Seminars in AI & Robotics (2021/2022)
Paper 1: Explainable Artificial Intelligence

Leandro Maglianella[1]

[1]La Sapienza, University of Rome

Abstract

This paper is about Explainable Artificial Intelligence (xAI), a specific new branch in the Machine Learning (ML) ecosystem, which aims at overcoming the intrinsic limitations of most of ML models and algorithms by providing explanations to them and, consequently, making their outcomes more reliable for users. In fact, it is not rare for some Machine Learning models to be described as 'black-box' models, meaning that the way in which they output their predictions or perform decisions on input data is not explicit or transparent enough or even understandable by humans. In this work, I explain the general xAI definitions and how its approaches are partitioned in the literature. Then, some of the current state-of-the-art techniques peculiar to this field of study are explored.

1. Introduction

The scientific and technological advancements crafted during the past decade have made very clear that Artificial Intelligence (AI) is quickly becoming more and more ubiquitous in any type of economic sector of our modern societies. As an example, we could consider the recent tremendous growth in global investment in AI: if in 2017 it amounted to about 10 billion US dollars, in 2021 this value has increased to about 80 billion US dollars (surpassing the estimated value, which was only 50 billion) and it is expected to skyrocket even more in the near future [1]. The wide use of AI has brought it to be highly prolific even in very different contexts where the decision-making process is trivially relevant, for instance in Advertisement, Finance, Healthcare, Legal, Military and Transportation applications. In critical domains as the ones just mentioned it is crucial for the used Machine Learning (ML) methods to be transparent, meaning that it must be possible to understand the reasons of an output given by it: this is important not only to ensure human control on AI but also to comprehend more deeply the reasoning modeled by a machine, enabling further improvements and discoveries. Unfortunately, however, too often this does not happen and we rely on unexplainable models in order to just get the highest possible performances neglecting their explainability; in fact usually the interpretability and the accuracy of a model are inversely proportional. For instance, usually Deep Neural Networks (DNNs) are capable of outperforming other kinds of Machine Learning models with respect to the most generally accepted evaluation metrics, but, unfortunately, they hugely lack on the interpretability aspects. Moreover, sometimes NNs can be very delicate architectures which predictions can be deeply corrupted even by the most insignificant variation [2]. On the other hand, simple and intrinsically-explainable architectures often guarantee a clear comprehension of their decision-making routine but they are also usually unreliable because of their poor performances in the related task and therefore not applicable in the real world [3]. It is now easy to understand how, from these premises, a research field that tries to solve all these problems was born: we are talking about Explainable Artificial Intelligence (xAI).

In the continuation of this paper, I will keep on explaining the general xAI definitions and the scientific community's efforts in its direction. Then, it will be described how xAI approaches are usually classified in literature following both a method and explanation partition fashion. Finally, I will study the current state-of-the-art techniques, in particular LIME, Kernel SHAP and CAM methods.

2. Explainable Artificial Intelligence

Explainable Artificial Intelligence (xAI) is the recently rising branch of ML which tries to fulfil the needs of implementation's transparency for critical application domains. No agreement has been yet reached on a formal definition for xAI, nevertheless, we can state that the general goal in this field of study consists in finding and developing techniques that can merge together high performances and high explainable capabilities for understanding the decisions made by ML algorithms. Therefore, a real xAI approach must contain some sort of system that allows humans to study and understand how an input is mathematically mapped to an output in the ML model.

Talking about xAI, we must also point out a different

maglianella.1792507@studenti.uniroma1.it (L. Maglianella)
www.linkedin.com/in/leandro-maglianella (L. Maglianella)

CEUR Workshop Proceedings (CEUR-WS.org)

topic: in fact, xAI is not just implemented to support humans' understanding of the machine, but can also be seen as a way to solve a more socio-ethical issue regarding the responsibility and accountability of an autonomous rational agent for the outcomes it produces and suggests. In fact, as known, in order for an artificial intelligence to become "general" and to be eventually considered as an evolved entity, in the scenario of serious damages generated by its application it is mandatory for it to possess a transparent way with which some party involved can be established guilty.

A lot of efforts are being done by the scientific communities worldwide to create projects which can serve this purpose: for instance, I am personally working for the European project "Multi-disciplinary Use Cases for Convergent new Approaches to AI explainability" (*MUCCA*) [4] on xAI in the CHIST-ERA context. This project aims at bringing together researchers from all over the world with different backgrounds to implement and test an array of ML explainability techniques relevant to a heterogeneous cluster of scientific matters (High Energy Physics, Medicine, Neuroscience, etc.).

Summarising, xAI's primary goal is the formulation of ML predictive models that to the maximum extent possible can be described as understandable, comprehensible, transparent, interpretable, intelligible, responsible, accountable, accurate and interactive [1].

3. xAI Partitions

There exists a lot of explainability methods, for this reason a lot of different possible categorisation ways have been proposed to partition them. Anyway, being xAI still an emerging field, it should be remarked that these classifications could neither be exclusive nor exhaustive.

3.1. Method types partition

This first partitioning approach is done according to three considered criteria: *Interpretability Complexity*, *Interpretability Scoop* and *Model Dependency Level* [1]. Figure 1 summarise this classification approach.

Interpretability Complexity describes how much the interpretation of a specific kind of model is difficult for humans. A model can be specifically designed to be explainable in an simple way, in this case it is said to be *Intrinsic Interpretable* (for instance, Decision Trees fall into this class). On the contrary, models based on very complex architectures (for instance, NNs belong to this class) to give a predictive outcome are usually not rapidly comprehensible: thus they belong to the *Post-hoc* class, meaning that they cannot be directly explained but instead need some further processing to provide information about their choices.

Interpretability Scoop instead is about the degree of understanding given by a model's explanation. If the explanation is capable of helping us directly comprehend the behavior of the entire model it is said to be *Global*, while instead if it makes us only find out the reasons for a single prediction given at a time it is *Local*.

Model Dependency Level, finally, is about the extent of applicability of a specific explainability technique to different predictive models. An explanation method is *Model-specific* if it can only be applied to one kind of model class (or to a restricted specific set of classes). *Intrinsic Interpretable* models belong by definition to this class. Alternatively, if an explanation method is not bound to some ML model and can be used indifferently every time, it is defined *Model-agnostic*.

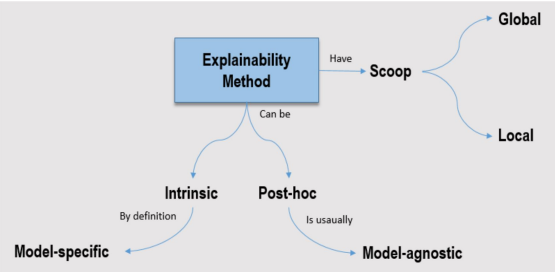

Figure 1: Explanability method partitions.

3.2. Explanation types partition

While the first approach considers more general attributes of the xAI methods, this second partitioning approach is conducted based on the nature of the generated explanations. This approach includes a huge amount of classes, here only the four main ones will be described: *Features Attribution*, *Learned Features*, *Explanation by Examples* and *Counterfactuals*.

Features Attribution methods try to give explanations by understanding which are the most relevant features in a certain given input sample for the prediction yielded by a ML architecture. In general, this methods work by computing an importance score for all the features of the sample to then emphasize the highest scoring features. This is the most popular and investigated category: a very large number of its approaches are being researched. In the following Section 4 (State-of-the-Art), I will give more details and go more in depth for some of the main methods of this class.

Learned Features methods try to give explanations by extracting what are the characteristics of the samples of a dataset that a ML model has learned during its training process. In particular for Neural Network architectures, it is usually investigated what are the input data point

that are able to maximise the response/activation of the network's inner neurons. One of the main paper in this area [5] helped to understand more on a high-level how Convolutional Neural Networks work internally claiming that the first layers of these architectures learn simple figurative concepts while last layers have the potential of merging them together so to form more complex shape that can retrieve the learnt figures. Moreover, they state that a model trained for an scene classification task would also be capable in recognising and detecting features on several levels of abstraction (for instance, edges, textures, objects and scenes) by extracting information directly from the intermediate layers and not only from the final one.

Explanation by Examples methods try to give explanations in a more "factual" manner. Furthermore, these approaches try to make the models' behaviour transparent by giving, together with the prediction for an input, a set of learned samples from the training dataset that the model believe to be similar to the input. Often, this procedure is performed by implementing and slightly modifying some classical methods as, for instance, K-means [6] and K-nearest neighbours [7].

Counterfactuals methods try to give explanations by finding out what is the minimal set of features for a input such that, if slightly modified, would result in a different prediction for that particular input. Clearly these approaches also focus on discovering only features which would make sense to modify (*actionable* edits) by studying particular properties between the input and the training dataset. Counterfactual candidate samples do not necessarily have to belong and be chosen from the training dataset, indeed some techniques have also been elaborated so to use generative ML models to build them ad hoc [8].

4. State-of-the-Art

Let us now describe in depth some of the most used Explainable Artificial Intelligence approaches, each one of them belonging to the *Features Attribution* partition: *Local Interpretable Model-agnostic Explanations* (LIME), *Kernel SHAP* and *Class Activation Maps* (CAM).

4.1. LIME and Kernel SHAP

We can state that ***LIME*** and ***SHAP*** are, for the time being, the two most important xAI techniques. Both are model-agnostic and data-agnostic, meaning that they can be applied and used with any kind of ML model fed with any kind of data (for instance, tables, text, images etc.).

LIME's procedure [9] is quite simple: the prediction's explanation for an input sample is generated by inspecting how the model's predictions vary slightly perturbing the input. Algorithmically speaking, LIME behaves as follows:

- It generates a new dataset consisting of perturbed samples of the chosen input and their corresponding predictions from a model f to be explained.
- It assigns a weight to each perturbated sample according to their features proximity to the input, where a high weight correspond to little perturbations being applied.
- It train a new, weighted, linear and interpretable model g on the generated dataset.
- Finally, explanations can be formulated for f's original predictions by interpreting g.

Formally speaking, in LIME an explanation model is found by minimising a certain objective function, formulated as:

$$arg\min_{g \in G} L(f, g, \pi_x) + \Omega(g) \tag{1}$$

where g is the explanation model, f is the original model, π_x are the local weights assigned to the perturbed samples, L is a loss function (generically a squared loss function), and Ω is a function representing the complexity of g.

Kernel SHAP [10] could be considered as an evolution of LIME: it is in fact based on LIME but also the concept of *Shapley values* is included in its definition. Shapley Values come from Cooperative Game Theory and are used to determine in a fair and perfectly unbiased manner how much effort some different players put into the achievement of a goal. In our xAI context, this concept is used to determine how much each feature contributed in the formulation of the final prediction. In Kernel SHAP the whole algorithm remains identical to LIME, the only variation carried out is in the way perturbations are weighted. In fact, the weights are here computed through an approximation of Shapley values where a high weight value is not only assigned to samples with little perturbations (as in LIME) but also to samples with very high perturbations. The reasoning behind this is quite simple: if the model is able to get the right predictions by only using a very small set of features, then those features must be very important for the completion of the task.

Mathematically, the main difference in Kernel SHAP is therefore how π_x (with reference to Equation 1) is computed, which is done with the following:

$$\pi_x(z') = \frac{M-1}{\binom{M}{|z'|}|z'|(M-|z'|)} \tag{2}$$

where M is the number of features and $|z'|$ is the number of non-zero features in the simplified input space $z' \in \{0,1\}$.

4.2. CAM

CAMs, introduced by [11], is a method used to locally explain the reasons behind some produced output of a CNN by highlighting the most important discriminative regions of an image (but also other kind of data) for its prediction. Figure 2 shows an example where the highlighted regions expose the relevant regions for a *"mouse"* prediction.

The key concept in the architectures implementing this method consists in the use of a *Global Averaging Pooling* (GAP) layer to connect the last convolutional layer and the output layer: doing so, each neuron in the GAP layer will contain the spatial average of the feature maps from the last convolutional layer, namely storing values expressing the neuron's involvement to the final prediction. The mentioned result, here defined S_c, produced in this way for a given class c by a last convolutional layer containing K feature maps $g_k(i, j)$ (where (i, j) is the spatial coordinate locating an entry in the feature map k) can be computed as:

$$S_c = \sum_{k=1}^{K} w_k \sum_{i,j} g_k(i, j) \tag{3}$$

where w_k is the weight of a neuron k for the output of the GAP layer.

Finally, these values are fed into a fully connected layer equipped with a softmax activation function, obtaining the final prediction

$$\hat{y} = \frac{\exp(S_c)}{\sum_c \exp(S_c)} \tag{4}$$

This method, although being very simple to be applied, has some fundamental drawbacks. First of all, it is very model-specific, meaning that it can only be applied on CNNs. In addition, to use the Activation Map approach, it is essential for the considered CNN to have a specific final predictive head, therefore limiting the available usable architectures even more. Finally, only relying on a CNN's final convolutional layer for an explanation may lead to the loss of other important information derivable from previous layers.

5. Conclusion

Concluding, by means of this paper I explained the reasons which led to the rapid growth of the xAI research field. I explored its definitions and a lot of methodologies usually used in literature to classify xAI methods. Some of the state-of-the-art (LIME, Kernel SHAP and CAM) approaches were discussed, both in a simple and formal fashion, to comprehensively demonstrate the capacity and potential of Explainable Artificial Intelligence.

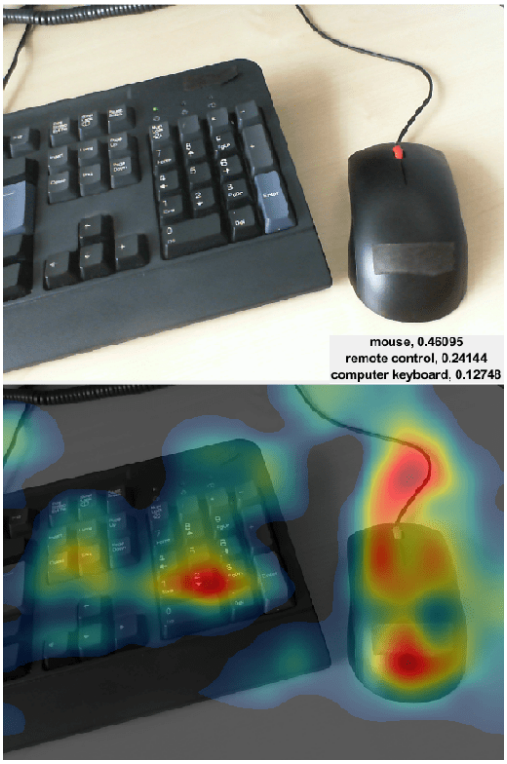

Figure 2: CAM example showing the regions of the input image contributing the most for the prediction.

References

[1] P. Adadi, M. Berrada, Peeking inside the black-box: A survey on explainable artificial intelligence (xai), IEEE Access 6 (2018). doi:10.1109/ACCESS.2018.2870052.

[2] A. Rahimi, Machine learning has become alchemy, 2018. URL: https://youtu.be/x7psGHgatGM?t=970.

[3] Y. Zhang, P. Tiňo, A. Leonardis, K. Tang, A survey on neural network interpretability, IEEE Transactions on Emerging Topics in Computational Intelligence 5 (2021) 726–742. doi:10.1109/TETCI.2021.3100641.

[4] CHIST-ERA, Mucca - multi-disciplinary use cases for convergent new approaches to ai explainability, 2021. URL: https://www.chistera.eu/projects/mucca.

[5] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, A. Torralba, Object detectors emerge in deep scene cnns, ICLR 2015 (2015). doi:10.48550/arXiv.1412.6856.

[6] S. P. Lloyd, Least squares quantization in pcm, Information Theory, IEEE Transactions on 28.2 (1982) 129–137. doi:10.1109/TIT.1982.1056489.

[7] N. S. Altman, An introduction to kernel and nearest-neighbor nonparametric regression, The American Statistician 46 (1992) 175–185. doi:10.1080/00031305.1992.10475879.

[8] S. Liu, B. Kailkhura, D. Loveland, Y. Han, Generative counterfactual introspection for explainable deep learning, GlobalSIP (2019). doi:10.1109/GlobalSIP45357.2019.8969491.

[9] M. T. Ribeiro, S. Singh, C. Guestrin, "why should I trust you?": Explaining the predictions of any classifier, CoRR abs/1602.04938 (2016). URL: http://arxiv.org/abs/1602.04938. arXiv:1602.04938.

[10] S. M. Lundberg, S. Lee, A unified approach to interpreting model predictions, CoRR abs/1705.07874 (2017). URL: http://arxiv.org/abs/1705.07874. arXiv:1705.07874.

[11] B. Zhou, A. Khosla, À. Lapedriza, A. Oliva, A. Torralba, Learning deep features for discriminative localization, CoRR abs/1512.04150 (2015). URL: http://arxiv.org/abs/1512.04150. arXiv:1512.04150.