Explainable AI

Vincenzo Colella 1748193

October 2022

Contents

1 Introduction

Machine learning and related topics have advanced significantly in the last ten years. Deep learning has given machine learning a new lease on life as a result of the growth in data and accessibility of affordable computer technology. Deep learning and AI's superior human ability in solving complicated issues has made them incredibly popular. The deep learning models' strength is also its weakness, since they contain millions of intricate internal calculations among millions of parameters, making it difficult for humans to understand them and effectively making them a "black box." The likelihood of its failure also rises in other sectors. The need for a tool that can explain these models to regular people, researchers, and domain experts has increased over the past ten years. The creation of such tools is the main emphasis of explainable AI research. We may need to construct a measurement metric in the future to convey the causes and quality of the explanation, even if this field is primarily concerned with the creation of the foundations and methodologies for the transparency and tractability of AI and deep learning models. This will enhance the AI models' interpretability and human expert augmentation in addition to improving the legal handling of the models. Robustness and comprehensibility are necessary for trusted AI, since in the future these neural networks will gain more and more responsibilities as well as have an increasing presence in our daily lives.

Two main types of methods have been suggested to improve the interpretability of AIs:

- Intrinsic (Model-specific)
- Post Hoc (Model-Agnostic)

In this paper, we'll study in depth a few of the methods from the two categories above.

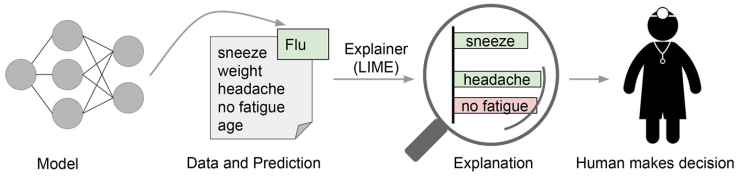

Figure 1: An example of explainability

2 Model-specific

2.1 Class Activation Map

Using CAMs enables the developer to see both the class that the network predicts and the area of the image that the network is most interested in. As

a result, the user receives an object localization of the expected class without having to manually name the bounding box for this object, which aids in both gaining greater insight into the network's learnings and simplifying troubleshooting. CAM can be added to an already working network as a weighted activation map, or CAM, that will be created for each image. It is useful to know what area a CNN is focusing on when classifying a picture. CAMs are trained in a loosely supervised manner rather than under supervision. This implies that the objects do not need to be explicitly labelled, and the localization is essentially "free" to learn. This is a big advantage since any developer can transform it's NN into an explainable one easily.

2.2 SmoothGrad

Very often sensitivity maps are used to explain which feature of a picture has been more important to get a particular result from a neural network. But results obtained are not always clear and easy to read. In this study is described SmoothGrad, a pretty straightforward technique that may be used with existing sensitivity map algorithms and, in fact, tends to reduce visual noise on gradient-based sensitivity maps. The fundamental concept is to start with an image of interest, sample related images by introducing noise, and then average the sensitivity maps for each sampled image. Then they contrasted two methods: SmoothGrad and one that adds noise during training (Bishop, 1995). By combining these two techniques, they discover an even better "de-noising" effect on sensitivity maps.

2.3 Real-time saliency map

As before, this is another strategy that is effective on Neural Networks that need readability from users and uses similar maps as in SmoothGrad to enhance this explicability. In particular, Piotr Dabkowski and Yarin Gal developed an efficient saliency detection technique that can be used with any classifier for differentiable image data. By masking important portions of the input image, they train a masking model to influence the classifier's scores. Their saliency detection model generalises well to unseen images and only needs one forward pass, making it appropriate for application in real-time systems. They claim to have tested this methodology on the CIFAR-10 and ImageNet datasets and demonstrated that the saliency maps it produced were clear, sharp, and error-free.

3 Model-Agnostic

3.1 Local Interpretable Model-Agnostic Explanations

LIME works by perturbing the input of a model and seeing how the predictions change. This turns out to be advantageous in terms of interpretability because, even if the model uses considerably more complex components as features, the

input can be changed by altering sections that are understandable to users (such as words or portions of an image).

We create an explanation by generating an interpretable model that estimate the underlying model using perturbations of the original instance (e.g., removing words or hiding parts of the image). The main idea underlying LIME is that it is far simpler to approximate a black-box model locally (in the vicinity of the prediction we wish to explain) by a simple model than it is to attempt to approximate it globally. This is accomplished by weighing the affected photos based on how closely they resemble the occurrence that needs explanation.

3.2 Layer-Wise Relevance Propagation

Layer-wise Relevance Propagation (LRP), a well-known explainability method created for complex computer vision models, generates intuitive, human-readable heat maps of the input images. The two phases of the suggested approach architecture are applied sequentially to each of the two datasets. A 1D-CNN is preprocessed and trained in the first phase. The 1D-CNN that is being suggested is trained on structured data. Then, the trained model is used in the second step, when XAI techniques are used to show the key features, once the network has been trained. This heat map is created through a strategy called deep Taylor decomposition, which means that the decision function of a decision will be broken into sub-functions, and explain each one separately.

3.3 KernelSHAP

This method, by Scott M. Lundberg and Su-In Lee, trains an interpretable model based on KernelSHAP and returns a representation of the interpretable model by attributing the output of the model with the specified target index to the inputs of the model In a way, it aggregates reasoning from LRP and LIME. Specifically, the variable I is added to the set "S" in this technique, and the effect on the function "es" value is examined to determine the relative relevance of the various variables or characteristics. The function "es" in this case can be thought of as a value function that clarifies the model "f" at a certain location "x*". The function is commonly described as the expected value for the conditional distribution where the conditioning holds true for all variables in a subset. The weighted average of every potential subset "S" is used to calculate the variable's contribution. In essence, the study of single ordering demonstrates how the value of the value function is altered by the addition of consecutive variables.

4 Conclusions

We studied many strategies needed nowadays to explain what's behind the choices made by neural networks. These methods are needed for various motivations, which include understanding how to enhance the accuracy of classifiers and even showing common people what reasoning a robot - or a NN - is going through before taking a decision. We are still at the beginning of the journey of getting AIs explainable, but these strategies must be developed further to get everyone able to accept machines to take big decisions or actions in our lives. There is still evidence of a lack of a global method that comprehends speed, efficiency, accuracy and reliability.

Authors

Marco Tulio Ribeiro, Sameer Singh and Carlos Guestrin "Local Interpretable Model-Agnostic Explanations (LIME): An Introduction", 2016.

Patrick Brus "Class Activation Mapping", 2020.

J. von Oswald, C. Henning, Benjamin F. Grewe, Joao Sacramento "Continual learning with hyper networks", 2019.

C. M. Bishop "Training with Noise is Equivalent to Tikhonov Regularization," in Neural Computation, vol. 7, no. 1, pp. 108-116, Jan. 1995.

Daniel Smilkov, Nikhil Thorat, Been Kim, Fernanda Viégas, Martin Wattenberg "SmoothGrad: removing noise by adding noise".

Scott M. Lundberg, Su-In Lee "A Unified Approach to Interpreting Model", 2017.

Ullah, H., Rios, A., Gala, Mckeever, S. "Explaining Deep Learning Models for Structured Data using Layer-Wise Relevance Propagation", 2020.