A Model-Agnostic Approach to Quantifying the Informativeness of Explanation Methods for Time Series Classification

Thu Trang Nguyen(✉), Thach Le Nguyen(✉), and Georgiana Ifrim(✉)

School of Computer Science, University College Dublin, Dublin, Ireland
thu.nguyen@ucdconnect.ie, {thach.lenguyen,georgiana.ifrim}@ucd.ie

Abstract. In this paper we focus on explanation methods for time series classification. In particular, we aim to quantitatively assess and rank different explanation methods based on their informativeness. In many applications, it is important to understand which parts of the time series are informative for the classification decision. For example, while doing a physio exercise, the patient receives feedback on whether the execution is correct or not (classification), and if not, which parts of the motion are incorrect (explanation), so they can take remedial action. Comparing explanations is a non-trivial task. It is often unclear if the output presented by a given explanation method is at all informative (i.e., relevant for the classification task) and it is also unclear how to compare explanation methods side-by-side. While explaining classifiers for image data has received quite some attention, explanation methods for time series classification are less explored. We propose a model-agnostic approach for quantifying and comparing different saliency-based explanations for time series classification. We extract importance weights for each point in the time series based on learned classifier weights and use these weights to perturb specific parts of the time series and measure the impact on classification accuracy. By this perturbation, we show that explanations that actually highlight discriminative parts of the time series lead to significant changes in classification accuracy. This allows us to objectively quantify and rank different explanations. We provide a quantitative and qualitative analysis for a few well known UCR datasets.

Keywords: Time series classification · Explainable machine learning · Evaluation · Comparing explanations · Saliency maps

1 Introduction

In the last decade, machine learning systems have become more ubiquitous and highly integrated with our daily life due to the increased availability of personal computing and wearable devices. Machine learning methods, including those dealing with time series data, have grown in complexity, performance,

V. Lemaire et al. (Eds.): AALTD 2020, LNAI 12588, pp. 77–94, 2020.
https://doi.org/10.1007/978-3-030-65742-0_6

and impact. Among many applications [22,23], Time Series Classification (TSC) algorithms are more commonly used nowadays in human activity recognition [3] tasks, which often require explanations for certain critical decisions [6,14]. This explanation is usually presented in the form of a *saliency map* [1], highlighting the parts of the time series which are *informative* for the classification decision.

Recent efforts both in designing intrinsically explainable machine learning algorithms, as well as building post-hoc methods explaining black-box algorithms, have gained significant attention [20,24,28,31,36]; yet, these efforts present us with a new challenge: *How to assess and objectively compare such methods?* In other words, if two methods give different explanations, e.g., two different saliency maps, which method and explanation should we trust more? Assessing and comparing explanations is a non-trivial problem and requires a solution.

In this work, we consider explanation and its informativeness within a defined computational scope, in which a more informative explanation means a higher capability to influence classifiers to correctly identify a class. With this definition, we aim to objectively quantify and compare the informativeness of different explanations, hence alleviating the need for, or at least reducing some of the effort for, conducting user-studies which are very difficult to reproduce [9]. We focus on quantitatively evaluating explanation methods for the TSC task. In this paper, we only consider methods that produce explanations in the form of saliency maps. In particular, we introduce a model-agnostic methodology to quantitatively assess and rank different saliency maps based on a concept of informativeness which we define in this paper.

In our experiments, we consider three popular and recent saliency-based explanation methods representing two approaches for generating explanations (i.e., model internals from an *intrinsically* explainable model and *post-hoc* explanations) and two scopes of explanations (i.e., *global* explanation for the entire dataset and *local* explanation for the prediction on a specific test example). As illustrated in Fig. 1, such methods often produce significantly different explanations and subsequently call for a methodology and evaluation measure for comparison.

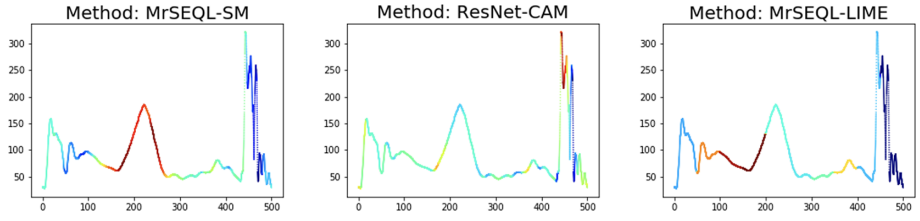

Fig. 1. Saliency map explanations for a motion time series obtained using different explanation methods. In this figure, the most discriminative parts are colored in deep red and the most non-discriminative parts are colored in deep blue. (Color figure online)

Our methodology stems from the idea that highly informative explanations correctly identify areas of the time series that are relevant for a classifier, thus perturbing such parts will result in a reduced capability of classifiers for making correct decisions. We focus on two scenarios in which the informativeness of explanation methods should be evaluated: when a *single explanation method* is presented and we want to know whether such method is actually informative, and when *multiple explanation methods* are presented and we wish to compare them.

The **evaluation of a single explanation method** compares the changes in classification performance under two settings: when the time series perturbation happens at either the discriminative and non-discriminative parts, as detected by the explanation method to be evaluated. If the method is informative, we expect that the accuracy will drop more significantly when the discriminative parts are perturbed. In contrast, for the **comparison of multiple explanation methods** we compare the classification performance only when the perturbation happens at the discriminative parts of the time series. The more informative method should trigger a more significant drop in accuracy. In both scenarios, we quantify the effect of change in performance by an **evaluation measure** which estimates the difference of the changes across multiple thresholds for identifying discriminative parts. We verify our experiment results with a sanity-check step, in which we visualize and compare the saliency maps for multiple examples of a dataset with known ground truth.

Our experiments show that explanations actually highlighting discriminative parts of the time series (i.e., that are more informative) lead to significant changes in classification accuracy, reflected by our proposed evaluation measure for quantifying this behaviour. While there is no one-size-fits-all ideal explanation method that perfectly highlights the discriminative parts in all TSC problems and datasets, our evaluation methodology provides a guideline to objectively evaluate any potential TSC saliency-based explanation methods for specific use cases, and safely reject those that fail both of the aforementioned steps.

Our main contributions are as follows:

1. We propose a new methodology and evaluation measure designed to enable us to objectively quantify and compare the informativeness of different explanation methods for the TSC task.
2. We empirically analyse our evaluation methodology with three representative explanation methods and three "referee" TSC algorithms.
3. We provide a discussion of the quantitative and qualitative assessment of various TSC explanation methods across several TSC benchmark datasets, and propose some directions for future work in this area.

2 Related Work

2.1 Time Series Classification

Although many TSC studies have been published in the past, very few of them focused on explainability. The list of TSC algorithms typically starts with the

famous baselines 1NN-Euclidean and 1NN-DTW [16]; both are a combination of a nearest neighbour classifier and a distance measure. In most of the literature in this field, they are the benchmark classifiers due to their simplicity and accuracy. For this type of classifier, one can explain the classification decision on a time series by examining its nearest neighbour in the training data. However, we are not aware of any TSC studies that have investigated this prospect in depth.

Recent TSC papers have explored many other directions which include interval-based, shapelet-based, dictionary-based, and autocorrelation-based methods [4]. Nevertheless, only shapelet-based and dictionary-based classifiers in this group have shown the potential for explainability. Shapelet-based classifiers revolve around the concept of shapelets, segments of time series which can generalize or discriminate the classes. Examples of shapelet-based classifiers include Shapelet Transform [5] and Learning Shapelets [11]. It is theoretically possible to use shapelets as an explanation mechanism for these classifiers, but this was not considered in depth in previous studies, beyond a high-level qualitative discussion. On the other hand, dictionary-based classifiers have made significant breakthroughs with the introduction of SAX-VSM [29], BOSS [26], WEASEL [27], and Mr-SEQL [18]. The SAX-VSM work, although inferior to the latter in terms of accuracy, presented some attempts to explain the classifier by highlighting the highest-scored subsequences of the time series, which is a form of saliency mapping. Similar bids to explain the classification decision were made by SAX-VFSEQL [21] and its successor Mr-SEQL which are also classifiers from this group. Two other important families of TSC algorithms are deep learning, e.g., ResNet [15] and ensemble methods, e.g., HIVE-COTE [4]; they are generally well-known for being highly accurate. While not many attempts have been made to explain ensemble TSC methods, deep neural networks with convolutional layers can produce a saliency map explanation of the time series classification by using the Class Activation Map (CAM) method [36]. This option was explored in [35] and [15].

2.2 Explanation in Time Series Classification

Saliency Maps. Saliency mapping is a visualisation approach to highlight parts of a time series that are important for the TSC model in making a prediction. Such mappings are often produced by matching a time series with a *vector of weights* (w) using a color map. This vector of weights has a corresponding weight value for each data point in the time series. The saliency map (characterized by the vector of explanation weights) and the method to produce the vector of weights for the mapping, are hereafter respectively called TSC *explanation* and *explanation method*. Figure 2 (bottom right) shows an example saliency map in which the vector of explanation weights is matched to the original time series using a heatmap. The explanation weight is *non-negative* since its magnitude reflects the discriminative *power* of the associated data point in the time series.

In this work, we explore three TSC explanation methods using the concept of explanation seen as a vector of weights: **MrSEQL-SM**, **CAM**, and **LIME**. These methods represent three distinct approaches for producing saliency-based

explanations in the form of highlighting the discriminative parts of a time series. We summarize the properties of these explanation methods in Table 1.

Table 1. Summary of TSC explanation methods properties.

Explanation method	Type	Model-specific	Explanation scope
MrSEQL-SM	Intrinsic	Yes	Global
CAM	Post-hoc	Yes	Local
LIME	Post-hoc	No	Local

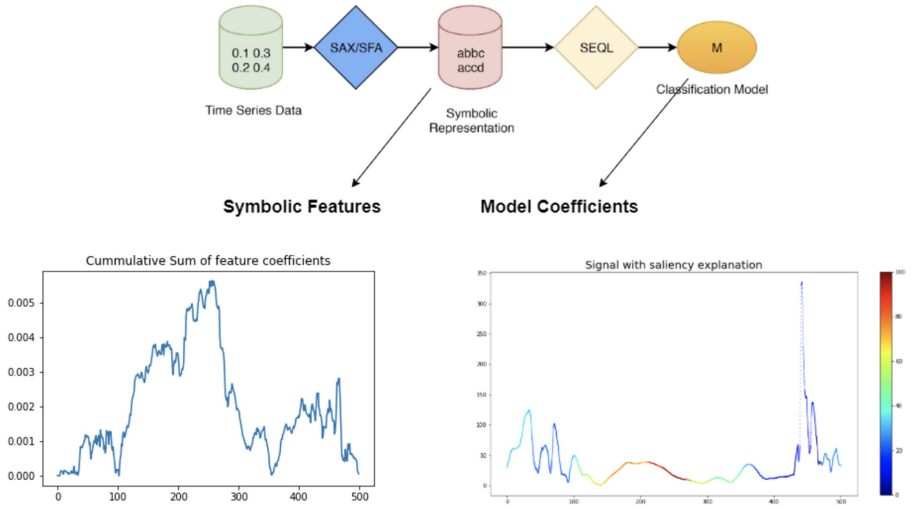

Fig. 2. The saliency map explanation MrSEQL-SM obtained from the MrSEQL linear classifier.

MrSEQL-SM. Mr-SEQL [18] is an efficient time series classification algorithm that is intrinsically explainable, since it learns a linear model for classification. The algorithm converts the numeric time series vector into strings, e.g., by using the SAX [19] transformation with varying parameters to create multiple symbolic representations of the time series. The symbolic representations are then used as input for SEQL [13], a sequence learning algorithm, to select the most discriminative subsequence features for training a classifier using logistic regression. The symbolic features combined with the classifier weights learned by logistic regression make this classification algorithm explainable (Fig. 2). For a time series, the explanation weight of each data point is the accumulated weight of the SAX features that it maps to. These weights can be mapped back to the original time series to create a saliency map to highlight the time series parts important for

the classification decision. We call the saliency map explanation obtained this way, MrSEQL-SM. For using the weight vector from MrSEQL-SM, we take the absolute value of weights to obtain a vector of non-negative weights.

CAM. CAM [36] is a post-hoc explanation method commonly used to explain deep networks that have convolutional layers and a global average pooling (GAP) layer just before the final output layer. With this very specific architecture, the weights from the GAP layer can be used to reveal the parts of the time series that are important for the classifier to make a prediction. Thus, these weights are used to produce the saliency mapping of the weight vector to the original time series.

LIME. LIME [24] is a post-hoc explanation method that can be used to explain a black-box classifier's decision for a local example. To explain the local decision of a model, LIME perturbs that local example (X) multiple times and weighs the perturbed examples (X') by their proximity to X. It finally gets the prediction of the original model for X' and fits an explainable classifier, usually a linear model, to estimate the local decision boundary of the original classifier. LIME does not explain the classification decision globally, but only locally around a specific example. Due to this aspect, this explanation method is computationally expensive as it has to be trained for each test example, hence we evaluate it with only a subset of the datasets used for experiments.

2.3 Explanation in Other Machine Learning Domains

Interpretable machine learning is a rapidly growing area of machine learning research. Besides inherently interpretable models (such as linear regression and decision trees), there are techniques developed for explaining complex machine learning models, ranging from feature-based [2,10], local surrogate [24], to example-based explanations [25,34]. In the context of this work, we focus on studying explanation methods within the scope of saliency map explanation. Saliency maps were originally used in Computer Vision to highlight certain properties of the pixels in an image [30]. The success of black-box deep neural networks in image recognition tasks [12,17,33] paved the way for the growth of post-hoc explanation methods designed to explain deep learning models. Notable works of this family include Class Activation Map [36], Gradient-weighted Class Activation Map (Grad-CAM) [28] and Guided Backpropagation (GBP) [32]. This growing list of techniques to explain deep learning models poses the challenge of assessing the quality of these explanation methods. The work by [1] attempts to visually and statistically evaluate the quality of a few saliency-based explanations for deep learning models, by tracking the changes of the saliency maps when the model parameters and test labels are randomized. Interestingly, they show that some explanation methods provide unchanged explanations even when both the model parameters and the data are random.

3 Research Methods

In this section, we first describe the perturbation process we propose for evaluating the informativeness of a TSC explanation method. We then outline two perturbation approaches and finally introduce a novel measure to quantify and compare the informativeness of TSC explanation methods.

3.1 Explanation-Driven Perturbation

The goal of providing a TSC explanation is to focus on the discriminative parts of the time series. If the explanation is truly informative, it should point out those parts of the time series that are most relevant for the classification decision. Consequently, if we perturb these parts, then the time series will be harder to classify. The more informative the explanation, the higher the decrease in accuracy we expect, since we knock out the important information contained in the data, for making a classification decision. In this section we provide an approach for quantifying the informativeness of an explanation, by perturbing the data points, as guided by the explanation.

Discriminative weights are identified by a threshold k ($0 \leq k \leq 100$) that represent the $(100 - k)$-percentile of the non-negative weight vector (w) that explains a time series. This threshold allows us to focus on the highest magnitude weights in the vector, e.g., $k = 10$ means that we focus on the top 10% highest weights in the vector. With a specific value of k, the **discriminative parts** of the time series are those parts where w_t belongs to the $(100 - k)$-percentile discriminative weights. This part is important because the weight magnitude captures information about the discrimination power of the corresponding data point in the time series. Similarly, with the same threshold k, the **non-discriminative** parts of the time series are parts which have w_t in the k-percentile of the time series (e.g., for $k = 10$ these are the bottom 10% weights with lowest magnitude) (Fig. 3).

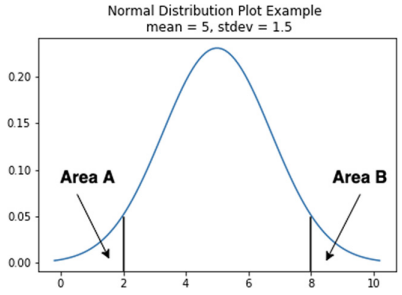

Fig. 3. Distribution of a hypothetical explanation weight vector with its non-discriminative weight area (Area A) and discriminative weight area (Area B).

We perturb a time series by adding Gaussian noise to its original signal. If the time series is represented by a vector x and the entire series is perturbed, the noisy time series would be represented by the new $x_{perturbed}$ vector

$$x_{perturbed} = x + \mathcal{N}(\mu, \sigma^2)$$

If a time series is normalized, the distribution for the Gaussian noise would be sampled from $\mathcal{N}(0, \sigma_1)$. The parameter σ_1 controls the magnitude of the noise.

In a similar fashion, the time series can also be selectively perturbed in accordance to a condition. In this case, we can perturb parts of the time series based on the corresponding weights in the explanation vector and keep the rest of the time series unchanged. With this logic of perturbing the time series (in accordance to a given weight vector), we selectively add noise to the time series as follows:

- *Type 1*: Perturbation applied only to discriminative region.
- *Type 2*: Perturbation applied only to non-discriminative region.

3.2 Method 1: Evaluating a Single Explanation Method

We propose an experiment to evaluate the informativeness of one explanation method. We aim to answer the question: *Is the explanation method truly informative?* In this experiment, we first build a time series classifier using the original, non-perturbed training time series. This classifier serves as the evaluation classifier for the explanation method, i.e., a *referee classifier*. In addition, we use the explanation method that we want to evaluate, to generate multiple versions of the test dataset, each corresponding to a value of the threshold k ($0 \leq k \leq 100$). For each value of k, we generate two perturbed test sets: one is only perturbed with *Type 1* noise, the other is only perturbed with *Type 2* noise. Using the referee classifier, we measure the accuracy in each perturbed test dataset. The entire process is summarized in Fig. 4.

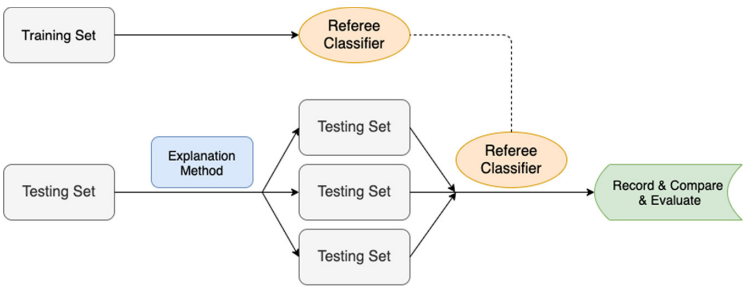

Fig. 4. Process of creating explanation-driven perturbed test sets and evaluating the explanation method using a referee classifier.

If the explanation method being evaluated is indeed informative, we expect that the perturbation of the discriminative parts (test datasets with *Type 1* noise) reduces the classifiers accuracy more than the perturbation of the non-discriminative parts (test datasets with *Type 2* noise).

3.3 Method 2: Comparing Multiple Explanation Methods

In contrast to the previous experiment, here we propose an experiment to compare multiple explanation methods by their informativeness. We follow the same process of creating noisy test sets as in Fig. 4, however, the perturbed test sets are now created differently. Instead of adding noise to both the discriminative and non-discriminative parts to create two different test sets for each k, we only add noise to the discriminative parts of the test time series. Since we have multiple explanation methods, at a same threshold k ($0 \leq k \leq 100$), we now have multiple versions of perturbed test datasets, each corresponding to a weight profile (i.e., explanation) obtained from one explanation method.

Among the evaluated explanation methods, a perturbation based on a more informative explanation should hurt the referee classifier more than the others. In Fig. 5, we hypothetically have two explanation methods with the *red* and *blue* lines representing the classification accuracy when test datasets are perturbed with either of the methods. Here, the explanation method controlling the perturbation of the test dataset with the resulting accuracy drawn in *red* is considered more informative, since perturbing the time series based on this explanation hurts the referee classifier more.

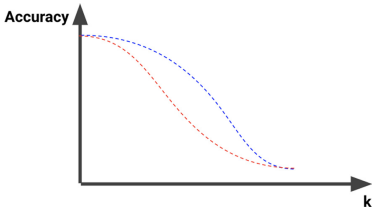

Fig. 5. Change of accuracy when the test set is perturbed with a threshold k.

As the vector of weights used as information to perturb the test dataset can be generated from *any* explanation method and independent of the referee classifier used to measure the change in accuracy, ***Method 1*** and ***Method 2*** are *model-agnostic* techniques to evaluate any TSC explanation method.

3.4 Informativeness of an Explanation: An Evaluation Measure

We quantify the informativeness of an explanation using the relationship between the accuracy of a referee classifier on test datasets perturbed at different levels k of noise. We calculate the impact of the explanation methods by estimating the area under the (explanation) curve described by accuracy at different perturbation levels k, using the trapezoidal rule. Since these values represents the reduction of the accuracy when noise is added to the time series, hereafter we call this metric *Explanation Loss* or *eLoss* for short. With this naming convention, one explanation method with lower *eLoss* will be considered better than another with higher *eLoss*.

$$eLoss = \frac{1}{2}k\sum_{i=1}^{t}(acc_{i-1} + acc_i)$$

where k denotes the values of each step normalized to 0–1 range; t denotes the number of steps ($t = \frac{100}{k}$); acc_i is the accuracy at step i. If we perturb the dataset with t steps, we will have a total of $t+1$ data points for accuracy scores. The step for $k = 0$ corresponds to the original test dataset, while the step for $k = 100$ corresponds to adding noise to the entire time series.

Evaluating a Single Explanation Method. The *eLoss* can serve as a measure to evaluate the informativeness of one explanation method. In particular, we estimate the *eLoss* of the accuracy curve produced by *Type 1* and *Type 2* noise. If the explanation method is informative, the *Type 1 eLoss* ($eLoss_1$) is expected to be less than *Type 2 eLoss* ($eLoss_2$). Alternatively, we can define this difference with Δ_{eLoss}:

$$\Delta_{eLoss} = eLoss_2 - eLoss_1.$$

If Δ_{eLoss} is positive, then the explanation method is computationally informative as captured by a referee classifier, otherwise the explanation method is deemed uninformative (i.e., the data points singled out by the explanation do not provide useful information to the classifier).

Comparing Multiple Explanation Methods. In the case where multiple explanation methods are presented for evaluation, we compare *Type 1 eLoss* ($eLoss_1$) for all explanation methods using an *independent referee classifier*. The explanation method that achieves the lowest $eLoss_1$ is the computationally most informative explanation method among the candidate methods.

4 Experiments

In this section, we present the results of applying our evaluation methodology using the following publicly available TSC datasets: CBF, Coffee, ECG200, GunPoint from UCR [7] and the CMJ dataset[1]. TSC explanations for these datasets have been examined in depth by the previous works [15,18,29], hence they are suitable for demonstrating our approach. Table 2 summarizes these datasets.

We evaluate three TSC explanation methods: *MrSEQL-SM*, CAM based on ResNet (*ResNet-CAM*) and LIME based on the Mr-SEQL classifier (*Mr-SEQL-LIME*). We also train three referee classifiers, *Mr-SEQL* [18], *ROCKET* [8], and *WEASEL* [27], in order to computationally evaluate the usefulness of these explanation methods. Due to a high computational cost for LIME, we evaluate LIME only with the CMJ and GunPoint datasets. The code and settings for all our experiments are available at https://github.com/mlgig/explanation4tsc.

[1] Retrieved from: https://github.com/lnthach/Mr-SEQL/tree/master/data/CMJ.

Table 2. Summary of TSC datasets used to evaluate explanation methods.

Dataset	Train size	Test size	Length	Type	No. classes
CBF	30	900	128	Simulated	3
CMJ	419	179	500	Motion	3
Coffee	28	28	286	SPECTRO	2
ECG200	100	100	96	ECG	2
GunPoint	50	50	150	Motion	2

4.1 Experiment 1: Evaluation of a Single Explanation Method

Table 3 summarizes the results for the evaluation of the three explanation methods with the three referee classifiers over the five TSC datasets. We calculate the difference between *Type 2 eLoss* and *Type 1 eLoss* (Δ_{eLoss}) with the explanation-driven perturbation approach. We expect Δ_{eLoss} to be positive when the explanation method is informative.

Table 3. Summary of Δ_{eLoss} of three explanation methods on five different TSC problems. Positive values suggest the findings of the explanation method are informative according to the referee classifier. Negative values suggest otherwise.

Dataset	Explanation method	Referee classifier		
		Mr-SEQL	ROCKET	WEASEL
CBF	MrSEQL-SM	0.0001	0.002	0.0126
	ResNet-CAM	**−0.0005**	0.0007	0.0141
CMJ	MrSEQL-SM	0.0045	0.0709	0.1151
	ResNet-CAM	**−0.0006**	**−0.0028**	0.0106
	MrSEQL-LIME	0.0084	0.0475	0.0531
Coffee	MrSEQL-SM	0.0286	0.0	0.0
	ResNet-CAM	0.0179	0.0	0.0143
ECG200	MrSEQL-SM	0.033	**−0.001**	0.024
	ResNet-CAM	**−0.011**	**−0.003**	0.038
GunPoint	MrSEQL-SM	0.0026	0.1373	0.0273
	ResNet-CAM	0.0067	0.0967	**−0.002**
	MrSEQL-LIME	0.002	0.0714	0.0007

To visualize the difference between *Type 1 eLoss* and *Type 2 eLoss*, we also present this information in the form of the accuracy curve for the GunPoint dataset (Fig. 6) and the CMJ dataset (Fig. 7). In each of the figures, we draw the accuracy curve in the case when noise is added to the most discriminative parts (*Type 1*) and non-discriminative parts (*Type 2*). We note that if the *Type 1* curve

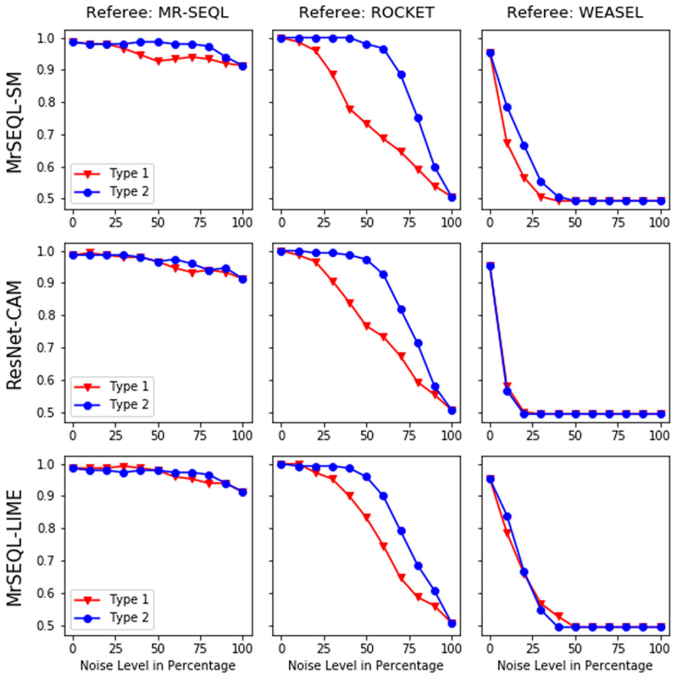

Fig. 6. Comparison of accuracy for *Type 1* (red) and *Type 2* (blue) perturbation for each explanation method and referee classifier for the GunPoint dataset. (Color figure online)

is below the *Type 2* curve, the explanation method is considered informative. If this trend is consistent across the referee classifiers, the evidence that the method is informative has more support. If we focus on evaluating the MrSEQL-SM explanation method for the GunPoint dataset, we observe that the *Type 1* curve is always below the *Type 2* curve for all three referee classifiers, thus we expect that this explanation method is informative. This information is consistent with the metric Δ_{eLoss} in Table 4, when Δ_{eLoss} is positive for all classifiers.

4.2 Comparison of Multiple Explanation Methods

In this experiment, we aim to compare the different explanation methods for a specific dataset. Instead of comparing the *eLoss* for the case when noise is added to the discriminative parts (*Type 1*) and non-discriminative parts (*Type 2*) of the time series for one explanation method, here we compare the *eLoss* for *Type 1* ($eLoss_1$) perturbation across different explanation methods. An explanation method is considered more informative if it has a smaller $eLoss_1$ for the same referee classifier.

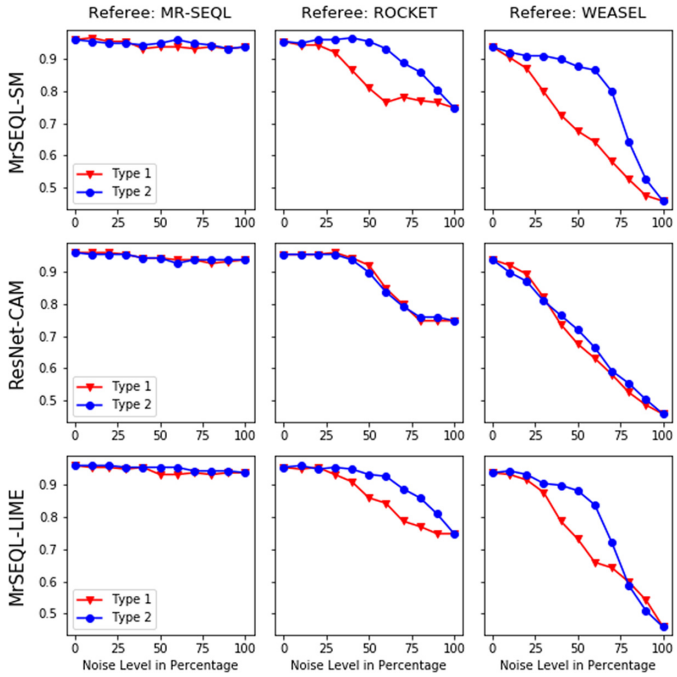

Fig. 7. Comparison of accuracy for *Type 1* (red) and *Type 2* (blue) perturbation for each explanation method and referee classifier for the CMJ dataset. (Color figure online)

We visualize this $eLoss_1$ in Fig. 8 in which the explanation curve of the three examined explanation methods are compared for the dataset CMJ (upper charts) and GunPoint (lower charts). We notice that the different in $eLoss_1$ is dependent on the referee classifier used to examine the change of the accuracy in the noisy test dataset. Given the same noisy datasets, the referee classifiers yield different classification accuracy. With the CMJ dataset, it is difficult to conclude which explanation method is most informative from Fig. 8, since the three lines are closely placed. This result is consistent with the comparison of $eLoss_1$ in Table 4. We can conclude that the three explanation methods are computationally similar in informativeness, although MrSEQL-SM is slightly more informative than the other two methods (its $eLoss_1$ is lowest for two referee classifiers).

4.3 Sanity Checks for Experiment Results

Although the evaluation measures show that one explanation method is more informative than another, we want to verify this conclusion by performing a sanity check step. In this step, we plot a few classification examples and their explanations by the methods evaluated previously. We choose to perform this

Table 4. Summary of $eLoss_1$ of three explanation methods on five different problems. Lower value (column-wise) suggests the explanation method is better in explaining the problem according to the referee classifier.

Dataset	Explanation method	Referee classifier		
		Mr-SEQL	ROCKET	WEASEL
CBF	MrSEQL-SM	**0.9991**	**0.9941**	**0.6018**
	ResNet-CAM	0.9993	0.9945	0.6041
CMJ	MrSEQL-SM	**0.9441**	**0.8422**	**0.6899**
	ResNet-CAM	0.9453	0.8735	0.6972
	MrSEQL-LIME	**0.9441**	0.8612	0.7385
Coffee	MrSEQL-SM	**0.9625**	1.0	**0.9786**
	ResNet-CAM	0.9696	1.0	0.9821
ECG200	MrSEQL-SM	**0.811**	0.9065	0.7565
	ResNet-CAM	0.838	**0.9035**	**0.7385**
GunPoint	MrSEQL-SM	**0.9477**	**0.7567**	0.543
	ResNet-CAM	0.961	0.7773	**0.5257**
	MrSEQL-LIME	0.9677	0.7953	0.573

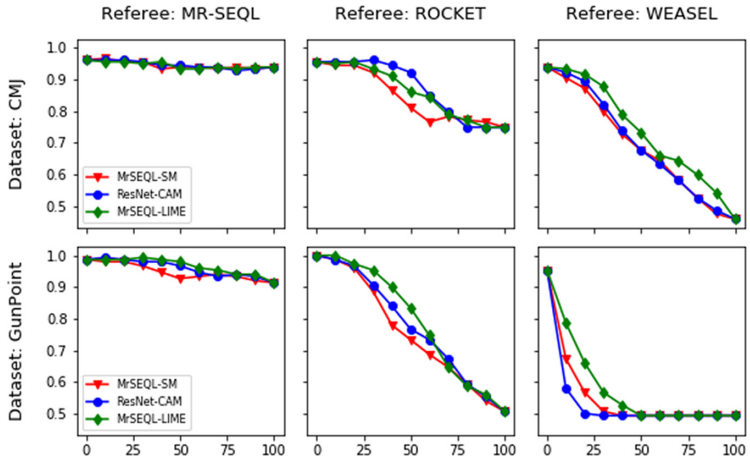

Fig. 8. Comparison of accuracy for *Type 1* perturbation based on three explanation methods (MrSEQL-SM, ResNet-CAM and Mr-SEQL-LIME) for GunPoint and CMJ datasets and three referee classifiers. Lower curve is better.

step with the CMJ dataset, for which the explanations are verified by a domain expert [18].

Figure 9 presents the saliency maps generated by three explanation methods for examples from the three motion classes in CMJ. Here we clearly see that these methods give different explanations. MrSEQL-SM seems to provide the

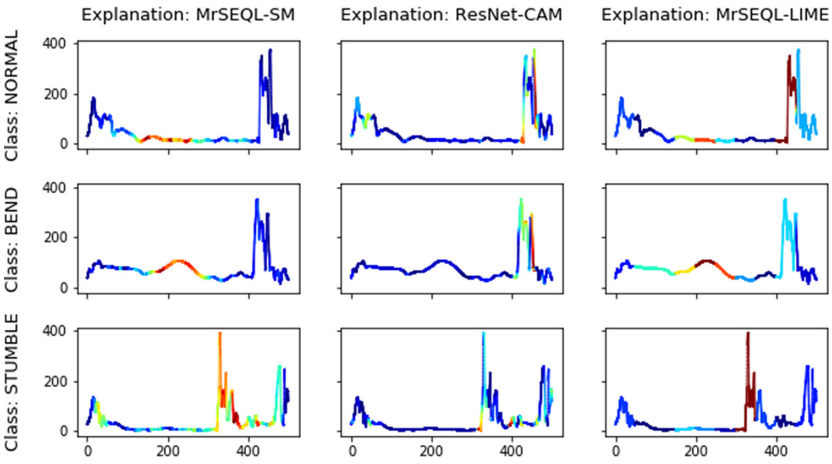

Fig. 9. Saliency maps produced by three explanation methods for example time series from the three classes of the CMJ dataset.

most informative/correct explanations that highlight the low, middle parts of the class NORMAL, the hump, middle parts of the class BEND, and the very high peak, middle parts of the class STUMBLE. MrSEQL-LIME gives a similar picture since it tries to explain the same classifier as MrSEQL-SM. ResNet-CAM does not clearly highlight similar parts in this dataset. This sanity check confirms the quantitative results in the previous experiments.

5 Discussion

In this section, we holistically interpret the experiment results with regard to informativeness and other perspectives. With the notion of informativeness, we set up the experiments based on an explanation-driven perturbation approach. This approach allows us to assess the contributing significance of the discriminative parts for a referee classifier. The results show that, with a given dataset, we are able to some extent evaluate and quantify the informativeness of different TSC explanation methods. There is scope though for further study of other perturbation approaches as well as the use of other referee classifiers in order to reach more significant differences in informativeness levels.

Stability of Explanation. Performing the experiment repeatedly, we notice that not all explanation methods provide consistent results. Methods that depend on certain level of randomization such as CAM (with randomized weight initialization) and LIME (with randomized local examples to estimate explanations) are generating slightly different explanations in different runs. For methods that are characterized by many hyperparameters like LIME, this stability of explanation is also dependent on these parameters, such as the number of local examples that it generates.

Robustness of Referee Classifier. We observe that some TSC methods are more sensitive to noise than others (Fig. 6, 7, 8). In our experiment, ROCKET and WEASEL seem to be more noise-sensitive than Mr-SEQL. This sensitivity results in higher value of Δ_{eLoss} when the method is tested with these noise-sensitive classifiers.

Computational Cost. It is worth mentioning that methods that locally generate explanations are computationally expensive. While MrSEQL-SM and CAM conveniently use the trained model internals to compute explanations for a new example, LIME generates multiple perturbations of the new example and reclassifies it to generate an explanation, which leads to high computational cost.

6 Conclusion

This work aims to provide an objective evaluation methodology to gauge the informativeness of explanation methods. Our experiment results show that it is feasible to quantitatively assess TSC explanation methods and the sanity checks visually confirm the experiment results. We envision that this technique is helpful when a user wants to assess an existing explanation method in the context of a given application, or wishes to evaluate different methods and opt for one that works best for a specific use case. In the scope of this work, we primarily evaluate three explanation methods which collectively represents different approaches to explain TSC decisions, though there are many other methods worth exploring. With the application of human activity recognition in mind, we believe that advancement in this area can potentially help many people who can thus conveniently access high quality technology to directly improve their lives.

Acknowledgments. This work was funded by Science Foundation Ireland through the SFI Centre for Research Training in Machine Learning (18/CRT/6183), the Insight Centre for Data Analytics (12/RC/2289_P2) and the VistaMilk SFI Research Centre (SFI/16/RC/3835).

References

1. Adebayo, J., Gilmer, J., Muelly, M., Goodfellow, I., Hardt, M., Kim, B.: Sanity checks for saliency maps. In: Proceedings of the 32nd International Conference on Neural Information Processing Systems, NIPS 2018, pp. 9525–9536. Curran Associates Inc., Red Hook (2018)
2. Apley, D.W., Zhu, J.: Visualizing the effects of predictor variables in black box supervised learning models (2016)
3. Avci, A., Bosch, S., Marin-Perianu, M., Marin-Perianu, R., Havinga, P.: Activity recognition using inertial sensing for healthcare, wellbeing and sports applications: a survey, pp. 167–176 (01 2010)
4. Bagnall, A., Lines, J., Bostrom, A., Large, J., Keogh, E.: The great time series classification bake off: a review and experimental evaluation of recent algorithmic advances. Data Mining and Knowledge Discovery, 1–55 (2016). https://doi.org/10.1007/s10618-016-0483-9

5. Bostrom, A., Bagnall, A.: Binary Shapelet transform for multiclass time series classification. In: Madria, S., Hara, T. (eds.) DaWaK 2015. LNCS, vol. 9263, pp. 257–269. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-22729-0_20
6. Bostrom, N., Yudkowsky, E.: The ethics of artificial intelligence (2011)
7. Dau, H.A., et al.: Hexagon-ML: The UCR time series classification archive, October 2018. https://www.cs.ucr.edu/~eamonn/time_series_data_2018/
8. Dempster, A., Petitjean, F., Webb, G.I.: Rocket: exceptionally fast and accurate time series classification using random convolutional kernels (2019)
9. Doshi-Velez, F., Kim, B.: Towards a rigorous science of interpretable machine learning (2017)
10. Fisher, A., Rudin, C., Dominici, F.: All models are wrong, but many are useful: learning a variable's importance by studying an entire class of prediction models simultaneously (2018)
11. Grabocka, J., Schilling, N., Wistuba, M., Schmidt-Thieme, L.: Learning time-series Shapelets. In: Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD 2014, pp. 392–401. ACM, New York (2014). https://doi.org/10.1145/2623330.2623613
12. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. CoRR abs/1512.03385 (2015). http://arxiv.org/abs/1512.03385
13. Ifrim, G., Wiuf, C.: Bounded coordinate-descent for biological sequence classification in high dimensional predictor space. In: Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD 2011, pp. 708–716. Association for Computing Machinery, New York (2011). https://doi.org/10.1145/2020408.2020519
14. Ismail Fawaz, H., Forestier, G., Weber, J., Idoumghar, L., Muller,P.A.: Accurate and interpretable evaluation of surgical skills from kinematic datausing fully convolutional neural networks. Int. J. Comput. Assist. Radiol. Surg. **14**(9), 1611–1617 (2019).https://doi.org/10.1007/s11548-019-02039-4
15. Ismail Fawaz, H., Forestier, G., Weber, J., Idoumghar, L., Muller,P.A.: Deep learning for time series classification: a review. Data Min. Knowl Disc. (2019). https://doi.org/10.1007/s10618-019-00619-1
16. Keogh, E., Ratanamahatana, C.A.: Exact indexing of dynamic time warping. Knowl. Inf. Syst. **7**(3), 358–386 (2005). https://doi.org/10.1007/s10115-004-0154-9
17. Krizhevsky, A., Sutskever, I., Hinton, G.E.: ImageNet classification with deep convolutional neural networks. In: Pereira, F., Burges, C.J.C., Bottou, L., Weinberger, K.Q. (eds.) Advances in Neural Information Processing Systems, vol. 25, pp. 1097–1105. Curran Associates, Inc. (2012). http://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.pdf
18. Le Nguyen, T., Gsponer, S., Ilie, I., O'Reilly, M., Ifrim, G.: Interpretable time series classification using linear models and multi-resolution multi-domain symbolic representations. Data Min. Knowl. Disc. **33**(4), 1183–1222 (2019). https://doi.org/10.1007/s10618-019-00633-3
19. Lin, J., Keogh, E., Wei, L., Lonardi, S.: Experiencing sax: a novel symbolic representation of time series. Data Min. Knowl. Disc. **15**(2), 107–144 (2007). https://doi.org/10.1007/s10618-007-0064-z
20. Lundberg, S.M., Lee, S.I.: A unified approach to interpreting model predictions. In: Guyon, I., Luxburg, U.V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., Garnett, R. (eds.) Advances in Neural Information Processing Systems, vol. 30, pp. 4765–4774. Curran Associates, Inc. (2017). http://papers.nips.cc/paper/7062-a-unified-approach-to-interpreting-model-predictions.pdf

21. Nguyen, T.L., Gsponer, S., Ifrim, G.: Time series classification by sequence learning in all-subsequence space. In: IEEE 33rd International Conference on Data Engineering (ICDE), pp. 947–958, April 2017. https://doi.org/10.1109/ICDE.2017.142
22. Petitjean, F., Forestier, G., Webb, G.I., Nicholson, A.E., Chen, Y., Keogh, E.: Dynamic time warping averaging of time series allows faster and more accurate classification. In: IEEE International Conference on Data Mining, pp. 470–479 (2014)
23. Ramgopal, S., et al.: Seizure detection, seizure prediction, and closed-loop warning systems in epilepsy. Epilepsy Behav. E&B **37C**, 291–307 (2014). https://doi.org/10.1016/j.yebeh.2014.06.023
24. Ribeiro, M.T., Singh, S., Guestrin, C.: Why should I trust you?: explaining the predictions of any classifier. CoRR abs/1602.04938 (2016). http://arxiv.org/abs/1602.04938
25. Ribeiro, M.T., Singh, S., Guestrin, C.: Anchors: High-precision model-agnostic explanations. In: AAAI (2018)
26. Schäfer, P.: The boss is concerned with time series classification in the presence of noise. Data Min. Knowl. Discov. **29**(6), 1505–1530 (2015)
27. Schäfer, P., Leser, U.: Fast and accurate time series classification with weasel. In: Proceedings of the 2017 ACM on Conference on Information and Knowledge Management, CIKM 2017, pp. 637–646. ACM, New York (2017). https://doi.org/10.1145/3132847.3132980
28. Selvaraju, R.R., Das, A., Vedantam, R., Cogswell, M., Parikh, D., Batra, D.: Grad-CAM: why did you say that? visual explanations from deep networks via gradient-based localization. CoRR abs/1610.02391 (2016). http://arxiv.org/abs/1610.02391
29. Senin, P., Malinchik, S.: SAX-VSM: interpretable time series classification using sax and vector space model. In: IEEE 13th International Conference on Data Mining (ICDM), pp. 1175–1180, December 2013. https://doi.org/10.1109/ICDM.2013.52
30. Simonyan, K., Vedaldi, A., Zisserman, A.: Deep inside convolutional networks: visualising image classification models and saliency maps. Preprint, December 2013
31. Smilkov, D., Thorat, N., Kim, B., Viégas, F.B., Wattenberg, M.: Smoothgrad: removing noise by adding noise. CoRR abs/1706.03825 (2017), http://arxiv.org/abs/1706.03825
32. Springenberg, J., Dosovitskiy, A., Brox, T., Riedmiller, M.: Striving for simplicity: the all convolutional net. In: ICLR (workshop track) (2015). http://lmb.informatik.uni-freiburg.de/Publications/2015/DB15a
33. Szegedy, C., et al.: Going deeper with convolutions. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1–9 (2015)
34. Wachter, S., Mittelstadt, B.D., Russell, C.: Counterfactual explanations without opening the black box: automated decisions and the GDPR. CoRR abs/1711.00399 (2017). http://arxiv.org/abs/1711.00399
35. Wang, Z., Yan, W., Oates, T.: Time series classification from scratch with deep neural networks: a strong baseline. In: International Joint Conference on Neural Networks (IJCNN), pp. 1578–1585, May 2017. https://doi.org/10.1109/IJCNN.2017.7966039
36. Zhou, B., Khosla, A., Lapedriza, A., Oliva, A., Torralba, A.: Learning deep features for discriminative localization. In: CVPR (2016)