Example or Prototype? Learning Concept-Based Explanations in Time-Series

Christoph Obermair OBERMAIR@TUGRAZ.AT
Graz University of Technology, Graz, Austria
CERN, Geneva, Switzerland
Alexander Fuchs FUCHS@TUGRAZ.AT
Franz Pernkopf PERNKOPF@TUGRAZ.AT
Graz University of Technology, Graz, Austria
Lukas Felsberger LUKAS.FELSBERGER@CERN.CH
Andrea Apollonio ANDREA.APOLLONIO@CERN.CH
Daniel Wollmann DANIEL.WOLLMANN@CERN.CH
CERN, Geneva, Switzerland

Editors: Emtiyaz Khan and Mehmet Gönen

Abstract

With the continuous increase of deep learning applications in safety critical systems, the need for an interpretable decision-making process has become a priority within the research community. While there are many existing explainable artificial intelligence algorithms, a systematic assessment of the suitability of global explanation methods for different applications is not available. In this paper, we respond to this demand by systematically comparing two existing global concept-based explanation methods with our proposed global, model-agnostic concept-based explanation method for time-series data. This method is based on an autoencoder structure and derives abstract global explanations called "prototypes". The results of a human user study and a quantitative analysis show a superior performance of the proposed method, but also highlight the necessity of tailoring explanation methods to the target audience of machine learning models.

Keywords: Explainable AI, Concept Explanations, Time-Series.

1. Introduction

Deep learning methods have conquered nearly every aspect of machine learning applications due to their flexibility and predictive power. However, they did not yet gain the same interest in safety-critical applications, due to their "black box" behavior (Beítez et al. (1997)). Especially in safety critical applications, wrong decisions can have severe impact on human health, *e.g.* in medical diagnosis (Reardon (2019); Weng et al. (2017)) or financial assets and reputation of large scale projects such as particle accelerators (Obermair et al. (2022)). In these cases, experts cannot simply rely on automatically generated predictions and are often legally obliged to state reasons for their decisions (Goodman and Flaxman (2017)). Therefore, the demand for methods that allow for the interpretation of black box models has been increasing and a wide variety of eXplainable Artificial Intelligence (XAI) algorithms were proposed in recent years.

The currently most popular XAI algorithms (Ribeiro et al. (2016); Bach et al. (2015); Shrikumar et al. (2016); Lundberg and Lee (2017); Simonyan et al. (2013); Chattopadhay et al. (2018)) are *relevance-based* methods capable of highlighting the parts of the data which are important for model predictions. Considering a handwritten digit classification problem using the MNIST dataset (Deng (2012)), for example, highlighting relevant pixels representing a particular number is an intuitive interpretation for humans.

Concept-based explanations (Kim et al. (2018); Yeh et al. (2020)) represent an alternative to highlighting important parts of the data. While there exist multiple definitions of *concepts* across the literature, we define a concept as *explanatory data containing all relevant properties that allow humans to make the same decisions as the black box model.* Typically, concepts are provided by (1) data examples, *i.e. explanation-by-example*, or (2) artificial data containing the most relevant information, *i.e. prototypes.* In the example of handwritten digit classification, showing the image of a typical digit 'one' from the available data would be an explanation-by-example, while showing an artificially created example of the digit 'one' with its main properties, *e.g.* the straight vertical line, would be a prototype.

In a recent empirical study conducted within a group of non-machine-learning experts, (Jeyakumar et al. (2020)) showed superior performance of a concept-based explanation method compared to relevance-based methods for time-series data. Explaining the non-intuitive nature of time-series data to non-machine-learning experts is a common task in safety critical applications, *e.g.* when explaining heart beat signals to medical professionals and patients. Consequently, concept-based explanations are an important tool in this domain. However, explanation-by-example and prototypes have not been compared in detail yet, although they belong to the main types of existing concept-based explanation methods.

Contribution. In this work, we investigate the advantages and disadvantages of explanation-by-example or prototypes for time-series explanations, depending on whether the target audience is users or model developers. Initially, we define a concept mathematically and denote concept properties to increase the explanation confidence. Consequently, we propose a model-agnostic concept-based XAI method[1], relying on an autoencoder using prototypes. We then compare our model-agnostic prototype (MAP) method to an explanation-by-example (EBE) (Jeyakumar et al. (2020)) and a model-specific prototype (MSP) (Gee et al. (2019)) explanation method with a human user study and a quantitative analysis.

Human User Study Details. For the conducted human user study, we utilized the ECG200 (Olszewski (2001)) dataset containing heartbeat signals and an artificial dataset reproducing signals from machine sensors in a noisy environment. Participants were asked to classify the time-series signals from the dataset, using the concept explanations which we provided. In total, 75 participants classified 3480 time-series signals based on explanation-by-example or prototypes derived from the different methods. The survey shows that our method is preferred, but also highlights the importance to distinguish between target audiences when comparing XAI methods.

Paper Structure. We first give an overview of related XAI work, followed by a formal definition of a concept and its properties. We then introduce our XAI method and our

1. https://github.com/cobermai/concep_based_explanations

study details. Finally, we discuss the results, and present future work in the domain of particle accelerators.

2. Related Work

In this section, we highlight the need for concept explanations, which are model-agnostic, applicable to time-series data, and tested and optimized for their target audience. With the increasing amount of time-series data available, hundreds of time-series classification methods have been recently proposed. Different methods are frequently based on nearest neighbors (Bagnall et al. (2017)), ensemble classifiers (Lines et al. (2018)), or convolutional neural networks (Fawaz et al. (2019)).

Many of the recently proposed XAI methods target the interpretation of such time-series classification methods (Rojat et al. (2021)). This is especially relevant for safety critical applications, where time-series data is a common data format. Tjoa and Guan (2020) provide a list of different XAI methods for medical applications as an example for safety critical applications. A recent summary from AlRegib and Prabhushankar (2022) highlights the small amount of model-agnostic XAI approaches and underlines the importance of human evaluation of such approaches. Amazons Mechanical Turk enables a relatively fast way to derive human non-expert evaluations without a bias, and is commonly used in XAI studies as in Jeyakumar et al. (2020), Lundberg and Lee (2017), Ribeiro et al. (2016), and Kim et al. (2018). It is more difficult to choose an intentional bias. For example, a bias towards the characteristics of the research community in safety-critical applications.

In the following subsections, we provide an overview of relevant concept explanation methods, distinguishing between methods using explanation-by-example and methods using prototypes to visualize their concepts. For each method, we emphasize whether the model is model-specific and whether the explanations are local or global. Local explanations, analyze the black-box predictions of each data sample, *i.e.* an *instance*, separately, while global explanations investigate all predictions at once.

2.1. Concept visualization with explanation-by-example

Kim et al. (2018); Yeh et al. (2020) describe concepts as a set of implicit vectors. To visualize a concept, the instance/example closest to the vector is extracted from the model specific architecture. Jeyakumar et al. (2020) cluster instances with similar activations in the last layer of a deep neural network. They use the cosine similarity as a similarity measure. The access of the activations makes this method model-specific. Explanation-by-example is frequently extended to show only relevant segments of examples. Chen et al. (2019); Das et al. (2020) show image patches like the ear of cat as examples. Guidotti et al. (2020) propose a model-agnosic method, which generates relevant example segments, using decision trees. These segments are frequently called shapelets. In Mochaourab et al. (2022) global explanations are derived from relevance-based explanation using Sobol's indices, *i.e.* a variance-based sensitivity analysis.

2.2. Concept visualization with prototypes

Prototype based methods aim at defining representative concept prototypes for model explanation (Bien and Tibshirani (2011)). Li et al. (2017); Gee et al. (2019) train prototypes with an autoencoder. A classifier is trained in parallel. This classifier utilizes the euclidean distance of the prototypes and the latent space of the autoencoder as an input. Here, classification and explanation are combined in the same model, which makes it model specific. In a similar way Zhang et al. (2020) derive one prototype per class with a model specific attention prototype network. Tang et al. (2020) generate time-series shapelets by combining concept-based and relevance-based methods.

The presented list of state-of-the-art methods, highlights the frequent use of prototypes and examples for visualization. For these methods, it has not been evaluated, which visualization technique is best in helping humans to reach similar accuracy as the black box model. This topic will be mathematically approached in the next section.

3. Concept Definition and Properties

Consider a training set of N instances $\mathbf{X} = \{\mathbf{x}_1, ..., \mathbf{x}_N\}$, where each instance $\mathbf{x}_n \in \mathbb{R}^p$ has a corresponding label $y_n \in \mathbb{N}^t$, and a black box model $f(\cdot)$, *e.g.* a pretrained deep neural network, which approximates these labels $\hat{y}_n = f(\mathbf{x}_n)$. An explainer model is then used to derive a set of M concept explanations $\hat{\mathbf{X}} = \{\hat{\mathbf{x}}_1, ..., \hat{\mathbf{x}}_M\}$ with predictions $\hat{\mathbf{Y}} = \{f(\hat{\mathbf{x}}_1), ..., f(\hat{\mathbf{x}}_M)\}$, where each explanation $m = 1, ..., M$ corresponds to a reconstructed concept $\hat{\mathbf{x}}_m$.

XAI methods are often evaluated through dedicated questionnaires (Holzinger et al. (2020)), asking its users to state their subjective assessment of the given explanation. To provide an objective evaluation of XAI methods, a human perceiver $s(\cdot)$ of an explanation should be able to find the correct label for unseen instances on their own. Showing all concept explanations $\hat{\mathbf{X}}$ and their corresponding labels $\hat{\mathbf{Y}}$ to users in the target audience, the *concept receptivity* is measured by the accuracy of the users when labeling new instances $\mathbf{x}_n$.

Definition 1 (Concept Receptivity) *A human perceiver $s(\cdot)$ has a concept receptivity r, which is the ability to find the label $\hat{y}_n$ for random instances $\mathbf{x}_n$ given the reconstructed concepts $\hat{\mathbf{X}}$ with labels $\hat{\mathbf{Y}}$*

$$r(\hat{\mathbf{X}}, \hat{\mathbf{Y}}) = \frac{1}{N} \sum_{n=1}^{N} \mathbb{1}_{\hat{y}_n = s(\mathbf{x}_n, \hat{\mathbf{X}}, \hat{\mathbf{Y}})}, \tag{1}$$

where $\mathbb{1}$ is an indicator function.

Human evaluations of explanations are labor-intensive. To this end, we further propose a quantitative evaluation method. Explainer models frequently use a transformation function to derive lower dimensional features $\mathbf{z}_i = g(\mathbf{x}_i)$, where $\mathbf{z}_i \in \mathbb{R}^q, \mathbf{x}_i \in \mathbb{R}^p$, and $q < p$. In order for this latent space to faithfully represent the input space, the relation of instances in the latent space, should be similar to the instances in the input space.

Definition 2 (Representability) *We specify the similarity of two instances* $\mathbf{x}_j$ *and* $\mathbf{x}_n$ *with the conditional probability*

$$p(\mathbf{x}_j|\mathbf{x}_n) = \frac{\exp(-||\mathbf{x}_n - \mathbf{x}_j||^2/2\sigma_n^2)}{\sum_{k\neq n}\exp(-||\mathbf{x}_n - \mathbf{x}_k||^2\sigma_n^2)}, \tag{2}$$

assuming a Gaussian distribution of data points with standard deviation σ *(Van der Maaten and Hinton (2008)). With this definition, we compare the conditional probabilities* $P_n = p(\mathbf{x}_j|\mathbf{x}_n)$ *and* $Q_n = p(\mathbf{z}_j|\mathbf{z}_n)$*, between input instances* $\mathbf{x}_j$ *and their latent space activations* $\mathbf{z}_j$*. We, therefore, determine the Kullback-Leibler (KL) divergence between conditional probabilities of one instance* n*, to all other* N *instances in the dataset. Notably,* $KL_{P_n=Q_n}(P_n||Q_n) = 0$ *indicates that distribution* P_n *equals* Q_n*. The sum of all KL divergences is the concept representability*

$$\phi_c = \sum_n^N KL(p(\mathbf{x}_j|\mathbf{x}_n) \parallel p(\mathbf{z}_j|\mathbf{z}_n)) = \sum_n^N \sum_j^N p(\mathbf{x}_j|\mathbf{x}_n) \log \frac{p(\mathbf{x}_j|\mathbf{x}_n)}{p(\mathbf{z}_j|\mathbf{z}_n)}. \tag{3}$$

Similarly, we determine how well the reconstructed concepts represent the input. In order to make the M *concepts comparable to the* N *input instances, we look for the nearest concept of each input instance in the latent space in terms of the* L_2*-Norm,* $\arg\min_{\hat{\mathbf{x}}_m} ||g(\mathbf{x}_n) - g(\hat{\mathbf{x}}_m)||_2$*. Hence, we obtain the reconstructed concepts in the input space,* $\hat{\mathbf{X}} = \{\hat{\mathbf{x}}_1, ..., \hat{\mathbf{x}}_N\}$*. We define the reconstructed concept representability as the sum of all KL divergences,*

$$\phi_{cr} = \sum_n^N KL(p(\mathbf{x}_j|\mathbf{x}_n) \parallel p(\hat{\mathbf{x}}_j|\hat{\mathbf{x}}_n)). \tag{4}$$

Fig. 1 depicts the concept representability ϕ_c and the reconstructed concept representability ϕ_{cr}. The input $\mathbf{x}_n$ consist of two blue signals of class one and one red signal of class two. Two concepts $\mathbf{c}_1, \mathbf{c}_2$ are derived from the latent space $\mathbf{z}_n$ with k-means. The red signal is reconstructed with $\mathbf{c}_2$. The two blue signals are closest to $\mathbf{c}_1$, and their reconstructed concept is therefore equal. Hence, the similarity of the input signals is well reflected by the concepts.

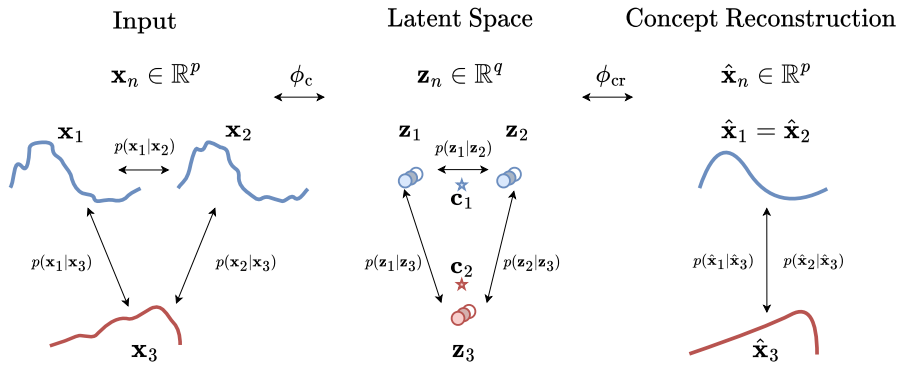


Figure 1: Example with three instances of two classes in red and blue. Two concepts have been reconstructed from the latent space.

4. Model-Agnostic Prototype Method

Our proposed method uses an autoencoder architecture, shown in Fig. 2, consisting of an encoder function $g(\cdot)$ that maps each instance n onto a latent space $\mathbf{z}_n \in \mathbb{R}^q$, and a decoder function $h(\cdot)$ that transforms the latent space back to the original input space $\mathbb{R}^p$. Using the latent space of the training set, we infer the concepts $\mathbf{C} = \{\mathbf{c}_1, ..., \mathbf{c}_M\}$ with k-means ($k = M$), where concepts are considered to be prototypes. A prototype enables the inference of M reconstructed concepts through $\hat{\mathbf{x}}_m = h(\mathbf{c}_m)$, using $\mathbf{c}_m$ within the latent space of the model.

Our method is trained independently of the black box model and is therefore model-agnostic. This not only enables to use any existing model without modifications, it also enables to derive explanations for already trained models. Furthermore, we argue that model-specific explanation methods, that access the activations of a hidden layer from a trained black box model, infer worse reconstructions, as detailed information necessary for the reconstruction is lost in the process of optimizing the weights for classification. Unlike other autoencoder methods (Gee et al. (2019); Li et al. (2017)), we derive our concepts directly from the latent space, *i.e.* the activations of the last encoder layer, instead of optimizing the concepts during training. We also employ a similarity loss for the latent space to diversify the concepts in Eq.7. In practice, this leads to more robust training, faster convergence, and more meaningful concepts.

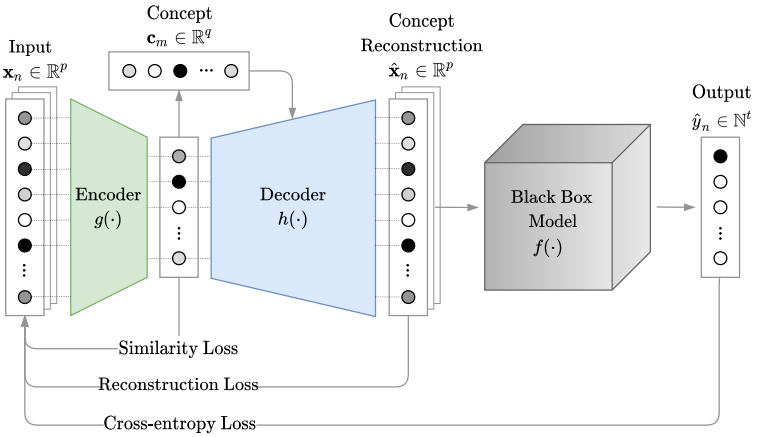


Figure 2: Model architecture used for the MAP explainer. Given a trained black box model, we fit an autoencoder to reconstruct the input data and to recreate the output of the black box model. The concepts are derived from the autoencoder latent space and are optimized to be diverse.

During optimization of the autoencoder weights, we maximize the ability to reconstruct both the input, and the exact prediction in terms of softmax outputs of the black box model. To regularize the concepts, we also employ a similarity loss during training. For the reconstruction loss, we use the mean-squared-error,

$$R(g, h, \mathbf{X}) = \frac{1}{N} \sum_{n=1}^{N} \left(\mathbf{x}_n - h\left(g(\mathbf{x}_n)\right)\right)^2 . \tag{5}$$

For classification tasks the ability to reconstruct the model prediction, is measured via the categorical-cross-entropy loss,

$$C(g, h, f, \mathbf{X}) = -\sum_{n=1}^{N} \left(\arg\max_{\hat{y}} f(\mathbf{x}_n) \right) \log f\left(h\left(g(\mathbf{x}_n)\right)\right). \tag{6}$$

Diverse concepts are obtained by penalizing non-orthogonality between two different concepts $\mathbf{c}_i, \mathbf{c}_j \in \mathbf{C}$. Specifically, we define the similarity loss as the scaled sum of their inner products (Yeh et al. (2020))

$$S(\mathbf{C}) = \frac{\sum_{i \neq j} \mathbf{c}_i^T \mathbf{c}_j}{M(M-1)}, \tag{7}$$

where the concepts are the cluster centers of the latent space, derived with k-means[2]. The complete learning objective is given as follows. Notably y_n is not required, which enables an unsupervised training of model-agnostic prototypes,

$$\mathcal{L}(g, h, \mathbf{X}) = R(g, h, \mathbf{X}) + \lambda_C C(g, h, f, \mathbf{X}) + \lambda_S S(\mathbf{C}). \tag{8}$$

4.1. Model Structure

We use two different autoencoder structures (for full details, see appended code[1]), based on an extensive sensitivity analysis, and relevant literature from Agarap (2018) and O'Shea and Nash (2015). In this section, we first validate our model and show the effect of the hyperparameters λ_C and λ_S in Section 4.2. For this task we use a three layer convolutional autoencoder architecture with a 3x3 kernel and a filter size of 32, 64, and 1 for both the encoder and the decoder. Additionally, both the encoder and decoder, use ReLU activations in the first two layers, and a sigmoid activation in the last layer.

For the explanation of time-series classification, in Section 5 and 6, we use a one layer encoder, L1-activity regularization and a normalized output. This means that our method finds a linear mapping of the input signal to the concepts. Multivariate time-series are flattened before the encoder. We set the hyperparameters to $\lambda_C = 1$ and $\lambda_S = 1$, and monitor that all loss terms converge. Furthermore, we used a three layer neural network, with 300 neurons per layer and a sigmoid activation in the second layer as a decoder.

For both autoencoders, we set the latent space size to five times the number of concepts and use the ADAM optimizer. This enables all loss terms to converge, while keeping the latent space small.

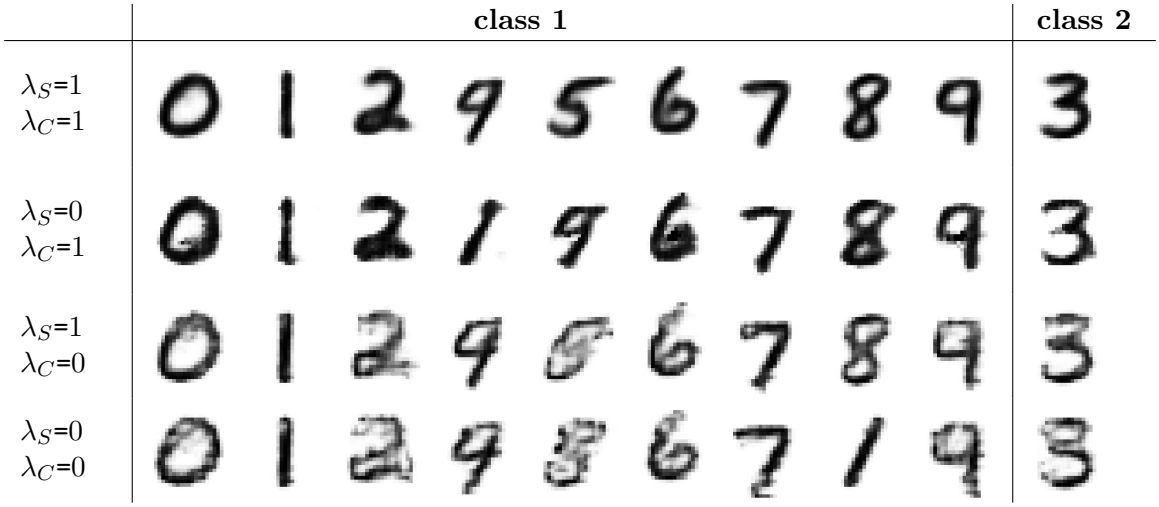
4.2. Model Validation

We validate our method by explaining a classifier, trained to predict whether an instance of the MNIST dataset of handwritten digits (Deng (2012)) contains the digit 'three'. As a classifier, we use a four layer neural network. Specifically, it consists of two convolutional layers of size 32 and 64, followed by two fully connected layers of size 128 and 10. All layers but the last use ReLU activations, where a softmax activation is used. In addition, we use max pooling after the second layer, and 0.2 and 0.4 dropout after the last two layers. The classifier is trained with the ADAM optimizer, achieving an accuracy of 99.8%.

2. In practice, this means that k-means is applied on each batch during training.

Ten model-agnostic prototypes ($M = 10$) were reconstructed, shown in Fig. 3, with different hyperparameters λ_C and λ_S. These prototypes, were manually sorted. The prototypes, calculated with $\lambda_C = 1$ and $\lambda_S = 1$, manage to represent all digits in the dataset, except for an overlap of digits 'four' and 'nine'. Without similarity loss ($\lambda_S = 0$), there are two concepts for the digit 'one', an overlap in the digits 'five' and 'nine', and digit 'four' is missing. Without classification loss ($\lambda_C = 0$), all numbers are represented with lower reconstruction performance compared to the first row. Finally, the effect of setting both hyperparameters to 0 is shown ($\lambda_C = 0$ and $\lambda_S = 0$). This means the reconstructions are blurry, the number 'eight' is missing, and the number 'one' occurs twice. With this simple example, we demonstrate the effectiveness of our model on an easily interpretable and well known dataset. The same approach will be applied to time-series classification problems, which are harder to interpret for humans.

Figure 3: Validation of our MAP method using MNIST with and without similarity and classification losses, with M=10. The black box model was trained to classify whether an image contains the digit 'three'.



5. Methodology

5.1. Modeling Methodology

We conduct a quantitative analysis on 12 datasets and a human user study, where we assess two of the datasets qualitatively. The selection of the datasets is based on signals frequently used in safety critical applications. We derive 11 of the datasets from the UCR archive Dau et al. (2019) and create one artificial dataset on our own. We address details for the human user study and the experimental methods in Section 5.2. The results of the quantitative analysis and a human user study, are reported in Section 6.1 and Section 6.2, respectively.

5.2. Human User Study[3]

We analyze the suitability of the three different methods in the context of safety critical applications empirically, with one survey[1] per method. For each survey, participants labeled 15 instances from two different datasets. These 15 instances were drawn randomly from the dataset to ensure fair comparison. To choose the right label, participants were given the concept explanations for each class in the dataset. Out of all correct answers, we then calculated the concept receptivity, as described in Eq. 1.

Participants. Our study was distributed within our research community, collecting a total of 3480 answers from 75 students and research staff working in the field of safety critical applications. The bias, introduced due to sampling in this research community, is intentional in order to optimize the models towards their future users. In the beginning of the survey, participants were asked to indicate whether they have prior knowledge in the field of machine learning. People who answered positively to this question were classified as typical *developers* of ML methods, and people who answered negatively to this question were classified as potential *users* of explanations in safety-critical system application.

Validating Responses. We applied two filtering criteria to eliminate non-reliable or biased answers. While people were allowed to fill in more than one survey, we only took into account the first survey for each person for the main results. This is to remove positively skewed responses resulting from familiarizing with the datasets, further discussed in the results. Additionally, we eliminated participants scoring worse than random, *i.e.* with less than 15 out of 30 correct answers. In total, 2190 answers from 73 participants were analyzed.

Datasets. For the survey, we selected two distinct datasets containing scaled signals of electrical activity and sensor data measured in Volt.

1. **ECG200:** We use the ECG200 (Olszewski (2001)) dataset from the UCR archive (Dau et al. (2019)) containing data of electrical activity measured during one heartbeat. Specifically, the latter part of a heart beat is shown in the signal, starting after the peak point R. The characteristic properties of a normal heart beat (class 1) compared to an ischemic heart beat (class 2) are the high peak point R and the limited recovery time from its minimum S to T. We show a scaled reconstruction of the ground truth in Fig. 4 together with the characteristic points R, S, T, and U.

2. **Artificial Dataset:** Furthermore, we created an artificial dataset, reproducing signals from machine sensors in a noisy environment.In particular, we used four basic time-series shapes, shown in the ground truth signals Fig. 5, and added multiplicative and additive noise with an amplitude of 0 to 1.1, drawn from a uniform distribution.

Black Box Model. We used a Fully Convolutional neural Network (FCN) (Fawaz et al. (2019)) to classify the signals. It consists of three convolutional layers with 128, 256, and 128 filters of kernel size 8, 5, and 3. The first two layers use ReLU activation and batch normalization. The last layer's output is globally averaged and fed into a softmax activation.

3. The study was conducted in compliance with the CERN (2022) Data Privacy Protection Policy and the CERN (2010) Code of Conduct.

Explanation Methods. We compare our MAP explanation method, described in Section 4, with two addition concepts explanation methods. Their implementation details are stated below. The number of concepts is set to two times the number of classes in all datasets $M = 4$, which allows all loss terms to converge. We also tried to keep the number of concepts low and even to improve the simplicity of our survey.

1. **Explanation-By-Example (EBE):** First, we used the idea of instance explanation-by-example methods (Jeyakumar et al. (2020); Papernot and McDaniel (2018)) to implement a global explanation method. Namely, we split the FCN into an encoder $g(\cdot)$ and a predictor, at the last convolutional layer. We then calculate the k-means cluster centers of the activations $g(\mathbf{x}_n)$. The instance with the closest euclidean distance to each cluster center was then used as a global explanation-by-example.
2. **Model-Specific Prototypes (MSP):** We implemented the model-specific prototype method from Gee et al. (2019). This method also learns prototypes from the output of an encoder. A softmax classifier then uses the distance of the encoder output to all prototypes for classification. Finally, the learned prototypes are reconstructed with a decoder. In addition to the cross-entropy loss and the reconstruction loss, the authors introduce a prototype diversity loss as a learning objective. As the method is model-specific, we used the convolutional layers of the FCN as an encoder on top of a fully connected layer with 20 neurons as an encoder $g(\cdot)$. Similar to our MAP model we used a three layer fully connected neural network, with 300 neurons per layer and a sigmoid activation function in the second layer as a decoder $h(\cdot)$. Similarly to the paper (Gee et al. (2019)), the predictor consists of a softmax layer, where decisions are inferred from the distance of input instances to the learned prototypes. All hyperparameters were taken from the original paper, after performing a detailed sensitivity analysis.

Training Stability. While the training of the FCN already converged after 200 epochs, we trained both autoencoder methods for 1500 epochs to ensure convergence of all regularization terms. We ensured that none of the models was stuck in local minimum, by training each model five times and selecting the one with the lowest overall loss.

Study Significance. Confidence intervals are calculated using the binomial proportion (Brown et al. (2001)) $\hat{p} \pm z\sqrt{(\hat{p}(1-\hat{p}))/n}$, where $\hat{p}$ is the proportion of successes in a binomial trial, *i.e.* the amount of all correctly classified instances divided by the amount of all classification samples n. Here, z is the quantile of a standard normal distribution $1 - \alpha/2$, where α is the target error rate. This means that for our 95% confidence interval $\alpha = 0.5$ and $z = 1.96$.

6. Results

6.1. Modeling Results

The quantitative modeling results of EBE, MSP, and MAP are shown in Table 1. Based on the definitions given in Section 3, the classification accuracy, the concept representability, and the reconstructed concept representability are shown by the mean and standard deviation (in brackets) over five training runs.

The EBE & MAP methods, use the same FCN classifier for the prediction of the classes. This FCN classifier is trained only with the cross-entropy loss, without a specific loss for

Table 1: Concept properties of EBE, MSP and our MAP. The model accuracy (higher scores are better), the representabilities (lower scores are better) are given by the mean of five independent training runs with standard deviation in brackets. The accuracy of EBE and MAP is equal, as they use the same FCN classifier for prediction.

Dataset	Model Accuracy [%]		Concept Representability			Reconstructed Concept Representability		
	FCN(ours)	MSP	EBE	MSP	MAP(ours)	EBE	MSP	MAP(ours)
ECG200	**84.0(0.8)**	79.7(10.8)	2.3(0.1)	2.1(1.1)	**0.2(0.1)**	8.1(4.6)	5.8(2.4)	**0.6(0.1)**
Artificial data	**99.9(0.1)**	93.6(17.7)	15.7(0.7)	15.7(4.8)	**7.8(0.8)**	4.7(2.7)	4.7(0.9)	**4.3(1.8)**
ACSF1	**85.9(2.5)**	**85.9(3.1)**	1.0(0.2)	1.4(0.1)	**0.5(0.1)**	1.8(0.9)	2.8(1.9)	**1.2(0.2)**
Computers	**83.5(2.0)**	73.8(11.0)	6.0(0.9)	12.0(4.8)	**1.5(0.2)**	**2.1(0.0)**	13.2(0.5)	9.3(1.0)
ECG5000	**92.9(0.2)**	92.7(0.7)	3.9(0.2)	13.0(1.5)	**2.0(0.3)**	8.9(0.3)	**5.6(1.3)**	6.3(0.5)
LargeKitchenAppliances	86.9(0.6)	**89.4(1.2)**	9.5(0.5)	15.5(0.3)	**1.3(0.5)**	15.5(1.2)	11.8(2.5)	**9.1(0.4)**
PowerCons	**91.9(0.8)**	78.7(19.1)	2.7(0.1)	4.7(2.2)	**1.2(0.2)**	9.6(1.5)	**5.3(2.6)**	7.4(1.7)
RefrigerationDevices	50.1(1.8)	**50.3(3.1)**	5.4(0.3)	11.5(0.7)	**4.1(1.0)**	**12.6(0.6)**	14.2(1.0)	18.3(1.4)
ScreenType	61.8(1.8)	**62.1(2.4)**	5.0(0.2)	10.9(0.9)	**1.8(0.2)**	**7.2(0.8)**	14.7(2.9)	12.4(1.1)
SmallKitchenAppliances	**78.5(1.2)**	74.7(3.3)	8.8(0.3)	14.9(1.0)	**1.5(0.7)**	**13.3(0.1)**	18.6(0.8)	15.5(0.6)
Plane	**99.4(1.1)**	95.6(7.7)	0.4(0.1)	0.9(0.8)	**0.2(0.1)**	1.0(0.3)	2.8(1.9)	**0.9(0.5)**
Trace	**100.0(0.0)**	**100.0(0.0)**	1.2(0.0)	1.9(0.1)	**0.1(0.0)**	2.6(0.9)	2.0(0.2)	**0.9(0.2)**
Win	9	5	0	0	12	4	2	6

explanation. The MSP method is trained to classify and explain at the same time. As a result of this combined objective, the model does not always converge to the global minimum of the cost function. This effect is also observed in the Artificial data, the Computers, the PowerCons, and the Plane dataset, where the standard deviation of the MSP is much higher compared to the standard deviation of the FCN. If the MSP does converge, then it reaches similar results compared to the FCN. In case of the LargeKitchenAppliances, the ScreenType, and the SmallKitchenAppliances datasets, the mean accuracies of MSP are even higher compared to the FCN mean accuracy.

The MAP reaches the highest concept representability in all cases. This shows that the distribution of the latent space is representing the input distribution most accurately for the MAP. While the MAP derives the latent space with a linear transformation, the decoder is still able to identify the correct concepts. This can be seen in the high reconstructed concept representability, where the MAP achieves best results for six datasets. The reconstructed concept representability of EBE is highest in four datasets. The similarity between input instances and explanation-by-example concepts is more similar for EBE compared to prototypes of MSP and MAP, where unimportant information, *e.g.* noise, is filtered out.

We further obtained the true reconstructed concept representability on the artificial dataset, where the ground truth is available. Here, we used the ground truth signal of each input instance as a concept. For this case we obtain a reconstructed concept representability of 3.7, while MSP and EBE reaches 4.7, and MAP reaches 4.3. Looking at Figure 5, the prototypes of MAP are closest to the ground truth, which validates the performance measure.

6.2. Human User Study Results

The results of the study including 73 Participants which classified a total of 2190 instances are presented in Table 2. Analyzing the answers of all participants from both datasets, our MAP method, showed the best results with 79.3% correct answers. This observation is valid also when taking into account the non-overlapping confidence intervals. When looking at the same quantity for individual datasets, one can observe a similar trend.

Figure 4: Ground truth and explanation of the ECG200 (Olszewski (2001)) dataset, showing the latter part of a heart beat, starting before the peak R. For each class, two concepts were extracted with different explanation methods.

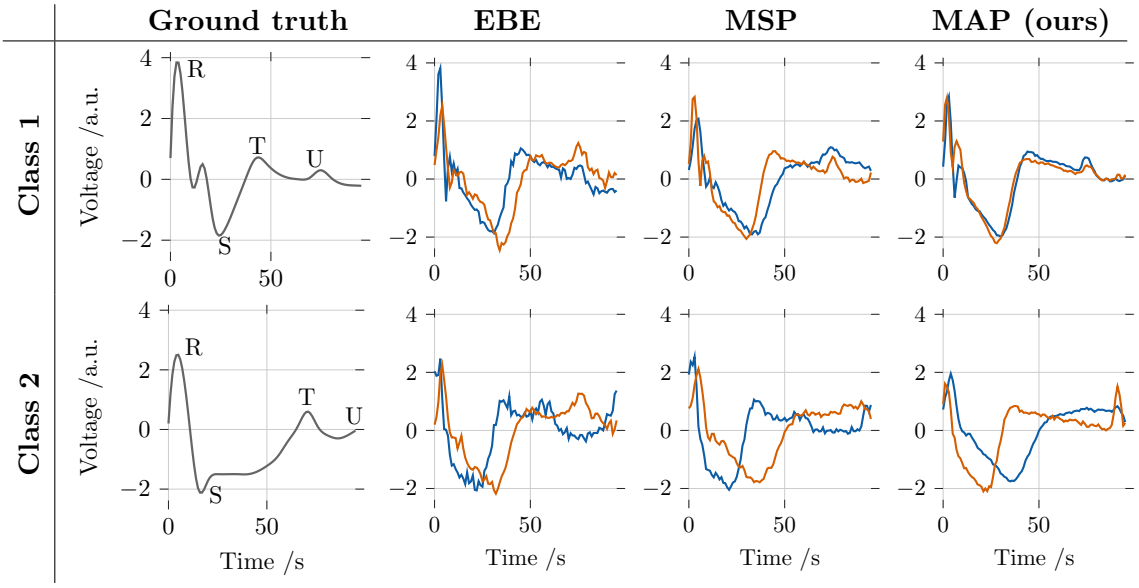
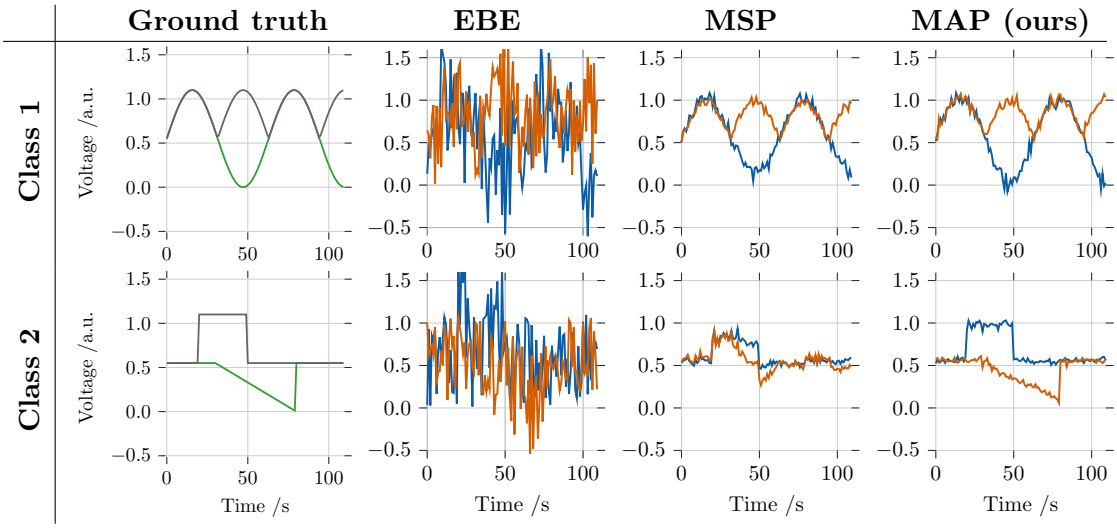


Figure 5: Explanations of artificially created dataset with two concepts per class, extracted from different explanation methods. The ground truth signal shows the four shapes within the dataset, to which multiplicative and additive noise with an amplitude of 0 to 1.1, drawn from a uniform distribution, was added.



In the artificial dataset, EBE was preferred by *users* over other methods. When considering Fig. 5, one would not expect this, as it seems that EBE is the most distinct from the ground truth signal. However, possibly participants not familiar with machine learning, were not able to establish the link between the pattern in the abstract concepts and the

Table 2: Results of the study comparing global EBE, MSP and our MAP. 73 Participants classified a total of 2190 instances, given the reconstructed concepts $\hat{\mathbf{x}}$ and their labels $\hat{y}$. The classification accuracy denotes their receptivity r (see Eq. 1) for the calculated concepts and is shown with the 95% confidence interval for a binomial proportion.

Participants	**Method**	**ECG200** [%]	**Artificial data** [%]	**Total** [%]
Developer	EBE	65.6 ± 6.7	79.0 ± 5.7	72.3 ± 4.4
	MSP	64.4 ± 8.1	74.1 ± 7.4	69.3 ± 5.5
	MAP (ours)	**74.8 ± 5.2**	**84.8 ± 4.3**	**79.8 ± 3.4**
User	EBE	71.3 ± 7.3	**80.0 ± 6.4**	75.7 ± 4.9
	MSP	64.3 ± 6.5	67.6 ± 6.3	66.0 ± 4.5
	MAP (ours)	**77.8 ± 7.0**	78.5 ± 6.9	**78.1 ± 5.0**
All participants	EBE	68.1 ± 4.1	79.4 ± 3.1	73.8 ± 2.4
	MSP	64.3 ± 4.0	70.1 ± 3.2	67.2 ± 2.4
	MAP (ours)	**75.8 ± 3.6**	**82.7 ± 2.9**	**79.3 ± 2.2**

time-series instances. Looking at the performance of the MAP method on the artificial dataset, prototypes that did not fit the shape of the ground truth, appear to be confusing for the *users*.

For the ECG200 dataset, *developers* and *users* were able to generalize best using our method with 74.8% and 77.8% correct answers, respectively. Looking at the class 2 signal in Fig. 4, the characteristic features of ischemic heartbeat signals are represented well by the derived concept. Specifically, the low amplitude in the spike R and the long recovery time from the points S to T is visualized, while showing much less noise than the other methods. A trend of *developers* giving worse results than *users* is visible, suggesting that *developers* are not necessarily able to generalize better than *users* utilizing concept-based explanations.

We further evaluated the effect of our filtering criteria (see Section 5.2), by looking at the results of the 1170 dropped answers from 39 participants who filled out more than one survey. Here, the learning effect outweighed the decision fatigue, as the performance increased on average by 6.8% for ECG200 and 5.1% for the artificial dataset in later attempts.

7. Conclusion

The quality of global, and model-agnostic concept explanation techniques is a key factor to help experts in safety critical domains gaining trust in predictions made by machine learning models. We demonstrated that our provided model-agnostic method fulfills these requirements by providing accurate and complete explanations, independent of the weight initialization or the concept numbers. We assessed the quality of our explanations quantitatively with 12 datasets, containing data common in safety critical applications. On two datasets, we further performed a human user study across 75 participants with, 2190 validated answers. The conducted survey showed that our proposed method helped participants to generalize explanations for classification tasks on time-series data across all datasets and target audiences. Specifically, participants reached 79.3% correct answers on average using our method, while reaching only 73.8% with explanation-by-example and 67.2% with model-specific prototypes. In the case of the artificial dataset, the prototype explanations show a significant visual discrepancy with respect to the signals presented in the survey, possibly leading to better results of the explanation-by-example method. In our domain,

i.e. predicting failures in particle accelerators, explanations are expected to be interpreted by domain experts. This makes the explanation by model-agnostic prototypes the preferred option in general, with explanation-by-example representing a valid alternative if prototypes become too abstract.

8. Future Work

Our future work will focus on the application of the proposed method to predict failures in superconducting electrical circuits in CERN's Large Hadron Collider (LHC Wenninger (2016)). The circuit data collected during several years of successful operation enables the use of data-driven methods to help experts find anomalies in the behavior of superconducting circuits and potentially also of protection systems. Our model-agnostic explanation technique will help in explaining existing deep learning models to system experts with no machine learning background. This will enable faster and more accurate fault diagnostics and optimized maintenance actions, further increasing safety and availability of the LHC. In addition, improvements of our method will aim at making more tailored variants of our explanation. Particularly, we plan to use Fourier analysis to correctly address the complexity of the behavior of superconducting circuits in the frequency domain.

References

Abien Fred Agarap. Deep learning using rectified linear units. *CoRR*, abs/1803.08375, 2018.

Ghassan AlRegib and Mohit Prabhushankar. Explanatory paradigms in neural networks: Towards relevant and contextual explanations. *IEEE Signal Processing Magazine*, 39(4): 59–72, 2022.

Sebastian Bach et al. On Pixel-Wise Explanations for Non-Linear Classifier Decisions by Layer-Wise Relevance Propagation. *PloS one*, 10(7):1–46, 2015.

Anthony Bagnall, Jason Lines, Aaron Bostrom, James Large, and Eamonn Keogh. The great time series classification bake off: a review and experimental evaluation of recent algorithmic advances. *Data mining and knowledge discovery*, 31(3):606–660, 2017.

José Manuel Beítez, Juan Luis Castro, and Ignacio Requena. Are artificial neural networks black boxes? *IEEE Transactions on neural networks*, 8(5):1156–1164, 1997.

Jacob Bien and Robert Tibshirani. Prototype selection for interpretable classification. *The Annals of Applied Statistics*, 5(4):2403–2424, 2011.

Lawrence D Brown, T Tony Cai, and Anirban DasGupta. Interval Estimation for a Binomial Proportion. *Statistical Science*, 16(2):101–133, 2001.

CERN. Code of conduct, 2010. URL https://procurement.web.cern.ch/system/files/document/cern-code-conduct.pdf.

CERN. Data privacy protection policy, 2022. URL https://home.cern/data-privacy-protection-policy.

Aditya Chattopadhay, Anirban Sarkar, Prantik Howlader, and Vineeth N Balasubramanian. Grad-cam++: Generalized gradient-based visual explanations for deep convolutional networks. In *Winter conference on applications of computer vision*. IEEE, 2018.

Chaofan Chen, Oscar Li, Daniel Tao, Alina Barnett, Cynthia Rudin, and Jonathan K Su. This Looks Like That: Deep Learning for Interpretable Image Recognition. In *Advances in Neural Information Processing Systems*. NeurIPS, 2019.

Subhajit Das, Panpan Xu, Zeng Dai, Alex Endert, and Liu Ren. Interpreting deep neural networks through prototype factorization. In *International Conference on Data Mining Workshops*. IEEE, 2020.

Hoang Anh Dau, Anthony Bagnall, Kaveh Kamgar, Chin-Chia Michael Yeh, Yan Zhu, Shaghayegh Gharghabi, Chotirat Ann Ratanamahatana, and Eamonn Keogh. The UCR time series archive. *IEEE Journal of Automatica Sinica*, 6(6):1293–1305, 2019.

Li Deng. The mnist database of handwritten digit images for machine learning research. *IEEE Signal Processing Magazine*, 29(6):141–142, 2012.

Hassan Ismail Fawaz et al. Deep learning for time series classification: a review. *Data Mining and Knowledge Discovery*, 33(4):917–963, 2019.

Alan H Gee, Diego Garcia-Olano, Joydeep Ghosh, and David Paydarfar. Explaining Deep Classification of Time-Series Data with Learned Prototypes. *CoRR*, abs/1904.0, 2019.

Bryce Goodman and Seth Flaxman. European Union regulations on algorithmic decision-making and a "right to explanation". *AI magazine*, 38(3):50–57, 2017.

Riccardo Guidotti, Anna Monreale, Francesco Spinnato, Dino Pedreschi, and Fosca Giannotti. Explaining any time series classifier. In *Second International Conference on Cognitive Machine Intelligence*. IEEE, 2020.

Andreas Holzinger, André Carrington, and Heimo Müller. Measuring the quality of explanations: the system causability scale (SCS). *KI-Künstliche Intelligenz*, 2020.

Jeya V. Jeyakumar, Joseph Noor, Yu-Hsi Cheng, Luis Garcia, and Mani Srivastava. How Can I Explain This to You? An Empirical Study of Deep Neural Network Explanation Methods. In *Advances in Neural Information Processing Systems*. NeurIPS, 2020.

Been Kim et al. Interpretability beyond feature attribution: Quantitative testing with concept activation vectors. In *International conference on ML*. PMLR, 2018.

Oscar Li, Hao Liu, Chaofan Chen, and Cynthia Rudin. Deep Learning for Case-based Reasoning through Prototypes. *CoRR*, abs/1710.04806, 2017.

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

Scott M. Lundberg and Su In Lee. A unified approach to interpreting model predictions. In *Advances in Neural Information Processing Systems*. NeurIPS, 2017.

Rami Mochaourab, Arun Venkitaraman, Isak Samsten, Panagiotis Papapetrou, and Cristian R Rojas. Post-hoc explainability for time series classification: Toward a signal processing perspective. *IEEE signal processing magazine*, 2022.

Christoph Obermair et al. Explainable Machine Learning for Breakdown Prediction in High Gradient RF Cavities. *CoRR*, abs/2202.05610, 2022.

Robert Thomas Olszewski. *Generalized feature extraction for structural pattern recognition in time series data*. PhD thesis, Ann Arbor, 2001.

Keiron O'Shea and Ryan Nash. An introduction to convolutional neural networks. *CoRR*, abs/1511.08458, 2015.

Nicolas Papernot and Patrick D McDaniel. Deep k-Nearest Neighbors: Towards Confident, Interpretable and Robust Deep Learning. *CoRR*, abs/1803.0, 2018.

Sara Reardon. Rise of robot radiologists. *Nature*, 576(7787):54–54, 2019.

Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. "Why Should I Trust You?": Explaining the Predictions of Any Classifier. *CoRR*, abs/1602.04938, 2016.

Thomas Rojat et al. Explainable artificial intelligence (XAI) on timeseries data: A survey. *CoRR*, abs/2104.0, 2021.

Avanti Shrikumar, Peyton Greenside, Anna Shcherbina, and Anshul Kundaje. Interpretable Deep Learning by Propagating Activation Differences. *CoRR*, abs/1605.01713, 2016.

Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Visualising image classification models and saliency maps. *CoRR*, abs/1312.6034, 2013.

Wensi Tang, Lu Liu, and Guodong Long. Interpretable time-series classification on few-shot samples. In *International Joint Conference on Neural Networks*. IEEE, 2020.

Erico Tjoa and Cuntai Guan. A survey on explainable artificial intelligence: Toward medical xai. *IEEE transactions on neural networks and learning systems*, 32(11):4793–4813, 2020.

Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(86):2579–2605, 2008.

Stephen F Weng et al. Can machine-learning improve cardiovascular risk prediction using routine clinical data? *PloS one*, 12(4):1–14, 2017.

Jörg Wenninger. Machine Protection and Operation for LHC. *Proceedings of the Joint International Accelerator School*, 2(1):377–401, 2016.

Chih-Kuan Yeh et al. On Completeness-aware Concept-Based Explanations in Deep Neural Networks. In *Advances in Neural Information Processing Systems*. NeurIPS, 2020.

Xuchao Zhang, Yifeng Gao, Jessica Lin, and Chang-Tien Lu. Tapnet: Multivariate time series classification with attentional prototypical network. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(04):6845–6852, 2020.