Interpretable Deep Learning Models

Vivek K [a] and Rengarajan A [a]

[a] *JAIN (Deemed-to-be University), Bengaluru, Karnataka, India*

Abstract

Deep Learning based models adopts a technique(s) to train computers to learn by data. Deep Learning models uses neural network architecture. However, their intrinsic design takes inputs and produces outputs without knowing the internals of the framework. In many scenarios, user wants to know the reasons behind the output. In this paper the need of interpretability component for deep learning models, formal definition Interpretable Deep learning (IDL) and components of IDL's are discussed. Also reviewed algorithms devised by researchers to build Interpretable Deep Learning Models.

Keywords

Interpretable learning, TrustyAI

1. Introduction

Deep Learning (DL) based models are inspired from human brain and the neurons mimic neurons of human brain. DL is used by Artificial Neural Network (ANN) and consists of nodes which are interconnected and are inspired by human brain.

Artificial Neural Networks can be described as layers of software units called neurons (also called node), connected with different neurons in a layered manner. These networks transform data from one neuron to another neuron until they can classify it as an output. Neural network is again a technique to build a computer program that learns from data.

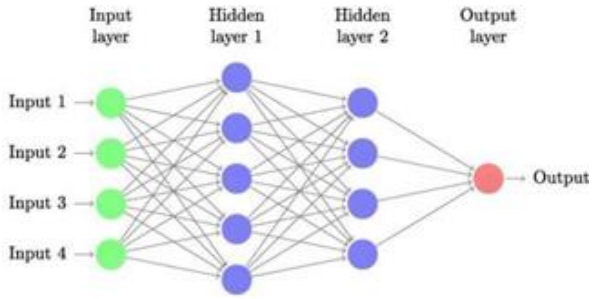

Figure 1: Artificial Neural Network

A typical ANN has three major components.

Input Layer Nodes: This layer receives the information from outer world to the network. The information is then passed onto the hidden node where computations can begin.

Hidden Node: There is no connection to the real world at this stage. This is the point where the machine uses the information received from the input node, it carries out computation and processing

WINS-2022: Workshop on Intelligent Systems, April 22 – 24, 2022, Chennai, India.
EMAIL: k.vivek@jainuniversity.ac.in (Vivek K)
ORCID: 0000-0002-1750-8236 (Vivek K)

on it. There can be more than one hidden layer.

Output Node: This is the final stage where the computations conclude, and data is made available to the output layer from where it gets transferred back into the real-world environment.

What the network does is it maps the inputs to the desired outputs. The information for this mapping is represented in the neuronal connection weights, which determine what computations will be performed on the input signal. Training the neural network entails adjusting these weights, using the back-propagation algorithm, to gradually make a structure that will process the inputs and get them to approximate the desired outputs. With enough training examples neural networks can, in principle, approximate any function, that is, any possible input-output mapping. In practice they never reach perfect generalization, but they often perform well enough for a large number of narrow applications, hence their rising popularity. When the number of hidden layers increases, we get "deep" neural networks, which have impressive capability in learning and representing input-output mappings. Deep neural networks are mostly used for classification tasks, assigning an input into a particular class from a set of possible classes. Many applications, from computer vision to machine translation, can be formulated as classification problems. In a sense, neural networks are used like a hammer in quest for nails: machine learning engineers are on a constant lookout for tasks that can be expressed as a class- assignment problem. In principle deep neural networks can approximate every conceivable input- output mapping, and in principle a huge number of cognitive tasks can potentially be formulated as a classification problem.

2. Need of Interpretability in deep learning

Building a deep learning model for high-risk scenarios deserve high interpretable DL models. Scenarios like medical image classification which miss-classifies or wrongly detecting traffic signals in an automated car not being able to understand the predictions without understanding internals of deep learning framework is not acceptable.

Black box signifies a system or device we do not know the internal working but can only see what inputs go in, and what outputs come out. In deep learning we use feature extraction and vectorization to represent the objects we want the model to process with numbers. The model only sees numbers and spots statistical regularities among these numbers [1]. It cannot register any qualitative relationships between the variables these numbers represent, like causality, hierarchy, and other abstractions [2] [3]. It only detects quantitative relationships among the numbers themselves, and therefore cannot explain its decisions in any human-meaningful way. It's a black-box. It has been shown that a machine learning model's interpretability is inversely proportional to its flexibility [4], and neural networks, with their mimicking brain plasticity, are arguably the most flexible models of all [5]. Debugging such an algorithm poses serious problems. Many applications use cascades of deep neural networks, one's output feeding the input of another to achieve complex tasks. The human brain is not merely a large neural network but it is a network of networks, and the quest for artificial general intelligence may take the direction of researching hierarchies of deep neural networks [4]. These systems might well be impossible to debug. As long as they remain black boxes, we can't trust neural networks to make important decisions in high-stake situations [3]. Facebook recommendations and automatic captioning of photos on blogs might be fine, but terrorism detection and forensic procedures shouldn't be entrusted on systems that cannot explain how they reach their conclusions. In Lipton [6] we learn that concerns over trust and other issues of interpretability may be "quasi-scientific." We learn that a lot of disagreement has been going about what makes a model interpretable, with candidate definitions often contradicting each other, therefore, we learn, concerns about interpretability are meaningless and "quasi-scientific". While it is true that interpretability is not always well-defined and there is high-variance of competing definitions, all of these definitions share one thing in common. Neural networks do not fulfill any one of them. No matter what standards for interpretability you set, neural networks do not meet them.

3. Interpretable Deep Learning

Interpretable Deep learning models can help us when we cannot formalize the ideas. Interpretable Deep learning models will help the users to understand the behaviour and predictions of deep learning systems. According to Marcus [2], neural networks "do not include any explicit representation of a relationship between variables. Instead, the mapping between input and output is represented through the set of connection weights. They replace operations that work over variables with local learning, changing connections between individual nodes without using 'global information."

There exist multiple reasons to interpret a DL model. Internal working of DL models is explained using additional set of algorithms called as interpretation algorithms, which are usually designed with different principles.

- DL models mainly relies on features or variables. These variables or features are derived from input data. Paying attention to important parts of input data will be the outcomes of this types of algorithms. This is achieved with perturbations, gradients, or proxy models called as explainable models.
- Deep understanding of inside deep learning models by investigation to understand the logic behind the decisions making capabilities of models.
- Calculating the weightage of each input variables of training data will help to interpret the training process.

4. Features of Interpretable Model

A typical Interpretable model will have following features.

- **Fairness:** All groups in data set will have equal representation, if predictions are unbiased. **Robustness:** Model is expected not to make major deviations in output for small changes in input. **Privacy:** By understanding the internal working of a model in terms of data that is used for trainingphase, can stop model from accessing sensitive information.
- **Causality of features & Debugging models:** An interpretable model helps to test the relationship between features with outcome i.e., the causality of the features, test its reliability and ultimately debug the model appropriately.
- **Trust:** if people understand how our model reaches its decisions, it's easier for them to trust it.

5. Interpretable Neural Networks

Quanshi Zhang et al.,2018 [8] introduced a taxonomy - "interpretable CNNs"-ICNN. This method modifies a traditional convolutional neural network (CNNs) by adding interpretable components to CNNs. In this method, high conv-layers of CNNs will have knowledge representations with clarifications. As per this approach, each filter in a high conv-layer represents a specific object part. This type of interpretable CNNs use the same training data as ordinary CNNs which does not require any additional annotations of object parts or textures for supervision. The central idea of interpretable CNN is to applying a high conv-layer with an object part during learning process and this will happen automatically. This approach can be applied to different types of CNNs. ICNN creates explicit knowledge representation and will help user to understand logic inside a CNN. CNN's can memorize these patterns for prediction.

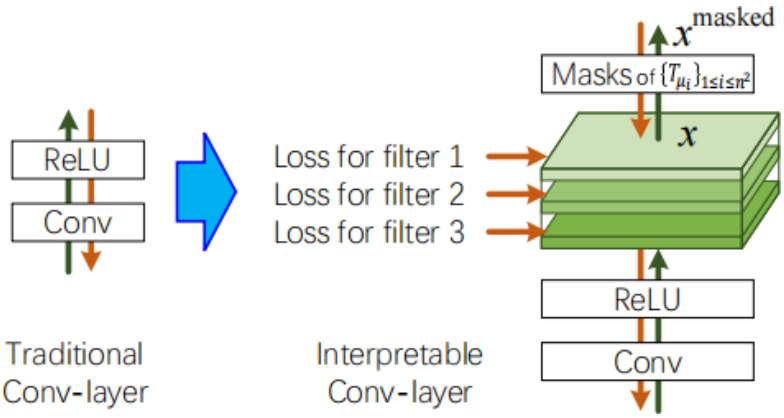

Figure 2: Architecture of convolutional neural network with Interpretable component

Interpretable models flexibility can be improved by adding new filters which can be used to describe discriminate textures of a category. Also, these new filters for object parts can also be shared by multiple categories.

Yinpeng Dong et al.,2017 [9] proposed a technique to improve the interpretability of Deep Neural Networks for image data which uses semantic information embedded in human descriptions. During video captioning, initially extract a set of semantically meaningful topics from the human descriptions that cover a wide range of visual concepts, and later integrate them into the model with an interpretive loss. With this approach, a prediction difference maximization algorithm can be used to interpret the learned features of each neuron. This approach can be extended for video captioning using the interpretable features. This technique can also be transferred to video action recognition. This will help to clearly understand the learned features and users can easily revise false predictions by keeping human in the overall procedure.

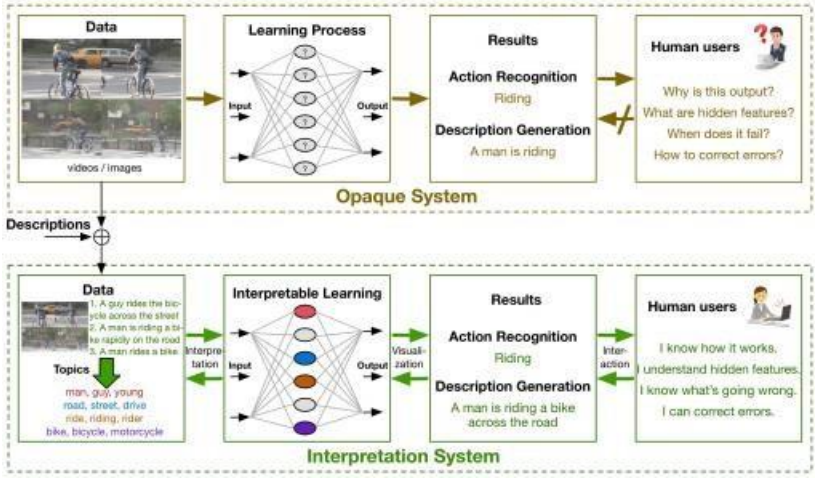

Figure 3: Deep Learning Opaque System Vs. Deep Learning with Interpretation System

A Deep Learning System, i.e., an opaque system often learns abstract and impenetrable features. Without ICNN, end users have to accept the decisions from the system passively, without understanding black-box logic without understanding the rationale of the decisions. Also, users cannot interact with DL system. Interpretability of Deep Neural Networks (DNNs) is improved by embedding topics in human descriptions as semantic information during the learning process. Each neuron can learn a topic. Topic could be riding related to bicycle, cart and horse. These interpretable features, can be used by human users to visualize and interaction with system smoother. Also, it allows a human element into learning procedure.

Xinyang Zhang and et al.,2019 [12] defined term interpretable deep learning system (IDLS) which

consists of classical DNN model component i.e., Classifier and an interpretation model called as interpreter. The enhanced interpretability of IDLS will increase the confidence of users who will be using the models for decision making [13]. However, DNN's are susceptible to Adversarial Deformation. Xinyang Zhang and et al.,2019 [12] proposed an adversarial training framework - Adversarial Interpretation Distillation (AID), which integrates Adversarial Deformation in training interpreters. AID improve the robustness of interpreters against Adversarial Deformation by reducing the prediction-interpretation gap.

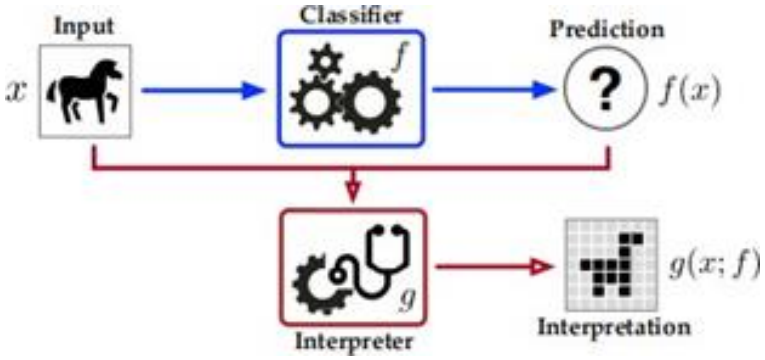

Figure 4: Core components of an interpretable deep learning system (IDLS)

Notation	Definition
f, g	target classifier, interpreter
x_o, x_*	benign, adversarial input
c_t, m_t	adversary's target class, interpretation
$x[i]$	i-th dimension of x
ε	perturbation magnitude bound
$\|\cdot\|$	vector norm
$\ell_{\text{int}}, \ell_{\text{prd}}, \ell_{\text{adv}}$	interpretation, prediction, overall loss
α	learning rate

Table 1. Symbols and notations.

6. DL platforms with Interpretation Libraries

There are several libraries which support interpretability component for Deep Learning frameworks. Following list gives the popular combinations.

- TF-Explainer library [18] based on Tensorflow framework [14]
- Captum library [19] based on PyTorch framework [15]
- InterpretDL [20] based on PaddlePaddle [16] and Shap [21] based on Anaconda [17]

7. Conclusion

We discussed architecture of Neural networks and need of interpretability component to neural networks. Further reviewed algorithms devised by researchers to build Interpretable Deep Learning Models. Also discussed land mark researches which handles the Adversarial Deformation.

8. References

[1] Chollet, F. (2017). Deep Learning with Python. Manning Publications.
[2] Marcus, G.F. (1998). Rethinking Eliminative Connectionism, COGNITIVE PSYCHOLOGY 37, 243–282
[3] Ribeiro, M. T., Singh, S., & Guestrin, C. (2016). "Why Should I Trust you?" Explaining the Predictions of Any Classifier. arXiv, cs.LG

[4] Lipton, Z. C. (2016). The Mythos of Model Interpretability. arXiv, cs.LG.
[5] M. Aubry and B. C. Russell. Understanding deep features with computer-generatedimagery. In ICCV, 2015
[6] M. Simon, E. Rodner, and J. Denzler. Part detector discovery in deep convolutional neural networks. In ACCV, 2014.
[7] K. Simonyan, A. Vedaldi, and A. Zisserman. Deep inside convolutional networks: visualizing image classification models and saliency maps. In arXiv:1312.6034, 2013.
[8] Quanshi Zhang, Ying Nian Wu, and Song-Chun Zhu. Interpretable Convolutional Neural Networks. In arXiv:1710.00935v4 [cs.CV] 14 Feb 2018
[9] Yinpeng Dong, Hang Su, Jun Zhu, Bo Zhang. "Improving Interpretability of Deep Neural Networks with Semantic Information", 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017
[10] N. Ballas, L. Yao, C. Pal, and A. Courville. Delving deeper into convolutional networks for learning video representations. In ICLR, 2016
[11] A. Karpathy, J. Johnson, and L. Fei-Fei. Visualizing and Understanding Recurrent Networks. In Proceedings of International Conference on Learning Representations (ICLR), 2016.
[12] Xinyang Zhang,Ningfei Wang,Hua Shen,Shouling Ji,Xiapu Luo,Ting Wang. Interpretable DeepLearning under Fire, arXiv:1812.00891,2019.
[13] Guanhong Tao, Shiqing Ma, Yingqi Liu, and Xiangyu Zhang. Attacks Meet Interpretability: Attribute-Steered Detection of Adversarial Samples. In Proceedings of Advances in Neural Information Processing Systems (NIPS), 2018
[14] Tensorflow Team. URL: https://www.tensorflow.org/
[15] PyTorch Team, https://pytorch.org/
[16] Deep Learning & Machine Learning Framework,URL:https://github.com/PaddlePaddle/Paddle
[17] Python & R Distribution. URL: https://www.anaconda.com/
[18] TF-Explainer library. URL: https://tf-explain.readthedocs.io/en/latest/
[19] Captum Library. URL: https://captum.ai/
[20] Interpret DL Library. URL: https://github.com/PaddlePaddle/InterpretDL
[21] Shap Library. URL: https://shap.readthedocs.io/en/latest/index.html