Guidelines to explain machine learning algorithms

Frédéric Boisnard, Ryma Boumazouza, Mélanie Ducoffe, Thomas Fel, Estèle Glize, Lucas Hervier, Vincent Mussot, Agustin Martin Picard, Antonin Poché, David Vigouroux

IVADO SAINT EXUPÉRY CRIAQ ANITI Porté par l'Université de Toulouse Institut intelligence et données

DEEL

DEpendable & Explainable Learning

Guidelines to explain machine learning algorithms

This document produced by the DEEL team aims at bridging the gap between AI's decision-making process and human understanding, by providing guidance on the use of explainability methods.

v1-23/09/13

Authors

F. Boisnard	Renault & Aniti
R. Boumazouza	Airbus & Aniti
M. Ducoffe	Airbus & Aniti
T. Fel	Carney Institute for Brain Science & SNCF & Aniti
E. Glize	Thales & Aniti
L. Hervier	IRT Saint Exupery & Aniti
V. Mussot	IRT Saint Exupery & Aniti
A. Picard	IRT Saint Exupery & Aniti
A. Poche	IRT Saint Exupery & Aniti
D. Vigouroux	IRT Saint Exupery & Aniti

Contacts

david.vigouroux@irt-saintexupery.com

agustin-martin.picard@irt-saintexupery.com

lucas.hervier@irt-saintexupery.com

antonin.poche@irt-saintexupery.com

vincent.mussot@irt-saintexupery.com

DEEL
DEpendable & Explainable Learning

Contents

Abstract

In the rapidly evolving and increasingly complex field of Artificial Intelligence(AI), understanding and interpreting the decision-making process of models is crucial. This document serves as an essential guide, aiming to bridge the gap between AI's internal operations and human understanding. It provides a comprehensive overview of several explainability methods: attribution methods, feature visualization and concept-based approaches. Other methods, like example based methods or subset minimum methods, are not included in this first version of the document but they could be integrated in a future version. This guide focuses on detailing and illustrating the usage of the explainability methods in an operational environment while also shedding light on their differences and inherent limitations. Emphasizing the critical role of human factors and expertise in interpreting the models' decisions, this document guides the reader towards a thoughtful and informed exploration of AI's intricate decision-making process.

1 Introduction

With the increasing complexity of AI models [33], their opacity has grown, giving rise to a need for new tools [5, 30, 42] to understand their decision-making process. Explainability methods [10, 25, 36, 53] have thus emerged to attempt to bridge the gap between the internal workings of models and the understanding of a specific task by a human.

In the case of an incorrect decision on a specific input, for example, it is natural to seek to understand what led to this decision, to comprehend why the model fails at its task, and find ways to improve it in similar cases. Even when the decision is correct, it remains relevant to ensure that the decision is "*right for the right reason*"[48], which will enhance the confidence a human can have in the model.

Moreover, datasets and learning processes generally contain biases that users must be aware of to ensure the best possible performances. Explainability methods can also be used in this context to try to detect the features that, according to a human, should not have been used by the model to perform the desired task. Conversely, they can be used to strengthen the confidence one can have in the final model. Still, in this case, it is necessary to approach explainability methods with in-depth knowledge of the subject to avoid human biases that may arise from misuse or misinterpretations.

In AI, the choice between several models is generally based on performance, often related to cost factors. However, explainability can assist in this decision-making process, for instance in the context of critical systems, or systems where the trust humans can place in their decisions is a major aspect.

Lastly, explainability can also lead to the discovery of new knowledge, by extracting previously unknown factors that contribute to a model's decision This aspect, however, requires a high level of expertise, both in the targeted domain and in the explainability methods used.

In Short...

- Explainability allows to identify potential issues and either enhance the overall model's performance or improve the confidence a human can have in its decisions
- Explainability methods entails several risks and should be approached with caution.

Figure 1 presents a high-level view of explainability, which can be seen as a way to represent an object in the mental model of a human target [8, 10, 18]. More specifically, it highlights that while explanations are often produced by methods and tools (like the ones presented in this document), they always rely on humans for *interpretation*.

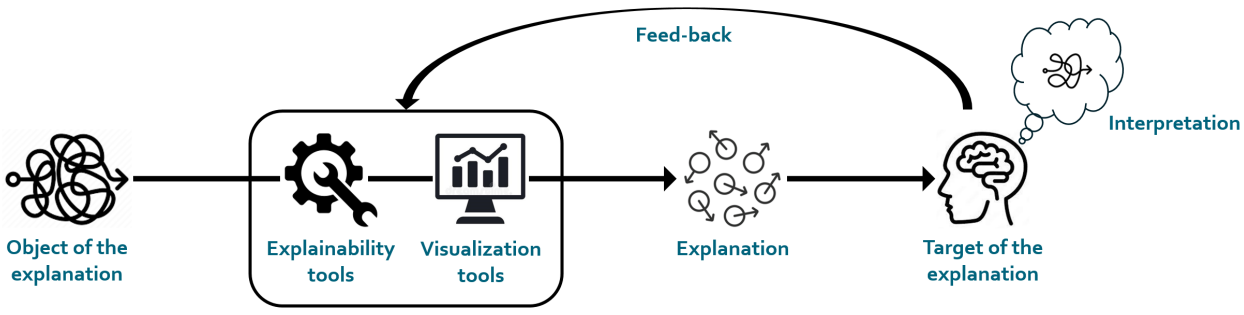


Figure 1 – Overview of explainability

1.1 Document motivations

The primary motivation behind this document is to provide guidance on the selection and application of explainability methods in various use cases. The objectives of this document can be summarized as follows:

- Give a general understanding of the main differences between the various approaches, their strengths, and their intrinsic limitations.
- Assist in choosing the appropriate explainability methods that suit specific use cases, which can lead to better insights and more effective model interpretation.
- Provide guidance to help validate models using explainability methods, for instance by identifying abnormal behaviors like biases, GDPR-related issues [27], and failure cases, as well as general model trends.
- Provide valuable insights and recommendations for a wide range of individuals involved in the model development and evaluation process, such as developers, quality auditors, investigators, and experts.

Methods recommendations throughout the document

Some recommendations provided in this document rely on researchers' experimental knowledge rather than a formal academic study. While these recommendations may serve as a starting point, their suitability for specific use-cases may vary. Consequently, it is advised to consider them as non-binding suggestions. The tag [ADVICE] is assigned to denote the presence of such recommendations.

What is this document *NOT* about?

Firstly, this document is not aimed at presenting the explainability methods. The methods will often be briefly introduced to give the reader the broad understanding required to be able to make informed choices, but we refer the reader to the original papers for a deeper understanding of the underlying mechanisms.

Going further...

This document is completed by a series of in-depth tutorials available in the Xplique library

Second, this document is not intended to help in tuning the methods and their hyperparameters. This is better done through practical applications and tutorials. For this, we recommend taking a look at the Xplique library [15] which provides a set of high-level tutorials for each explainability method in their readme, as well as in-depth tutorials.

Finally, it is worth noting that this document focuses solely on *post-hoc* explainability methods due to their popularity in the domain and their ease-of-use. Therefore we will not consider other types of explainability methods such as *by-design methods* [4, 31, 49, 50], or *data-centric methods* [28].

1.2 Target audience

As illustrated in Figure 1, the explanations we need to produce may vary depending on the target of this explanation [36]. Therefore this document is designed for a diverse target audience, including:

- **[DEVELOPERS & QUALITY AUDITORS]**: The content is aimed at guiding developers during the validation of their models, enabling them to use explainability as part of their quality process.

This role is closely related to quality auditors, which can leverage explainability to assess the respect of the quality process.

- **[AUDIT - INVESTIGATION]**: The document also serves as a resource for those conducting audits or investigations, providing them with valuable insights and methodologies to examine the model's decision process.
- **[EXPERTS]**: Domain experts who seek to uncover general rules from complex data sets that may be challenging for human understanding can also benefit from the information provided in this document.

All recommendations provided in this document primarily target developers and quality auditors. However, specific sections or elements meant for "AUDIT - INVESTIGATION" or for "EXPERTS" will be explicitly identified throughout the document, ensuring that each audience can readily find the information most relevant to their needs.

1.3 Overview of methods

This document is organized around the three following families of explainability methods: Section 3 presents *attribution* methods [1, 7, 12, 13, 18, 35, 39, 41, 45, 51, 52, 54, 58, 59, 64, 65, 66] and their associated metrics [1, 2, 3, 6, 8, 17, 19, 21, 23, 24, 32, 34, 47, 56, 60, 62], Section 4 describes *feature visualization* methods [37, 38, 40], and finally Section 5 covers and illustrates *concept-based* methods [16, 20, 29, 31].

These families of methods can be summarized as follows:

- **Attribution methods** focus on understanding the contributions of individual input features to a model's output, providing *local* explanations for specific instances or predictions. They can be model-specific[7, 14, 44, 51, 64] or model-agnostic [12, 39, 41, 45, 46, 54, 55, 57, 59, 65, 66] and are applicable to various data types and tasks, such as image, tabular, text, and time series data, as well as classification, regression, and other predictive tasks.
- **Feature visualization methods**[37, 38, 40, 63] primarily deal with image data and classification tasks, aiming to understand the overall logic and functioning of neural networks, thus producing only *global* explanations. They generate images that maximize the activation of a neuron or layer for a specific class. It provides a representation of what the model interprets about specific data (what it considers as a bear for example), which offers insights into the model's decision process.
- **Concept-based methods**[16, 20, 29, 31] aim to provide a more intuitive understanding of a model's behavior by connecting its internal representations to human-understandable concepts. They explore the relationships between the model's components and concepts, offering a higher-level understanding of the model's decision-making process and can produce both *local* and *global* explanations. Concepts can either be predefined by the user or automatically extracted depending on the method.

These families of methods all come with their qualities and limitations, and they may provide complementary results for the same problem, which is why we recommend using several of them whenever it is possible. However, we provide the following necessary information to make a first selection of methods according to your own constraints.

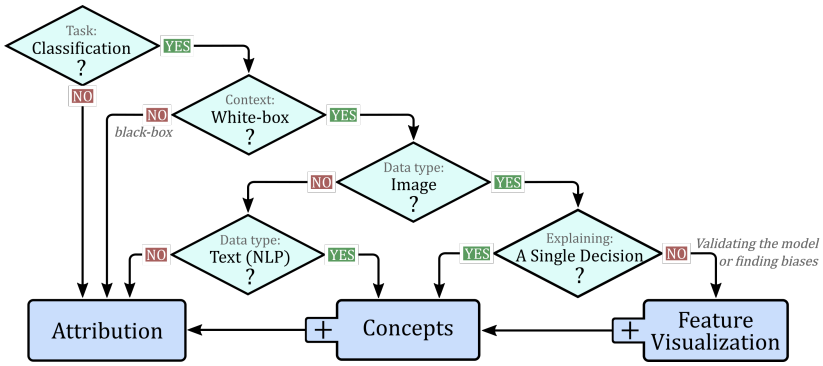


Figure 2 – Choosing an explainability method.
When Feature visualization could be used, Concepts and Attribution could be used too.
When Concepts could be used, Attribution could be used aside of concepts methods.

Method's maturity

We consider the maturity in relation with the Technology readiness level (TRL) of the methods.

- Feature visualization and Concept-based methods are considered low TRL
- Attribution methods are considered medium TRL

1.4 Choosing a family of methods

Figure 2 provides a generic view on how to choose an explainability method depending on several constraints. Note that all families should be jointly used whenever it is possible. This diagram will be updated when new methods will be extended to new data types or tasks.

The first aspect to consider is the task itself, which may already limit the applicable methods. Indeed, for all tasks other than classification, the only methods that are mature enough to be recommended are based on attributions. Similarly, this family of methods is also the only one that can produce results in a *black-box* context, where no information about the model or the training is known. The type of data may also be an important factor to consider, as various methods only support images for now.

Furthermore, in the specific context of an image classification task, the objective behind the use of explainability methods may also be restrictive: *Feature visualization* for instance can only be used for *global* explanations, which means detecting model biases and trends for groups of samples. Feature vizualization shall be used in addition to concepts based methods which could provided global explainations and local. If the aim is to investigate specific failure cases, attribution and some of the concept-based methods will be more appropriate than Feature vizualization.

The computation time could be an important aspect in some cases as well. For instance, white-box methods are typically faster than most black-box methods. We provide a few elements of response on this subject in the presentation of the attribution methods in section 3.

However, in any case, and as mentioned earlier, we recommend combining multiple methods whenever it is possible both to obtain complementary explanations, but also to cross-validate the results and mitigate errors that could come from the explainability methods themselves.

1.5 Limitations and challenges

The field of explainable AI faces several limitations and challenges, which must be acknowledged to ensure appropriate interpretation of model explanations and effective decision-making based on these explanations. These limitations include:

- **Inherent limitations of methods**: Each family of methods has its own limitations that need to be taken into account when interpreting results. We detail these limitations in the next sections.
- **Human factors**: As illustrated in Figure 1, explainability always requires humans in the loop, which opens the door to potential biases that must not be overlooked. One particular concern in this regard is confirmation bias, which refers to our tendency to interpret information in a way that confirms our pre-existing hypotheses, while ignoring information that challenges them. For instance, when interpreting an explanation about the model, it is important to keep in mind that just because the explanation seems to make sense, it does not necessarily mean that it accurately reflects the underlying decision-making process of the model.
- **Necessity of expertise**: This applies to both domain experts and XAI practitioners. On one hand, interpretations often requires the involvement of domain experts to understand the explanations themselves, and on the other hand, deep knowledge of the explainability method used is highly valuable to prevent biases during interpretation.
- **Model performance limitation**: Explainability methods attempt to represent the inner decision-making process of a model. Therefore, a prerequisite is to have a model with good performance before attempting to use these methods. Otherwise, the explanations produced could be noisy and misleading, and the conclusions drawn from their interpretation would be highly disputable.
- **Scientific maturity**: Although there is extensive knowledge and experience regarding attribution methods, other emerging techniques are still in the process of gaining scientific maturity. The guidelines for these methods may be less precise due to the current state of scientific knowledge, and a more expert-level understanding may be required.
- **Rapid evolution of the field**: The field of explainable AI is rapidly evolving, with new techniques and approaches being developed and existing ones being refined. As a result, recommendations and guidelines based on the current state of the literature may change over time. This is particularly relevant for feature visualization and concept-based methods, where the current level of understanding of these technologies may not yet be sufficient.

In light of these challenges, it is essential for practitioners to stay up-to-date with the latest advancements in the field and be prepared to adapt their approaches as new insights and techniques emerge. Furthermore, close collaboration between domain experts and explainability experts can help ensure accurate interpretation of model explanations and more effective decision-making based on these insights.

In Short...

- Stay up-to-date with new XAI methods
- Make domain experts and XAI practitioners collaborate

2 Impact of data choices on explanations

Theoretically, for a given problem, the entire training dataset should be used for explainability studies. However, it is often necessary to select a subset of the training data, especially when the dataset is too large. Moreover, this selection of a subset of data is required at two specific stages of an explainability analysis: when calibrating explainability methods and when interpreting the explainability results.

2.1 Impact of data on the calibration of explainability methods

It is common practice to calibrate and optimize the hyperparameters of explainability methods for a given problem on a subset of existing data, for a reduced computation time but also to ensure consistency between explanations. This step is generally not done using a single sample but rather a subset of existing data (or all available data if the dataset is not too large). Indeed, the optimization of hyperparameters per sample is an area that is not yet settled in the literature. It would be interesting to investigate whether the issue is merely due to computation time constraints and other factors or if there are more fundamental reasons behind it. Therefore, while we encourage further exploration in this direction to address this scientific question, our current recommendation is to follow the existing approaches in the literature: the choice of data split for methods calibration can be done simply by random selection from the training dataset [ADVICE].

2.2 Impact of data on the interpretation of explanations

Explainability methods can be applied with various objectives in mind, which can impact the selection of the data subset on which to perform the analyses.

2.2.1 Investigate failure cases

If the goal is to investigate the model's failure cases (significant loss, poor classification predictions,...), particularly in the context of an [AUDIT - INVESTIGATION] activity, it will naturally be interesting to select the corresponding test examples to understand the model's errors' origins. In this case, it is also recommended to randomly select data from this subset. In particular, when considering a classification task, this will ensure a representativity of errors between all classes. However, it is worth noting that if the model is wrong, but the explainability results are correct (they assess that the model focuses on the right information to make its decision), it will not be possible to confidently conclude anything.

2.2.2 Sampling strategies and interpretations

In the context of validating models and detecting model biases or general trends, it is recommended to select a representative set of samples from the dataset. [ADVICE] The number of samples will often depend on the methods and will be detailed in the next sections, but in the context of a classification task for example, there should be at least a dozen of instances of each class analysed before drawing first conclusions.

Whenever sampling is needed, there are several ways to proceed. We recommend starting with random sampling as it is the simplest method. And if a deeper analysis is required, we recommend following the order of this list to use the other methods, as they are ranked by complexity of application:

- **Random sampling**: A basic method consists in sampling randomly from the dataset. However in this case, even if explainability methods seem to produce good results, it is not possible to conclude that the model works well "everywhere," especially if the number of samples is small. Furthermore,

it is recommended to investigate more deeply the samples similar to the failures identified by this method to better characterize the detected biases.

- **ODD sampling [EXPERTS]**: Collecting a subset of data deemed representative of the Operational Design Domain (ODD) is a good practice, and performing explainability analysis on this subset allows to validate that the model behaves correctly in its ODD. However, when faced with abnormal explanations in this subset, it is recommended to over-constrain the ODD around the concerned samples to draw new data in order to better characterize the source of the poor results and detect potential model biases.
- **Influence data-point detection [EXPERTS]**: Some methods allow the identification of important points in the training dataset from the point of view of the learned model (Influenciae library). These methods can provide both the most and the least influential data points, which can both be used as a starting point to obtain insights on the model:
  - For the most influential data point, an error identified by an explainability method may suggest that the model did overfit on these specific samples, which may stem from the under-representation of other groups in the training dataset. In this case, it is recommended to identify the semantics behind these influential data points and to compensate for the missing data to overcome the problem.
  - For the least influential data point, an error detected through explainability may indicate that the model has an important bias on a characteristic sample, in the same fashion as what prototypes may indicate.
- **Prototype-based sampling [EXPERTS]**: Methods like MMD-critic [28] allows identifying samples characteristic of groups which are called *prototypes*. In this case, if explainability methods identify model failures on a particular group, it may indicate that the model fails on a primary logic of this group. It is therefore recommended to investigate this group in detail to correct the source of this bias, by selecting more samples from this group or samples with different characteristics.

3 Attribution methods

Attributions are *post-hoc* local explanation methods that highlight the importance of each input feature for a given decision. In the case of images, they are most of the time represented by heatmaps superimposed over the image as shown in figure 3 where important pixels are hot. In the case of text, important words are highlighted with more or less intense colors, for tabular data there exist several possibilities, one of them is a bar plot, and for time series no fixed visualization was proposed to our knowledge but heatmap could also be used. The motivations behind those explanations are provided in section 3.1. Figure 3 also shows that different methods may bring different explanations. The selection of the methods for a given model with the help of metrics is described in section 3.2, then we dive into the interpretation of abnormal explanations section 3.3 before concluding with the attribution methods' limitations 3.4.

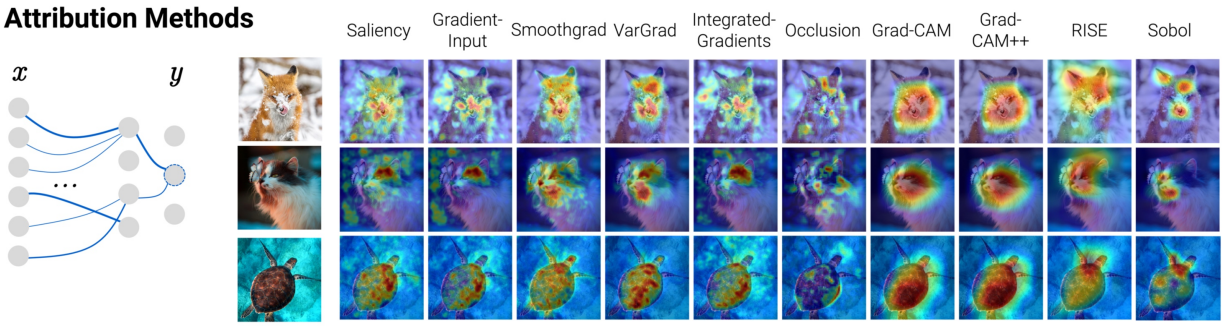


Figure 3 – Illustration of attributions methods

3.1 Motivations

Attribution methods are the most popular explanation methods because they are easy to use and the idea is easy to understand. Therefore, there exists a large variety of attribution methods, and metrics to evaluate them (16 methods and 6 metrics are available in Xplique). Furthermore, such methods allow to:

- **Detect bias**: The figure 4.1 with the example of husky and wolf shows that the model used the background to make its decisions.
- **Understand failure cases**: with the same example, when faced with a wolf with a grass background the model would predict a husky. Attribution methods can allow us to understand where the errors are coming from.
- **Validate and compare models**: With enough coverage, we can validate the model behavior overall and compare models' explainability.
- **Knowledge discovery**: like most explanation methods, if the model performs better than humans and uses unknown information to us, then explainability may be able to extract this for us [8].

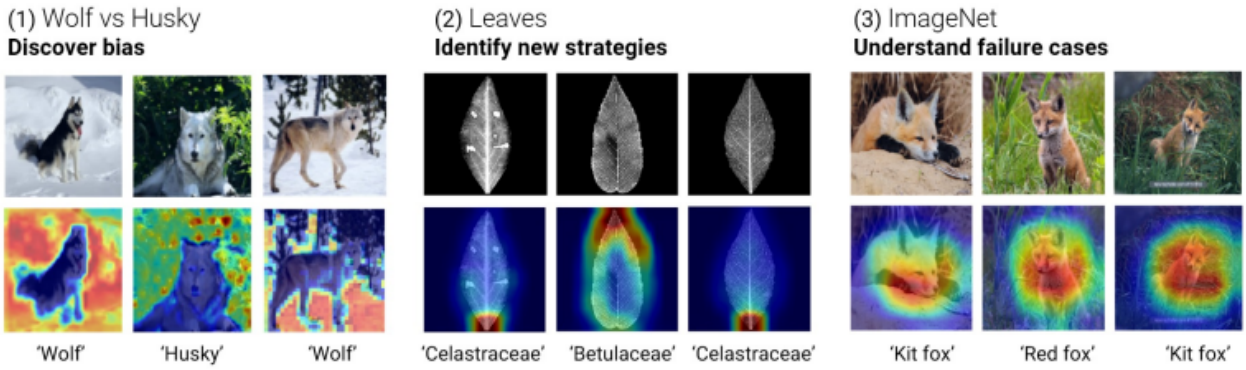


Figure 4 – Illustration of possible uses of attributions from Colin et al. [8]. It was demonstrated that attributions are useful to detect bias in the case of (1) and (2). However, in some case like (3), human participants are not able to understand the bias of the model thanks to attribution methods.

3.2 Selection of the right explainability method(s)

Source of the provided elements

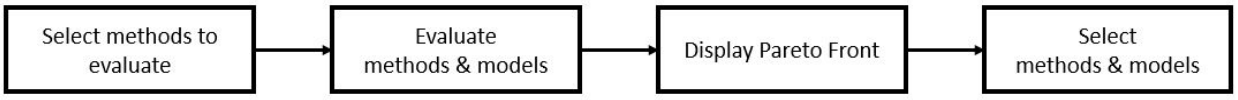
The provided elements in the "Selection of the right explainability method(s)" and "Interpretation of abnormal explanations" sections that are not explicitly associated with a reference can be considered a recommendation/advice from the redaction team coming from our experience.
If you know of papers supporting our claims, please inform us, it would give better grounding to the document.

There are many explanation methods proposed in the literature and available in Xplique. Hence, choosing the most appropriate method becomes challenging, fortunately, there are metrics to evaluate and compare those methods for a given decision or model. In this section, we present the evaluation strategy of explainability methods, then detail the use of metrics, and describe how those metrics can also be used to evaluate the model's explainability.

3.2.1 Evaluation strategy

Before choosing specific explainability methods or hyperparameters, users must thoroughly understand the method's and hyperparameters' workings. Numerous tutorials are available in libraries like Xplique to help users grasp the methods and the impacts of hyperparameters. Some methods need access to the model's gradient (white-box methods) and others are limited to images. Users should select methods applicable to their specific problem and conduct a hyperparameter search accordingly.

A Step-by-Step Evaluation Strategy:



a) **Select method and hyperparameters:** Choose appropriate explainability methods and a set of hyperparameters for your problem. To do so, users should thoroughly read the different unit method tutorials.

b) **Evaluate methods and/or models:** Assess the selected methods and models using explainability metrics such as fidelity (insertion, deletion mu-fidelity), stability, and consistency. Those metrics should be evaluated on what could be called an explanation evaluation dataset, it should be designed in the same way as the one for explanation application (see section 2). Ideally, this dataset should not overlap with the dataset on with we will study explanations. This step helps identify which methods provide the most reliable explanations. To know how to apply metrics, please refer to Xplique metrics tutorial, and for interpretation and decisions, see section 3.2.2 for method selection and 3.2.3 for model selection.

c) **Include all relevant metrics:** All metrics evaluate different properties of the explanation, they should hence all be included in the evaluation.

d) **Display the Pareto front:** Plot the Pareto front, representing the trade-offs between various explainability metrics. The Pareto front illustrates the performance of different methods and models on multiple metrics simultaneously. A simple example can be found in figure 5. (For model selection, make sure also to consider non-explainability metrics such as accuracy or robustness.) See 3.2.2 for more details.

e) **Select the most suitable methods and models:** Based on the Pareto front of explainability methods for a given a model, pick the methods/models that best represent the desired trade-offs between the metrics. We could divide the selected methods into levels depending on the Pareto front, our analysis would be focused on the first level, and the second level method should be kept if further analysis is needed. As an example, in figure 5, we could select the red points as the first level of methods.

f) **Consider multiple explainability methods:** Due to the complexity of the Pareto front, it can be challenging to pick the best methods and models, one method may be better on a metric and worse on another. In such cases, consider using multiple explainability methods to ensure robust and reliable explanations. Furthermore, using several methods allow us to cross-check explanation coherence between methods. If different methods provide drastically different results, we cannot make conclusions based on them. Note that the same method with different hyperparameters is still considered the same method. Indeed, when selecting several methods, we want diversity in the explanation and different methods may provide it, while explanations from the same method with different hyperparameters are often similar.

By following this comprehensive evaluation strategy, users can effectively choose the most suitable explainability methods and models for their specific problem, ensuring reliable explanations and paving the path toward interpretability.

3.2.2 Metrics to select the best explanations for a given model

Intuitively, the best explanation of a decision could decided by asking to the final user if the explanation is the best for his point of view. However, it was demonstrated in the literature that this approach could be biased and several additional properties shall be satisfy to select the best explanation. To evaluate such properties and compare explainability methods, calculable metrics was proposed. Many properties

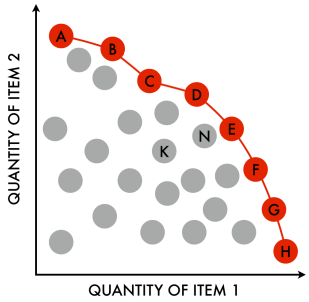


Figure 5 – A production-possibility frontier. The red line is an example of a Pareto-efficient frontier, where the frontier and the area left and below it are a continuous set of choices. The red points on the frontier are examples of Pareto-optimal choices of production. Points off the frontier, such as N and K, are not Pareto-efficient, since there exist points on the frontier which Pareto-dominate them. (from wikipedia.)

were presented in various papers and surveys. We give for example the five from Fel *et al.* [17] and Utility from Colin *et al.* [8], and the associated metrics in Xplique in parenthesis:

- **Fidelity**: How representative of the model behavior is an explanation. (Insertion, Deletion, Mu-fidelity)
- **Stability**: Is the explanation robust to small perturbations on the samples? Note that the explanation's stability also depends on the model's stability or robustness, hence for non-robust models this property is hard to evaluate. (AverageStability)
- **Comprehensibility**: How interpretable are the explanations? (Human subjective evaluation)
- **Generalizability**: To what extent does the explanation truly reflect the underlying decision process? (MeGe)
- **Consistency**: How logically similar are the explanations of two different predictors trained on the same task? It may not be wanted if predictors have different behaviors. (ReCo)
- **Simulatability**: How much explanations help users identify rules driving a model's predictions (correct or incorrect) that transfer to unseen data [16]. (Utility)

To choose the metric to evaluate the attribution methods, one should keep in mind the following points:

- **Plurality of samples**: In statistics and machine learning, to estimate a value, we need multiple samples. Explainability metrics are also empirical evaluations, at least 30 samples should be used.
- **Robust metrics values**: In fact, one should evaluate the metric values on several sets of at least 30 samples and verify that the ranking between methods is stable. One could look at the standard deviation. We suggest doing it only to compare methods, not during hyperparameters optimization.
- **Same context**: The samples used to evaluate each method should be the same.

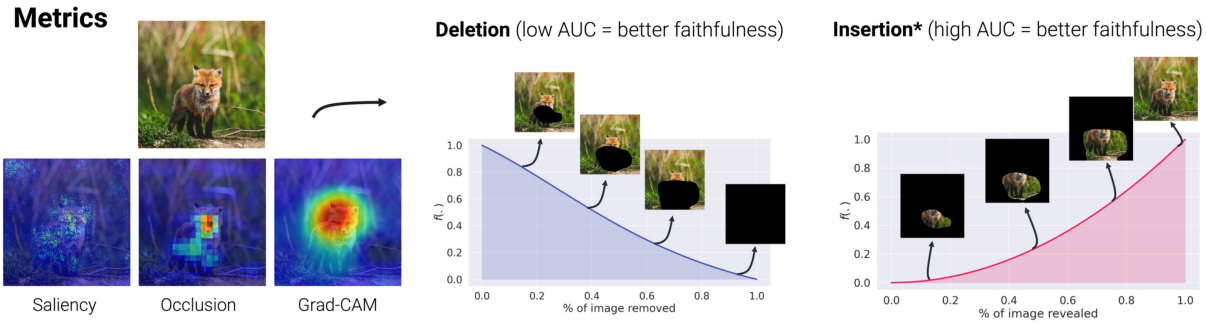


Figure 6 – Examples of insertion and deletion fidelity metrics for attribution methods. They gradually delete or insert the most important pixels (according to the attribution method under study) and look at the prediction score's evolution. Removing/Inserting the most important pixels for the model shall have the most impact on the prediction of the model. The AUC (Area Under the Curve) reflects the faithfulness of the explainability methods.

- **Metrics' parameters consistency**: The metrics' parameters should not be modified between each method evaluation. Furthermore, in the same way as methods, some metrics rely on baselines, which represent the complete lack of information (such as black or gray pixels for images, like in data augmentation occlusions). Therefore, we suggest that the user ensures consistency and coherence between all baselines used.
- **Variety of metrics**: It is essential to recognize that no single metric can comprehensively assess explainability. Each metric captures a unique aspect of explainability that the method should possess. Therefore, considering multiple metrics and avoiding overemphasizing a single one, such as fidelity, is vital.
- **Adaptability**: Explainability metrics are adaptable to different data types and problem domains. Although some libraries only support specific data types, the selected metrics should be adjustable to accommodate any problem.
- **Computation costs**: Consider the computation time and ease of use when selecting metrics. Some metrics may be less expensive (e.g., Deletion, Insertion, Mu-fidelity) while others can be more expensive and/or complex to apply (e.g., AverageStability, Consistency, Representativity, Robustness, Comprehensibility). Choose metrics based on your needs, operational constraints, and resources.
- **Comprehensibility property**: The interpretability of an explanation cannot be automatically evaluated through metrics as it involves the human's evaluation of comprehensibility. Including human-centered metrics can provide valuable insights but requires extensive additional effort and resources. Hence, we cannot easily include this metric in the Pareto front, nonetheless, this property cannot be ignored as it is the primary objective of explainability. Depending on the situation, one can opt for constructing human attribution maps or conducting human experiments:
  - Create human attribution maps [8], then compare them with the method's attribution. Note that this method takes confirmation bias into account but requires expert knowledge of the dataset. With this method, we could automatize the process rigorously. However, it possesses a high development cost, hence, we propose a second solution with a lower cost.

  - First, construct the Pareto front for the other metrics and select the best methods for a given model. Then, make a user study to evaluate the comprehensibility of the obtained methods-model pairs.

- **Optimal metrics value**: Metrics can be used to rank and compare different explainability methods. However, it is essential to remember that reaching the "optimal" value for a given metric may not always be possible, and knowing the maximum value for a specific problem may be impossible.

3.2.3 Choose the best model regarding explainability

Warning...

Insertion and Deletion are faithfulness metrics for XAI methods, and thus, cannot evaluate the extent to which a model is explainable/interpretable.

In the case of model selection, there is a way to include explainability in the comparison. Once the best method is selected for each model, some metrics (Mu-fidelity, Consistency, Representativity, Stability, Robustness) can also be used to evaluate the model's explainability. To apply such metrics to model selection, the different models should solve the same problem and the metrics should be computed on the same samples. Similarly to method selection, there is no metric better than the other and it creates a new Pareto front. In this Pareto front, one should include other model's evaluation metrics such as accuracy or robustness. We may face the often mentioned in the literature "explainability-accuracy" trade-off, but recent papers [8] have shown that this trade-off may not always hold.

Note that one should not forget to include interpretability or comprehensibility in the evaluation and comparison of the models.

We previously encouraged the selection of several explainability methods for a given model. However, in the case of model selection, we should only select the best explainability method for each model, because the model's Pareto front would otherwise be too complex to analyze with several explainability methods for each model.

3.3 Interpretation of abnormal explanations

When one applies explainability methods to a model, an explanation could either match our expectations or be considered an abnormal explanation. Abnormal explanations are not necessarily incoherent explanations, it could be interpretable yet different from the expectations. Furthermore, we expect abnormal explanations to be faithful to the model's decision process but reflect that the model's behavior deviates from our expectations. Some examples of what abnormal explanations could be and possible conclusions:

- The highlighted part is not where we expected it to be.
  - The model seems to have detected a link between an unimportant element and the label. It may come from a lack of diversity or bias in the dataset, like with wolfs and huskies in figure 4.
- The highlighted part is smaller or has a different form. (In the case of images).
  - The model's decision was based only on a part of what we would judge as important. For the model, those parts seem characteristic and sufficient to make its prediction.
- What we expect to have a positive influence on a class prediction has a negative influence.

  - In most cases, there is a difference between the model's prediction and the label. In this case, the input features (like pixels) with negative influence indeed pushed toward the right decision.

- The explanations are too noisy or fuzzy.

  - The model may have a noisy behavior and low performance, or the data distribution deviates from the training distribution (outside of the ODD).

When interpreting explanations produced by explainability methods, it is crucial to pay attention to abnormal or unexpected results. Abnormal explanations can occur due to various reasons, such as biases in the training data, biases in the model, or issues with the explanation method itself. These unexpected explanations can provide valuable insights into potential problems within the model.

Warning...

Interpretations are done by humans, they could be subject to many different biases.

For interpretability, it is essential to take into account the following factors:

1. **Validate explanation**: Make sure the obtained explanations are the best explanation one could obtain with each given method. Please refer to the unit methods tutorials to do so.

2. **Cross-validation**: An incoherent explanation for a given method does not necessarily mean all explainability methods will fail. Hence, it is vital to compare several explanations for a given decision.

3. **Data quality**: Examine the quality of the data used to train the model and generate explanations. Abnormal explanations could arise due to noisy, incomplete, biased data, or data outside of the training set distribution (model's ODD). Addressing data quality issues can help improve the accuracy and interpretability of the model and its explanations.

4. **Model limitations**: Assess whether the abnormal explanation is a result of an issue with the underlying model. This could include overfitting, underfitting, biases introduced during the training process, or due to samples outside of the model's ODD. Investigating these potential issues can help identify areas for improvement in the model's architecture, training data, or training process.

5. **Failure cases**: One should compare the model's prediction to the expected prediction if available (and/or vice-versa). If the model is correct, making sure the model based its decision on the expected element should be simple. However, in the case of an incorrect prediction, the interpretation is more complex and requires more analysis.

6. **Human factors**: Acknowledge that human interpretation is subject to cognitive biases and limitations. The collaboration between data scientists, domain experts, and other stakeholders to evaluate the abnormal explanations from multiple perspectives, ensures a more comprehensive understanding of the model's behavior. Confirmation bias requires particular precautions.

7. **Over-interpretation**: Furthermore, humans have the need to understand something and sometimes draw conclusions without bases. Indeed, there are cases when we cannot make conclusions based on attribution methods [8], this is further described in the limitations (section 3.4).

3.4 Limitations

It is essential to understand the limitations of explainability methods and the limitations of the associated metrics to make informed decisions when selecting appropriate explanation methods and interpreting model behavior. Here's a detailed overview of some key limitations:

- **Baseline dependence**: Perturbation-based methods and some metrics such as Mu-fidelity, insertion, or deletion rely on the concept of a "baseline" (a value representing the absence of information). Defining this non-information can be challenging, and different baseline choices can lead to varying explanations. These metrics which use baselines are also inherently biased due to their baseline dependence. When using these metrics, one should consider the choice of baseline, which can be informed by the data augmentation techniques used by the model (for example, whether baseline masks were used during training). We also suggest the baseline value used for metrics matches the baseline value used by perturbation methods.

- **Interpretability evaluation**: The comprehensibility property is key for model interpretability. However, its evaluation is expensive because humans are needed, and ensuring that no bias is introduced further increase complexity.

- **Human limitations**: Attribution methods can help identify significant biases in the model, but they have inherent limitations related to human interpretation. A human may not always be able to pinpoint the source of the model's bias, even if the attribution method identifies a problem. This is because people may not have the necessary domain knowledge, dataset knowledge, or model knowledge to understand the underlying cause of the identified bias. Additional explanatory methods, such as concept-based and feature visualization methods, should be employed in parallel to gain more insights.

- **Where**: The attribution method provides information on the "where" the model looked (location in an image, for example) and not on the what" [16] (the semantic information it recognized, what it actually saw). This means that attributions are explanations of which input features (like pixels) were used for the decision but not what was recognized in those features. The problem comes from the format and not the method, as explanations from humans with the same format have a similar constraint [8]. This is tackled by other types of explanations such as concepts (see section 5).

- **Complementary explanations**: Other explanation formats coupled with attributions may bring more developed explanations and simplify the interpretation. As an example, figure 10 shows a classic attribution on the left, and two concepts feature attributions in the middle.

4 Feature visualization

Feature visualizations are generated images that maximize one of the model's outputs. Feature visualization methods typically start from a random image (only noise) and gradually modify it to obtain the desired feature visualization. The updates are applied along the gradient ascent direction which gradually maximizes the value for a given output. It allows generating images that represent the model's interpretation of the class associated with the output. This mechanism can also be applied to neurons, layers, or directions in the latent space. Nonetheless, this section will focus on output feature visualization as it is arguably easier to interpret and recommendations can be extrapolated to other types of feature visualizations.

Warning...

Methods with low TRL maturity!
Many open questions remain.

Although they haven't become as popular as attribution methods, there exists a clear line of research on how to perform feature visualization [37, 38, 61], with Olah et al.'s method being the most widely used [40]. Despite the efforts of the research community, they are oftentimes challenging interpretation and their scalability issues push us to consider these methods as not mature enough from the point of view of the TRL. This section will describe the motivations behind feature visualization 4.1 before presenting an evaluation strategy 4.2. Then this section explores the interpretation of feature visualization explanation 4.3 and finally, the limitations of such methods 4.4.

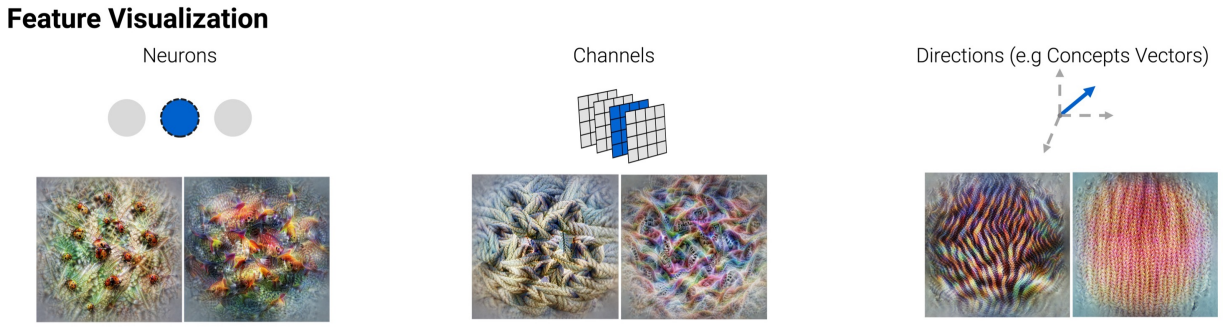


Figure 7 – Illustration of feature visualization methods

4.1 Motivations

The generated feature visualization images for a class represent what the model associates with the given class. On the first hand, this method allows the extraction of the key patterns learned by the model. Those patterns are shapes, textures, and colors that the model associates with a class. The feature visualization on the left of figure 7 represents the class *ladybug*. We can recognize several red insects with sometimes a black head or black dots on their body. Those are patterns of ladybugs and we can see them on the feature visualization. On the second hand, feature visualizations could allow for detecting bias. Indeed, when a pattern that is identifiable by humans but would be considered to be background information appears on the generated image, then the model could be exhibiting biased behavior. As an example, in the left image of figure 7, the background is green, which means that for the model, ladybugs are more easily classified on grass or leaves. Hence, feature visualization is a powerful technique to understand the overall logic and functioning of a neural network. It offers valuable insights into how the network processes and extracts patterns from input data. Here, we provide a global recommendation for using feature visualization methods:

- **Applications**: Feature visualization is, to our knowledge, only employed in image data and classification tasks. Although it could potentially be extended to tabular or time series data, our recommendations are limited to image data and classification tasks due to limited experience with these techniques in other domains.
- **Interpretation**: When using feature visualization, consider the following points:
  a *Class-level interpretation*: The method generates an image that maximizes the selected class logit, *i.e.* the prediction of the model regarding the selected class. To get the intuition take a model trained to identify hair color on portraits, like blond, ginger, brown, etc. In this case, feature visualization for the blond class could look like an image covered by blond hairs curly, for instance.
  b *Intermediate model layers*: In the same way as a model output, we could maximize an intermediate neuron, layers, or direction similarity. When a neuron's activation is maximized, the generated image shows what kind of shapes, textures, colors, etc. lead to the neuron's activation. A direction's cosine similarity maximization can be used for concept visualization as a concept can be seen as a direction in the latent space (examples in figure 9).

4.2 Evaluation strategy

Source of the provided elements

The provided elements in the "Evaluation strategy" and "Interpretation of abnormal explanations" sections that are not explicitly associated with a reference can be considered a recommendation/advice from the redaction team coming from our experiences.
If you know of papers supporting our claims, please inform us, it would give better grounding to the document.

In this section, we discuss how to validate feature visualization and the choice of method in the literature.

4.2.0.1 Validation criteria

: When validating feature visualization, consider the following criteria:

- **Effective optimization process**: Ensure that the optimization process performs well, achieving optimal results for the class under consideration. It should maximize the corresponding class's softmax value. It may not be exactly the value of 1, but a value that is close to 1 is to be expected.
- **Interpretability by humans**: Ideally, the resulting image from the feature visualization should exhibit some visual properties that humans relate to the target class. An understandable visualization will provide more valuable insights into the model's behavior and help identify potential issues or biases. Nonetheless, as with other explanations, feature visualizations can be subject to human cognitive bias and should be used carefully. Indeed, humans tend to fill in the blank and extrapolate patterns from insignificant information.

4.2.0.2 *Feature visualization methods in the literature*

What we call feature visualization and the method implemented in Xplique correspond to the one described by Olah *et al.* [40] in their Distill blog post. Also, the DEEL team also published a recent paper on the subject by Fel *et al.* [11]. And several other works also treated the subject [37, 38, 61].

4.3 Interpretation of abnormal explanations

Feature visualizations provide global explanations with the model understanding of classes or the visualization of concepts. The added value of interpretations of feature visualization is dual:

- **Identify key patterns**: Patterns represented in the generated image can be associated with what the model considers to be the key to classifying the class.
- **Detect bias**: We saw with the ladybug example in Figure 7 that feature visualization allows the detection of bias in the model. Those biases are often caused by a lack of diversity in the dataset. We encountered other examples such as the apparition of helmets for the "assault rifle" class or cobwebs for each spider's class. Nonetheless, the apparition of unexpected key patterns in feature visualization does not mean that the model needs such a pattern to make its prediction. It only means that the apparition of such patterns pushes toward the studied class. Recognizing these abnormalities can assist in improving data diversity and refining the training process. Nonetheless, the detection of such unexpected patterns is subject to human interpretation and those patterns may be necessary for the predictions.

4.4 Limitations

Even if feature visualization allows the identification of key patterns for the model and the detection of biases thanks to those key patterns, it should not be used as sole method for model validation. If biases cannot be detected through feature visualizations, it does not mean that such biases do not exist. Furthermore, the low TRL maturity of such methods prevents model validation using feature visualizations alone, it should be used as a complement to other explainability methods.

5 Concepts

Concept-based methods for model explainability come in two main forms: methods based on labeled concept datasets and methods for automatic extraction of concepts from the neural network. Following a brief overview of these methods, section 5.1 outlines the motivations for selecting a particular type of method, while sections 5.2 and 5.3 dive into each type of methods.

Warning...

Methods with low TRL maturity!
Many open questions remain.

What is a concept?

A concept is an abstraction of common elements between samples and it corresponds to a direction in the latent space. Figure 8 shows the visualization of six different concepts that the Craft method [16] associated with the given image. In this example, the detected concepts for the class "chain saw" seem to be: the chainsaw engine, the saw blade, the human head, the vegetation, the jeans and the tree trunk.

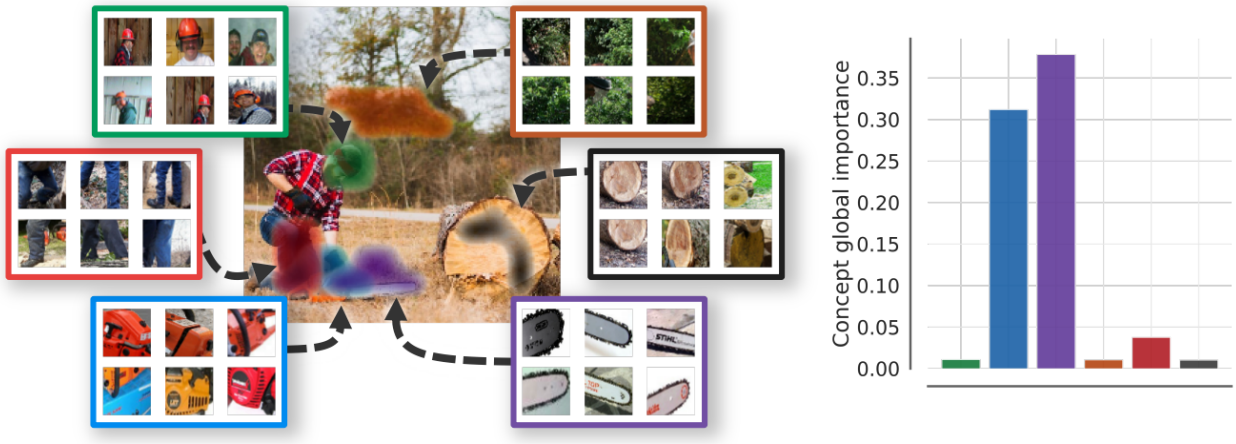


Figure 8 – Illustration of six concepts associated to an image by the Craft method.

Labeled Concept Dataset

Methods, like TCAV-CAV [29], rely on an additional dataset containing labeled concepts to verify if the decision of a model are consistent with what humans would use for the same task. In TCAV-CAV, a concept classifier CAV (Concept Activation Vector) helps to identify the presence of specific concepts in the network's internal representations and TCAV (Testing with Concept Activation Vectors) provides a measure of how much the model recognized the given concept in a sample/image.

Automatic Concept Extraction

Automatic extraction methods, such as ACE [20], and Craft [16], aim to identify concepts directly from the model's network by manipulating the latent space and without relying on an additional labeled dataset. Craft has shown promising results compared to ACE - *utility* metric used in [16] shows that Craft outperforms ACE -, therefore we focus on it in the following sections.

5.1 Motivations (Choosing between types of methods)

Each concept-based approach has its own benefits and limitations, which are crucial to understanding before deciding on the most suitable method for a specific use case.

Labeled Concept Dataset [AUDIT - INVESTIGATION]

This type of methods generate a local explanations (activated concepts per sample), and they should be used to verify that a concept is correctly used to classify a sample. Thus they describe a correlation relationship instead of a causality one. They have several advantages and disadvantages:

- *Strengths*: The concept database is an additional labeled dataset used to create the concepts. It is powerful, as it allows for defining "expert" concepts and incorporating this knowledge into the explainability and interpretability of the model.
- *Weaknesses*: The concept database can be resource-intensive to create and maintain. Moreover, users themselves can introduce biases into concepts through their construction. Finally, methods based on labeled concept datasets cannot entirely detect biases in the model, as the model may still focus on concepts that are not present in the concept database, leading to irrelevant outcomes (correlation explanation).

Automatic Concept Extraction [AUDIT - INVESTIGATION, EXPERTS]

Craft has been applied to image classification tasks [16] and NLP [26] in the literature. It generates both local and global explanations. Concepts can give a local explanations by highlighting concepts activated in a sample (see Figure 8). It can also provide a global explanation by showing the most representative concepts of a class (see Figure 9). Thus, it is used to ensure that a class is appropriately covered by a set of concepts or that an image's classification is consistent. Craft has several advantages and disadvantages:

- *Strengths*: Craft is a powerful method for extracting concepts, offering the potential for deep insights into the model's decision-making process.
- *Weaknesses*: The automated extraction of concepts can result in concepts that are more challenging to interpret compared to those constructed by users. Finally, despite its potential, there is not yet enough experience with Craft on diverse datasets to conclusively determine its effectiveness for all types of problems. More research and experimentation are needed to fully understand its capabilities and limitations.

In Short...

- TCAV-CAV for local explanations and Craft for both local and global explanations.
- Creation of a labeled concept dataset for TCAV-CAV.
- Low TRL maturity on the concept database for TCAV-CAV and low TRL maturity on the Craft applications.

Figure 9 – Global explanation from CRAFT for two classes of ILSVRC2012 [9] for a pre-trained ResNet50V2 [22]. For each class, it shows crop images that activate the concept the most of the top 3 most important concepts. It also shows feature visualizations of the associated concepts.

5.2 Methods based on labeled concept dataset

This section encompasses techniques for utilizing the explainability of methods based on labeled concept dataset. This includes the conception of the concept database in subsection 5.2.1, the evaluation strategy in subsection 5.2.2 and the interpretation of abnormal explanations in subsection 5.2.3. Finally, the subsection 5.2.4 presents its limitations.

5.2.1 Building the concept database

When using concept-based methods, it is essential to carefully build the concept database and make the most appropriate choices of concepts. It is crucial to construct the dataset in such a way that it avoids concepts that are only meaningful to humans and not the machine.

- **Types of samples to construct a concept**: To construct a CAV, the method needs to have two distinct datasets: one representing the concept and one representing random examples.
- **Types of concepts**: The method should comprise two types of concepts:
  - *Expert concepts*, which are meaningful to humans and should be the basis for the machine's decision-making process. To ensure exhaustiveness of a concept with respect to the use-case, it is necessary to verify its coverage of the Operational Design Domain (ODD) while constructing the dataset. For example, if a concept *"tree"* is defined based on a *"tree"* dataset that exclusively contains trees with leaves, the concept may not be activated when presented with an image of a leafless tree. Therefore, if the ODD includes trees with and without leaves, it is crucial to ensure that the concept of *"tree"* is comprehensive and encompasses all possible variations of the object of interest. Expert concepts can also be defined by experts to find "coherent" bias in the model (e.g. use the concept of *spider's web* to detect spiders in images).
  - *Diversified concepts*, which may not necessarily be related to elements that a human would consider in decision-making (for instance the concept of *cat* to detect spiders in images).

They should have no impact on the model's decision and should be present to check for biases. The diversified concepts should cover all possible concepts that may be present in the inputs. However, identifying a complete set of "biased" concepts is challenging. Therefore, it is necessary to include as many as possible even if there is no certainty that the set of corresponding biases will be complete.

- **Size of the concept database**[ADVICE]: Depending on the specific use-case, it is generally advisable to employ a range of four to ten expert-defined concepts per class, while ensuring a sufficient representation of approximately one hundred distinct concepts. Additionally, a minimum of ten samples per concept (ideally closer to thirty) is recommended for adequate model training. Therefore, a dataset consisting of ten concepts would require a total of one hundred samples to be considered suitable for training a model.

- **Choosing concepts**: Experts are responsible for selecting the concepts and, together with data scientists, creating the dataset for the concepts, including labeling (positive or negative) and the source of the concepts.

- **Type of samples in the concept database**[ADVICE]: The source of the concepts can include textures in an image, patterns, image crops, and so on. However, it appears that not properly representing the type of concept from the training dataset in the concept database is not a recommended practice (e.g. using a sketch to represent a concept when the dataset only contains real images), although there are no concrete experiments in the literature.

- **Using the training dataset to construct the concept database**: Currently, there is no consensus in the literature on how to construct the dataset in relation to the training dataset (e.g. using stripes to recognize a zebra). We recommend to first use the reference dataset as a basis for constructing both the diversified and expert concepts. For instance, this can be accomplished by cropping background images, selecting words by gender in natural language processing (NLP), or utilizing random sampling techniques.

- **Concept validation**: It is possible to introduce a bias in the concept during its creation. For instance, if a wheel classifier is trained solely on images of wheels with hubcaps, it can introduce the bias of detecting a wheel only when it is accompanied by a hubcap in the image. Therefore, upon completion of the construction of the concept database, it is advised to validate the defined concepts, particularly through an analysis of the CAV concept classifier (see 5.2.2). A study on the limitations of the concept dataset can be found in [43].

In Short...

- **Use expert concepts** (between 4 to 10 by class) defined by experts which are directly linked to the use-case and extracted from the training dataset.
- **Use diversified concepts** (around 100 concepts in total) covering concepts that may be present in the dataset but without influence on the model's decision.
- **Use the training dataset** at first to create the concept database.
- Use the same type of data for the concept database than the training dataset.

5.2.2 Evaluation Strategy

The main takeaway from using concepts methods like TCAV is the ability to validate whether a human-defined concept is used by the model or not. Therefore, an important step in the evaluation of the method is to validate the accuracy of the defined concepts.

5.2.2.1 Validation of the CAV concept classifier [ADVICE]

The validation of the CAV concept classifier aims to determine if CAV accurately predicts the desired concepts. There are two distinct validation methods, with the first being faster, albeit less rigorous to employ than the latter:

Warning...

The choice of the dataset utilized for the concepts is quite important and can have a considerable impact on the quality of the explanations [43].

- uses a dataset divided into training and testing subsets to simply validate the CAV classifier. In this case, train the CAV classifier with the training subset and compute the TCAV score with the testing subset. The score should be higher for the tested concept compared to the other concepts.
- uses a dataset divided into training, validation, and testing subsets to validate the CAV classifier with classical method. The requirement is to have a concept classifier with high accuracy.

If the classifier does not meet the TCAV score/accuracy requirement, it is important to first confirm that the initial model has good accuracy, and then consider redefining the concept database. Note that the CAV method can also be validated by a feature-attribution method on the concept database, except for textural concept.

In Short...

To validate TCAV-CAV and address its limitations, it is essential to consider the following steps when evaluating models using concepts methods:

- **Validate the CAV concept classifier**, either by TCAV score or accuracy on concept's samples.
- **Use a diverse set of concepts**, including both expert and diversified concepts, to ensure a comprehensive evaluation.
- **Continuously update and refine the concept base** to include any newly discovered concepts that may be relevant to the model's decision-making process.
- **Consider combining concepts methods with other explainability techniques** to gain a more in-depth understanding of the model's behavior and to identify potential biases.

5.2.3 Interpretation of abnormal explanations

In case of sufficient model accuracy and a concept database validated (see Section 5.2.2), abnormal explanations can provide valuable insights into potential problems:

- **Detect bias**: unwanted concept activations can show bias in the model (e.g. spider's web and spider).
- **Data quality**: a desired concept that's inactive could indicate an issue with the underlying dataset.

5.2.4 Limitations for TCAV-CAV

- **No guarantee of finding bias**: even if the concepts are validated, there is no guarantee that all the concepts useful for the model have been discovered or that there isn't a correlated, yet unidentified concept that influences the importance of the described concept. An example of this is the correlation between a wheel classified thanks to the presence of a hubcap in an image.

5.3 Methods based on automatic concept extraction

As previously said in the section 5.1, this section refers to the Craft method as it is the most promising method in automatic concept extraction. It covers the requirements to use Craft in subsection 5.3.1, the evaluation strategy in subsection 5.3.2 and the interpretation of abnormal explanations in subsection 5.3.3. Finally, the subsection 5.3.4 presents its limitations.

5.3.1 Craft requirements

Craft needs to be used on a sufficient number of images per class, typically having 100 elements in each class is a minimum.

Remark: On another hand, having too many samples can be resource-intensive, making it necessary to select a representative subset from the dataset. A random selection of a certain number of elements, such as 1000 images per class, should be sufficient to create this subset.

5.3.2 Evaluation strategy

The evaluation strategy for Craft-like methods involves several aspects: ensuring that the extracted concepts are intelligible, validated, and sufficient for explaining the classification.

- **Concept Size**: Choose a concept size (rank of the matrix) which corresponds to the number of concepts extracted to explain a class and that enables effective classification (using the Sobol metric of Craft). The selected concepts should be sufficient to explain a class, this requires examining the reconstruction error of the outputs for each class.

- **Intelligible Concepts**: To assess an extracted concept individually, a group of people should assess its understandability. Concepts may be too general or too specific, so it is crucial to validate the method by a human experience, following the experiment outlined in the Craft paper [16]. The concepts are evaluated based on their effectiveness in helping participants improve their ability to predict model decisions for unseen images. Participants undergo three training sessions containing five samples with associated explanations, followed by tests where they predict model decisions on new samples without explanations.

- **Feature Visualization**: To assess an extracted concept individually, Craft provides users with a set of examples linked to this concept for better comprehension. Nonetheless, humans are prone to confirmation bias, which can affect their interpretation. Therefore, feature visualization serves as a supplement to explanations by providing an impartial "summary" of the concept for the model, allowing for verification that the concept corresponds to the one extracted by humans from the given examples (see Figure 9).

- **Validating Attribution Methods**: The attribution methods used by Craft can be validated using other attribution methods, as discussed in the section 3. It is essential to note that the attribution method is calculated at a different level of the neural network than the classic attribution methods,

so the choice of the attribution method (and its hyperparameters) may differ when used for Craft or for explaining the classification itself. For instance, integrated gradients are better than Sobol when working on the latent space.

To validate Craft:

- Use the feature visualization and the samples extracted from a concept to better describe it.
- Ensure the extracted concepts of Craft are intelligible for experts.
- Validate the attribution method used by Craft.

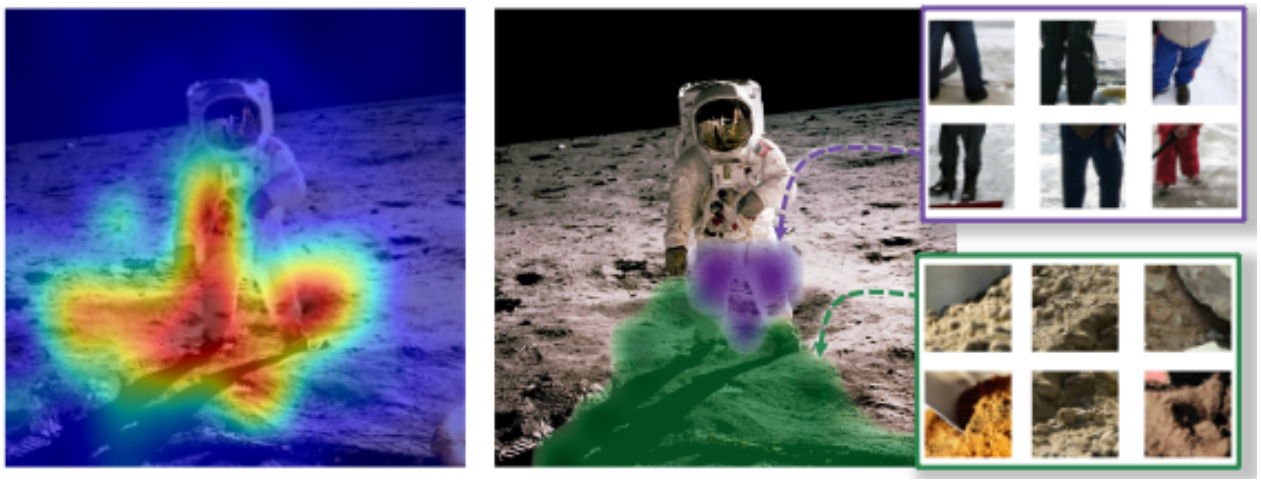


Figure 10 – **The "Man on the Moon" incorrectly classified as a "shovel" by an ImageNet-trained ResNet50.** *Heatmap generated by a classic attribution method (left) vs. concept attribution maps generated with the proposed CRAFT approach (right) which highlights the two most influential concepts that drove the ResNet50's decision along with their corresponding locations. CRAFT suggests that the neural net arrived at its decision because it identified the concept of "dirt" • commonly found in members of the image class "shovel" and the concept of "ski pants" • typically worn by people clearing snow from their driveway with a shovel instead the correct concept of astronaut's pants (which was probably never seen during training).*

5.3.3 Interpretation of abnormal explanations

Craft allows to have a better explanation than attribution methods, as illustrated in Figure 10. In some instances, it is possible to identify the biases within the model and understand the root causes of these issues, as well as the primary logic followed by the model. By doing so, users can gain a more precise understanding of the origin of the bias. For example, they might determine that specific concepts associated with an object (e.g., a spider web with a spider, or a plant and flower with a bee) are over-represented for a particular class, causing the model to rely on these concepts when making decisions. Recognizing these abnormal explanations and biases can help users make necessary adjustments to improve the model's performance and fairness.

5.3.4 Limitations

Craft has certain limitations that should be taken into consideration:

- **Applicability challenges**: Craft has been only applied for image classification and NLP sentiment analysis tasks.
- **Evaluation challenges**: Evaluation of Craft's performance is constrained by a limited number of benchmark datasets.
- **Number of samples per class**: As previously said in section 5.3.1, the number of samples needed per class varies depending on the specified problem.

Glossary

post-hoc Method applied to the already trained model. (Opposed to by-design and intrinsic methods and either global or local explanation.

black-box A method that only inputs and outputs of a model and treat the model as a prediction box. (Opposed to white-box methods and a *post-hoc* method by definition).

by-design A method to build inherently explainable models. It should be taken into account during model construction and often affect the structure of the given model. Note that some models called transparent models are inherently interpretable. (Opposed to intrinsic and *post-hoc methods)*.

data-centric A method that explains the dataset but gives no information on the model. (Opposed to global and local methods).

global A method that explains the whole model behavior and decision process. (Opposed to data-centric and local methods).

intrinsic A method that needs to be taken into account during model training without affecting the final state. (Opposed to by-design and *post-hoc methods)*.

local A method that explains a given decision, it explains the decision process behind one inference at a time. (Opposed to data-centric and global methods).

model-agnostic A method that can be applied to any model or a large group of models. (Opposed to model-specific methods).

model-specific A method that can only be applied to one model or a smaller group of models. (Opposed to model-agnostic methods).

white-box A method that needs access to either the gradients, the weights of the model, or both. (Opposed to black-box methods and usually a *post-hoc* method).

References

[1] Julius Adebayo, Justin Gilmer, Michael Muelly, Ian Goodfellow, Moritz Hardt, and Been Kim. Sanity checks for saliency maps. In *Advances in Neural Information Processing Systems (NIPS)*, 2018.

[2] Marco Ancona, Enea Ceolini, Cengiz Öztireli, and Markus Gross. Towards better understanding of gradient-based attribution methods for deep neural networks. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2018.

[3] Umang Bhatt, Adrian Weller, and José M. F. Moura. Evaluating and aggregating feature-based model explanations. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, 2020.

[4] Moritz Böhle, Mario Fritz, and Bernt Schiele. B-cos networks: alignment is all we need for interpretability. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10329–10338, 2022.

[5] Joy Buolamwini and Timnit Gebru. Gender shades: Intersectional accuracy disparities in commercial gender classification. In *Conference on fairness, accountability and transparency*. PMLR, 2018.

[6] Diogo V. Carvalho, Eduardo M. Pereira, and Jaime S. Cardoso. Machine learning interpretability: A survey on methods and metrics. *Electronics*, 2019.

[7] Aditya Chattopadhay, Anirban Sarkar, Prantik Howlader, and Vineeth N Balasubramanian. Grad-cam++: Generalized gradient-based visual explanations for deep convolutional networks. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, 2018.

[8] Julien Colin, Thomas Fel, Rémi Cadène, and Thomas Serre. What i cannot predict, i do not understand: A human-centered evaluation framework for explainability methods. *Advances in Neural Information Processing Systems (NeurIPS)*, 2021.

[9] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

[10] Finale Doshi-Velez and Been Kim. Towards a rigorous science of interpretable machine learning. *ArXiv e-print*, 2017.

[11] Thomas Fel, Thibaut Boissin, Victor Boutin, Agustin Picard, Paul Novello, Julien Colin, Drew Linsley, Tom Rousseau, Rémi Cadène, Laurent Gardes, et al. Unlocking feature visualization for deeper networks with magnitude constrained optimization. *arXiv preprint arXiv:2306.06805*, 2023.

[12] Thomas Fel, Remi Cadene, Mathieu Chalvidal, Matthieu Cord, David Vigouroux, and Thomas Serre. Look at the variance! efficient black-box explanations with sobol-based sensitivity analysis. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021.

[13] Thomas Fel, Melanie Ducoffe, David Vigouroux, Remi Cadene, Mikael Capelle, Claire Nicodeme, and Thomas Serre. Don't lie to me! robust and efficient explainability with verified perturbation analysis. *Workshop on Formal Verification of Machine Learning, Proceedings of the International Conference on Machine Learning (ICML)*, 2022.

[14] Thomas Fel, Melanie Ducoffe, David Vigouroux, Remi Cadene, Mikael Capelle, Claire Nicodeme, and Thomas Serre. Don't lie to me! robust and efficient explainability with verified perturbation analysis. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

[15] Thomas Fel, Lucas Hervier, David Vigouroux, Antonin Poche, Justin Plakoo, Remi Cadene, Mathieu Chalvidal, Julien Colin, Thibaut Boissin, Louis Bethune, Agustin Picard, Claire Nicodeme, Laurent Gardes, Gregory Flandin, and Thomas Serre. Xplique: A deep learning explainability toolbox. *Workshop on Explainable Artificial Intelligence for Computer Vision (CVPR)*, 2022.

[16] Thomas Fel, Agustin Picard, Louis Bethune, Thibaut Boissin, David Vigouroux, Julien Colin, Rémi Cadène, and Thomas Serre. Craft: Concept recursive activation factorization for explainability. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

[17] Thomas Fel and David Vigouroux. Representativity and consistency measures for deep neural network explanations. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, 2022.

[18] Ruth C. Fong and Andrea Vedaldi. Interpretable explanations of black boxes by meaningful perturbation. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2017.

[19] Amirata Ghorbani, Abubakar Abid, and James Zou. Interpretation of neural networks is fragile. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, 2017.

[20] Amirata Ghorbani, James Wexler, James Y Zou, and Been Kim. Towards automatic concept-based explanations. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

[21] Leilani H. Gilpin, David Bau, Ben Z Yuan, Ayesha Bajwa, Michael Specter, and Lalana Kagal. Explaining explanations: An overview of interpretability of machine learning. In *Proceedings of the IEEE International Conference on data science and advanced analytics (DSAA)*, 2018.

[22] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Identity mappings in deep residual networks. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part IV 14*, pages 630–645. Springer, 2016.

[23] Sara Hooker, Dumitru Erhan, Pieter-Jan Kindermans, and Been Kim. A benchmark for interpretability methods in deep neural networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

[24] Cheng-Yu Hsieh, Chih-Kuan Yeh, Xuanqing Liu, Pradeep Ravikumar, Seungyeon Kim, Sanjiv Kumar, and Cho-Jui Hsieh. Evaluations and methods for explanation through robustness analysis. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2021.

[25] Alon Jacovi, Ana Marasović, Tim Miller, and Yoav Goldberg. Formalizing trust in artificial intelligence: Prerequisites, causes and goals of human trust in ai. In *Proceedings of the 2021 ACM conference on fairness, accountability, and transparency*, pages 624–635, 2021.

[26] Fanny Jourdan, Agustin Picard, Thomas Fel, Laurent Risser, Jean Michel Loubes, and Nicholas Asher. Cockatiel: Continuous concept ranked attribution with interpretable elements for explaining neural net classifiers on nlp tasks. *arXiv preprint arXiv:2305.06754*, 2023.

[27] Margot E Kaminski. The right to explanation, explained. In *Research Handbook on Information Law and Governance*. Edward Elgar Publishing, 2021.

[28] Been Kim, Rajiv Khanna, and Oluwasanmi O Koyejo. Examples are not enough, learn to criticize! criticism for interpretability. *Advances in neural information processing systems*, 29, 2016.

[29] Been Kim, Martin Wattenberg, Justin Gilmer, Carrie Cai, James Wexler, Fernanda Viegas, et al. Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (tcav). In *International conference on machine learning*. Proceedings of the International Conference on Machine Learning (ICML), 2018.

[30] Svetlana Kiritchenko and Saif M Mohammad. Examining gender and race bias in two hundred sentiment analysis systems. *Proceedings of the 7th Joint Conference on Lexical and Computational Semantics (*SEM)*, 2018.

[31] Pang Wei Koh, Thao Nguyen, Yew Siang Tang, Stephen Mussmann, Emma Pierson, Been Kim, and Percy Liang. Concept bottleneck models. In *International Conference on Machine Learning*, pages 5338–5348. PMLR, 2020.

[32] Isaac Lage, Emily Chen, Jeffrey He, Menaka Narayanan, Been Kim, Sam Gershman, and Finale Doshi-Velez. An evaluation of the human-interpretability of explanation. In *Workshop on Correcting and Critiquing Trends in Machine Learning, Advances in Neural Information Processing Systems (NIPS)*, 2019.

[33] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *Nature*, 2015.

[34] Zhong Qiu Lin, Mohammad Javad Shafiee, Stanislav Bochkarev, Michael St Jules, Xiao Yu Wang, and Alexander Wong. Do explanations reflect decisions? a machine-centric strategy to quantify the performance of explainability algorithms. In *Advances in Neural Information Processing Systems (NIPS)*, 2019.

[35] Scott Lundberg and Su-In Lee. A unified approach to interpreting model predictions. In *Advances in Neural Information Processing Systems (NIPS)*, 2017.

[36] Tim Miller. Explanation in artificial intelligence: Insights from the social sciences. *Artificial Intelligence*, 267:1–38, February 2019.

[37] Anh Nguyen, Jason Yosinski, and Jeff Clune. Multifaceted feature visualization: Uncovering the different types of features learned by each neuron in deep neural networks. *Visualization for Deep Learning workshop, Proceedings of the International Conference on Machine Learning (ICML)*, 2016.

[38] Anh Nguyen, Jason Yosinski, and Jeff Clune. Understanding neural networks via feature visualization: A survey. *arXiv preprint arXiv:1904.08939*, 2019.

[39] Paul Novello, Thomas Fel, and David Vigouroux. Making sense of dependence: Efficient black-box explanations using dependence measure. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

[40] Chris Olah, Alexander Mordvintsev, and Ludwig Schubert. Feature visualization. *Distill*, 2017.

[41] Vitali Petsiuk, Abir Das, and Kate Saenko. Rise: Randomized input sampling for explanation of black-box models. In *Proceedings of the British Machine Vision Conference (BMVC)*, 2018.

[42] Inioluwa Deborah Raji and Joy Buolamwini. Actionable auditing: Investigating the impact of publicly naming biased performance results of commercial ai products. In *Proceedings of the 2019 AAAI/ACM Conference on AI, Ethics, and Society*, pages 429–435, 2019.

[43] Vikram V Ramaswamy, Sunnie SY Kim, Ruth Fong, and Olga Russakovsky. Overlooked factors in concept-based explanations: Dataset choice, concept learnability, and human capability. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10932–10941, 2023.

[44] Sylvestre-Alvise Rebuffi, Ruth Fong, Xu Ji, and Andrea Vedaldi. There and back again: Revisiting backpropagation saliency methods. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[45] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. "why should i trust you?": Explaining the predictions of any classifier. In *Knowledge Discovery and Data Mining (KDD)*, 2016.

[46] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. Anchors: High-precision model-agnostic explanations. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, 2018.

[47] Laura Rieger and Lars Kai Hansen. Irof: a low resource evaluation metric for explanation methods. In *Workshop, Proceedings of the International Conference on Learning Representations (ICLR)*, 2020.

[48] Andrew Slavin Ross, Michael C. Hughes, and Finale Doshi-Velez. Right for the right reasons: Training differentiable models by constraining their explanations. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI-17*, pages 2662–2670, 2017.

[49] Cynthia Rudin. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature machine intelligence*, 2019.

[50] Zohaib Salahuddin, Henry C Woodruff, Avishek Chatterjee, and Philippe Lambin. Transparency of deep neural networks for medical image analysis: A review of interpretability methods. *Computers in biology and medicine*, 2022.

[51] Ramprasaath R. Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2017.

[52] Junghoon Seo, Jeongyeol Choe, Jamyoung Koo, Seunghyeon Jeon, Beomsu Kim, and Taegyun Jeon. Noise-adding methods of saliency map as series of higher order partial derivative. In *Workshop on Human Interpretability in Machine Learning, Proceedings of the International Conference on Machine Learning (ICML)*, 2018.

[53] Thomas Serre. Deep learning: The good, the bad, and the ugly. *Annual review of vision science*, 2019.

[54] Avanti Shrikumar, Peyton Greenside, and Anshul Kundaje. Learning important features through propagating activation differences. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2017.

[55] Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. In *Workshop, Proceedings of the International Conference on Learning Representations (ICLR)*, 2013.

[56] Leon Sixt, Maximilian Granz, and Tim Landgraf. When explanations lie: Why many modified bp attributions fail. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2020.

[57] Daniel Smilkov, Nikhil Thorat, Been Kim, Fernanda Viégas, and Martin Wattenberg. Smoothgrad: removing noise by adding noise. In *Workshop on Visualization for Deep Learning, Proceedings of the International Conference on Machine Learning (ICML)*, 2017.

[58] Jost Tobias Springenberg, Alexey Dosovitskiy, Thomas Brox, and Martin Riedmiller. Striving for simplicity: The all convolutional net. In *Workshop Proceedings of the International Conference on Learning Representations (ICLR)*, 2014.

[59] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2017.

[60] Richard Tomsett, Dan Harborne, Supriyo Chakraborty, Prudhvi Gurram, and Alun Preece. Sanity checks for saliency metrics. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, 2019.

[61] Guangrun Wang and Philip HS Torr. Traditional classification neural networks are good generators: They are competitive with ddpms and gans. *arXiv preprint arXiv:2211.14794*, 2022.

[62] Chih-Kuan Yeh, Cheng-Yu Hsieh, Arun Sai Suggala, David I. Inouye, and Pradeep Ravikumar. On the (in)fidelity and sensitivity for explanations. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

[63] Jason Yosinski, Jeff Clune, Anh Nguyen, Thomas Fuchs, and Hod Lipson. Understanding neural networks through deep visualization. *Proceedings of the International Conference on Machine Learning (ICML)*, 2015.

[64] M. D. Zeiler, G. W. Taylor, and R. Fergus. Adaptive deconvolutional networks for mid and high level feature learning. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2011.

[65] Matthew D Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. In *Proceedings of the IEEE European Conference on Computer Vision (ECCV)*, 2014.

[66] Matthew D Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. In *Proceedings of the IEEE European Conference on Computer Vision (ECCV)*, 2014.