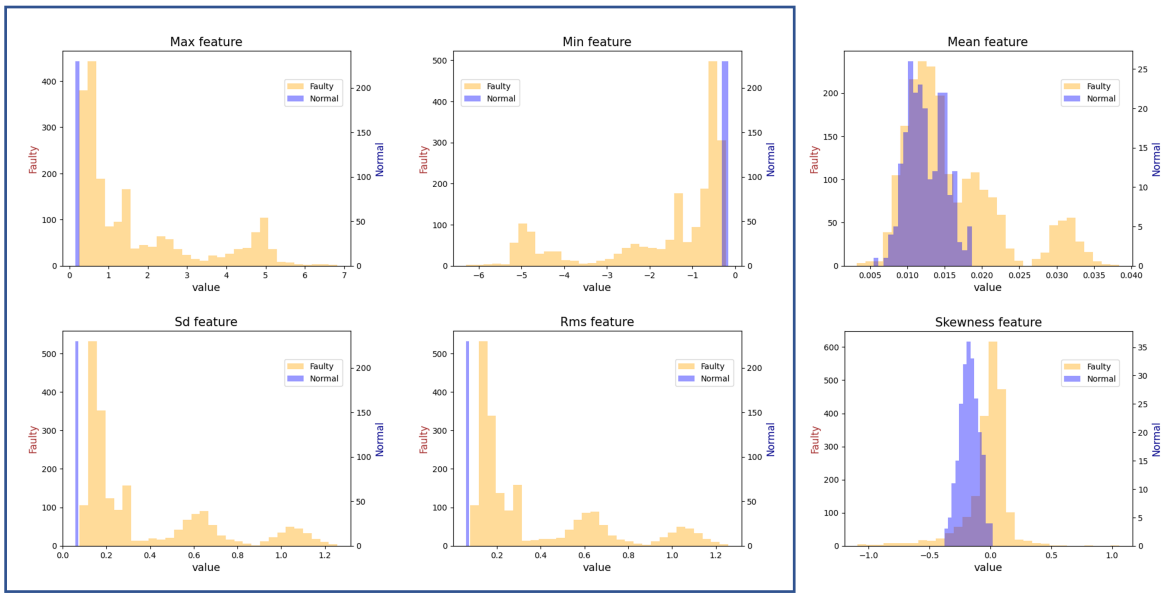


Figure 3. Distribution of engineered features of the CWRU dataset for normal (blue) and faulty (orange) bearings. Important features which allow for a simple distinctive line between the classes are circled.

3.4. Quantitative Evaluation

No omniapplicable quantitative metric has been introduced in the literature to assess the saliency method's performance. This is partly due to the lack of a universal definition of the properties that a saliency method must possess. Controlled environment experiments on synthetic data allow us to rely confidently on obtained visual clues. Loeffler et al. [3] proposed a time-series-specific evaluation framework consisting of six orthogonal metrics for post hoc saliency methods applied to classification problems. We employ two of the fundamental metrics, i.e., faithfulness and sanity.

Faithfulness for temporal importance is based on the decrease in accuracy when salient input features are perturbed. The reported score ranging from zero (not faithful) to one (highly faithful) is the average score for all dataset samples. For a saliency method m, a classifier f_{XY}, and a dataset D consisting of N samples, the faithfulness score is calculated as

$$\text{Faith}(m, f_{XY}, D) = 1 - \frac{1}{N \cdot L} \sum_{X \in D} \sum_{l=1}^{L} \text{softmax}^c(X_l^{'}), \tag{1}$$

where $X_l^{'}$ depicts the perturbation of $X = (x_1, ..., x_T)$, in which the values x_t with the highest l saliency scores are replaced by the mean value of all sequences in the test dataset at time t. The softmax prediction is performed with respect to the class label of interest c. In our experiments, we set the maximum perturbation length L to $0.3 \cdot T$.

The sanity score measures the dependency of the generated saliency maps on the classification model. Specifically, the evaluation metric calculates the change in saliency maps when the classifier's parameters (weights) are randomized layer-wise. Since we exclusively utilize shallow networks, we measure sanity via a cascading randomization of all network weights instead of a layer-wise procedure. Starting from the last layer, we randomize weights in steps of 25% until all network weights are altered. Sanity is evaluated based on the structural similarity index (SSIM) [38] between the original and post-randomization saliency pattern. Let K be the number of randomization steps, $M_T(X)$ the saliency map on sample X produced by method m, and $M_{T_k}^{'}(X)$ the produced saliency map after weight randomization step k. Then, the sanity score for the saliency method m, a classifier f_{XY}, and a dataset D is calculated as

$$\text{Sanity}(m, f_{XY}, D) = -\frac{1}{N \cdot K} \sum_{X \in D} \sum_{k=1}^{K} \text{SSIM}(M_T(X), M_{T_k}^{'}(X)), \tag{2}$$

where SSIM$(\cdot)$ is a structural similarity index measure. For better visualization, we scale the average sanity score across all dataset samples to the $[0, 1]$ range (higher is better).

4. Results

In this section, we present the experimental results for the methods in Table 1 using the simulated datasets explained in Table 2 as well as the CWRU dataset. The saliency methods are only expected to perform well when the classifiers are effective. Therefore, we first report the performance evaluation results for the classifiers. Subsequently, we illustrate and analyze the output of the saliency methods. All the obtained results are interpreted and discussed in Section 5.

4.1. Classification Performance

Table 3 presents the results of the average accuracy and F1 score across all synthetic datasets grouped by the type of experiments. The experiments are introduced in Table 2. In [5], we noted that the LSTM classifier relatively underperforms in the phase shift experiments. This is possibly due to the vanishing gradient problem which hinders a proper classification when the informative patterns are placed in the early time points. The saliency-guided training procedure had a slightly negative effect on the performance of the LSTM. On the other hand, the AttentionLSTM showed a significant improvement

over the LSTM classifier, especially in the phase shift experiments. Unlike the LSTM, the CNN benefited from the saliency-guided training procedure. Finally, the TCN showed consistent notable performance across experiments. Overall, the input-cell attention LSTM, the temporal convolutional network, and the CNN trained via saliency-guided training achieved the best classification performances. Based on this observation, we consider only these classifiers for the experiments on real-world data.

Table 3. Average classification performance on test data across all synthetic datasets. The bold values are higher than the counterparts with a statistical significance.

Classifier	Shapelet		Frequency		Phase Shift		Amplitude	
	Accuracy	F1	Accuracy	F1	Accuracy	F1	Accuracy	F1
LSTM	0.8535	0.8466	0.9749	0.9470	0.5157	0.4914	0.9981	0.9981
LSTM + SGT	0.8242	0.8417	0.9082	0.9117	0.5352	0.4145	0.9160	0.9230
CNN	0.6221	0.7439	**0.9610**	**0.9633**	0.9629	0.9625	0.9981	0.9981
CNN + SGT	0.8721	0.9138	**0.9610**	**0.9633**	0.9649	0.9634	**1.0000**	**1.0000**
TCN	0.7442	0.8275	**0.9610**	**0.9633**	**0.9844**	**0.9842**	0.9981	0.9981
AttentionLSTM	**0.9307**	**0.9361**	0.9375	0.9406	0.9507	0.8985	**1.0000**	**1.0000**

Table 4 provides the experimental results on the CWRU dataset for the three chosen classifiers. Due to the high data class imbalance, we also report the balanced accuracy for the classifiers. We note that all classifiers achieved satisfying results, with no classifier outperforming others significantly.

Table 4. Classification performance on test data of the CWRU Bearing dataset. All classifiers perform highly satisfactorily.

Classifier	Accuracy	F1	Balanced Accuracy
AttentionLSTM	0.9816	0.9896	0.9613
TCN	0.9990	0.9994	0.9994
CNN + SGT	0.9928	0.9960	0.9798

4.2. Saliency Method Evaluation in Simulation Experiments

To evaluate the explainability of the saliency methods visually, we investigate the importance heat maps of the saliency scores provided by the post hoc methods, as well as the attention scores assigned by the input-cell attention mechanism, which is overlaid by the original time series (Figures 4–6). This allows for a direct assessment of the relevance of each input time step for the prediction outcome. We expect the heat maps to highlight the following specific patterns for the respective experiment groups: the time steps in which the shape occurs (shapelet experiments), oscillations focusing either on the peaks or the valleys of the time series (frequency and amplitude experiments) or the beginning of the time series (phase shift experiments). To present and discuss the results, we visualize one representative sample per experiment (refer to [5]).

Figure 4 presents the saliency maps produced by four of the aforementioned post hoc saliency methods (IG, DL, LIME, SHAP) plotted for one sample per experiment group (shapelet and latent experiments). The explanations provided by IG and DL follow the expected visual pattern and are comparatively easy to interpret. On the other hand, the results of SHAP and LIME neither align with our expectations nor permit easy interpretability [5].

Figure 5 depicts examples of the saliency map given by the method IG for the standard LSTM, the LSTM + SGT, and the ante hoc AttentionLSTM (left to right). The LSTM appears to suffer significantly from the vanishing saliency problem since the scores are uniform across the entire time series except for the last time points. Contrary to the observations by Ismail et al. [35], the saliency-guided training procedure did not help to diminish this problem. On the other hand, the input-cell attention mechanism strongly improved the performance of the employed gradient-based saliency method. If the label is based on

amplitude, only on the input-cell attention LSTM can the method IG correctly highlight amplitude-related patterns. The saliency method IG could not highlight the shapelet in the middle of the time series on the LSTM and LSTM trained via saliency-guided training. On the input-cell attention LSTM, the method correctly identified the start of the shapelet as important.

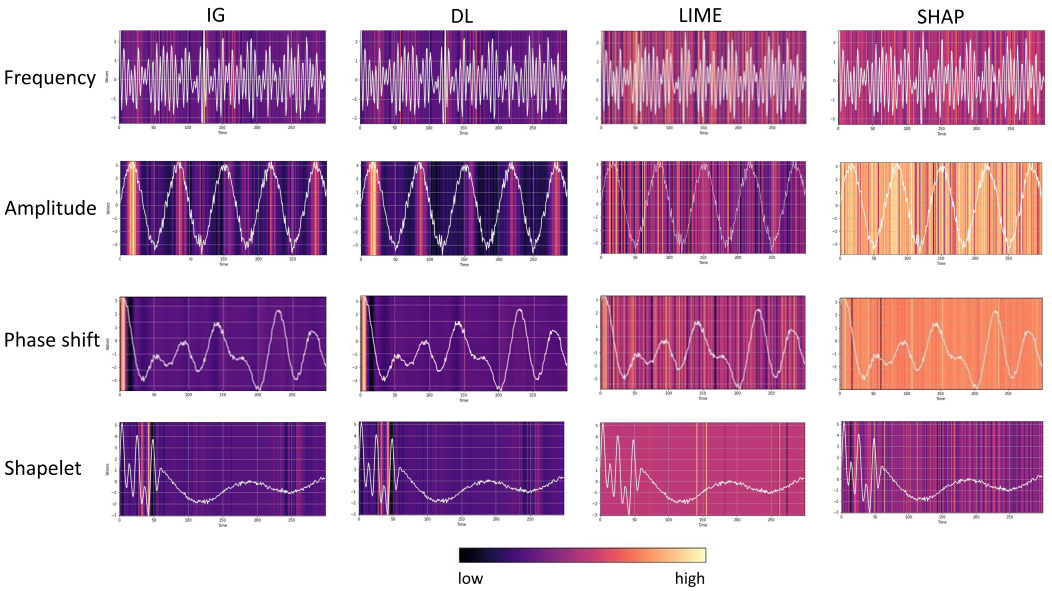


Figure 4. Comparison of importance heat maps from feature attribution methods IG, Deep-Lift, Lime, and SHAP for the CNN + SGT on a frequency, amplitude, phase shift, and shapelet experiment, respectively. Explanations provided by IG and Deep-Lift clearly focus on aspects related to the latent feature (peaks and valleys for amplitude and frequency, beginning of time sequence for phase shift) and the shapelet, respectively. Maps of Lime and SHAP are visually uninterpretable [5].

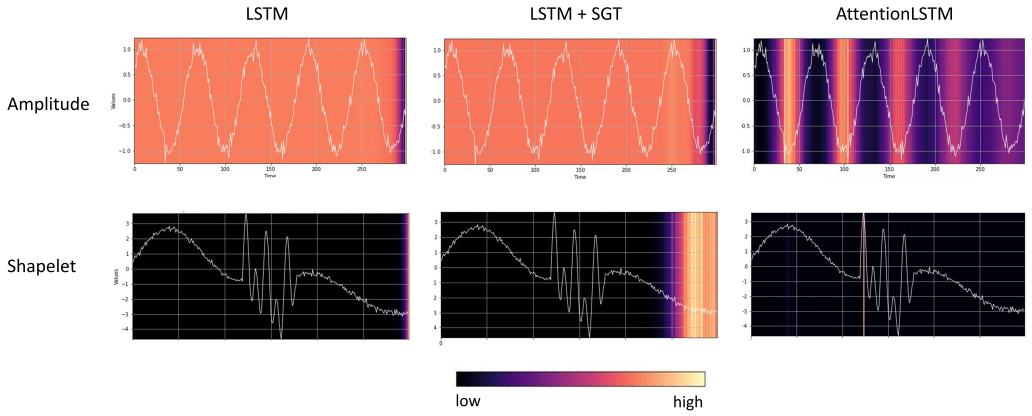


Figure 5. Comparison of explanations provided by the gradient-based saliency method IG on the LSTM, LSTM + SGT, and input-cell attention LSTM if the latent feature amplitude (**top**) or a shapelet at a fixed position (**bottom**) is class-distinctive.

Considering the solid classification performance of the AttentionLSTM in Tables 3 and 4, we compare the saliency scores provided by attention scores with the results of IG in Figure 6. In this experiment (no. 6 in Table 2), samples of class 0 have lower frequencies compared to class 1. Both methods highlight the valleys of the time series, thus correctly implicating a transformation of the concept frequency. IG, however, shows a slight vanishing phenomenon in later time points.

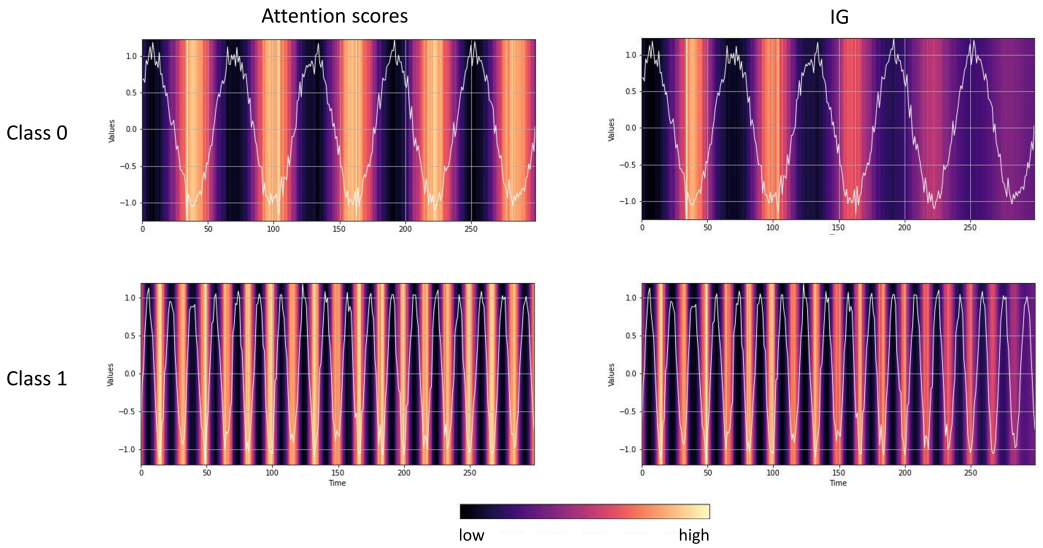


Figure 6. Comparison of importance heat maps from attention scores (**left**) and IG (**right**) for the two classes of experiment 6. The attention mechanism clearly focuses on frequency-related patterns in the time series. The identified importance patterns of both saliency maps coincide.

Figure 7 presents the average sanity and faithfulness scores for post hoc methods across the three well-performing classifiers (AttentionLSTM, TCN, and CNN + SGT) and the synthetic datasets. As depicted, no method significantly outperforms other counterparts in both scores. In a closer examination, Figure 8 presents faithfulness scores per experiment averaged across all classifiers. IG and DL scored slightly higher in the shapelet experiments, especially when a shapelet was introduced at a fixed position (experiments 2–4). The SHAP and LIME methods, whose maps were visually uninterpretable in previous steps, obtained a heightened score on the phase shift experiments (7 and 8). Although these results roughly align with our original hypothesis and the visual evaluation, small differences among the experiments must be pointed out.

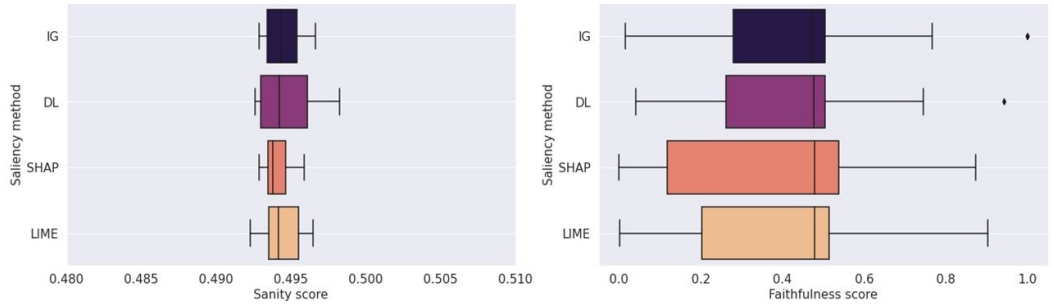


Figure 7. Average saliency performance of the tested post hoc methods evaluated through sanity (**left**) and faithfulness (**right**).

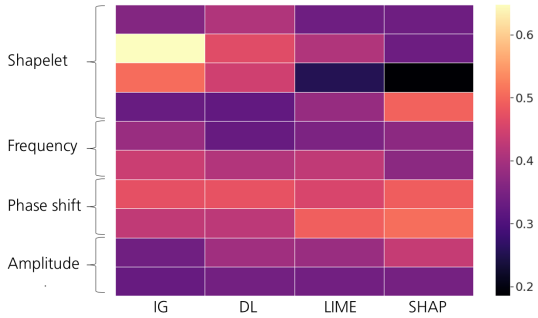


Figure 8. Heat map of faithfulness results split by experiments.

With a focus on more successful classifiers based on Table 3, we measure faithfulness scores for CNN+SGT and TCN individually. As presented in Figure 9, scores are relatively similar for CNN+SGT across different experiments and saliency methods. On the other hand, for the TCN classifier, the effective IG method achieved significantly higher scores in the shapelet experiments than in the latent experiments. Significant differences are also visible in the case of other saliency methods, however not in favor of the shapelet experiments. As an extreme case, SHAP achieves lower scores in the shapelet experiments. In general, the faithfulness score in the latent experiments is close to the average (Figure 7).

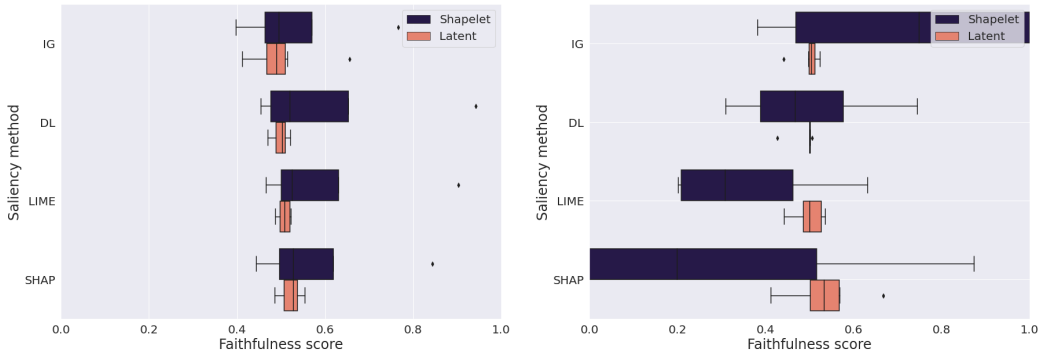


Figure 9. Faithfulness of saliency maps split by type of class-distinctive feature (latent feature vs. shapelet) for the two well-performing classifiers CNN + SGT (**left**) and TCN (**right**).

4.3. Saliency Method Evaluation on the CWRU Dataset

For the CWRU experiment, we employed the CNN + SGT classifier based on the results in Table 4. To present the saliency results, we measured the faithfulness and sanity scores of the four post hoc saliency methods. Figure 10 presents the quantitative evaluation results. As shown in the left plot, IG and DL achieved higher faithfulness scores, while sanity scores were similar across the methods. In a closer look, the right plot shows the detailed sanity test on the methods. This plot visualizes the gradual drop in similarity score between saliency maps before and after the randomization of the weights. According to Equation (2), a sane saliency method must show a continuous decrease in the similarity score as more weights are randomized [3]. Based on this analysis, IG and DL show a slightly better reaction to weight randomization than other methods. Therefore, we selected IG and DL to pair with CNN + SGT classifier for this experiment.

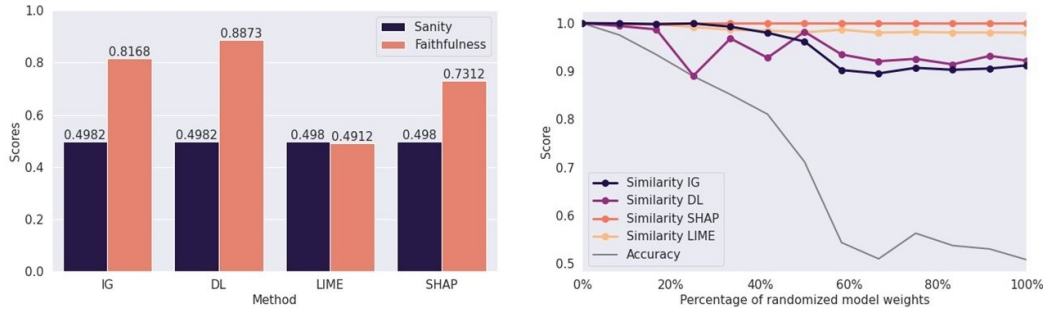


Figure 10. Quantitative saliency evaluation for the CNN trained via saliency-guided training on the CWRU Bearing dataset depicted as a comparison of all four post hoc saliency methods based on sanity and faithfulness scores (**right**) and drop in similarity between saliency maps when randomizing different percentages of network weights (**right**).

Figure 11 presents saliency maps produced by IG and DL for positive and negative samples in the CWRU dataset (a reproducibility comment: by rerunning the provided code, the user will obtain inverted heat maps, i.e., dark color on peaks and light color elsewhere). Both saliency methods consistently highlight the time series' peaks, i.e., the maximal values. This result aligns with the importance analysis of the nine extracted features reported in

Section 3. Further examination reveals a vast difference in the overall amplitude of the signal between the two classes. The maximum value thus effectively works as a proxy for the class-distinctive feature amplitude. From an explainability perspective, however, the resulting scores do not immediately offer any implication about the assumed latent features. This challenge will be more aggravated in real-world datasets with more complex latent models.

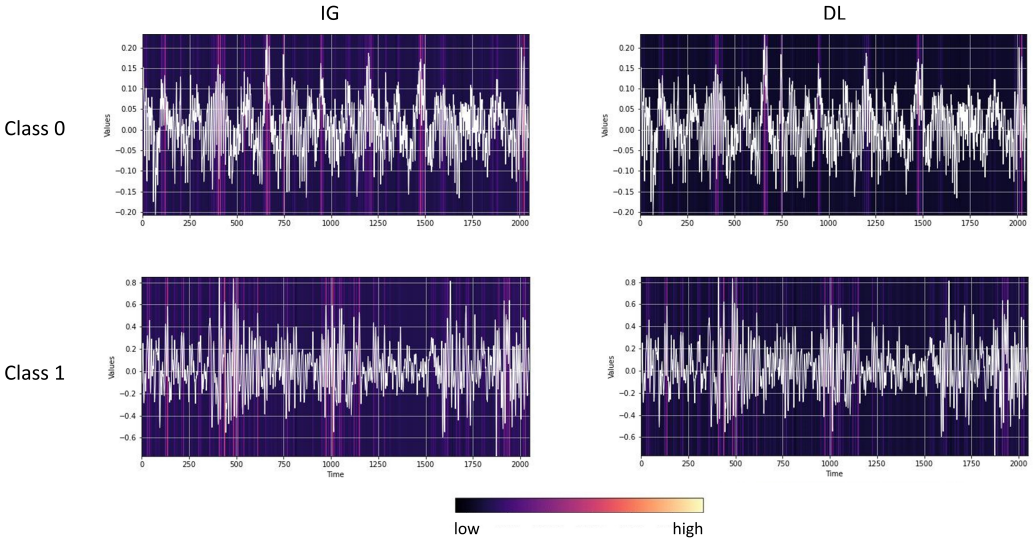


Figure 11. Explainability heat maps of Integrated Gradients (**left**) and Deep Lift (**right**) for each one sample of the positive (**top**) and the negative (**bottom**) class of the classification of the CNN trained via saliency-guided training on the CWRU Bearing dataset.

5. Discussion

In this section, we discuss and analyze the findings from the experiments regarding the classification performances and interpretation of the saliency methods.

5.1. Classification Performance–Explainability Relation

As discussed in Section 1, the desired classification performance does not necessarily couple with functional saliency scores. In other words, a classifier might achieve high accuracy scores, while the paired saliency method (either ante hoc or post hoc) outputs improper scores. Analyzing the obtained accuracy and F1 scores in Tables 3 and 4 alongside saliency results presented throughout Sections 4.2 and 4.3 empirically supports this cautionary message.

5.2. Effectiveness of the Tested Saliency Methods

Throughout our experiments, some state-of-the-art saliency methods underperformed even in the straightforward shapelet scenarios (Figure 4). In particular, LIME and SHAP methods failed to provide useful and enlightening scores in the shapelet experiments. This finding is in accordance with the results in [39]. This observation raises caution regarding the use of image saliency methods for time series data, as previously pointed out by Loeffler et al. [3], Schröder et al. [5], Parvatharaju et al. [40], Schlegel and Keim [41]. Based on our findings, the methods IG and DL can be expected to provide reliable performances if paired with effective classifiers. Nevertheless, as multiple explanations can coexist [42], we suggest considering the results of various saliency methods to find the best interpretation.

Regarding ante hoc methods, the classification performance scores reported in Tables 3 and 4 and saliency results in Figures 5 and 6 support the use of the AttentionLSTM method. These findings show that attention weights are highly correlated with gradient-based feature attribution scores, contradicting the findings of Jain and Wallace [43]. We conclude that the input-cell attention mechanism is a promising candidate for explana-

tion purposes in time series classification. This finding can be considered in the ongoing discussion of whether or not "attention is explanation".

5.3. Need for Development of Methods Able to Detect Latent Feature Saliency for Time Series Classification

The obtained results strongly emphasize the need for developing latent feature saliency methods for time series classification. Current saliency methods adopted from the image domain are unable to assign interpretable saliency scores in the presence of class-distinctive latent patterns. Similar to in [5,44], we hypothesize that due to the independence assumption between neighboring data points, neglecting the relative temporal ordering of input features, the model is not able to detect temporal dependencies.

In addition to IG and DL methods, which were also identified as "promising" in the previous work [5], the ante hoc AttentionLSTM also consistently yielded useful saliency maps for the shapelet- and latent-related problems. Nevertheless, the need to directly assess the latent parameters' importance remains since the expected proxy patterns in the time domain might be similar for two or more distinct latent parameters. For instance, the "promising" methods produce oscillating patterns for frequencies, which can be easily related to the frequency parameter after visual inspection. However, amplitude and phase shift saliency maps can be misinterpreted as implying positional information. For example, if the maps highlight a single high peak, one might mistake the peak as the presence of a unique shapelet instead of the implication of the amplitude feature. The latent feature saliency detection problem is likely to aggravate more complex latent-related problems in terms of interpretable associations with positional information.

5.4. Note on Sanity and Faithfulness Evaluation

Throughout all experiments, sanity scores remained relatively uninformative of the evaluated methods. For all methods across datasets and experiments, we achieved scores close to 0.5, which only partially supports the visual findings and results of qualitative evaluation. We hypothesize that this is a direct cause of employing the structural similarity index (SSIM) as a time series similarity measure (Equation (2)). Originally developed to measure image similarity, SSIM evaluates three main aspects of images: luminance, contrast, and structure. These concepts are not directly applicable to time series. Finally, to support our decision to consider faithfulness scores despite equally weak sanity scores, we note that these metrics are designed to be orthogonal [3]; hence, they can be used and analyzed separately.

5.5. Recommendations

We summarize our findings into a list of suggestions to follow step by step when selecting saliency methods for time series classification problems:

- **Step 0: Vanishing saliency problem in RNNs.** This work supports the findings of Bai et al. [36] and Cui et al. [45] regarding the inefficiency of LSTM models for time series classification. CNN and TCN models exhibit more reliable performance throughout the experiments. The vanishing saliency problem should be considered when employing recurrent neural networks, as Figure 5 depicts how gradient-based methods might not yield desired results.
- **Step 1: Visual evaluation of multiple saliency methods.** Even though some saliency methods outperformed the counterparts on average, we still observed varying scores throughout the experiments. Thus, we recommend utilizing different methods from different subcategories (e.g., gradient-based, model-agnostic) to capture different aspects of feature importance. A visual evaluation of the provided heat maps gives the first intuition on the salient domain, i.e., the time domain or latent space. If only a specific segment is highlighted as important, the methods will likely highlight the importance of a distinctive shapelet. If the heat maps are not interpretable, we recommend employing counterfactual methods.

As discussed in Section 3, for a given sample, counterfactual methods identify a similar but counterclass sample to compare and thus visually depict the important and class-distinctive features of the time series. Figure 12 presents examples of Native Guide's outputs in amplitude- and phase-shift-related experiments. As depicted, counterfactual samples (orange) correctly depict the difference in the peak for amplitude and phase shift difference. Upon visual inspection, counterfactual explanation methods might provide more insights than feature attribution methods.

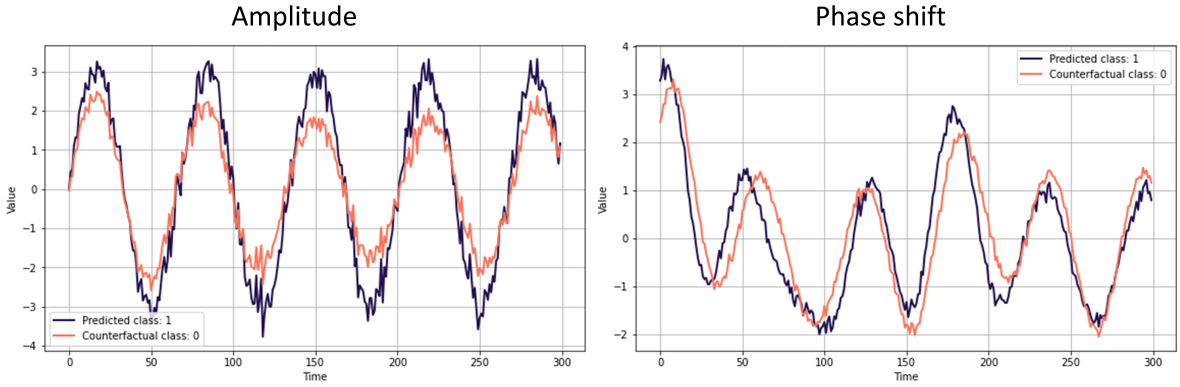


Figure 12. Explanations provided by the counterfactual method Native Guide when the class label depends on one of the latent features amplitude (**left**) or phase shift (**right**).

- **Step 2: Quantitative evaluation.** For an effective evaluation of the saliency maps' trustworthiness, we encourage using the faithfulness score. Furthermore, when evaluating the methods by sanity scores, we hypothesize that focusing on the structural similarity only will represent this property of time series more precisely than the complete structural similarity index, since the other similarity components are data-type-specific and only meaningful for image data. The proposed similarity measure then is

$$s(X, X') = \frac{\sigma_{XX'} + c}{\sigma_X \sigma_{X'} + c}$$

  where $\sigma_X, \sigma_{X'}$ represent the standard deviations of two normalized time series X and X', $\sigma_{XX'}$ is the correlation coefficient, and c is a constant, added to prevent numerical difficulties.
- **Step 3: Usage of ante hoc explainable methods.** Suppose the saliency methods achieve low faithfulness and sanity scores and their results remain uninformative after employing multiple methods and counterfactual explanations. In that case, we recommend considering the use of ante hoc methods. In particular, we found AttentionLSTM results to be consistently reliable throughout the experiments.
- **Step 4: Feature engineering.** Suppose the prior knowledge about the classification problem implies a specific latent model. In that case, an alternative solution is to utilize the prior knowledge directly by a preprocessing feature extraction step based on the latent model. The resulting engineered features should be employed instead of the raw time series. This way, saliency scores for the output of the feature extraction layer can be analyzed, and thus explainability is achieved. This approach is studied and advocated in time series literature for many latent models [46–49].

5.6. Future Work

To extend the empirical studies of our work, we suggest investigating different latent models for time series. Such work will improve the use of standard saliency methods for time series to achieve explainability. Furthermore, developing methods able to incorporate multiple feature spaces into the saliency analysis should be of high importance. If the target latent model is known, there is a potential for extracting latent saliency scores directly from positional scores. Our experiments show that the outputs of IG, DL, and the

input-cell attention mechanism are closely associated with the Fourier series latent model. Designing a mapping from positional to latent scores can serve as a baseline approach for future research.

The analyses in this paper were conducted at the sample level, i.e., saliency maps for individual samples were examined. An intraclass examination of the variability and variance of the generated maps might reveal further information about the classification mechanism. For example, heat maps might reveal several inconsistent patterns for individuals of the same class, which might indicate the existence of a latent model. For this purpose, quantitative evaluation such as intraclass and interclass robustness scores [3] can be used along with sanity and faithfulness scores.

6. Conclusions

Explainability of deep time series models is a trending and crucial research topic. For creating a wide acceptance of in AI models in real-world applications and aiding the identification of artifacts, interpretation and explanation of the black-box classifiers is essential. Multiple saliency methods, originally developed for the image domain, have been adapted to time series classification problems in the literature. These methods focus on positional information of the input features, providing spatial explanations. In the time series domain, however, class labels are not bound to depend on positional information but might be subject to a latent model. To the best of our knowledge, such settings have been neglected in the literature studying the performance and behavior of saliency methods so far. We extended our preliminary work in outlining this problem by showing that if the class label is associated with the latent features of the time series instead of the presence of a specific shape, commonly applied saliency methods do not provide accurate or human-interpretable explanations. We provided in-depth experiments, visually and quantitatively evaluating the resulting saliency maps on synthetic and real-world datasets.

Direct application of image explainability methods has been criticized in the literature before. We add a new aspect to this line of work as we reveal the cruciality of considering the latent models for time series alongside spatial information. We summarized our findings in a list of cautionary notes and suggestions to utilize these methods, as well as an outline for developing extensions of existing saliency methods that map time-step-wise saliency results to latent feature importance scores. In general, our work emphasizes the need for further research in the field of latent feature saliency detection for deep time series classification.

Author Contributions: Conceptualization, M.S., A.Z. and N.A.; methodology, M.S.; software, M.S. and A.Z.; formal analysis, A.Z.; validation, A.Z.; investigation, M.S.; resources, N.A.; writing—original draft preparation, M.S. and A.Z.; writing—review and editing, N.A.; visualization, M.S.; supervision, N.A.; project administration, N.A.; funding acquisition, N.A. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Bavarian Ministry for Economic Affairs, Regional Development and Energy, as part of a project to support the thematic development of the Fraunhofer Institute for Cognitive Systems.

Data Availability Statement: Publicly available datasets were analyzed in this study. These data can be found at https://engineering.case.edu/bearingdatacenter/download-data-file (accessed on 1 May 2022).

Acknowledgments: We thank Oleksandr Zadorozhnyi for his valuable support throughout the course of the research project. We thank Ruijie Chen and Adrian Schwaiger for proofreading the manuscript and providing instructive feedback.

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

Appendix A

The appendix presents supplementary notes and explanations on the implementation details and the synthetic data generation. It is directly adapted from our preliminary work. For further information please refer to [5].

Appendix A.1. Implementation Details

Since our goal was to investigate the performance of explainability methods when the class labels depend on latent features rather than positional information, all classifiers only consisted of one-layer networks. Furthermore, we did not employ dropout or any other form of additional regularization. By keeping the architecture simple, we intended to objectively evaluate and compare the explainability methods without the influence of optional variations, preventing overfitting or boosting the network performance.

All algorithms were implemented in the Python programming language. The classifiers were implemented using the deep learning library PyTorch [50] with the help of the wrapper PyTorch Lightning [51]. The authors of [6] provide a publicly available implementation of the input-cell attention module which we adapted. Hyperparameter optimization was performed through the library Optuna [52]. For the feature attribution techniques, the implementations from the PyTorch-based model interpretability library Captum [53] were employed. A GitHub repository containing the complete code base can be found at https://github.com/m-schroder/TSExplainability (accessed on 1 April 2023).

Appendix A.2. Synthetic Data Generation

We generated a total of ten experiments to understand the response of different saliency methods to different patterns. Our ten experiments included four experiments with temporal shapelet patterns, two with latent amplitude patterns, two with latent frequency patterns, and two with latent phase shift patterns. In each experiment, we built a dataset containing 2560 time series samples of equal length divided into two equally sized classes. For the shapelet experiments, each sample in the dataset was generated by first randomly sampling from the latent space and then applying a Fourier transformation to reconstruct its temporal signal from the latent space matrix. Afterward, the time series samples in class 1 were superimposed with a dominant shapelet pattern positioned either at a random location (experiment 1), the end (experiment 2), middle (experiment 3), or start (experiment 4) of the time series. For the latent feature experiments, the latent space matrices for class 0 were sampled from a latent space different than the latent space for class 1. The difference was defined in terms of sampling intervals for frequency, amplitude, or phase shift. For each experiment, the training, validation, and testing sets were generated by random sampling without replacement with a ratio of 80%, 10%, and 10%, respectively.

For assigning the labels to the data samples, we induced a simple linear relation between the latent or temporal patterns and the class labels. In the latent scenarios, two classes were distinguishable using a single decision boundary defined as $Z^* = const.$, meaning that only one latent feature is class-distinctive. Likewise, in shapelet-related scenarios, the presence or absence of a specific shapelet decides the label of the data. This allowed us to study the latent features individually and in a controlled manner. In such settings, potential poor results can be confidently attributed to the intrinsic weakness of the saliency methods rather than inappropriate classifiers. The data generation mechanism and the resulting datasets are presented and described in detail in the following.

Based on the Fourier series latent model, a time series $x_t, t = 1, ..., T$ was modeled as

$$\begin{aligned} x_t &= a_0 + \sum_{n=1}^{\infty} a_n \cos(\omega_n t) + \sum_{n=1}^{\infty} b_n \sin(\omega_n t) \\ &= a_0 + \sum_{n=1}^{\infty} A_n \cos(\omega_n t + \phi_n) \\ &= a_0 + \sum_{n=1}^{\infty} A_n \sin(\omega_n t + \phi_n + \frac{\pi}{2}). \end{aligned}$$

To simulate data, we let $\tilde{n}$ represent the number of amplitudes present in the series, i.e., $\forall i > \tilde{n}, A_i = 0$. For simplicity, we considered centered stationary periodic time series in the data generation process, i.e., $a_0 = 0$. In this case, the value at every time step t was calculated as

$$x_t = \sum_{i=1}^{\tilde{n}} A_i \sin(\omega_i t + \phi_i + \frac{\pi}{2}). \tag{A1}$$

We refer to the notions amplitude A, frequency ω, and phase shift ϕ as "concepts". The separate Fourier coefficients A_i, ω_i, ϕ_i for $i = 1, ..., \tilde{T}$ are referred to as latent features. The latent features frequency ω_i and phase shift ϕ_i were each sampled from a uniform distribution. The sampling intervals were chosen with respect to the specific intention in the experiment design. To simulate the amplitude parameters A_i, a dominant amplitude A_1 was sampled. The next amplitudes were calculated considering an exponential decay with a fixed rate *dec*:

$$A_i = A_1 \exp(-i \cdot dec), \quad i = 1, ..., \tilde{n}.$$

This makes the first frequency, i.e., ω_1, the dominant frequency of the Fourier series. Throughout the experiments, all time series were generated with an equal length of 300 time steps, i.e., $T = 300$.

For assigning class labels to the time series samples, we considered the following two scenarios.

- Scenario 1: Label based on the presence of a shapelet

For assigning shape-based labels to the time series, a shapelet was inserted at a random or fixed position into all time series $X \in D$ belonging to one class. The shapelet was a second simulated Fourier series of length $l \leq T$, where $l = \text{window-ratio} \cdot T$ for a chosen window ratio. We defined the sampling intervals for the latent features of the shapelet to be nonintersecting with the sampling intervals of the latent features of the original time series X. The resulting shapelet replaced the original time series in the interval $[j, j + l]$, where

$$j \sim \mathcal{U}(1, T - l).$$

- Scenario 2: Label based on differences in the latent features

Following the investigation of the effectiveness of explainability methods for latent features, we introduced a second simulation scenario where the labels depended on a difference in the sampling distribution of latent features of the time series. This scenario highlights the main focus of this project, and represents our novel view of explainability methods for time series. Similar to the first scenario, the time series were sampled as discretized Fourier series with latent variables ω, A and ϕ. The latent-dependency was induced as follows:

1. Two normal distributions with different means (based on Table A1) were selected for classes 0 and 1. For positive parameters, the distributions were log-normal.
2. Per each class, $N/2$ Fourier parameters were sampled from the given distributions.
3. The rest of the parameters were sampled from the same distribution for both classes.

4. Sampled parameters were given to the deterministic Fourier series in Equation (A1) to generate the temporal samples. Rows were then labeled with the associated class, from the corresponding distribution of which the informative parameters were sampled.

Based on the data generation method described above, we designed ten different mechanisms. In four experiments, the label was based on a shapelet at random and fixed positions in the start, middle, and end of the time series, respectively. Each two datasets were designed such that the label was based on one of the latent Fourier concepts. For the scope of this project, we designed the datasets to only include one label-making feature at a time. Table A2 lists the parameters and algorithms for assigning labels to each sample. In Table A1, the parameters used for sampling the Fourier series are presented. The complete simulation code base can be found in the GitHub repository at https://github.com/m-schroder/TSXplainability (accessed on 1 April 2023).

Table A1. Overview of simulation parameters of the Fourier series. If two entries are present in one cell, each of the classes were sampled from different distributions. The first entry in each cell corresponds to the sampling parameter of class 0, and the second entry to class 1.

Exp.	Number of Sines	Freq. Low	Freq. High	Phase Low	Phase High	Dominant Amplitude	Decay Rate	Noise Ratio
1	10	$\frac{\pi}{300}$	$\frac{\pi}{60}$	$\frac{-\pi}{4}$	$\frac{\pi}{4}$	1	0.3	0.1
2	10	$\frac{\pi}{300}$	$\frac{\pi}{20}$	$\frac{-\pi}{4}$	$\frac{\pi}{4}$	1	0.3	0.1
3	10	$\frac{\pi}{300}$	$\frac{\pi}{20}$	$\frac{-\pi}{4}$	$\frac{\pi}{4}$	1	0.3	0.1
4	10	$\frac{\pi}{300}$	$\frac{\pi}{20}$	$\frac{-\pi}{4}$	$\frac{\pi}{4}$	1	0.3	0.1
5	10/10	$\frac{\pi}{300}$ / $\frac{\pi}{100}$	$\frac{\pi}{20}$ / $\frac{\pi}{2}$	$\frac{-\pi}{4}$ / $\frac{-\pi}{4}$	$\frac{\pi}{4}$ / $\frac{\pi}{4}$	1/1	0.3/0.3	0.1/0.1
6	1/1	$\frac{\pi}{300}$ / $\frac{\pi}{100}$	$\frac{\pi}{20}$ / $\frac{\pi}{2}$	$\frac{-\pi}{4}$ / $\frac{-\pi}{4}$	$\frac{\pi}{4}$ / $\frac{\pi}{4}$	1/1	0.3/0.3	0.1/0.1
7	1/1	$\frac{\pi}{300}$ / $\frac{\pi}{300}$	$\frac{\pi}{20}$ / $\frac{\pi}{20}$	0/ $\frac{-\pi}{4}$	$\frac{\pi}{4}$ / $\frac{\pi}{2}$	1/1	0.3/0.3	0.1/0.1
8	10/10	$\frac{\pi}{300}$ / $\frac{\pi}{300}$	$\frac{\pi}{20}$ / $\frac{\pi}{20}$	0/ $\frac{-\pi}{4}$	$\frac{\pi}{4}$ / $\frac{\pi}{2}$	1/1	0.3/0.3	0.1/0.1
9	10/10	$\frac{\pi}{300}$ / $\frac{\pi}{300}$	$\frac{\pi}{20}$ / $\frac{\pi}{20}$	0/ $\frac{-\pi}{4}$	$\frac{\pi}{4}$ / $\frac{\pi}{4}$	1/3	0.3/0.3	0.1/0.1
10	1/1	$\frac{\pi}{300}$ / $\frac{\pi}{300}$	$\frac{\pi}{20}$ / $\frac{\pi}{20}$	$\frac{-\pi}{4}$ / $\frac{-\pi}{4}$	$\frac{\pi}{4}$ / $\frac{\pi}{4}$	1/3	0.3/0.3	0.1/0.1

Table A2. Label-making features per experiment. The overlapping ranges refer to the sampling intervals for frequency and phase shift.

Experiment	Label Feature	Description of Shapelet
1	Shapelet	Random position, window length of 0.2 × sequence length
2	Shapelet	Fixed position, last 0.2 × sequence length time steps
3	Shapelet	Fixed position, starting at time step 0.4 × sequence length with window length 0.2 × sequence length
4	Shapelet	Fixed position, first 0.2 × sequence length time steps
5	Frequency	Overlapping frequency ranges
6	Frequency	Overlapping frequency ranges
7	Phase shift	Nonoverlapping phase shift ranges
8	Phase shift	Nonoverlapping phase shift ranges
9	Amplitude	Different dominant amplitude
10	Amplitude	Different dominant amplitude

References

1. Guidotti, R.; Monreale, A.; Ruggieri, S.; Turini, F.; Giannotti, F.; Pedreschi, D. A Survey of Methods for Explaining Black Box Models. *ACM Comput. Surv.* **2018**, *51*, 1–42. [CrossRef]
2. Ismail, A.A.; Gunady, M.K.; Corrada Bravo, H.; Feizi, S. Benchmarking Deep Learning Interpretability in Time Series Predictions. In Proceedings of the 34th Conference on Neural Information Processing Systems (NeurIPS), Online, 6–12 December 2020; pp. 6441–6452.
3. Loeffler, C.; Lai, W.C.; Eskofier, B.; Zanca, D.; Schmidt, L.; Mutschler, C. Don't Get Me Wrong: How to apply Deep Visual Interpretations to Time Series. *arXiv* **2022**, arXiv:2203.07861. [CrossRef]
4. Schlegel, U.; Oelke, D.; Keim, D.A.; El-Assady, M. An Empirical Study of Explainable AI Techniques on Deep Learning Models For Time Series Tasks. In Proceedings of the Pre-Registration Workshop NeurIPS (2020), Vancouver, BC, Canada, 11 December 2020.
5. Schröder, M.; Zamanian, A.; Ahmidi, N. Post-hoc Saliency Methods Fail to Capture Latent Feature Importance in Time Series Data. In Proceedings of the ICLR 2023 Workshop on Trustworthy Machine Learning for Healthcare, Online, 4 May 2023.
6. Ismail, A.A.; Gunady, M.; Pessoa, L.; Corrada Bravo, H.; Feizi, S. Input-Cell Attention Reduces Vanishing Saliency of Recurrent Neural Networks. In Proceedings of the 32nd Conference on Neural Information Processing Systems (NeurIPS 2019), Vancouver, BC, Canada, 8–14 December 2019.
7. Ye, L.; Keogh, E. Time series shapelets: a new primitive for data mining. In Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD09), Paris, France, 28 June–1 July 2009; pp. 947–956.
8. Simonyan, K.; Vedaldi, A.; Zisserman, A. Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps. *arXiv* **2014**, arXiv:1312.6034. [CrossRef]
9. Shrikumar, A.; Greenside, P.; Shcherbina, A.; Kundaje, A. Not Just a Black Box: Learning Important Features through Propagating Activation Differences. In Proceedings of the 33rd International Conference on Machine Learning (ICML'16), New York, NY, USA, 19–24 June 2016; Volume 48.
10. Zeiler, M.D.; Fergus, R. Visualizing and understanding convolutional networks. In Proceedings of the Computer Vision (ECCV 2014), Zurich, Switzerland, 6–12 September 2014; pp. 818–833.
11. Springenberg, J.T.; Dosovitskiy, A.; Brox, T.; Riedmiller, M.A. Striving for Simplicity: The All Convolutional Net. *arXiv* **2015**, arXiv:1412.6806. [CrossRef]
12. Smilkov, D.; Thorat, N.; Kim, B.; Kim, B.; Viégas, F.B.; Wattenberg, M. SmoothGrad: Removing noise by adding noise. *arXiv* **2017**, arXiv:1706.03825. [CrossRef]
13. Fong, R.; Patrick, M.; Vedaldi, A. Understanding Deep Networks via Extremal Perturbations and Smooth Masks. In Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Republic of Korea, 27 Ocotber–2 November 2019; pp. 2950–2958.
14. Shrikumar, A.; Greenside, P.; Kundaje, A. Learning Important Features Through Propagating Activation Differences. In Proceedings of the 34th International Conference on Machine Learning (ICML'17), Sydney, Australia, 6–11 August 2017; Volume 70.
15. Sundararajan, M.; Taly, A.; Yan, Q. Axiomatic Attribution for Deep Networks. In Proceedings of the 34th International Conference on Machine Learning (ICML'17), Sydney, Australia, 11–15 August 2017; Volume 70, pp. 3319–3328.
16. Bastings, J.; Filippova, K. The elephant in the interpretability room: Why use attention as explanation when we have saliency methods? In Proceedings of the 2020 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP, Online, 11–12 November 2020.
17. Bach, S.; Binder, A.; Montavon, G.; Klauschen, F.; Müller, K.; Wojciech, S. On Pixel-Wise Explanations for Non-Linear Classifier Decisions by Layer-Wise Relevance Propagation. *PLoS ONE* **2015**, *10*, e130140. [CrossRef] [PubMed]
18. Montavon, G.; Lapuschkin, S.; Binder, A.; Samek, W.; Müller, K.R. Explaining Nonlinear Classification Decisions with Deep Taylor Decomposition. *Pattern Recognit.* **2017**, *65*, 211–222. [CrossRef]
19. Carrillo, A.; Cantú, L.F.; Noriega, A. Individual Explanations in Machine Learning Models: A Survey for Practitioners. *arXiv* **2021**, arXiv:2104.04144. [CrossRef]
20. Petsiuk, V.; Das, A.; Saenko, K. RISE: Randomized Input Sampling for Explanation of Black-box Models. In Proceedings of the 29th British Machine Vision Conference (BMVC), Newcastle, UK, 3–6 September 2018.
21. Fong, R.C.; Vedaldi, A. Interpretable Explanations of Black Boxes by Meaningful Perturbation. In Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017; pp. 3449–3457.
22. Ribeiro, M.T.; Singh, S.; Guestrin, C. "Why Should I Trust You?" Explaining the Predictions of Any Classifier. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD 16), New York, NY, USA, 13–17 August 2016; pp. 1135–1144.
23. Datta, A.; Sen, S.; Zick, Y. Algorithmic Transparency via Quantitative Input Influence: Theory and Experiments with Learning Systems. In Proceedings of the 2016 IEEE Symposium on Security and Privacy (SP), San Jose, CA, USA, 23–26 May 2016; pp. 598–617.
24. Lipovetsky, S.; Conklin, M. Analysis of regression in game theory approach. *Appl. Stoch. Model. Bus. Ind.* **2001**, *17*, 319–330. [CrossRef]
25. Štrumbelj, E.; Kononenko, I. Explaining Prediction Models and Individual Predictions with Feature Contributions. *Knowl. Inf. Syst.* **2014**, *41*, 647–665. [CrossRef]

26. Shapley, L.S. A value for n-person games. In *Contributions to the Theory of Games II*; Princeton University Press: Princeton, NJ, USA, 1953; pp. 307–317.
27. Lundberg, S.M.; Lee, S.I. A Unified Approach to Interpreting Model Predictions. In Proceedings of the 31st International Conference on Neural Information Processing Systems (NIPS'17), Long Beach, CA, USA, 4–9 December 2017; pp. 4768–4777.
28. Guidotti, R.; Monreale, A.; Spinnato, F.; Pedreschi, D.; Giannotti, F. Explaining Any Time Series Classifier. In Proceedings of the 2020 IEEE Second International Conference on Cognitive Machine Intelligence (CogMI), Atlanta, GA, USA, 28–31 October 2020; pp. 167–176.
29. Karlsson, I.; Rebane, J.; Papapetrou, P.; Gionis, A. Locally and Globally Explainable Time Series Tweaking. *Knowl. Inf. Syst.* **2020**, *62*, 1671–1700. [CrossRef]
30. Wang, Z.; Samsten, I.; Mochaourab, R.; Papapetrou, P. Learning Time Series Counterfactuals via Latent Space Representations. In Proceedings of the 24th International Conference on Discovery Science (DS 2021), Halifax, NS, Canada, 11–13 October 2021; pp. 369–384.
31. Ates, E.; Aksar, B.; Leung, V.J.; Coskun, A.K. Counterfactual Explanations for Multivariate Time Series. In Proceedings of the 2021 International Conference on Applied Artificial Intelligence (ICAPAI), Halden, Norway, 19–21 May 2021; pp. 1–8.
32. Delaney, E.; Greene, D.; Keane, M.T. Instance-Based Counterfactual Explanations for Time Series Classification. In Proceedings of the Case-Based Reasoning Research and Development: 29th International Conference (ICCBR 2021), Salamanca, Spain, 13–16 September 2021; pp. 32–47.
33. Hochreiter, S.; Schmidhuber, J. Long Short-Term Memory. *Neural Comput.* **1997**, *9*, 1735–1780. [CrossRef] [PubMed]
34. Le Cun, Y.; Jackel, L.; Boser, B.; Denker, J.; Graf, H.; Guyon, I.; Henderson, D.; Howard, R.; Hubbard, W. Handwritten digit recognition: Applications of neural network chips and automatic learning. *IEEE Commun. Mag.* **1989**, *27*, 41–46. [CrossRef]
35. Ismail, A.A.; Corrada Bravo, H.; Feizi, S. Improving Deep Learning Interpretability by Saliency Guided Training. In Proceedings of the 35th Conference on Neural Information Processing Systems (NeurIPS2021), Online, 6–14 December 2021; pp. 26726–26739.
36. Bai, S.; Kolter, J.Z.; Koltun, V. An Empirical Evaluation of Generic Convolutional and Recurrent Networks for Sequence Modeling. *arXiv* **2018**, arXiv:1803.01271. [CrossRef]
37. CWRU Bearing Dataset. Available online: https://engineering.case.edu/bearingdatacenter/download-data-file (accessed on 10 October 2022).
38. Wang, Z.; Bovik, A.; Sheikh, H.; Simoncelli, E. Image quality assessment: from error visibility to structural similarity. *IEEE Trans. Image Process.* **2004**, *13*, 600–612. [CrossRef] [PubMed]
39. Neely, M.; Schouten, S.F.; Bleeker, M.J.R.; Lucic, A. Order in the Court: Explainable AI Methods Prone to Disagreement. In Proceedings of the ICML Workshop on Theoretic Foundation, Criticism, and Application Trend of Explainable AI, Virtual Event, 23 July 2021.
40. Parvatharaju, P.S.; Doddaiah, R.; Hartvigsen, T.; Rundensteiner, E.A. Learning Saliency Maps to Explain Deep Time Series Classifiers. In Proceedings of the 30th ACM International Conference on Information & Knowledge Management, Virtual Event, 1–5 November 2021; pp. 1406–1415.
41. Schlegel, U.; Keim, D.A. Time Series Model Attribution Visualizations as Explanations. In Proceedings of the 2021 IEEE Workshop on TRust and EXpertise in Visual Analytics (TREX), New Orleans, LA, USA, 24–25 October 2021; pp. 27–31.
42. Wiegreffe, S.; Pinter, Y. Attention is not Explanation. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), Hong Kong, China, 3–7 November 2019; pp. 11–20.
43. Jain, S.; Wallace, B.C. Attention is not Explanation. In Proceedings of the 2019 Annual Conference of the North American Chapter of the Association for Computational Linguistics, Minneapolis, MN, USA, 2–7 June 2019.
44. Lim, B.; Arik, S.; Loeff, N.; Pfister, T. Temporal Fusion Transformers for Interpretable Multi-horizon Time Series Forecasting. *Int. J. Forecast.* **2021**, *37*, 1748–1764. [CrossRef]
45. Cui, Z.; Chen, W.; Chen, Y. Multi-scale convolutional neural networks for time series classification. *arXiv* **2016**, arXiv:1603.06995. [CrossRef]
46. Bagnall, A.; Lines, J.; Hills, J.; Bostrom, A. Time-series classification with COTE: The collective of transformation-based ensembles. *IEEE Trans. Knowl. Data Eng.* **2015**, *27*, 2522–2535. [CrossRef]
47. Tseng, A.; Shrikumar, A.; Kundaje, A. Fourier-transform-based attribution priors improve the interpretability and stability of deep learning models for genomics. In Proceedings of the 34th Conference on Neural Information Processing Systems (NeurIPS 2020), Online, 6–12 December 2020; pp. 1913–1923.
48. Kazemi, S.M.; Goel, R.; Eghbali, S.; Ramanan, J.; Sahota, J.; Thakur, S.; Wu, S.; Smyth, C.; Poupart, P.; Brubaker, M. Time2vec: Learning a vector representation of time. *arXiv* **2019**, arXiv:1907.05321. [CrossRef]
49. Rangapuram, S.S.; Seeger, M.W.; Gasthaus, J.; Stella, L.; Wang, Y.; Januschowski, T. Deep state space models for time series forecasting. In Proceedings of the 32nd Conference on Neural Information Processing Systems (NeurIPS 2018), Montréal, QC, Canada, 2–8 December 2018.
50. Paszke, A.; Gross, S.; Massa, F.; Lerer, A.; Bradbury, J.; Chanan, G.; Killeen, T.; Lin, Z.; Gimelshein, N.; Antiga, L.; et al. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In Proceedings of the 32nd Conference on Neural Information Processing Systems (NeurIPS 2019), Vancouver, BC, Canada, 8–14 December 2019; pp. 8024–8035.
51. Falcon, W. *PyTorch Lightning*. Available online: https://github.com/Lightning-AI/lightning (accessed on 5 June 2022).

52. Akiba, T.; Sano, S.; Yanase, T.; Ohta, T.; Koyama, M. Optuna: A Next-generation Hyperparameter Optimization Framework. In Proceedings of the 25st ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '19), Anchorage, AK, USA, 4–8 August 2019.
53. Kokhlikyan, N.; Miglani, V.; Martin, M.; Wang, E.; Alsallakh, B.; Reynolds, J.; Melnikov, A.; Kliushkina, N.; Araya, C.; Yan, S.; et al. Captum: A unified and generic model interpretability library for PyTorch. *arXiv* **2020**, arXiv:2009.07896. [CrossRef]