JOHANNES KEPLER
UNIVERSITY LINZ

Submitted by
Viktor Maximilian Loreth
k12006268

Submitted at
Institute of Computational Perception

Supervisor
Katharina Hoedt, PHD

August 27, 2023

EVALUATION OF IMAGE RECOGNITION NEURAL NETWORK INTERPRETATION METHODS: AN IN-DEPTH LOOK

Bachelor Thesis

to obtain the academic degree of

Bachelor of Science

in the Bachelor's Program

Artificial Intelligence

JOHANNES KEPLER UNIVERSITY LINZ
Altenbergerstraße 69
4040 Linz, Österreich
www.jku.at
DVR 0093696

JOHANNES KEPLER
UNIVERSITY LINZ

Sworn Declaration

I hereby declare under oath that the submitted Bachelor Thesis has been written solely by me without any third-party assistance, information other than provided sources or aids have not been used and those used have been fully documented. Sources for literal, paraphrased and cited quotes have been accurately credited.

The submitted document here present is identical to the electronically submitted text document.

Linz, August 27, 2023

JOHANNES KEPLER
UNIVERSITY LINZ

Abstract

This paper focuses on the accuracy of interpretability methods for machine learning models. The main research problem is the lack of ground truth for evaluation method in interpretability methods. While there are inherent interpretative models, black-box networks perform better and have developed rapidly. Existing interpretability methods for neural networks such as ROAR, KAR, BAM and Real Time Image Saliency for Black Box Classifiers provide a numerical evaluation method but they do still suffer from ambiguity. This paper summarizes the different evaluation methods, compares them and also calculate ROAR on 2 different saliency methods.

JOHANNES KEPLER
UNIVERSITY LINZ

Contents

List of Figures

1 Introduction

Artificial Intelligence (AI) has undergone rapid development in the last years. In today's modern era of mobile phones and computers, algorithm's are used on a daily basis to have quick access to information and improve the efficiency of the daily life.

While various Algorithms (e.g.: Decision Trees, Linear Regression, Support Vector Machines, etc.), which are comprehensible by design, have been developed, the spotlight has turned to Deep Neural Networks(DNN). This shift is attributed to the increase in computational power and the exponential increase in accessible data. Despite their remarkable accuracy, DNN remain opaque black-boxes, which we struggle to understand. Nevertheless, the immense improvement in performance and their ability to handle massive datasets have led to widespread adoption in contemporary devices. It is predicted, that algorithms based on Neural Networks will be becoming increasingly popular in the next years.

However, one of the primary difficulties with Neural Networks is the lack of reliable interpretation techniques. Numerous interpretation methods exist, yet a universally reliable method remains missing. Particularly in the domain of image analysis, encompassing critical applications like automated driving and facial recognition, no solution is present. The decision-making rationale of neural networks remain unclear, attributed to factors like background elements, peripheral objects or lighting conditions. Efforts to address this issue have given rise to gradient methods, aiming to assign significance values to pixels and represent their importance on neural network decisions. Another alternative option to mitigate the black-box nature of algorithms involves employing model-agnostic methods. These methods offer an computational linkage between inputs and outputs, irrespective which model is used. Although highly effective for smaller datasets, they begin to struggle as the data size and their complexity increases. Because of this, they do not offer a reliable way to quickly make Neural Networks interpretable.

In light of these prevalent problems, the object of this thesis is to recapitulate interpretation algorithms Neural networks in computer vision. Emphasis is placed on the evaluation of post-hoc interpretability techniques, forecasting potential future developments and focusing on the strengths and weaknesses of distinct techniques. Concluding the theoretical segment, a practical demonstration showcasing the application of ROAR is shown.

1.1 Structure of the thesis

1. The first part presents a overview of contemporary machine learning algorithms, categorizing them into two main groups: algorithms with inherent interpretability and those without. The goal is to show why interpretation methods are required.

2. Subsequent sections delve into interpretability, emphasizing global and local model-agnostic techniques. These methods offer insights into overall model behavior, regardless of algorithm specifics.

3. Additionally, in the domain of interpretability techniques for Neural Networks, the paper explores ad-hoc methods for Neural Networks. Features visualization and Gradient-focused methods are explained.

4. After introducing existing interpretation methods, the paper's focus transitions to evaluating post-hoc interpretation methods. Various approaches to assess the effectiveness and dependability of these methods in offering meaningful insights into intricate models are introduced and discussed. Additionally, the advantages and disadvantages of these approaches are carefully examined to provide a comprehensive understanding of their applicability.

5. To exemplify the discussed concepts, the practical application of the ROAR methodology using the food101 data set [2] and MNIST dataset [3] is presented. This real-world instance illustrates effective employment of interpretability techniques in image recognition.

2 Machine Learning and their Interpretability

This paper focuses on two types of machine learning methods: Unsupervised and Supervised. However, we'll only look at supervised methods here because interpreting unsupervised methods works differently at a basic level.

In supervised machine learning, there are several base methods of classification. They are shortly introduced and analyzed for their interpretability. This should give readers a simply structure on why this research is needed.

2.1 Supervised Methods

This section analyzes the base functionality of each singular model. Furthermore, an analysis of the interpretability from a human perspective is made. Methods are developing rapidly, therefore only the most common methods are included.

2.1.1 Linear Models

When it comes to predicting outcomes, a simple method is to use a linear regression model. This model predicts by adding up different features, each multiplied by a weight. The linear nature of this model makes it easy to understand. Mathematically, the predictive output, denoted as $\hat{y}$, is captured in the equation:

$$\hat{y} = \alpha_0 + \alpha_1 x_1 + \alpha_2 x_2 + ... + \alpha_n x_n + \epsilon$$

The alphas α_i indicate the significance of each feature. The initial coefficient α_0 is known as the intercept, signifying the baseline. The noise ϵ encapsulates the inevitable errors stemming from inherent non-linearity in real-world dynamics or measurement inaccuracies.
To train model, the MSE-Loss or the absolute loss can be applied. When using regularization methods, the absolute loss is taken to be more resilient to outliers.

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

$$\text{ABS} = \frac{1}{n}\sum_{i=1}^{n} y_i - \hat{y}_i$$

The interpretability of the model is very simple. The factors are given through the coefficient matrix. Each feature has a distinctive importance to the model and it can be seen easily how important each factor is, when the data is normalized.

$$\alpha = \begin{bmatrix} \alpha_0 \\ \alpha_1 \\ \dots \\ \alpha_n \end{bmatrix}$$

Although linear models possess comprehensibility and provide a straightforward method for prediction and are inherent understandable, their application is limited to linear relationships and small datasets. In the domain of image recognition it is not applicable.

2.1.2 Distance-Based methods

K-Nearest Neighbors is used for classification and uses the nearest neighbars as classification. KNN is not interpretable by default, as there are no parameters to learn and analyze. One can argue, that KNN is interpretable by the fact, that it just describes if there are similar samples. However KNN also struggles with big datasets.

Support Vector Machines (SVM) aim to find a hyperplane that maximizes the margin between different classes of data points.

$$\text{Minimize } \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n} \xi_i$$

$$\text{Subject to } y_i(w \cdot x_i + b) \geq 1 - \xi_i, \quad \xi_i \geq 0 \text{ for } i = 1, \dots, n$$

With using the kernel trick, SVM can be used for non-linear data. In higher dimensionality, SVM becomes non-interpretable, as displaying the weight matrix is not understandable. SVM also struggles with big datasets. Therefore also SVM is not applicable for image datasets.

2.1.3 Decision Tree-Based Methods

Decision Trees based on minimizing the gini-index are inherently interpretable. As the depth increases, the models become less understandable. One disadvantage is the lack of smoothness. If one boundary is reached, the model classification changes.

Random forests are an ensemble of multiple decision trees. Their advantage is a smoother predictive power. But it suffers from being less interpretable. Using special techniques like SHAP values [6] or partial dependence trees can make them more understandable.

Gradient boosting like XGBoost, LightGBM and CatBoost are similar to random forest and decision trees, but with differential learned weights for each decision. They suffer from the same interpretability issues as random forests.

While decision tree-based methods can be applied for image classification, their accuracy is poor.

2.1.4 Computational intensive optimization Problems, LDA, QDA

LDA (Linear Discriminant Analysis): LDA aims to maximize the distance between classes while minimizing the variance within each class. It assumes that the data within each class follows a multivariate normal distribution with equal covariance matrices across all classes. LDA is useful when there's a linear separation between classes in the feature space. It can be effective when the class distributions are well-separated.

QDA (Quadratic Discriminant Analysis): QDA is an extension of LDA that relaxes the assumption of equal covariance matrices across classes. It allows each class to have its own covariance matrix, which can capture more complex relationships between features. QDA is suitable when the class distributions have different covariance structures or when the assumption of equal covariance is not met.

Both LDA and QDA have been applied to image recognition tasks with varying degrees of success. Their performance can be influenced by factors such as the amount of training data available, the choice of features, and the complexity of the underlying data distribution.

2.1.5 Neural Networks

The rise of neural networks and their strong predictive power makes them a common choice for classification task. But as they increase in size, understandability by taking a look at the weights becomes impossible. With the rise of CNN in 2012 [**krizhevsky2012**] Neural Networks have become state of art for image prediction. Special interpretability methods considering the learned weights are looked in detail in 3.3.

3 Interpretation of Image Recognition Neural Networks

In this chapter, evaluation method are described to analyze the behavior of not inherently interpretable models. We only introduce evaluation methods, which are used to analyze neural networks.

3.1 Global Model-Agnostic Methods

Global model-agnostic methods describe expected outcomes based on the distribution of the data. They can show a correlation between singular or multiple features and an outcome.

A quick evaluation on applicability for Neural Networks is done:

1. **Partial Dependency Plots:** Partial Dependency Plots (PDP) are easy to interpret but using it with several features becomes increasingly difficult. Therefore, using a PDP is not possible in Neural Networks.
2. **Accumulated Local Effects:** Accumulated Local Effects (ALE) are an an advancement of PDP. While it solves some problems PDP suffers from, it can't be applied to a Neural Network because of the complexity of the datasets.
3. **Feature Interaction:** Feature Interaction also analyzes how features correlate. Doesn't work due to the same problem of the complexity of the dataset.
4. **Functional Decomposition:** Functional Decomposition is commonly used in Neural Networks. See Chapter 3.3.1.
5. **Permutation Feature Importance:** Permutation Feature Importance is regularly used in visual machine learning tasks.
6. **Prototype and Criticism:** Prototype and Criticism is used as adversial attacks in Neural Networks.

3.2 Local Model-Agnostic Methods

To explain individual predictions, Local model-agnostic methods are used. A single outcome is correlated to some features and explain the model.

1. **LIME: Local Interpretablle Model-agnostic Explanations:** Lime generates locally faithful explanations by training interpretable models on perterurbed instances of the original data.
2. **Local surrogate models:** Local surrogate models can be programmed to select a singular instance to explain.
3. **Scoped Rules (Anchors):** Find so called anchors to explain the predictions.
4. **Individual conditional expectation curves:** Practically not useful in image recognition, as the computation is too high.
5. **Counterfactual explanations:** Try to find a change while still resembling the original image.
6. **SHAPly:** Calculate SHAP values for an image prediction to determine how each pixel contributes to the prediction's deviation from the average prediction across all images. Positive SHAP values indicate pixels that push the prediction up, while negative values indicate pixels that pull it down.

3.3 Neural Network Interpretation

In the domain of Natural Language Processing (NLP) and Computer Vision, Deep Learning has proven very successful. By passing the input data through a sequence of layers, characterized by matrix-multiplications with kernel weights and nonlinear transformations functions, a prediction is computed. Depending on the specific task, additional elements like Long Short-Time Memory(LSTM) layers and Convolutional layers (CNN) are utilized. Given the immense amount of mathematical operations underlying a single prediction, humans are not fit to apprehend the mapping. To interpret predictions, we would have to decipher the intricate learned knowledge of numerous different kernels and weights. Recognizing that it's impossible for humans to grasp millions of weights, the demand for evaluation methods is high. To assess the behavior and predictions of Deep Neural networks, specific interpretation methods were developed. These methods calculate the likelihood of an input entry being responsible for the result.

While model-agnostic methods offer an approach to understand Neural Networks, the sheer size of the data used to train and test Neural Networks make this task extremely hard. For instance, an image with the dimensions of 3x224x224, as commonly encountered in Food-101,the data entries

exceed 150.000. In NLP tasks, where vocabularies often encompass around 20.000 words, the computational complexity renders most model-agnostic techniques as too expensive.
In the pursuit of comprehending the complex dynamics of Deep Neural Networks it makes sense to utilize the underlying weights in the model. The information saved in the hidden layers as learned weights can be used to evaluate the network. Moreover, the gradients can be taken into consideration as well. In the following subsections several concepts for understanding Deep Neural Networks are introduced.

3.3.1 Feature visualization and Network Dissection [7]

Modern Neural Networks like ResNet50 or Bard consist of several million layers. Network dissection attempts to overcome this challenge by breaking down separate layers and connecting them with ideas.

The higher-level features in these networks relate to clear concepts, shown in Figure 3.1. As the input image moves through layers, it changes at each layer. In each convolutional layer, the network gains new and more complex features. The smooth joining of fully connected layers then changes image-based data into predictions.

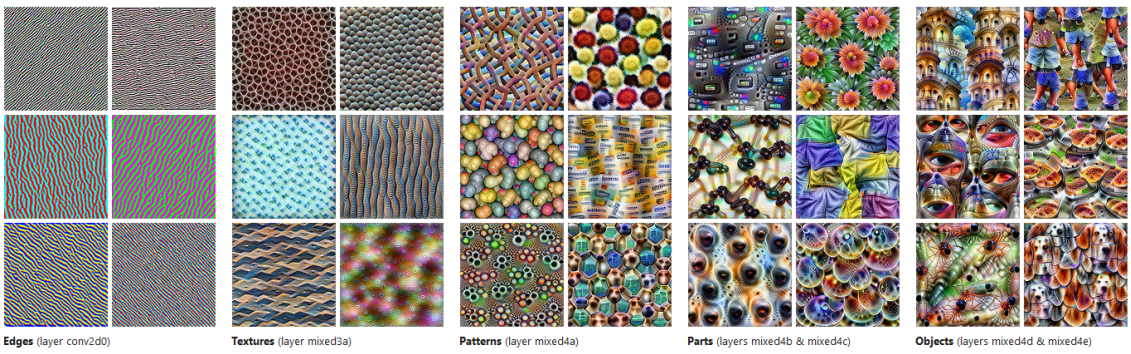

Figure 3.1: Feature Visualization [7]

The image explains this process. The first convolutional layers find simple features like edges and basic textures. Later, they recognize more detailed patterns. The deepest layers learn about parts and objects. This object information passes to the other hidden layers, which then finally make a prediction.

In the pursuit of understanding feature visualization, the focus lies on the activation of a single unit within the neural network. This involves maximizing the activation of a specific neuron, mathematically speaking (Visible in 3.2). There are two methods for achieving this. First, we can make use of the training image that triggers the highest activation. Yet, this approach faces a significant problem. When an image contains multiple objects, it's hard to pinpoint which object

causes the activation. Because of this an alternative route is adopted: generating new images from random noise. This is accomplished through methods like Generative Adversarial Networks (GANs) or other diffusion-based techniques.

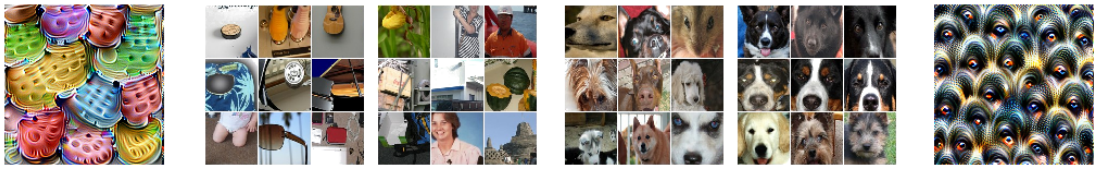

Figure 3.2: Activation Maximization [7]

Advantages of Feature Visualization:

1. **Initial Model Insights:** Feature visualization offer an initial view into a model's behavior, improving the understanding of its inner layers.
2. **Enhanced Domain Understanding:** It has the potential to enrich domain understanding by aligning learned features with domain-specific knowledge. An example can be seen in the medical industry
3. **Debugging and Improvement:** Feature visualization assists in debugging and refining models, contributing to their overall performance enhancement.

Disadvantages of Feature Visualization:

1. **Unclear Decision-Making:** While activations are evident, understanding the meaning behind them and how they contribute to decision-making remains challenging.
2. **Subjective Interpretation:** The interpretation of visualized features can be subjective, potentially leading to differing conclusions among observers.
3. **Limited Applicability to Visual Data:** Feature visualization's applicability is limited to visual data types.

3.3.2 Saliency Maps

Saliency maps are visualizations that highlight the regions of an input image that have the most significant impact on a model's output. By revealing the areas that strongly influence a prediction, saliency maps bridge the gap between the model's "black-box" nature and human understanding.

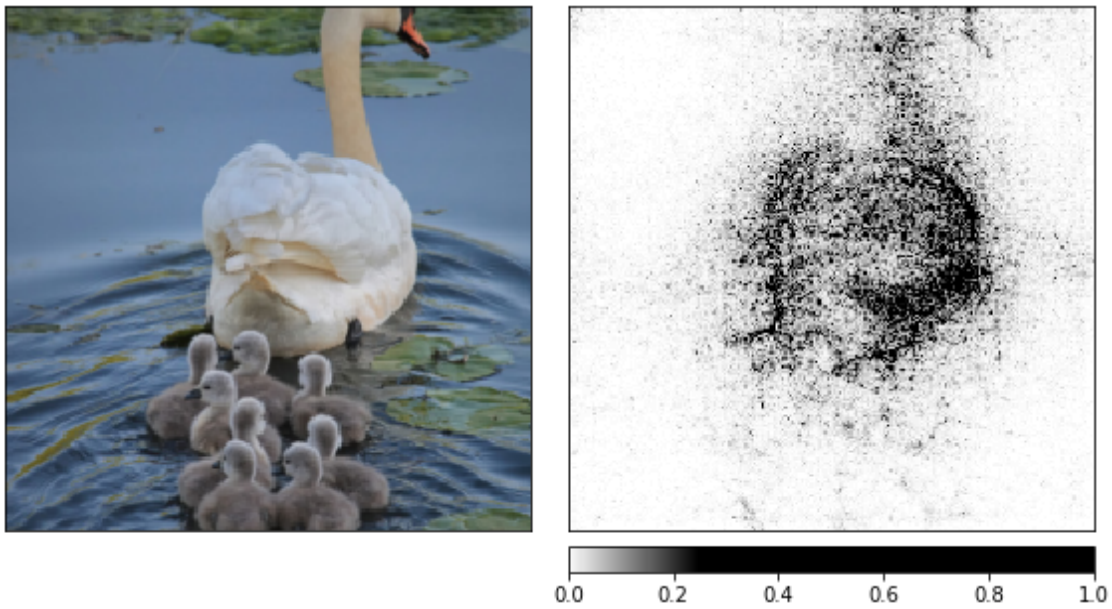

Figure 3.3: Saliency Map - Source: https://captum.ai/tutorials/Resnet_TorchVision_Interpret

Saliency maps are commonly calculated using SHAP [6] or gradient methods. Saliency maps provide a direct and intuitive way to understand which parts of an input data are influencing a model's decision. The main difficulty is the generation of reliable saliency maps.

3.3.3 Gradient-focused methods

Option 1: Vanilla Gradient & DeconvNet

Vanilla Gradient [8] focuses on computing the gradients of the network. A forward pass of an image is generated. The gradients of the class score is computed. Then the gradients are visualized. When ReLU is used and the gradients are negative, then information is lost. Deconv Net [9] takes care of this problem.

Option 2: Grad-CAM & Guided Grad-CAM

Grad-CAM only takes care of non-CNN maps. Guided Grad-CAM computes Grad-CaAM with another method to have a better localization

4 Evaluation of post-hoc interpretability methods

Gradient-methods which generate saliency maps are hard to measure. While humans can evaluate the maps by giving a general statement, this is not a scientific statem<ent and can also not be applied to thousands of images. Despite many significant recent contributions to saliency maps, the valuable effort of explaning machine learning models face this methodological challenge: the difficulty of assessing the scope and quality of model explanations [1].

4.1 A benchmark for interpretability methods in deep neural networks

RoaR is a algorithm which estimates the effectiveness of saliency maps. This is done, by removing supposedly informative features from the input and observing the reaction of the neural network.

As shown in the paper, KAR seems to be a bad method to evaluate the efficiency.

4.2 New Definitions and Evaluations for Saliency Methods: Staying intrinsic, complete and sound [4]

Explaining Completeness:

Logical reasoning: All correct statements are proveable. Example: The dog is responsible for the nets output as dog.

Soundness: Incorrect statements cannot be proved. Example: The sun is responsible for the nets output as dog.

4.3 Sanity checks for Saliency maps

4.4 Comparison of the Evaluation Methods & critique

While the described methods both offer a numerical evaluation method, they do still lack a clear evaluation structure. They show, that some evaluation methods are indeed correct, but they still suffer ambuigity from different datasets.

5 Project work

5.1 Project Goal

In RoaR[5] the paper does not list the standard deviation of the trained nets. We expect to validate the results by achieving similar results.

As training 25 image nets requires high computational power we do not have right now, we limited our research to evaluating food-101 [2] using only 2 interpretation methods and comparing it to the baseline.

Additionally, we also add a mini evaluation using the MNIST dataset.

5.1.1 Project Setup

5.1.2 Results and Plots

5.1.3 Discussion of results

Bibliography

[1] Julius Adebayo et al. *Sanity Checks for Saliency Maps*. 2020. arXiv: 1810.03292 [cs.CV].

[2] Lukas Bossard, Matthieu Guillaumin, and Luc Van Gool. "Food-101 – Mining Discriminative Components with Random Forests". In: *European Conference on Computer Vision*. 2014.

[3] Li Deng. "The mnist database of handwritten digit images for machine learning research". In: *IEEE Signal Processing Magazine* 29.6 (2012), pp. 141–142.

[4] Arushi Gupta et al. *New Definitions and Evaluations for Saliency Methods: Staying Intrinsic, Complete and Sound*. 2022. arXiv: 2211.02912 [stat.ML].

[5] Sara Hooker et al. *A Benchmark for Interpretability Methods in Deep Neural Networks*. 2019. arXiv: 1806.10758 [cs.LG].

[6] Scott Lundberg and Su-In Lee. *A Unified Approach to Interpreting Model Predictions*. 2017. arXiv: 1705.07874 [cs.AI].

[7] Chris Olah, Alexander Mordvintsev, and Ludwig Schubert. "Feature Visualization". In: *Distill* (2017). https://distill.pub/2017/feature-visualization. DOI: 10.23915/distill.00007.

[8] Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. *Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps*. 2014. arXiv: 1312.6034 [cs.CV].

[9] Matthew D Zeiler and Rob Fergus. *Visualizing and Understanding Convolutional Networks*. 2013. arXiv: 1311.2901 [cs.CV].