Explaining Therapy Predictions with Layer-wise Relevance Propagation in Neural Networks

Yinchong Yang, Volker Tresp
Ludwig-Maximilians-Universität München
Siemens AG, Corporate Technology, Munich, Germany
{yinchong.yang, volker.tresp}@siemens.com

Marius Wunderle, Peter A. Fasching
Department of Gynecology and Obstetrics,
University Hospital Erlangen, Germany
{marius.wunderle, peter.fasching}@uk-erlangen.de

***Abstract*—In typical data analysis projects in biology and healthcare, simpler predictive models, such as regressions and decision trees, enjoy more popularity than more complex and expressive ones, such as neural networks. One reason for this is that the functioning of simpler models is easier to explain, which greatly increases user acceptance. A neural network, on the contrary, is often regarded as a black box model, because its very strength in modeling complex interactions also makes its operation almost impossible to explain. Still, neural networks remain very interesting tools, since they have demonstrated promising performance in a variety of predictive tasks, such as medical image classification and segmentation, as well as clinical event prediction, i.e., in the modeling of therapy decisions and survival time. In this work, we attempt to improve the explainability of neural networks applied in healthcare. We propose to apply the Layer-wise Relevance Propagation algorithm to explain clinical decisions proposed by deep modern neural networks. This algorithm is able to highlight the features that lead to the probabilistic prediction of therapy decisions for each individual patient. We evaluate the feature-oriented explanations generated by the algorithm with clinical experts. We show that the features, which are identified by the algorithm to be relevant, largely agree with clinical knowledge and guidelines. We believe that being able to explain machine learning based decisions greatly improves transparency and acceptance of neural network models applied in the clinical domain.**

***Keywords*-explainable machine learning, layer-wise relevance propagation, healthcare, decision support**

I. Introduction

The constantly increasing data volume and variety pose novel challenges for predictive data analysis. Especially in the task of processing data features of high dimensionality and complexity, deep neural networks have shown excellent performances. They outperform more traditional methods that rely on hand-engineered representations of data on a wide range of problems varying from image classification [1, 2, 3], machine translation[4, 5, 6] to playing video games [7, 8, 9]. To a large extent, the success of deep neural networks is attributable to their capability to represent the raw data features in a new and latent space that facilitates the predictive task [10, 11].

Also in the domain of healthcare informatics, deep neural networks have found multiple promising applications. Convolution neural networks, for instance, can be applied for the classification and segmentation of medical imaging data [12, 13, 14]. In addition, recurrent neural networks prove to be efficient in processing clinical event data [15, 16, 17, 18, 19]. The predictive prowess of these methods may assist the physicians in repetitive tasks, such as annotating radiology images and reviewing health records, and enable them to concentrate on more intellectually challenging and creative tasks [20, 21]. This new way of human-machine collaboration may greatly improve clinical service and therefore patient experience.

However, healthcare remains a problematic area where machine learning models have to be applied with great caution [22]. The fact that (not necessarily deep) neural networks lack explainability is greatly limiting their application in this domain. In May 2018, the European Union's new General Data Protection Regulation (GDPR) will take effect and restrict automated decision making produced by, e.g., algorithms [23]. Article 13 *Information to be provided where personal data are collected from the data subject* specifies that the data controller (e.g. clinics) should provide the data subject (e.g. patients) with *"meaningful information about the logic involved"*. In Article 22 *Automated individual decision-making, including profiling* states that *"The data subject shall have the right not to be subject to a decision based solely on automated processing"*, unless, e.g. the data subject explicitly consents with it (paragraph 2.c). These new clauses have a large impact on the application of machine learning methods as long as personal data are involved. A data subject will have the right to demand an explanation not only of the decision but also of the algorithm that generates the decision [24]. For clinics in the European Union, it will be a mandatory component of clinical services to provide an explanation, as long as machine learning or any algorithmic logic is applied to propose decisions. It is, therefore, a pressing task to be able to explain the predictions on the one hand, and preserve as much as possible of the expressiveness of complex neural network architectures, on the other hand.

The term *explainability* or *interpretability* in machine learning is in fact not well defined, and there are multiple ways to claim a model to be explainable [25]. A first category of models is either designed in a fashion such that it is interpretable to a human, or directly inspired by human decision making. Representative examples are decision tree models, which learn to hierarchically split the data at different cutoff values in each feature. This approach leads to intuitive and interpretable hierarchical decision rules. Another example is the k-nearest-

neighbor approach, where the prediction is based on one or multiple training samples that are most similar to a test sample. The second category of explainable models includes linear and logistic regression. One could make a distribution assumption for the regression coefficients, and perform statistical tests to quantify whether a coefficient is significantly different from 0. If that is the case, the corresponding feature can be interpreted to be relevant.

Neural networks can be seen as an extension to regression models and a Gaussian prior of the weights may be applied. However, one cannot easily explain a neural network as a regression by reading the significance of an input feature, because a multidimensional hidden layer models multiple interactions between all input features. In other words, a single input feature has multiple paths to influence the output, and the number of such paths increases exponentially with the number of layers in a network. To this end, there have been multiple works that attempt to increase the explainability of neural network models.

Essentially, there are three classes of approaches to explain a neural network. *i)* One could simplify a trained, complex neural network model. The mimic learning [26] paradigm suggests training a simple, e.g., linear regression or decision tree [27], model against the predicted value produced by a trained deep model until the simple model converges. This approach thus provides a simple and interpretable model with almost the same expressiveness as a deep neural network. However, finding a shallow regression model or decision tree for high dimensional and complex data may turn out to be challenging, because these works consider three-layered network as "simple", such as in [26]. *ii)* In contrast to *i)*, one could also complicate a neural network model further still by including attention mechanisms. For instance, attention mechanisms in RNNs [28, 29, 30] and CNNs [31, 32] are representatives of this class of approaches. They include additional modules in the network that learn to assign an attention score on each time step or pixel groups, respectively. This approach provides interpretation of the relevance of the input features, and can sometimes increase prediction quality as well. One drawback is that, by introducing additional modules, the neural networks become more complex and this would typically require longer training time and more labeled data. *iii)* In this work, we focus on a third class of approaches for explaining neural networks. Here we start with a trained neural network and try to explain and visualize its functioning as a postprocessing step. Specifically, by applying the Layer-wise Relevance Propagation (LRP) algorithm, we analyze the weight parameters in the model and attempt to figure out how much influence each input feature has w.r.t. the final prediction. A closely related method is sensitivity analysis [33], which calculates the partial derivative of each feature overall w.r.t. the target.

When we look at the p-values of regression coefficients of a simplified network as in paradigm *i)*, we make statements that a specific feature is *in general* relevant for the prediction. But in case of *ii)* the attention modules as well as *iii)* the relevance propagation and sensitivity analysis, the influence or relevance of each feature is derived for *a specific data point*.

The essential idea in the LRP algorithm is to decompose the predicted probability of a specific target into a set of relevance scores and redistribute them onto the neurons of the previous layer. The relevance scores are defined in terms of 1) the strength of the connection represented by the weight, and 2) the activation of the neuron in the previous layer. In each layer of a neural network, the relevance score can be seen as a kind of contribution that each input neuron gives to each output neuron. When this approach is applied iteratively, i.e., from the output layer down to the input layer, one would have a relevance score for each feature neuron. Especially in image data and language data, this approach has been demonstrating promising explainability in combination with CNNs and RNNs, respectively. In this work, we apply this method to real-world healthcare data. We first train an RNN-based model to predict therapy decisions, whose prediction quality is close to that of a clinical expert. We further explain these decisions with LRP and show that the derived explanations largely agree with the actual clinical knowledge and guidelines.

The remaining part of this paper is organized as follows: In Sec. II we give an overview of relevant works in i) explaining machine learning in healthcare and ii) layer-wise relevance propagation. In Sec. III we present information about our cohort and data processing steps. In Sec. IV we introduce the layer-wise relevance propagation in detail, including the original algorithm for fully-connected layers, gating neurons and relevance propagation in time. We also perform a simulation study in Sec. V on benchmark data, in order to prove our concept as well as to verify the implementation. We report our experimental results on real-world clinical data in Sec. VI, including the performance of our predictive neural network model as well as the relevance scores calculated by the propagation algorithm.

II. Related Works

a) Explaining Machine Learning in Healthcare: Explainability of machine learning models is highly desirable and encouraged in healthcare informatics. It remains, however, quite a challenging task that only a few works have addressed. Ref. [34, 35] apply knowledge distillation to predict mortality and ventilator-free days for patients with acute hypoxemic respiratory failure. The distillation is realized by training a gradient boosting tree that mimics the predicting behavior of LSTM, thus identifying relevant features that can support and contribute to robust decision making. In another work, [36], one attempts to simplify a classification model that serves as a scoring system for sleep apnea screening. It implements a super-sparse linear integer model which can provide feature selection that is in accordance with general medical findings. The work of [17] augments recurrent neural networks with two-way attention modules that are claimed to mimic decision making by the physicians. The implementation is tested on a large EHR dataset including over 200K patients with risk of heart failure. The additional sophisticated attention learning module turns out to also improve prediction quality. Ref. [18] attempts to

explain a clinical decision predicted by deep neural networks by studying latent representations. It is shown that, with the capability of RNNs to encode sequential features of variable lengths into a fixed-size vector, one could, in fact, compare patients in the learned latent space, whilst such comparison would otherwise have been impossible in the raw input space. One can thus explain and support the predicted decision by showing that the patients identified to be similar have received similar, if not the same, therapies. For an overview of the topic of interpreting or explaining machine learning in general, one could also refer to [25, 37, 38].

b) Layer-wise Relevance Propagation: A framework defining the layer-wise relevance propagation can be found in [39]. The most generic idea is to Taylor-expand a prediction made by a function $f(\boldsymbol{x})$ with respect to the input $\boldsymbol{x}$. The score is related to the increase in the contribution to the cost function for a particular data instance, when a particular input or hidden neuron is replaced by a default value. Various approaches to identify the root point in a Taylor setting yield different rules to perform the decomposition, i.e., the relevance propagation. A more detailed description of the propagation rules can also be found in [40]. Ref. [41] further provides multiple applications by demonstrating the pixels identified to be relevant in an image classification task. The specific relevance propagation rule for RNNs is discussed in [42]. It covers a couple of new rules necessary to capture the specific recurrent connection patterns, and conducts experiments in sentinel prediction of natural language data. With this method, one can derive words that contribute to a positive and negative sentiment score, respectively.

III. Cohort and Feature Processing

Our data are provided by the PRAEGNANT [43] study network and were collected on recruited patients suffering from metastatic breast cancer. We selected 1048 patients for training and 150 for testing, all of which had met the first line of medication therapy and had positive hormone receptor and negative HER2. Physicians are often not in agreement on the prescription of antihormone therapy and chemotherapy and thus our study is of significant clinical relevance.

Similar to [18], we retrieve on each patient 199 *static* features that encode, 1) demographic information, 2) the primary tumor and 3) metastasis *before being recruited in the study*. These features form for each patient i a feature vector $\boldsymbol{m}_i \in \{0,1\}^{199}$. We further include the patients' time-stamped clinical event data as *sequential* features, such as 4) local recurrences, 5) radiotherapy, 6) medication therapy, 7) diagnosed metastasis *during the study*, 8) surgery and 9) clinic visits. For the i-th patient, we encode these sequential features using an ordered set $\{\boldsymbol{x}_i^{[t]}\}_{t=1}^{T_i}$ where each $\boldsymbol{x}_i^{[t]}\{0,1\}^{189}$. T_i denotes the number of clinical events observed on the patient i, i.e., the length of the sequence. In our cohort, T_i from 0 to 15, and is on average 3.03.

Among the static features, there are originally four numerical values, including the age, the number of positive cells of estrogen receptor, the number of positive cells of progesterone receptor and the Ki-67 IHC[1]. This poses a novel challenge to the application of LRP algorithm: According to [39], the *consistency* of the relevance propagation is only guaranteed, if all input features are in the same space. In our experiments, we can confirm that the outcome of the LRP algorithm is much less explainable if the input feature consists of a mixture of numerical and binary coded categorical features, even if the numerical ones are normalized. To this end, we apply two kinds of stratification to transform the numerical features. For the feature of age, we stratify all patients into three groups of almost identical size, using the 33.3% and 66.7% quantiles. The other three features are represented following clinical practice. The number of positive cells of estrogen receptor, for instance, is stratified in two groups using one threshold of 20%. Because a percent smaller than this threshold can be a hint for chemotherapy if a number of other criteria are fulfilled as well. The same also applies to the Ki-67 IHC with a threshold of 30%, often suggesting a fast growth of the tumor cell.

IV. Method: Layer-wise Relevance Propagation

First, we denote a fully connected layer prior to activation as

$$\begin{aligned} \boldsymbol{z} &= \boldsymbol{W}\boldsymbol{x} + \boldsymbol{b} \\ \text{with } \boldsymbol{x} \in \mathbb{R}^m, \boldsymbol{z}, \boldsymbol{b} \in \mathbb{R}^n, \boldsymbol{W} &\in \mathbb{R}^{n\times m}. \end{aligned} \tag{1}$$

If it is the first layer in a neural network, the input vector $\boldsymbol{x}$ denotes the data features, otherwise $\boldsymbol{x}$ consists of the activated output neurons, i.e., $\xi(\boldsymbol{z})$ from the previous layer. Here we use $\xi(\cdot)$ to denote a generic activation function. And for the sake of notation convenience, we omit the index of the layer, by agreeing that all terms in Eq. 1 are defined within one layer.

Now, we assume that for each of the output neuron z_k, there exists a known relevance score, denoted as R_k, which is to be decomposed as

$$R_k = \sum_j R_{k\to j}, \tag{2}$$

where j indexes an input neuron x_j. This step of *decomposition* is illustrated in Fig. 1. Thus, $R_{k\to j}$ describes how much of R_k should be propagated onto x_j. This ratio is denoted as $p_{k,j}$:

$$R_{k\to j} = p_{k,j}\cdot R_k, \text{ with } \boldsymbol{P} \in \mathbb{R}^{n\times m}. \tag{3}$$

Once $R_{k\to j},\ \forall k$ are known, one collects all the relevance scores bound to be assigned to input neuron j:

$$R_j = \sum_k R_{k\to j}, \tag{4}$$

which is illustrated in Fig. 2. Calculating Eq. 3 iteratively for all input neurons $\forall j \in [1,m]$, as illustrated in Fig. 3, one redistributes the relevance scores from the output down to the input layer. Now the only term that remains to be defined is $\boldsymbol{P} \in \mathbb{R}^{n\times m}$, the matrix which contains the ratio of relevance of each output neuron $k \in [1,n]$ to be assigned to input neuron $j \in [1,m]$.

[1]KI-67 immunohistochemistry, a marker of proliferating cells.

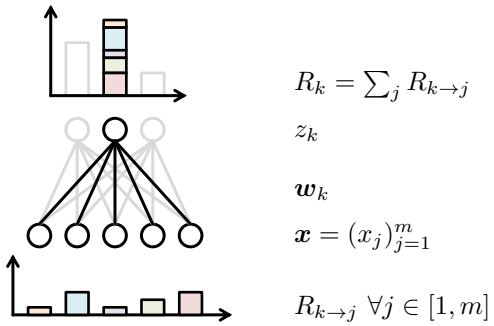

Figure 1. The decomposition of a relevance score in the output layer, corresponding to Eq. 2.

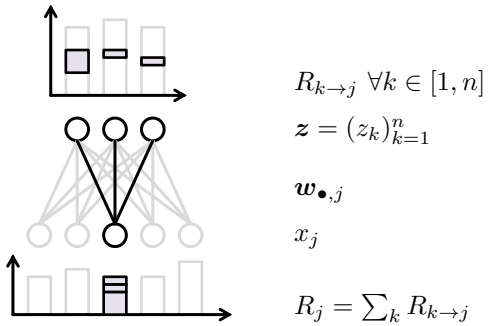

Figure 2. The collection of all relevance scores in the output, which are to be assigned to a input neuron j, corresponding to Eq. 3

Ref. [39] gives a systematic overview of various ways to define $p_{k,j}$, depending on the domain shared by all neurons in $\boldsymbol{x}$. One of the most intuitive, and empirically efficient way, according to [44], is

$$p_{k,j} = \frac{x_j \cdot w_{k,j}}{z_k} = \frac{x_j \cdot w_{k,j}}{\boldsymbol{x}^T \boldsymbol{w}_k}. \tag{5}$$

The most relevant term in Eq. 5 is the numerator $x_j \cdot w_{k,j}$, which quantifies the contribution each input neuron x_j makes to the pre-activated output neuron z_k. A highly active input neuron x_j, and a strong connection $w_{k,j}$ imply that a large contribution has been made by x_j, which in return deserves more relevance to be propagated upon [44]. The denominator, on the other hand, is exactly the pre-activated neuron z_k. It serves the normalizing purpose since z_k remains the same for all input x_j and that

$$z_k = \sum_j x_j \cdot w_{k,j} \text{ ,i.e., } \sum_j \frac{x_j \cdot w_{k,j}}{z_k} = 1, \tag{6}$$

so that Eq. 2 always holds. Furthermore, it guarantees the relevance scores are propagated without loss through the entire network, in contrast to the propagation of gradients [45, 46]. The reason is that at each layer, the LRP algorithm merely decomposes and redistributes the relevance from the upper layer onto the lower layer while preserving the total sum of the relevance scores constant. In order to stabilize the numerical

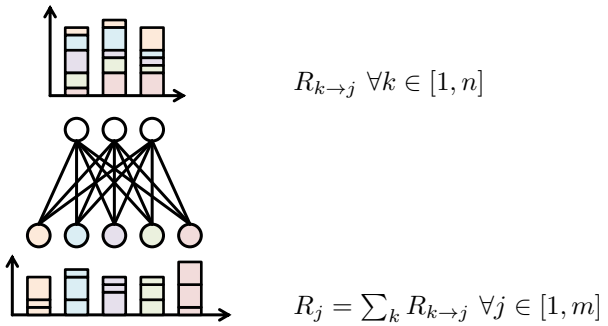

Figure 3. The redistribution of all decomposed relevance scores from the output to the input layer. It is easy to see that the redistribution is to perform both calculations illustrated in Fig. 1 and Fig. 2 for all j's and k's.

computation, [42, 44] include further stabilizers as:

$$p_{k,j} = \frac{x_j \cdot w_{k,j} + \epsilon \cdot sign(z_k)/m}{\boldsymbol{x}^T \boldsymbol{w}_k + \epsilon \cdot sign(z_k)}, \tag{7}$$

where ϵ is set to .0001 in our experiments.

We made the assumption that for each layer, the relevance score R_k is known for each output neuron z_k. If this is the topmost layer in a network that produces a probability for each class $\mathbb{P}(Y_k = 1)$, then R_k is by definition equivalent to the predicted probability. Applying the LRP algorithm to the topmost layer yields the relevance scores for the neurons in the last hidden layer, from which we can derive the relevance scores for the lower layer. Therefore, one applies iteratively the LRP algorithm from the topmost layer down to the input layer, propagating and redistributing the relevance scores from the predicted probability to input features.

LSTM (and also GRU) recurrent neural networks, which is used in our experiments, consists of multiple gating operations in the general form of

$$\boldsymbol{z} = \boldsymbol{z}^s \circ \sigma(\boldsymbol{z}^g) \tag{8}$$

$$\text{with } \boldsymbol{z}^s = \boldsymbol{W}^s \boldsymbol{x} + \boldsymbol{b}^s,\ \boldsymbol{z}^g = \boldsymbol{W}^g \boldsymbol{x} + \boldsymbol{b}^g, \tag{9}$$

where we denote the source neurons with $\boldsymbol{z}^s$, the gating neurons with $\boldsymbol{z}^g$, and the sigmoid function with $\sigma(\cdot)$. We further refer to $\boldsymbol{z}$ in this context as gated neurons. The weights $\boldsymbol{W}^g$ are trained to assign a percentage, i.e. the gating neuron, $\sigma(z_k^g)$, to every source neuron z_k^s. In the forward pass, the gating neurons $\sigma(\boldsymbol{z}^g)$ determine with the Hadamard product $\circ$ how much each input neuron should contribute to the output neurons. The gating neurons serve in fact as a special mapping layer of $\mathbb{R}^n \to \mathbb{R}^n$, in that $z_k = z_k^s \cdot \sigma(z_k^g)$. That is to say, each output neuron z_k is only connected to the input z_k^s via the weight $\sigma(z_k^g)$. Therefore, applying Eq. 5 one gets

$$p_{k,k} = \frac{z_k^s \cdot \sigma(z_k^g)}{z_k^s \cdot \sigma(z_k^g)} = 1, \tag{10}$$

$$p_{k,k'} = 0 \quad \forall k' \neq k, \tag{11}$$

$$\text{i.e., } \boldsymbol{P} = \boldsymbol{I}_n. \tag{12}$$

Exploiting the fact in Eq. 3 we can get $R_{k\to k^s} = R_k$. That is to say, a source neuron z_k^s receives the full relevance score,

Algorithm 1 Layer-wise Relevance Propagation

Input: a data point as $\boldsymbol{x} \in \mathbb{R}^{m^{(0)}}$,
a L-layered neural network with weights $\{\boldsymbol{W}^{(l)}\}_{l=1}^{L}$,
the relevance score of last layer $(R_k)_{k=1}^{n^{(L)}}$;

Output: $\{R_j^{(0)} \in \mathbb{R}\}_{j=0}^{m^{(0)}}$;

for $l = L$ **to** 1 of each layer **do**
  if Eq. 8 is satisfied: **then**
    $\boldsymbol{W}^{(l)} := \boldsymbol{W}^{s,(l)}$, $\boldsymbol{z}^{(l)} := \boldsymbol{z}^{s,(l)}$
  end if
  for $k = 1$ **to** n in each output vector **do**
    for $j = 1$ **to** m in each input vector **do**
      $p_{k,j}^{(l)} = \frac{x_j^{(l)} \cdot w_{k,j}^{(l)} + \epsilon \cdot sign(z_k^{(l)})/m^{(l)}}{(\boldsymbol{x}^{(l)})^T \boldsymbol{w}_k^{(l)} + \epsilon \cdot sign(z_k^{(l)})}$;
      $R_{k \to j}^{(l)} = p_{k,j}^{(l)} \cdot R_k^{(l)}$;
    end for
  end for
  for $j = 1$ **to** m in each input vector **do**
    $R_j^{(l-1)} = \sum_k R_{k \to j}^{(l)}$
  end for
end for

R_k, from the *gated* neuron z_k. Ref. [42], stating that the gating neurons have already determined the percentage of source neurons that are propagated in the forward pass, arrives at the same conclusion to copy the relevance scores from *gated* neurons to the source neurons.

We summarize the layer-wise relevance propagation approach in the algorithm 1. In this algorithm description, we specify the index of each layer with l. The data feature, specifically, is indexed with 0, and the output prediction with L. Each layer is a mapping function defined in $\mathbb{R}^{m^{(l)}} \to \mathbb{R}^{n^{(l)}}$. Essentially, for each $l = [L, L-1, ..., 1, 0]$, the algorithm takes as input the relevance score $(R_k)_{k=1}^{n^{(l)}}$ of the output layer $\boldsymbol{z}^{(l)}$. Each R_k is then decomposed and redistributed, to form the relevance scores of the input layer $\boldsymbol{x}^{(l)}$. These scores are therefore the outcome of each current l-th step and the input to the next $(l-1)$-th step.

In Algorithm 2, we extend the propagation to the time axis. At each time step $t \in [T, T-1, ..., 2, 1]$, we apply the Algorithm 1 twice: First, we calculate the relevance scores $\boldsymbol{R}_{\boldsymbol{x}}^{[t]}$ w.r.t. the input, by using the input-to-hidden network. In case of simple RNN, it is a fully-connected layer, but includes multiple gatings in LSTM and GRU. Second, we calculate the relevance scores $\boldsymbol{R}_{\boldsymbol{h}}^{[t-1]}$ w.r.t. the previous hidden state. The responsible module is the hidden-to-hidden network in a recurrent architecture. As in simple RNN, LSTM and GRU, it is a fully-connected layer mapping from $\boldsymbol{h}^{[t-1]}$ to $\boldsymbol{h}^{[t]}$. The output of this step is the relevance score of the hidden state of time step $[t-1]$, which forms the input of the next step in the time loop.

Algorithm 2 Layer-wise Relevance Propagation in Time

Input: a sequence of input $\{\boldsymbol{x}^{[t]} \in \mathbb{R}^m\}_{t=1}^{T}$,
a recurrent neural network $\lambda(\cdot)$,
the relevance score of the last hidden state $\boldsymbol{R}_{\boldsymbol{h}}^{[T]}$;

Output: a sequence of relevance scores $(\boldsymbol{R}_{\boldsymbol{x}}^{[t]})_{t=1}^{T}$;

for $t = T_i$ **to** 1 of each time step t **do**
  $\boldsymbol{R}_{\boldsymbol{x}}^{[t]}$:= Algorithm 1(
    current input $\boldsymbol{x}^{[t]}$,
    the input-to-hidden network,
    relevance scores of the hidden state $\boldsymbol{R}_{\boldsymbol{h}}^{[t]}$),
  if $t > 1$ **then**
    $\boldsymbol{R}_{\boldsymbol{h}}^{[t-1]}$:= Algorithm 1(
      last hidden state $\boldsymbol{h}^{[t-1]}$,
      the hidden-to-hidden network,
      relevance scores of the hidden state $\boldsymbol{R}_{\boldsymbol{h}}^{[t]}$),
  end if
end for

V. Simulation Study

In our simulation study, we validate, both qualitatively and quantitatively, that the implemented LRP algorithm is able to identify temporal patterns that indeed contribute to the prediction.

We sample random MNIST [47] digits of size 28×28 to form a sequence. The length of the sequence is a uniformly distributed number between 1 and 32. The sequence is labeled 1 if it contains one or more 0's, and is labeled 0 if it does not. We train an LSTM model for this binary classification task with 10,000 of such random sequences and test its performance on another mutually exclusive set of 1,000 sequences. We expect the LSTM to be able to learn the classification task and, more importantly, the LRP algorithm should assign a high relevance score to the 0's in the sequence. A qualitative visualization of our results can be found in Fig. 4, depicting four random sequences from the test sets.

We repeat the experiment 5 times with different seeds for sampling and model initialization and report the average of the classification accuracy. In order to evaluate the performance of the LRP algorithm quantitatively, we report the AUROC and AUPRC calculated between the ground truth locations of the zeros and the calculated relevance scores in the test sets. We denote the ground truth using binary vectors. For instance, we use a vector $[1, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0]$ to represent the ground truth of the second exemplary sequence in Fig. 4, i.e., that the 1st and 10th digits in the sequence are 0's. We measure the AUROC and AUPRC w.r.t. the ground truth vector and relevance scores generated by LRP algorithm for each test sequence [2].

[2]We publish the source codes for this simulation study at `https://github.com/Tuyki/TT_RNN/blob/master/MNISTSeq.py`, and use the same implementation for the experiments in Sec. VI.

Table I
RESULTS OF THE SIMULATION STUDY. CLASSIFICATION ACC IS THE BINARY PREDICTION ACCURACY OF THE LSTM MODEL. THE RELEVANCE SCORES ARE MEASURED IN BOTH AUROC AND AUPRC.

Classification accuracy	Relevance AUROC	Relevance AUPRC
0.972 ± 0.001	0.987 ± 0.003	0.935 ± 0.019

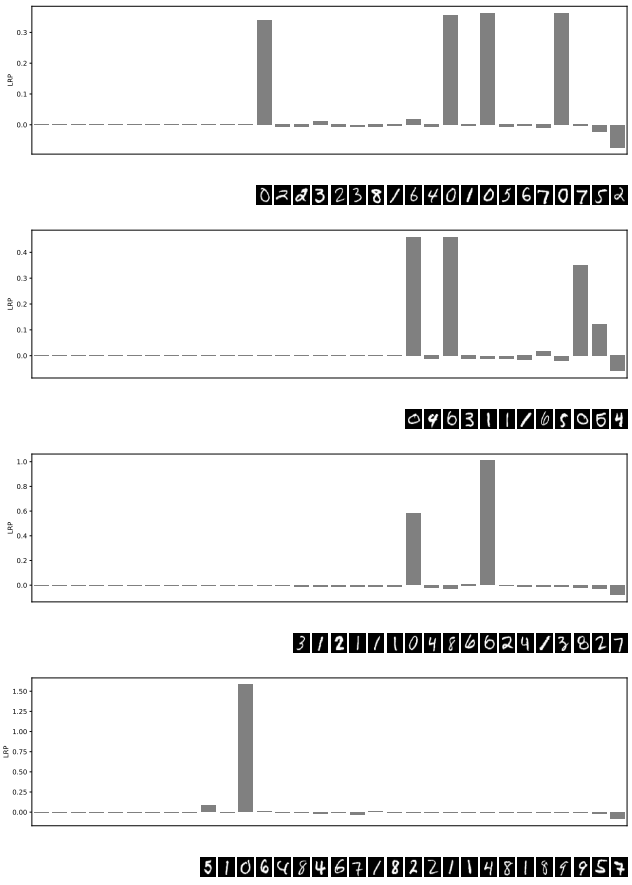

Figure 4. Visualization of the relevance scores of four random test sequences in our simulation study.

This experiment simulates the real-world patient data situation to a large extent, where we expect certain clinical events to have a relevant impact on the decision made at the end of the sequence of events, just as the 0 in the sequence decides the label. Note that for this sequence classification task, it is unimportant to study the relevance of a single pixel feature. Instead, we are more interested in the sum of relevance scores of all pixels of a digit at each time step. In the clinical data, however, we shall later cover both aspects, studying both the sequential features as well as the event type.

We can see in Fig. 4 that the 0's receive significantly higher relevance scores than the other digits do. Furthermore, it is interesting to note that, in the second sequence, the LRP assigns a high relevance score to the 3rd digit, which turns out to be a 6. The reason for this mistake is simply that this 6 is written in a fashion that strongly resembles a 0. This is, however, the empirical proof that our LSTM can indeed capture the pattern of the input in a sequence and remember it over a long period of time.

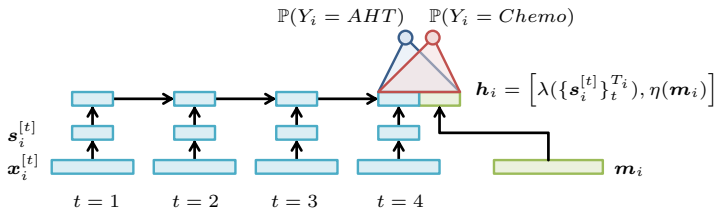

Figure 5. An illustration of the model architecture, defined in Eq. 13 14 15 and 16.

VI. EXPERIMENTS

A. The Model

The model we apply to predict the therapy decision consists of a LSTM with embedding layer and a feed-forward network:

$$\mathbb{P}(Y_i = \text{'AHT'}) = \sigma(\boldsymbol{w}_{AHT}^T \boldsymbol{h}_i + \boldsymbol{b}_{AHT}) \quad (13)$$

$$\mathbb{P}(Y_i = \text{'Chemo'}) = \sigma(\boldsymbol{w}_{Chemo}^T \boldsymbol{h}_i + \boldsymbol{b}_{Chemo}) \quad (14)$$

$$\text{with } \boldsymbol{h}_i = \left[\lambda(\{\boldsymbol{s}_i^{[t]}\}_t^{T_i}), \eta(\boldsymbol{m}_i)\right] \quad (15)$$

$$\boldsymbol{s}_i^{[t]} = \gamma(\boldsymbol{x}_i^{[t]}) \ \forall t \in [1, T_i]. \quad (16)$$

Recall that we use $\boldsymbol{x}_i^{[t]}$ to denote the sequential features observed on patient i at time step t, and $\boldsymbol{m}_i$ to denote the patient's static features. In Eq. 16, due to the sparsity and dimensionality of $\boldsymbol{x}_i^{[t]}$, we first deploy an embedding layer, denoted with function $\gamma(\cdot)$, which is expected to learn a latent representation $\boldsymbol{s}_i^{[t]}$ from $\boldsymbol{x}_i^{[t]}$ [15, 16]. An LSTM $\lambda(\cdot)$ then consumes these sequential latent representations as input. It generates at the last time step T_i another representation vector, which is expected to encode all relevant information from the entire sequence. Please note that recurrent neural networks, such as LSTMs, are able to learn a fixed-size vector from sequences of variable sizes [11, 18]. From the static features $\boldsymbol{m}_i$, which are also sparse and high dimensional, we learn a representation with a feed-forward network $\eta(\cdot)$. We concatenate both representations and form a vector $\boldsymbol{h}_i$ as in Eq. 15, which represents all relevant information on patient i up to time step T_i [15]. Finally, in Eq. 13 and 14, the vector $\boldsymbol{h}_i$ serves as input to a logistic regression with softmax output that predicts the probability that the patient should receive either antihormone ('AHT') or chemotherapy ('Chemo'). We illustrate the complete model architecture in Fig. 5.

B. Prediction Evaluation:

We first split the training set into 5 mutual exclusive sets. In turn, we tune the hyper parameters and train a model on the union of 4 sets, while evaluate the model on the

Table II
THE WEAK BASELINES: RANDOM AND MOST-POPULAR PREDICTIONS

	Log Loss	Accuracy	AUROC
Random	1.00	0.477	0.471
Most-popular	0.702	0.500	0.500

Table III
THE STRONG BASELINE: AN MLP WITH AGGREGATED SEQUENTIAL FEATURES

	Log Loss	Accuracy	AUROC
5-fold Validation sets	0.602± 0.012	0.724 ±0.015	0.798±0.011
Test set	0.589	0.715	0.806

remaining validation set. We apply the model with the best validation performance in terms of accuracy on the test set. The performances are reported in Tab. IV. With the same schema, we also report a strong baseline model, which is a two-layered feed-forward network consuming the concatenation of $\boldsymbol{m}_i$, and the aggregated sequential features $\frac{1}{T_i}\sum_{t=1}^{T_i}\boldsymbol{x}_i^{[t]}$. We report the results in Tab. III. We also include weak baselines such as random prediction and the most-popular prediction in Tab. II. The latter one constantly predicts the more popular decision in the training set for all test cases.

Furthermore, a clinician re-evaluated 69 of the 150 test cases. 75.4% of the re-evaluations turned out to agree with the ground truth, while our model achieves 81.2% accuracy for this subset of test patients. This clinical validation is based on a relatively small patient set. However, it reveals that there is often disagreement between medical experts. More importantly, we realize that, while it is extremely expensive and demanding for physicians to (re-)evaluate so many patient cases at once, a computer program can be required to perform the task anytime necessary, and could yield comparable performances to a human expert.

C. Explaining the Prediction with Relevance Scores:

Following the schema proposed in [42], we calculate the relevance score w.r.t. the correctly predicted class. Please note that we calculate the relevance scores for the *test* patient cases, which makes our experiments more challenging and realistic. Tab. V and VI summarize the static features that are most frequently identified to have contributed to the prediction of antihormone and chemotherapy, respectively.

Recall that the patients are known to have positive hormone receptors, and thus antihormone therapy seems to be the default decision. This fact is supported, for instance, by the 2nd feature "positive estrogen receptor status" and the 5th feature "positive cells of estrogen receptor $\geq 20\%$" in Tab. V. The 8th feature, the age group, suggests that the old patients should receive antihormone therapy. This also agrees with the clinical knowledge that chemotherapy often results in severe side-effects and should be prescribed with caution to elder patients.

However, it is much more interesting to study which features lead to a deviation from the default antihormone therapy and

Table IV
PREDICTION QUALITY OF OUR LSTM-BASED MODEL.

	Log Loss	Accuracy	AUROC
5-fold Validation sets	0.536 ± 0.026	0.749 ±0.035	0.834 ± 0.021
Test set	0.545	0.762	0.828

Table V
RELEVANCE RANKING OF STATIC FEATURES FOR ANTIHORMONE THERAPY

	Features	Frequencies
1	neoadjuvant therapy as first treatment: no	41
2	positive estrogen receptor status	39
3	no anti-HER2 therapy as part of first treatment	37
4	positive progesterone receptor status	31
5	positive cells of estrogen receptor $\geq 20\%$	28
6	Ki-67 IHC not identified	22
7	no chemotherapy as part of first treatment	21
8	age group: old	20
9	overall evaluation: cT2	17
10	estrogen receptor status positive cells unknown	6

choose the chemotherapy. In Tab. VI, we find features such as the 1st one of "primary tumor malignant invasive", the 8th feature of "Ki-67 IHC $\geq 30\%$", which describe an invasive primary tumor that suggests chemotherapy. Features like the 6th "G3 grading" and the metastasis in lungs, liver and lymph nodes (3rd, 4th, and 5th) depict a late stage of the metastasis. The 2nd feature of "age group: young" is also identified to have contributed to the prediction. All these factors agree with the clinical knowledge, as well as guidelines, in handling metastatic breast cancer with chemotherapy.

Tab. VII and VIII list the sequential features are frequently marked as relevant for the respective prediction. In the tables, we include the event type to which an event feature belongs using a colon. For instance, "medication therapy: antihormone therapy" means a medication therapy that has a feature of antihormone type. In Tab. VII the features "curative radiotherapy" (1st) and surgeries (2nd, 4th, and 5th) indicate an early stage of the cancer, because the patients have undergone therapies that aim at curing the primary tumor. The features of "no metastasis in liver" (7th) and "first lesion metastasis in lungs" (8th) also suggest an early phase in the development of the metastasis, which again indicates an optimistic therapy situation.

In Tab. VIII, however, we observe sequential features that support a decision for chemotherapy. Specifically, "a complete remission of metastasis" (2nd) and "local recurrence in the breast" (3rd) are hints of progressing cancer which, considering other patient features in Tab. VI, would lead to a decision for chemotherapy, from a clinical point of view.

In Tab. IX, we summarize for each event type, such as local recurrence, radiotherapy, etc., all relevance scores for antihormone and chemotherapy, respectively. This is similar to [42] and our simulation study, in that we calculate one relevance score for each time step by aggregating the relevance scores of all features observed at the time step. Because one observes only one event type at a time step, the sum of relevance scores of all

Table VI
RELEVANCE RANKING OF STATIC FEATURES FOR CHEMO THERAPY

	Features	Frequencies
1	primary tumor malignant invasive	37
2	age group: young	23
3	metastasis in lungs	23
4	metastasis in liver	23
5	metastasis in lymph nodes	18
6	G3 grading	17
7	neoadjuvant chemotherapy as part of first treatment	15
8	Ki-67 IHC $\geq$ 30%	12
9	no surgery for primary tumor	11
10	positive cells of progesterone receptor $>$ 20%	8

Table VII
RELEVANCE RANKING OF SEQUENTIAL FEATURES FOR ANTIHORMONE THERAPY

	Features	Frequencies
1	radiotherapy: curative	25
2	surgery: Excision	25
3	visit: ECOG status: alive	13
4	surgery: Mastectomy	11
5	surgery: breast preservation	9
6	radiotherapy: percutaneous	6
7	metastasis: none in liver	3
8	metastasis: first lesions of unclear dignity in lungs	2
9	medication therapy: ended due to toxic effects	2
10	medication therapy: regularly ended	2

features is in fact equivalent to the sum of relevance scores of the observed event type. The first row in the table, for instance, can be interpreted that if the patients have experienced a local recurrence, she/he should receive a chemotherapy instead of an antihormone therapy (0.772 v.s. -0.193). Another dominating decision criterion is given by the metastasis (4th row): according to the LRP algorithm, the fact that metastasis is observed in the past also strongly suggests a chemotherapy instead of an antihormone therapy (3.657 v.s. -1.192), which again agrees with clinical guidelines.

D. Patient Case Studies

So far we have studied the features that are relevant *in general* for the therapy decisions in our cohort. We further analyze the features that are relevant for *specific* patients. We demonstrate two representative patient cases: one with antihormone and the other with chemotherapy as ground truth.

The patient case A received an antihormone therapy, which our model correctly predicts with a probability of 0.754. One observes 4 events before this decision was due. The top ranking features based on relevance scores are summarized in Tab. X. The LRP algorithm assigns high relevance scores to the fact that she had a bone metastasis before being recruited in the study. Bone metastasis is seen as an optimistic metastasis because there exists a variety of bone-specific medications that effectively treat this kind of metastasis. Also, the event of curative radiotherapy, which is assigned with a high relevance score, hints at a good outcome of the therapy. Considering that the patient is in the age group of being old as well, antihormone therapy would often be recommended. In conclusion, for this

Table VIII
RELEVANCE RANKING OF SEQUENTIAL FEATURES FOR CHEMO THERAPY

	Features	Frequencies
1	medication therapy: type of following a surgery	15
2	metastasis: type of complete remission	12
3	local recurrence: in the breast	11
4	medication therapy: no surgery before or after	7
5	medication therapy: antihormone therapy	5
6	tumor board: first line met	4
7	medication therapy: for cM0/local recurrence	4
8	local recurrence: recurrence in axilla	2
9	local recurrence: invasive recurrence	2
10	medication therapy: bone specific therapy	2

Table IX
RELEVANCE SCORES SUMMARIZED W.R.T. CLINICAL EVENTS

Event type	Antihormone therapy	Chemotherapy
local recurrence	-0,193	0,772
radiotherapy	1,064	-0,398
medication therapy	2,023	-1,137
metastasis	-1,192	3,657
surgery	0,697	-0,883
visit	-0,058	0,676

specific patient, the LRP algorithm turns out to have identified relevant features that accord with clinical guidelines.

Patient B was prescribed chemotherapy, which our model predicted with probability: 0.916. 7 events have been observed before this therapy decision was due. The top ranking features based on relevance scores are summarized in Tab. XI. The static features that have been identified as relevant for the chemotherapy show a strong pattern of metastasis, including the brain, lung, and other locations. The identified sequential features include invasive local recurrences in the breast and axilla. Based on general clinical knowledge and guideline, for such a young patient with quite a malignant tumor, a chemotherapy seems indeed appropriate.

Furthermore, it is also interesting to see that the feature of being postmenopausal has a negative relevance for the decision antihormone therapy in case A, while a positive one for the chemotherapy in case B. In other words, being postmenopausal always supports the decision of chemotherapy, which agrees with clinical knowledge and guidelines.

VII. SUMMARY

It is commonly accepted that there exists a trade-off between model expressiveness and model explainability [24, 25, 38]. Especially deep and/or recurrent neural networks, though demonstrating convincing modeling capacity in a variety of machine learning tasks, suffer from lack of explainability. In healthcare, providing explanations for decisions generated by algorithms is not only a desirable characteristic but will soon become legally required. The layer-wise relevance propagation provides a solution to increase model explainability, while fully preserving model expressiveness. For each data instance, the algorithm can assign a relevance score to each feature, and thus is able to explain which feature has contributed to what degree to the final decision. To verify our implementation

Table X
THE FEATURES OF PATIENT CASE A THAT ARE IDENTIFIED BY LRP AS RELEVANT FOR A PREDICTION OF ANTIHORMONE THERAPY.

Features	Relevance score
Static	
bone metastasis	0.728
age group: old	0.160
two pregnancies	-0.030
postmenopausal	-0.057
ever hormone replacement therapy	-0.131
Sequential	
radiotherapy: curative	0.061
surgery: excision	0.061
radiotherapy: adjuvant	0.050
radiotherapy: percutaneous	0.036
medication: regularly ended	0.033
medication: first treatment	0.018
medication: antihormone therapy	0.011
surgery: breast preservation	0.010

Table XI
THE FEATURES OF PATIENT CASE B THAT ARE IDENTIFIED BY LRP AS RELEVANT FOR A PREDICTION OF CHEMOTHERAPY.

Features	Relevance score
Static	
metastasis in lungs	0.286
metastasis in brain	0.276
1st age group	0.184
other metastasis	0.139
postmenopausal	0.024
Sequential	
local recurrence: in the breast	0.048
local recurrence: invasive	0.046
local recurrence: in axilla	0.017
medication: treatment of local recurrence	0.008
medication: not related to a surgery	0.006
radiotherapy: palliative	0.005
medication: antihormone	0.005

of the LRP algorithm, we performed a simulation study on a high dimensional sequence classification task. In our experiments, we compared the generated relevance scores with general clinical knowledge and treatment guidelines, which demonstrated a large degree of agreement.

REFERENCES

[1] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[2] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 1–9.

[3] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2014, pp. 580–587.

[4] K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using rnn encoder-decoder for statistical machine translation," *arXiv preprint arXiv:1406.1078*, 2014.

[5] Y. Kim, Y. Jernite, D. Sontag, and A. M. Rush, "Character-aware neural language models." in *AAAI*, 2016, pp. 2741–2749.

[6] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," *arXiv preprint arXiv:1409.0473*, 2014.

[7] V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, and M. Riedmiller, "Playing atari with deep reinforcement learning," *arXiv preprint arXiv:1312.5602*, 2013.

[8] T. Salimans, J. Ho, X. Chen, and I. Sutskever, "Evolution strategies as a scalable alternative to reinforcement learning," *arXiv preprint arXiv:1703.03864*, 2017.

[9] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski *et al.*, "Human-level control through deep reinforcement learning," *Nature*, vol. 518, no. 7540, pp. 529–533, 2015.

[10] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *IEEE transactions on pattern analysis and machine intelligence*, vol. 35, no. 8, pp. 1798–1828, 2013.

[11] Y. Bengio, I. J. Goodfellow, and A. Courville, "Deep learning," *Nature*, vol. 521, pp. 436–444, 2015.

[12] H.-C. Shin, H. R. Roth, M. Gao, L. Lu, Z. Xu, I. Nogues, J. Yao, D. Mollura, and R. M. Summers, "Deep convolutional neural networks for computer-aided detection: Cnn architectures, dataset characteristics and transfer learning," *IEEE transactions on medical imaging*, vol. 35, no. 5, pp. 1285–1298, 2016.

[13] P. Kisilev, E. Sason, E. Barkan, and S. Hashoul, "Medical image captioning: learning to describe medical image findings using multi-task-loss cnn," 2011.

[14] B. Kayalibay, G. Jensen, and P. van der Smagt, "Cnn-based segmentation of medical imaging data," *arXiv preprint arXiv:1701.03056*, 2017.

[15] C. Esteban, O. Staeck, S. Baier, Y. Yang, and V. Tresp, "Predicting clinical events by combining static and dynamic information using recurrent neural networks," in *Healthcare Informatics (ICHI), 2016 IEEE International Conference on*. IEEE, 2016, pp. 93–101.

[16] E. Choi, M. T. Bahadori, and J. Sun, "Doctor ai: Predicting clinical events via recurrent neural networks," *arXiv preprint arXiv:1511.05942*, 2015.

[17] E. Choi, M. T. Bahadori, J. Sun, J. Kulas, A. Schuetz, and W. Stewart, "Retain: An interpretable predictive model for healthcare using reverse time attention mechanism," in *Advances in Neural Information Processing Systems*, 2016, pp. 3504–3512.

[18] Y. Yang, P. A. Fasching, and V. Tresp, "Predictive modeling of therapy decisions in metastatic breast cancer

with recurrent neural network encoder and multinomial hierarchical regression decoder," in *Proceedings of the IEEE International Conference on Healthcare Informatics (ICHI)*. Park City, Utah, USA: IEEE, 23–26 Aug 2017.

[19] ——, "Modeling progression free survival in breast cancer with tensorized recurrent neural networks and accelerated failure time model," in *Machine Learning for Healthcare 2017*, ser. Proceedings of Machine Learning Research, vol. 68. Northeastern University, Boston, USA: JMLR, 18–19 Aug 2017.

[20] V. Tresp, S. Zillner, M. J. Costa, Y. Huang, A. Cavallaro, P. A. Fasching, A. Reis, M. Sedlmayr, T. Ganslandt, K. Budde *et al.*, "Towards a new science of a clinical data intelligence," *arXiv preprint arXiv:1311.4180*, 2013.

[21] V. Tresp, M. Overhage, M. Bundschus, S. Rabizadeh, P. Fasching, and S. Yu, "Going digital: A survey on digitalization and large scale data analytics in healthcare," *arXiv preprint arXiv:1606.08075*, 2016.

[22] R. Caruana, Y. Lou, J. Gehrke, P. Koch, M. Sturm, and N. Elhadad, "Intelligible models for healthcare: Predicting pneumonia risk and hospital 30-day readmission," in *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 2015, pp. 1721–1730.

[23] Parliament and C. of the European Union, "General data protection regulation," 2016.

[24] B. Goodman and S. Flaxman, "European union regulations on algorithmic decision-making and a" right to explanation"," *arXiv preprint arXiv:1606.08813*, 2016.

[25] Z. C. Lipton, "The mythos of model interpretability," *arXiv preprint arXiv:1606.03490*, 2016.

[26] G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," *arXiv preprint arXiv:1503.02531*, 2015.

[27] N. Frosst and G. Hinton, "Distilling a neural network into a soft decision tree," *arXiv preprint arXiv:1711.09784*, 2017.

[28] V. Mnih, N. Heess, A. Graves *et al.*, "Recurrent models of visual attention," in *Advances in neural information processing systems*, 2014, pp. 2204–2212.

[29] K. Cho, A. Courville, and Y. Bengio, "Describing multimedia content using attention-based encoder-decoder networks," *IEEE Transactions on Multimedia*, vol. 17, no. 11, pp. 1875–1886, 2015.

[30] J. K. Chorowski, D. Bahdanau, D. Serdyuk, K. Cho, and Y. Bengio, "Attention-based models for speech recognition," in *Advances in Neural Information Processing Systems*, 2015, pp. 577–585.

[31] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," *arXiv preprint arXiv:1706.03762*, 2017.

[32] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhudinov, R. Zemel, and Y. Bengio, "Show, attend and tell: Neural image caption generation with visual attention," in *International Conference on Machine Learning*, 2015, pp. 2048–2057.

[33] Y. Dimopoulos, P. Bourret, and S. Lek, "Use of some sensitivity criteria for choosing networks with good generalization ability," *Neural Processing Letters*, vol. 2, no. 6, pp. 1–4, 1995.

[34] Z. Che, S. Purushotham, R. Khemani, and Y. Liu, "Distilling knowledge from deep networks with applications to healthcare domain," *arXiv preprint arXiv:1512.03542*, 2015.

[35] ——, "Interpretable deep models for icu outcome prediction," in *AMIA Annual Symposium Proceedings*, vol. 2016. American Medical Informatics Association, 2016, p. 371.

[36] B. Ustun and C. Rudin, "Supersparse linear integer models for optimized medical scoring systems," *Machine Learning*, vol. 102, no. 3, pp. 349–391, 2016.

[37] F. Doshi-Velez and B. Kim, "Towards a rigorous science of interpretable machine learning," 2017.

[38] Y. Lou, R. Caruana, and J. Gehrke, "Intelligible models for classification and regression," in *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2012, pp. 150–158.

[39] G. Montavon, S. Bach, A. Binder, W. Samek, and K.-R. Müller, "Explaining nonlinear classification decisions with deep taylor decomposition," *Pattern Recognition*, vol. 65, pp. 211–222, 2017. [Online]. Available: http://dx.doi.org/10.1016/j.patcog.2016.11.008

[40] S. Bach, A. Binder, G. Montavon, F. Klauschen, K.-R. Müller, and W. Samek, "On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation," *PLoS ONE*, vol. 10, no. 7, p. e0130140, 07 2015. [Online]. Available: http://dx.doi.org/10.1371%2Fjournal.pone.0130140

[41] A. Binder, S. Bach, G. Montavon, K.-R. Müller, and W. Samek, "Layer-wise relevance propagation for deep neural network architectures," in *Information Science and Applications (ICISA) 2016*, ser. Lecture Notes in Electrical Engineering, K. J. Kim and N. Joukov, Eds. Singapore: Springer Singapore, 2016, vol. 376, pp. 913–922. [Online]. Available: http://dx.doi.org/10.1007/978-981-10-0557-2_87

[42] L. Arras, G. Montavon, K.-R. Müller, and W. Samek, "Explaining recurrent neural network predictions in sentiment analysis," in *Proceedings of the EMNLP'17 Workshop on Computational Approaches to Subjectivity, Sentiment & Social Media Analysis (WASSA)*. Association for Computational Linguistics, 2017, pp. 159–168. [Online]. Available: http://www.aclweb.org/anthology/W17-5221

[43] P. Fasching, S. Brucker, T. Fehm, F. Overkamp, W. Janni, M. Wallwiener, P. Hadji, E. Belleville, L. Häberle, F. Taran, D. Luftner, M. Lux, J. Ettl, V. Muller, H. Tesch, D. Wallwiener, and A. Schneeweiss, "Biomarkers in patients with metastatic breast cancer and the praegnant study network," *Geburtshilfe Frauenheilkunde*, vol. 75, no. 01, pp. 41–50, 2015. [Online]. Available: http://www.praegnant.org/

[44] W. Samek, T. Wiegand, and K.-R. Müller, "Explainable artificial intelligence: Understanding, visualizing and interpreting deep learning models," *arXiv preprint arXiv:1708.08296*, 2017.

[45] Y. Bengio, P. Simard, and P. Frasconi, "Learning long-term dependencies with gradient descent is difficult," *IEEE transactions on neural networks*, vol. 5, no. 2, pp. 157–166, 1994.

[46] S. Hochreiter, "The vanishing gradient problem during learning recurrent neural nets and problem solutions," *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, vol. 6, no. 02, pp. 107–116, 1998.

[47] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.