Integration of local and global features explanation with global rules extraction and generation tools

Victor Contreras[1][0000−0002−6189−0217], Michael Schumacher[1,2][0000−0002−5123−5075], and Davide Calvaresi[1][0000−0001−9816−7439]

[1] University of Applied Sciences Western Switzerland (HES-SO), Switzerland
victor.contrerasordonez@hevs.ch
[2] Sense Innovation and Research Center, Switzerland

Abstract. Widely used in a growing number of domains, Deep Learning predictors are achieving remarkable results. However, the lack of transparency (i.e., opacity) of their inner mechanisms has raised trust and employability concerns. Nevertheless, several approaches fostering models of interpretability and explainability have been developed in the last decade. This paper combines approaches for local feature explanation (i.e., Contextual Importance and Utility – CIU) and global feature explanation (i.e., Explainable Layers) with a rule extraction system, namely ECLAIRE. The proposed pipeline has been tested in four scenarios employing a breast cancer diagnosis dataset. The results show improvements such as the production of more human-interpretable rules and adherence of the produced rules with the original model.

Keywords: Local explainability · Global explainability · Feature ranking · rule extraction.

1 Introduction

Deep Learning (DL) predictors are Machine Learning (ML) models that can learn directly from data with a "minimal" human intervention. Such predictors are widely used due to their high performance in complex tasks like image recognition [33], natural language processing [22], recommender systems [5], and autonomous control agents [28]. Despite their success and high performance, DL predictors (so-called black-boxes) are opaque — the decision-making process leading to a given outcome is unclear [24, 25, 34]. The predictors' opaqueness harms their trust and employability. Indeed, they cannot be (easily) debugged, and their complete understanding cannot be achieved. Explainable Artificial Intelligence (XAI) has emerged as a research field to provide interpretations and explanations of opaque models, shedding some light on the decision process [17].

XAI has been successfully applied to general ML techniques. Indeed, some of them could be defined as *explainable-by-design* (i.e., decision trees and linear regression) [27]. Usually, explainable-by-design models are employed as a proxy

model to explain the behavior of opaque models (*surrogate models*) [29]. Other XAI techniques are model agnostic and can be applied in the same way to any estimator like *Local Interpretable Model-Agnostic Explanations* (LIME) [15], SHAP values [6], Contextual Importance and Utility (CIU) [13], gradient-based explanations [30], explainable layers [39] and histograms of activations [37]. These techniques provide explanations in terms of feature importance and sensitivity. On the other hand, methods such as interpretable decision sets [21], RX [26], ECLAIRE [40] and TREPAN [7] perform rule extraction attempting to transform black-box neural networks models to white-box rule sets. However, explaining DL predictors is still an open research topic, which is more challenging w.r.t traditional ML models. This is due to the nature of the knowledge in DL predictors being sub-symbolic, implicit, and tacit (connectionist), which is stored (ingrained) in the estimator's architecture, weights, neurons' biases, activation functions, and gradients.

This paper proposes a new pipeline combining local and global feature explanation methods with a global rule extraction tool. In particular, the DL model is pruned with Contextual and Importance Utility (CIU) [4] – local – and Explainable Layers (ExpL) [39] – global – explanation methods and successively processed by ECLAIRE [40], the rule extraction tool. By doing so, the process produces more concise human-understanding rule sets with high adherence to the original model.

The rest of the paper is organized as follows: Section 2 presents the state of the art of DL rule extraction and local explainability features. Section 3 describes and motivates our proposed method and pipeline. Section 4 presents and analyses the results. Section 5 discusses the overall study. Finally, Section 6 concludes the paper.

2 State of the Art

On the one hand, ML models such as decision trees or linear regression can be *explainable-by-design* — they have interpretable structures, parameters, and statistics [11]. However, explainable-by-design models have limitations/constraints such as inability to deal with linear relationship (decision threes) and the sole capability of representing linear relationships (linear regression) that make them suitable only for some specific tasks/datasets [27]. On the other hand, DL models overcome such limitations. However, they are complex non-linear connectionist models which cannot be directly explained by looking through their internal parameters [8] — known as *black-boxs*. XAI methods for DL explanation can be classified as *model agnostic* if they can be applied to any model or *model specific* if they are limited to a particular model [32]. Moreover, XAI tools for DL can provide local or global explanations. Global explanation methods aim to explain the overall behavior of the model [34], whereas local explanations are limited to explaining specific data points [1].

A surrogate model is an approximation to explain complex models possibly composed of rule sets, structural models (i.e., decision trees) or feature impor-

tance (i.e., coefficients in a linear regression model). Indeed, explainable-by-design models can be employed to approximate *explanations* of DL models [27]. The quality of a surrogate model's explanation depends on how well it reflects the behavior of the original model. In XAI, this measure is known as *fidelity*. Moreover, other interesting approaches to be mentioned are feature importance analysis [10] and gradient attribution [3]. Among the most relevant tools, we can mention the following.

- Local Interpretable Model-agnostic Explanation (LIME) provides local explanations, employing random perturbations on features and sensitivity analysis to describe the relationship between the input features and the model's output [15, 41, 23]. LIME is widely used to explain classification models and has been successfully applied to explain deep learning models [9].
- *Contextual Importance and Utility* (CIU) produces local explanations based on random perturbations, Monte-Carlo simulations, and sensitivity analysis to provide the importance of features (coverage of feature variations) and utility (contextual typicity of features for a given output) [13]. CIU can produce multimodal explanations, which are textual and visual explanations. CIU's explanations are suitable for (non)experts [4, 12, 14].
- SHapley Additive exPlanations (SHAP) is another widely adopted explanation method able to produce local and global explanations through the analysis of multiplicative contributions of Shapley values, a concept inspirited by game theory [6, 20, 38]. Despite their similarity, CIU presents several advantages over LIME and SHAP, since CIU does not assume a linear relationship between features contributions to output and provides contextual utility information, which is missing in the other methods [4]. Despite its advantages over other local explanation methods, CIU presents some drawbacks like high simulation times, no inference explanations, and, like for the other local methods, the explanations produced by CIU are limited to one sample at a time.
- Explainable Layers (ExpL) are a global explanation method suitable for explaining neural networks that produce a feature importance ranking. This method introduces a new layer without bias after the input layer, connected one-on-one with the input features, acting as a measurement element which activation threshold quantifies the relative importance of input features. However, this method is only applicable for binary classification tasks on shallow neural networks, and in some cases, its values are not self-explanatory. This means that it requires additional processing to interpret them [39].

The rule extraction process (transforming a black box into a white box predictor) can be carried out using approaches such as:

- *decompositional approach*: it splits the network at a neuron level and extracts local rules neuron-by-neuron and layer-by-layer to combine them in the final rule set,

- *the pedagogical approach*: it extracts rules from a global interpretable surrogate model like decision trees or random forest, and
- *eclectic approach*: it combines the pedagogical and decompositional approaches in different phases [18].

Concerning rule extraction methods, we can mention:

- FERNN – a decompositional method to extract rules for regression neural networks with one single hidden layer. For every neuron in the hidden layer, the activation function is discretized using linear segments, from which the rules are extracted [35, 36]. Despite its effectiveness, FERNN presents several limitations. For example, and similar to other decompositional methods, this algorithm requires a pruning process on the hidden units and inputs to reduce the generated rule set complexity, which implies re-training the model several times (with a high computational cost). Additionally, FERNN applicability is limited to shallow neural networks and regression tasks, which limits its coverage.
- TREPAN: it is a pedagogical rule extraction method called TREPAN that employs decision trees to represent the whole network, generates partitions on the input space using queries, and extracts rules with the form M-of-N from the decision tree partitions. TREPAN was tested in a neural network with only one hidden layer, but like other pedagogical methods, it can be extended to multi-layer neural networks, being more flexible than FERNN in this dimension [7].
- ECLAIRE is an eclectic rule extraction method suitable for the classification task and multi-layer neural networks that produce a global set of logical rules as a white-box explanation. ECLAIRE uses intermediate representations learned in each hidden layer to create an augmented dataset, which is used to extract a set of intermediate rules that relate the intermediate representation with the output. Eventually, all intermediate rule sets are merged and substituted by feature values, composing a reconciling rule set used to make predictions or explain the estimator's decision function [40]. ECLAIRE presents several advantages like self-explainable rule sets, accurate results, and explanatory inference. However, it also presents some limitations like architecture dependency and complex sets of rules.
- RX is an eclectic method suitable for shallow feed-forward neural networks, based on clustering and genetic algorithms reporting high accuracy values, but with high computational cost [19].

This work targets the typical issues on decompositional models related to the complexity of the pruning process, local feature representation, and lack of generalization to improve the rule extraction process. Improving local representation allows understanding features better and more efficiently.

3 Methodology

Figure 1 shows the methodology we propose to overcome the limitations mentioned above. It consists of an augmented pipeline that combines local/global fea-

ture explanation (ExpL and CIU) with the rule extraction tool named ECLAIRE. Combining CIU and ExpL it is possible to provide additional information such as feature ranking, importance, and utility to prune the original model and complement the rule set explanations.

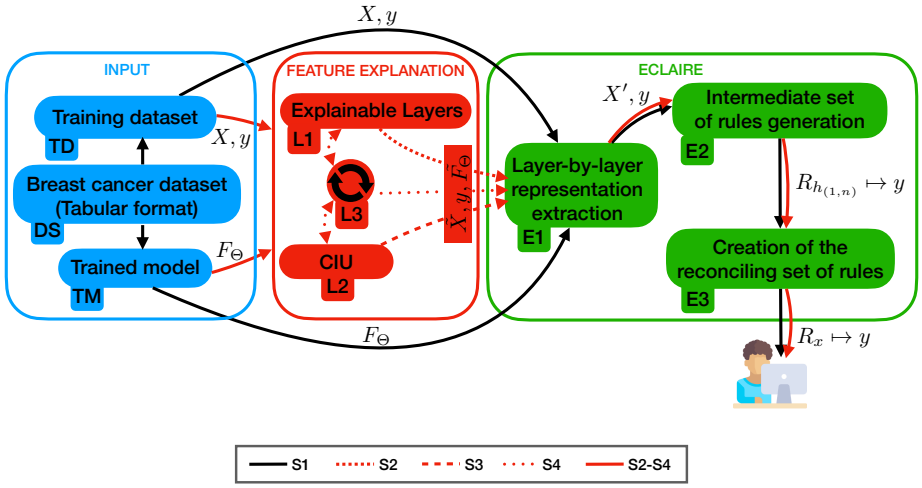

Fig. 1. Methodology and pipeline schematization.

We set up four scenarios involving three feature explanation approaches (L1, L2, L3) feeding the rule extraction process (E1 - E3). In particular:

S1 - ECLAIRE
This scenario intends to define the baseline of the original rule extraction process with ECLAIRE. The training dataset is a tuple $(\mathbf{X}, \mathbf{y})$ where $\mathbf{X} \in \mathbb{R}^{M \times N}$ is a matrix with M samples of N features and $\mathbf{y} \in \mathbb{R}^{M \times L}$ is the label matrix with M samples with L labels. The trained model F_θ is a trained DL estimator with n hidden layers. In this scenario, $(\mathbf{X}, \mathbf{y})$ and F_Θ are the sole inputs of ECLAIRE. For each data sample in $\mathbf{X}$, its corresponding class label $\hat{\mathbf{y}} = \{\hat{y}^{(1)}, \hat{y}^{(2)}, \cdots, \hat{y}^{(M)}\}$ is calculated using the "predict" method of F_Θ. In particular, in the step E1 (see Figure 1), the intermediate representations of the input data $\mathbf{X}' = \{x'^{(1)}, x'^{(2)}, \cdots, x'^{(M)}\}$, where $\mathbf{X}' = h_{1..n}(\mathbf{X})$ are extracted, and a new dataset $(\mathbf{X}', \hat{\mathbf{y}})$ is assembled. In turn, in E2, a set of intermediate rules are extracted from the intermediate dataset $(\mathbf{X}', \hat{\mathbf{y}})$ with the C5.0 tree expansion algorithm. Finally, E3 merges intermediate rule sets into the final rule set using wise substitution [40].

S2 - ExpL (L1) and ECLAIRE
In this scenario, we intend to assess if adding a global feature explanation be-

fore the rule extraction can improve the accuracy and fidelity of the obtained rule set. In particular, ExpL is applied to learn the features' weight with non-negativity constraints and rank them. Then, the original model is pruned and retrained based on the produced ranking. The pruned model ($\tilde{F_\Theta}$) is considered explainable-by-design, and it can guide the rule extraction process performed by ECLAIRE. The outcome of this step is a feature ranking (see Table 4.2) generated from $\tilde{F_\Theta}$.

Additionally, it is possible to set threshold restrictions in the form of $RELU(x_i + b_i)$ where x_i is the i^{th} feature, and b_i is the i^{th} threshold. From the threshold restriction, we could directly set binary rules with the form $If\ x_i \ \geq \ -b_i\ then\ x_i$, and this ensemble of rules is useful to guide the rule extraction process. To do so, however, further modification/merging with the internal ECLAIRE's pipeline is required.

S3 - CIU (L2) and ECLAIRE
In this scenario, we intend to assess if adding a local feature explanation before the rule extraction can improve the accuracy and fidelity of the obtained rule set. In particular, CIU is applied to determine the average importance and utility values of a set of samples ($\sim 10\%$) (randomly selected) per class. Such a selection is necessary due to the computational demand of CIU[3]. In turn, the CIU values are employed to rank the features. According to such a ranking, the most important features are kept, and the model is pruned and retrained. The expected outcome of this step is the pruned model $\tilde{F_\Theta}$, the pruned dataset $\tilde{X}$, and the features ranking.

S4 - ExpL ∩ CIU (L3) and ECLAIRE
This scenario intends to test how explainable methods differ or overlap in the feature ranking and how combining different local and global explanations methods affects the rule extraction process. After executing L1 and L2, the results are elaborated in L3 before feeding the pruned model to ECLAIRE. In particular, we select the top-ranked feature produced in L1 and L2 and intersected. Based on the resulting feature set, the model is pruned, retrained, and fed to ECLAIRE.

Overall, by adding ExpL and CIU to the pipeline, we expect the following benefits:

B1 In E1, pruned models produce clean inputs and better intermediate representations. Indeed, since the set of selected features results from a feature explanation process, the noise introduced by less relevant features is reduced.
B2 Cleaner intermediate representations improve the intermediate rule sets' accuracy produced in E2.

[3] Future studies will be conducted on more performing hardware allowing bigger sample selection. Nevertheless, CIU is a local explainable method. Thus, increasing the samples' size does not guarantee better results.

B3 The global rule set produced in E3 is more concise, human-readable, and accurate — due to the higher quality of the intermediate rule sets.

The following section presents and discusses the experimental results obtained in each scenario.

4 Results & Analysis

The methodology presented in the previous section proposes four scenarios (S1 - S4). This experimentation aims to test and compare the effect of combining three explainable methods (ExpL/CIU and ECLAIRE) in a complete pipeline to find complementarities between them and improve the accuracy and completeness of the DL models' explanations.

We selected the Breast Cancer Wisconsin (Diagnostic) Data set to execute our experiments. Concerning such a data set, the task is to perform a binary classification, predicting whether a breast tumor is benign (B) or malignant (M). The data set consists of 569 samples with 30 features (in a tabular form) extracted from digitalized images of a breast mass [2].

The model aims to diagnose (predict) if a set of features in a sample describes a malignant or benign breast tumor. The baseline model is a feed-forward neural network with two hidden layers. Before the training, the model features were scaled to the normal standard distribution (Z-score normalization). Then, the dataset was divided into train 60%, validation 20%, and test 20% with stratified sampling; this partition was employed in all the scenarios to make them comparable.

To assess and compare the experimental results on scenarios S1 to S4, we employ *accuracy* and *fidelity* as performance measures. *Accuracy* measures the quality of a classification model and can be defined as the number of correct predictions over the total number of predictions, as is shown in the equation 1. *Fidelity* measures how reliable the explanations are in reflecting the underlying model's behavior. Both accuracy and fidelity measures have an important impact on explanation trust [31]. Below, the results were organized per scenario.

$$Accuracy = \frac{Number\ of\ correct\ predictions}{Total\ number\ of\ predictions} \tag{1}$$

4.1 S1 – ECLAIRE

The trained model F_Θ is explained through a rule set produced by ECLAIRE. In this scenario, all features are employed to train the model and extract the rule set. Table 1 presents the feature importance values extracted from F_Θ model using weight connection interpretation for binary classification problems [16]. Table 2 shows the rule set generated by ECLAIRE with sequential Keras model, and Table 3 shows the non-degenerated rule set produced after the implementation of F_Θ in the Keras functional APIs [40].

Weight connection Analysis			
Feature	Value	Feature	Value
area_se	1.48	compactness_mean	0.37
radius_mean	1.45	perimeter_se	-0.31
concavity_worst	1.22	concavity_se	0.31
area_mean	1.16	concave points_se	-0.24
texture_worst	0.92	perimeter_worst	0.20
concavity_mean	0.88	texture_se	0.20
concave points_mean	0.87	radius_worst	0.17
fractal_dimension_se	-0.80	smoothness_worst	0.14
perimeter_mean	0.75	symmetry_se	0.13
radius_se	0.62	symmetry_mean	0.12
symmetry_worst	0.59	area_worst	0.11
smoothness_mean	-0.49	compactness_se	-0.09
fractal_dimension_mean	0.48	compactness_worst	0.07
fractal_dimension_worst	0.47	smoothness_se	0.06
texture_mean	0.38	concave points_worst	0.02

Table 1. Feature ranking for F_Θ model using weight connection interpretation for binary classification task.

Rule
IF radius_mean $>$ 12.76 THEN benign
IF radius_mean $\leq$ 12.76 THEN malign

Table 2. Degenerated rule set extracted by ECLAIRE in S1.

Rule
IF ('perimeter_worst'$\leq$ 115.90) $\vee$ ('concave points_worst'$\leq$ 0.14) THEN benign
IF ('perimeter_worst' $>$ 115.90) $\vee$ ('concave points_worst' $>$ 0.14) THEN malign

Table 3. Non-degenerated rule set extracted by ECLAIRE in S1.

Table 10 shows the performance obtained in this scenario: the average time for execution $\pm$ the standard deviation, fidelity (value between 0 and 1), measures the similarity of the predictions provided by the F_Θ w.r.t those of the rule set, and accuracy (the total correct over total predictions).

The performance obtained in this scenario is low fidelity and low accuracy (i.e., 42% fidelity and 46% accuracy) for the degenerate rule set (see Table 2) — composed of only one feature. By modifying the structure of the model, we can obtain non-degenerate rule sets. With such a model, we executed additional experiments, achieving a fidelity value of 91% and an accuracy of 90%.

Comparing the rule sets in Tables 2 and Table 3, the degenerate rule set contains only one feature and the non-degenerate one contains two. Although the number of features used in the rules is similar, the difference lies in the quality of features. For example, features like "perimeter_worst" and "concave points_worst" (selected in the non-degenerate case) are more informative than "radius_mean" (selected in the degenerate case) even if this feature is at the top of the ranking.

4.2 S2 – ExpL & ECLAIRE

In this scenario, F_Θ model is pruned based on the feature ranking calculated using a post-hoc ExpL with non-negativity constraints. The feature ranking shown in Table 4.2 is obtained from the weights learned by the ExpL layer. Features with near-zero ranking values have been removed from the dataset. The pruned model $\tilde{F_\Theta}$ is obtained by introducing an interpreted-by-design ExpL layer with threshold constraints. Table 5 shows the rule set extracted from $\tilde{F_\Theta}$ with the ExpL layer.

Features importance in ExpL			
Feature	Value	Feature	Value
area_mean	0.29	smoothness_worst	0.11
concave points_worst	0.24	compactness_mean	0.11
perimeter_mean	0.24	concave points_se	0.08
concave points_mean	0.21	fractal_dimension_worst	0.07
concavity_mean	0.19	compactness_se	0.03
radius_se	0.19	concavity_se	0.02
fractal_dimension_se	0.18	radius_mean	0.0
texture_worst	0.18	smoothness_mean	0.0
area_se	0.17	symmetry_mean	0.0
texture_se	0.16	fractal_dimension_mean	0.0
perimeter_se	0.15	symmetry_se	0.0
texture_mean	0.15	radius_worst	0.0
smoothness_se	0.14	perimeter_worst	0.0
compactness_worst	0.14	area_worst	0.0
symmetry_worst	0.13	concavity_worst	0.0

Table 4. Feature importance ranking from ExpL with non-negativity constrains.

Rule
IF ('concave points_worst' $\leq$ 0.14) $\vee$ [('area_mean' $\leq$ 747.19) $\wedge$ ('concave points_worst' $\leq$ 0.17)] $\vee$ ('concave points_mean' $\leq$ 0.05) THEN benign
IF ('concave points_mean' $>$ 0.05) $\vee$ [('area_mean' $>$ 747.19) $\wedge$ ('concave points_worst' $>$ 0.13)] $\vee$ ('concave points_worst' $>$ 0.17) THEN malign

Table 5. Rule set from pruned model with ExpL layer.

The rule set in Table 5 is compact and composed of three features including"concave point_worst", "area_mean", and "concave points_mean" (ranked in the top 5 features). Post-hoc ExpL appeared to be an accurate estimator for feature importance. The performance obtained in S2 is characterized by high fidelity (i.e., 98%) and high accuracy (96%). However, high accuracy and fidelity degraded the execution time due to the introduction of the ExpL threshold layer, which passed from $\sim 454ms$ to $\sim 1200ms$.

4.3 S3 – CIU & ECLAIRE

In this scenario, the F_Θ model is pruned based on the feature importance ranking calculated by the CIU method. As CIU is a local explainable method, the contextual importance (CI) and contextual utility (CU) have significant variability even with samples that belong to the same class. To produce a global approximation for CI and CU, we conducted stratified sampling over the train set and selected $\sim 10\%$ of the training set. For each selected sample, the values of CI and CU have been calculated and then averaged to produce the feature ranking (see Table 6). According to such a ranking, the least important features have been removed. In turn, $\tilde{F_\Theta}$ has been trained and fed to ECLAIRE. Table 7 shows the rule set obtained from $\tilde{F_\Theta}$.

Features	Mean Contextual Importance	Mean Contextual Utility
area_worst	0.60	0.81
fractal_dimension_se	0.60	0.87
radius_mean	0.59	0.84
smoothness_se	0.52	0.87
concavity_se	0.52	0.89
area_mean	0.50	0.77
concavity_mean	0.49	0.87
radius_worst	0.48	0.80
texture_mean	0.46	0.80
concave points_worst	0.42	0.87
smoothness_worst	0.41	0.89
fractal_dimension_worst	0.39	0.85
symmetry_se	0.39	0.88
texture_se	0.37	0.69
texture_worst	0.36	0.86
area_se	0.36	0.69
compactness_se	0.24	0.72
concave points_se	0.23	0.85
smoothness_mean	0.23	0.79
concave points_mean	0.22	0.78
perimeter_worst	0.21	0.90
symmetry_mean	0.19	0.84
radius_se	0.19	0.74
compactness_worst	0.18	0.76
symmetry_worst	0.18	0.83
perimeter_se	0.18	0.83
concavity_worst	0.17	0.73
compactness_mean	0.13	0.74
fractal_dimension_mean	0.08	0.65
perimeter_mean	0.07	0.79

Table 6. Feature importance ranking obtained with CIU.

Rule
IF ('concavity_mean' $\leq$ 0.09) $\vee$ [('radius_worst' $\leq$ 16.97) $\wedge$ ('concavity_mean' $\leq$ 0.11)] $\vee$ [('radius_worst' $\leq$ 16.97) $\wedge$ ('concave points_worst' $\leq$ 0.13)] THEN benign
IF ('radius_worst' $>$ 16.97) $\vee$ ('concavity_mean' $>$ 0.11) $\vee$ [('radius_worst' $>$ 16.84) $\wedge$ ('concavity_mean' $>$ 0.09)] THEN malign

Table 7. Rule set from pruned model with CIU.

The resulting rule set (see Table 7) is compact (with only three features) and it includes "concave point_worst", "radius_worst", and "concativity_mean" are

in the top 10 features ranked by CIU. The performance obtained in S3 has an accuracy of 90% and fidelity of 94%. Differently from S2, achieving such high-quality results positively affected the time performance recording an average execution time of $391ms$ w.r.t the $454ms$ of S1.

4.4 S4 – $ExpL \cap CIU$ & ECLAIRE

The feature rankings obtained intersecting the outcomes of CIU and ExpL are used to prune F_Θ and produced the set of features shown in Table 8). Differently from S2, $\tilde{F}_\Theta$ does not include an ExpL as a layer. Table 9 shows the rule set extracted from the $\tilde{F}_\Theta$ with the features in common among CIU and ExpL.

Features	Mean CI	Mean CU	ExpL
fractal_dimension_se	0.60	0.87	0.18
smoothness_se	0.52	0.87	0.14
area_mean	0.50	0.77	0.29
concavity_mean	0.49	0.87	0.19
texture_mean	0.46	0.81	0.15
concave points_worst	0.42	0.87	0.24
smoothness_worst	0.41	0.89	0.11
texture_se	0.37	0.69	0.16
texture_worst	0.36	0.86	0.18
area_se	0.36	0.69	0.17

Table 8. Feature importance ranking for common features in $CIU \cap ExpL$.

Rule
IF ('concave points_worst' $\leq$ 0.14) THEN benign
IF ('concave points_worst' $>$ 0.14) THEN malign

Table 9. Rule set from pruned model with $CIU \cap ExpL$.

The rule set produced in this scenario is shown in Table 9. It is extremely compact. Indeed, it contains only the feature "concave points_worst", which is highly informative for this data set. The performance obtained in this scenario is high fidelity (92%) and high accuracy (90%). Differently from S3, the high-quality results affect the execution time, incrementing the average time and its variance (i.e., $\sim 607ms$). Table 10 summarizes the performance of the four scenarios.

Scenario	Time ± std	Fidelity	Accuracy
S1 (Eclaire)	188 ms ± 20.4 ms	0.42	0.46
S1 Non-degenerate (Eclaire)	454 ms ± 34.6 ms	0.91	0.90
S2 (ExpL + Eclaire)	1200 ms ± 42.9 ms	0.98	0.96
S3 (CIU + Eclaire)	391 ms ± 34.9 ms	0.94	0.90
S4 ($CIU \cap ExpL$ + Eclaire)	607 ms ± 253 ms	0.92	0.90

Table 10. Performance measure for scenarios S1 - S4.

5 Discussion

This study has tested three feature ranking methods in four scenarios. These methods are weight connection interpretation (see Table 1 for S1), post-hoc ExpL (see Table 4.2 for S2), CIU (see Table 6 for S3) and the intersection of ExpL and CIU (see Table 8 for S4). Comparing the feature rankings and the rule sets produced in each scenario, we found that the feature "concave points_worst" is present in 4 out of 5 rule sets (except in S1 - degenerate). Moreover, in the scenarios where "concave points_worst" is present, accuracy and fidelity values are $\leq$ 90%. Thus, we can conclude that such a feature is highly informative and discriminative for the dataset under assessment. Comparing the position of "concave points_worst" in feature rankings, we found it in the top 10 in S2 - S4 and in the last position in S1. Such a difference can be explained by several factors:

H1 The weight connection interpretation method used to rank features in S1 does not accurately describe the feature importance for the model.
H2 The F_Θ model does not consider "concave points_worst" informative enough.

To test hypotheses H1 and H2, we performed additional experimentation explaining F_Θ model with ExpL. The results are shown in Table 11. In this feature ranking, "concave points_worst" is in the top 5. Additionally, CIU with a representative sample experiment was executed on F_Θ, and the feature importance is shown in Table 12 where "concave points_worst" appears in the top 10. These two experiments support H1.

To test hypothesis 2, we evaluated the performance of model F_Θ on the test set, obtaining an accuracy value of 96%. Then, we removed the feature "concave points_worst" and re-train the model, obtaining an accuracy of 95%. This supports H2. In such a case, other features can provide similar information replacing "concave points_worst".

Adding an explainable layer by design with threshold constrain, as done in S2, improved the model's accuracy and fidelity compared with the other scenarios on the expenses of the execution time. Even though the performance improvement is minimal ($\sim$ 5%), this improvement might suggest a promising research path once the redundancies are removed (i.e., between L1, L2, L3, and E1).

Comparing performance values between scenarios, we found that fidelity is higher ($\sim$ 4%) in pruned models than in the baseline. The fidelity improves in

a pruned model because removing noisy features improves the rule extraction process.

Features importance in ExpL			
Feature	Value	Feature	Value
area_se	0.09	texture_mean	0.0
area_worst	0.08	perimeter_mean	0.0
perimeter_se	0.08	area_mean	0.0
radius_mean	0.08	smoothness_mean	0.0
concave points_worst	0.08	compactness_mean	0.0
radius_worst	0.08	concavity_mean	0.0
concave points_mean	0.08	symmetry_mean	0.0
compactness_worst	0.08	radius_se	0.0
concavity_worst	0.08	smoothness_se	0.0
symmetry_worst	0.08	concavity_se	0.0
smoothness_worst	0.07	concave points_se	0.0
compactness_se	0.07	symmetry_se	0.0
fractal_dimension_se	0.03	texture_worst	0.0
texture_se	0.02	perimeter_worst	0.0
fractal_dimension_mean	0.01	fractal_dimension_worst	0.0

Table 11. Feature ranking for S1 with ExpL.

Features importance in CIU			
Feature	Value	Feature	Value
area_worst	0.99	fractal_dimension_worst	0.02
fractal_dimension_se	0.94	symmetry_se	0.02
area_se	0.48	perimeter_worst	0.01
concavity_se	0.41	perimeter_se	0.01
area_mean	0.35	smoothness_worst	0.01
radius_mean	0.23	texture_se	0.01
concave points_mean	0.13	perimeter_mean	0.01
radius_worst	0.10	compactness_mean	0.01
radius_se	0.09	concave points_se	0.01
concave points_worst	0.06	symmetry_mean	0.0
symmetry_worst	0.05	compactness_worst	0.0
concavity_mean	0.04	concavity_worst	0.0
texture_worst	0.03	fractal_dimension_mean	0.0
texture_mean	0.03	compactness_se	0.0
smoothness_se	0.02	smoothness_mean	0.0

Table 12. Feature ranking for S1 with CIU.

6 Conclusions

This paper proposed a new methodology to overcome limitations in decompositional rule extraction processes related to the complexity of the pruning process, local feature representation, and lack of generalization. The proposed pipeline combines local/global feature explanation tools (ExpL and CIU) with the rule extraction tool named ECLAIRE. The results indicate that

- Different rule sets can be equally valid and reach similar performance. Indeed, there is more than one valid solution to complex problems like describing the decision boundaries of DL models.
- Introducing an Explainable layer (ExpL) in the model can guide ECLAIRE during the rule extraction process, increasing the accuracy and fidelity scores. However, it requires more execution time, reflecting a trade-off between quality and execution time.
- A concise and accurate rule set can be obtained using CIU and Explainable layers (ExpL) in combination with ECLAIRE. Moreover, combining CIU and ExpL produces a short and accurate feature explanation that reduces the number of intermediate rules reaching a shorter and more accurate rule set.
- Feature importance rankings generated with ExpL and CIU add contextual information to the logic rule set generated by ECLAIRE, complementing the logic explanation with reasons that describe why certain features were selected over others.

The future work is two-folded. In particular,

FW1 Improvement of other rule extraction algorithms that require an iterative pruning and re-training (i.e., FERNN).
FW2 ExpL limits its explanation to the first hidden layer. We envision amending the inner mechanism to produce explanations for all the hidden layers as an element to distill a logic model from a DNN.
FW3 To integrate the revised ExpL (see FW2) within the step E1 of ECLAIRE.

Acknowledgments

This work is supported by the Chist-Era grant CHIST-ERA19-XAI-005, and by *(i)* the Swiss National Science Foundation (G.A. 20CH21_195530), *(ii)* the Italian Ministry for Universities and Research, *(iii)* the Luxembourg National Research Fund (G.A. INTER/CHIST/19/14589586), *(iv)* the Scientific, and Research Council of Turkey (TÜBİTAK, G.A. 120N680).

References

1. Adebayo, J., Gilmer, J., Goodfellow, I., Kim, B.: Local explanation methods for deep neural networks lack sensitivity to parameter values. arXiv preprint arXiv:1810.03307 (2018)

2. Agarap, A.F.M.: On breast cancer detection: an application of machine learning algorithms on the wisconsin diagnostic dataset. In: Proceedings of the 2nd international conference on machine learning and soft computing. pp. 5–9 (2018)
3. Ancona, M., Ceolini, E., Öztireli, C., Gross, M.: Gradient-based attribution methods. In: Explainable AI: interpreting, explaining and visualizing deep learning, pp. 169–191. Springer (2019)
4. Anjomshoae, S., Främling, K., Najjar, A.: Explanations of black-box model predictions by contextual importance and utility. In: International Workshop on Explainable, Transparent Autonomous Agents and Multi-Agent Systems. pp. 95–109. Springer (2019)
5. Batmaz, Z., Yurekli, A., Bilge, A., Kaleli, C.: A review on deep learning for recommender systems: challenges and remedies. Artificial Intelligence Review **52**(1), 1–37 (2019)
6. Van den Broeck, G., Lykov, A., Schleich, M., Suciu, D.: On the tractability of shap explanations. In: Proceedings of the 35th Conference on Artificial Intelligence (AAAI) (2021)
7. Craven, M.W., Shavlik, J.W.: Using sampling and queries to extract rules from trained neural networks. In: Machine learning proceedings 1994, pp. 37–45. Elsevier (1994)
8. Dağlarli, E.: Explainable artificial intelligence (xai) approaches and deep meta-learning models. Advances and applications in deep learning **79** (2020)
9. Di Cicco, V., Firmani, D., Koudas, N., Merialdo, P., Srivastava, D.: Interpreting deep learning models for entity resolution: an experience report using lime. In: Proceedings of the Second International Workshop on Exploiting Artificial Intelligence Techniques for Data Management. pp. 1–4 (2019)
10. Dickinson, Q., Meyer, J.G.: Positional shap (poshap) for interpretation of deep learning models trained from biological sequences. bioRxiv (2021)
11. Du, M., Liu, N., Hu, X.: Techniques for interpretable machine learning. Communications of the ACM **63**(1), 68–77 (2019)
12. Fouladgar, N., Alirezaie, M., Främling, K.: Decision explanation: applying contextual importance and contextual utility in affect detection. In: Italian Workshop on Explainable Artificial Intelligence, XAI. it 2020, co-located with 19th International Conference of the Italian Association for Artificial Intelligence (AIxIA 2020), Online Event, November 25-26, 2020. pp. 1–13. Technical University of Aachen (2020)
13. Främling, K.: Explaining results of neural networks by contextual importance and utility. In: Proceedings of the AISB'96 conference. Citeseer (1996)
14. Främling, K.: Contextual importance and utility: atheoretical foundation. arXiv preprint arXiv:2202.07292 (2022)
15. Garreau, D., Luxburg, U.: Explaining the explainer: A first theoretical analysis of lime. In: International Conference on Artificial Intelligence and Statistics. pp. 1287–1296. PMLR (2020)
16. Garson, D.G.: Interpreting neural network connection weights (1991)
17. Gunning, D., Stefik, M., Choi, J., Miller, T., Stumpf, S., Yang, G.Z.: Xai—explainable artificial intelligence. Science Robotics **4**(37), eaay7120 (2019)
18. Hailesilassie, T.: Rule extraction algorithm for deep neural networks: A review. arXiv preprint arXiv:1610.05267 (2016)
19. Hruschka, E.R., Ebecken, N.F.: Extracting rules from multilayer perceptrons in classification problems: A clustering-based approach. Neurocomputing **70**(1-3), 384–397 (2006)

20. Kokalj, E., Škrlj, B., Lavrač, N., Pollak, S., Robnik-Šikonja, M.: Bert meets shapley: Extending shap explanations to transformer-based classifiers. In: Proceedings of the EACL Hackashop on News Media Content Analysis and Automated Report Generation. pp. 16–21 (2021)
21. Lakkaraju, H., Bach, S.H., Leskovec, J.: Interpretable decision sets: A joint framework for description and prediction. In: Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining. pp. 1675–1684 (2016)
22. Le, H., Vial, L., Frej, J., Segonne, V., Coavoux, M., Lecouteux, B., Allauzen, A., Crabbé, B., Besacier, L., Schwab, D.: Flaubert: Unsupervised language model pretraining for french. arXiv preprint arXiv:1912.05372 (2019)
23. Lee, E., Braines, D., Stiffler, M., Hudler, A., Harborne, D.: Developing the sensitivity of lime for better machine learning explanation. In: Artificial Intelligence and Machine Learning for Multi-Domain Operations Applications. vol. 11006, p. 1100610. International Society for Optics and Photonics (2019)
24. Lei, D., Chen, X., Zhao, J.: Opening the black box of deep learning. arXiv preprint arXiv:1805.08355 (2018)
25. London, A.J.: Artificial intelligence and black-box medical decisions: accuracy versus explainability. Hastings Center Report **49**(1), 15–21 (2019)
26. Lu, H., Setiono, R., Liu, H.: Effective data mining using neural networks. IEEE transactions on knowledge and data engineering **8**(6), 957–961 (1996)
27. Molnar, C.: Interpretable machine learning. Lulu. com (2020)
28. Niroui, F., Zhang, K., Kashino, Z., Nejat, G.: Deep reinforcement learning robot for search and rescue applications: Exploration in unknown cluttered environments. IEEE Robotics and Automation Letters **4**(2), 610–617 (2019)
29. Nóbrega, C., Marinho, L.: Towards explaining recommendations through local surrogate models. In: Proceedings of the 34th ACM/SIGAPP Symposium on Applied Computing. pp. 1671–1678 (2019)
30. Pan, D., Li, X., Zhu, D.: Explaining deep neural network models with adversarial gradient integration. In: Thirtieth International Joint Conference on Artificial Intelligence (IJCAI) (2021)
31. Papenmeier, A., Englebienne, G., Seifert, C.: How model accuracy and explanation fidelity influence user trust. arXiv preprint arXiv:1907.12652 (2019)
32. Quinn, T.P., Gupta, S., Venkatesh, S., Le, V.: A field guide to scientific xai: Transparent and interpretable deep learning for bioinformatics research. arXiv preprint arXiv:2110.08253 (2021)
33. Recht, B., Roelofs, R., Schmidt, L., Shankar, V.: Do imagenet classifiers generalize to imagenet? In: International Conference on Machine Learning. pp. 5389–5400. PMLR (2019)
34. Samek, W., Montavon, G., Lapuschkin, S., Anders, C.J., Müller, K.R.: Explaining deep neural networks and beyond: A review of methods and applications. Proceedings of the IEEE **109**(3), 247–278 (2021)
35. Setiono, R., Leow, W.K.: Fernn: An algorithm for fast extraction of rules from neural networks. Applied Intelligence **12**(1), 15–25 (2000)
36. Setiono, R., Leow, W.K., Zurada, J.M.: Extraction of rules from artificial neural networks for nonlinear regression. IEEE transactions on neural networks **13**(3), 564–577 (2002)
37. Stano, M., Benesova, W., Martak, L.S.: Explaining predictions of deep neural classifier via activation analysis. arXiv preprint arXiv:2012.02248 (2020)
38. Sundararajan, M., Najmi, A.: The many shapley values for model explanation. In: International conference on machine learning. pp. 9269–9278. PMLR (2020)

39. Zach, J.: Interpretability of deep neural networks (2019)
40. Zarlenga, M.E., Shams, Z., Jamnik, M.: Efficient decompositional rule extraction for deep neural networks. arXiv preprint arXiv:2111.12628 (2021)
41. Zhang, Y., Song, K., Sun, Y., Tan, S., Udell, M.: " why should you trust my explanation?" understanding uncertainty in lime explanations. arXiv preprint arXiv:1904.12991 (2019)

A Appendix Feature description

Feature	**Description**
area_se	Area standard error
area_worst	Average of three largest area values
perimeter_se	Perimeter standard error
radius_mean	Average of cell's radius value
concave points_worst	Average of three largest concave points in the contour values
radius_worst	Average of three largest radius values
concave points_mean	Average of concave points in the contour
compactness_worst	Average of three largest compactness values
concavity_worst	Average of three largest concavity values
symmetry_worst	Average of three largest symmetry values
smoothness_worst	Average of three largest smoothness values
compactness_se	Compactness standard error
fractal_dimension_se	Fractal dimension standard error
texture_se	Texture standard error
fractal_dimension_mean	Fractal dimension mean
texture_mean	Texture mean
perimeter_mean	Perimeter mean
area_mean	Area mean
smoothness_mean	Smoothness mean
compactness_mean	Compactness mean
concavity_mean	Concavity mean
symmetry_mean	Symmetry mean
radius_se	Radius standard error
smoothness_se	Smoothness standard error
concavity_se	Concavity standard error
concave points_se	Concavity points standard error
symmetry_se	Symmetric standard error
texture_worst	The average of the largest three texture values
perimeter_worst	The average of the largest three texture values
fractal_dimension_worst	The average of the largest three fractal dimension values

Table 13. Breast Cancer Dataset Feature description.