A Study of Explainable AI

The Concept of Zero Information

Thodoris Lymperopoulos

National and Technical University of Athens
Electrical and Computer Engineering
Master of Data Science and Machine Learning

October 26, 2023

MSc Thesis

A study of Explainable AI

Thodoris Lymperopoulos

AM: 03400140

Supervisor: Prof. Stefanos Kollias

Thesis Examination Committee

Dr. Yannis Avrithis
Principal Investigator, Institute of Advanced Research on Artificial Intelligence (IARAI)

Prof. Athanasios Voulodimos
Assistant Professor, School of Electrical and Computer Engineering (NTUA)

Prof. Stefanos Kollias
Professor, School of Electrical and Computer Engineering (NTUA)

October 26, 2023

Copyright

This thesis is an original work, and I have taken care to respect copyright and intellectual property rights throughout the research and development process. I declare that the methods and techniques developed in this research are a product of my original work, and I affirm that I do not infringe upon any copyrights or intellectual property rights of others.

I encourage and welcome anyone with an interest in these methods and techniques to explore and utilize them freely. If you have any questions or wish to discuss these methods further, please feel free to contact me. Discussion and further exploration of these concepts are openly encouraged.

Περίληψη

Η ταχεία ανάπτυξη σύνθετων μοντέλων βαθιάς μάθησης οδήγησε σε μεγάλα και σημαντικά επιτεύγματα, σε μια πληθώρα εφαρμογών, υποδεικνύοντας την ικανότητά τους στην αναγνώριση σημαντικών προτύπων στα δεδομένα. Ωστόσο, αυτή η επιτυχία συνοδεύτηκε από τη μείωση της ερμηνευσιμότητας των μοντέλων. Αυτές οι περίπλοκες αρχιτεκτονικές θεωρούνται ¨μαύρα κουτιά¨, διότι πραγματοποιούν πολλούς χαμηλού επιπέδου, μη γραμμικούς υπολογισμούς. Παρόμοια με τον ανθρώπινο εγκέφαλο, κατανοούμε τον εκάστοτε αλληλεπίδραση των νευρώνων, αλλά δυσκολευόμαστε να κατανοήσουμε πώς το μοντέλο συνδυάζει πληροφορίες για τη δημιουργία υψηλότερων έννοιων. Αυτό το κενό γνώσης οδήγησε στον τομέα της Επεξηγηματικής Τεχνητής Νοημοσύνης (ΕΤΝ).

Η Επεξηγηματική Τεχνητή Νοημοσύνη (ΕΤΝ) βρίσκεται στο στάδιο της ανάπτυξης, και μια σειρά ερωτημάτων για την κατανόηση της λειτουργίας του μοντέλου δεν έχει ακόμα διατυπωθεί. Ωστόσο, η έρευνα ξεκίνησε με ένα θεμελιώδες ερώτημα: ¨ποιοι παράγοντες επηρεάζουν την απόφαση ενός μοντέλου για ένα δεδομένο είσοδο¨, οδηγώντας στην ανάπτυξη των ¨μεθόδων Ανάθεσης¨. Αυτές οι μέθοδοι στοχέυουν στην απόδοση της απόφασης ενός μοντέλου πίσω στα χαρακτηριστικά εισόδου, με την ανάθεση σημασίας σε κάθε χαρακτηριστικό. Για να επιτευχθεί αυτό, χρησιμοποιούν διάφορα μαθηματικά εργαλεία, στενά σχετιζόμενα με την έννοια της σημασίας. Η εξάπλωση τέτοιων μεθόδων απαίτησε τη δημιουργία ¨μετρικών Αξιολόγησης' για να μετρήσουν την αποτελεσματικότητά τους. Ωστόσο, τέτοιες μετρικές εμφάνισαν περιορισμούς, οδηγώντας μερικούς ερευνητές στην ανάπτυξη συνόλων ¨Αξιωμάτων και Κριτηρίων' που θεωρούνταν απαραίτητα για την αξιόπιστη ανάθεση. Αν και αντιμετωπίζουν παρόμοιους περιορισμούς, παρέχουν μια πιο εννοιολογικά ορθή προσέγγιση για αξιόπιστες αναθέσεις.

Σε απάντηση σε αυτές τις προκλήσεις, αυτή η διατριβή εξερευνά το έννοια της ¨Μηδενικής Πληροφορίας¨, καθώς μπορεί να προσφέρει σημαντική βοήθεια σε αυτές τις ερωτήσεις. Αυτή η έννοια αποσκοπεί στο να κρύψει όλες τις πληροφορίες που περιέχονται σε τμήματα μιας εικόνας για ένα συγκεκριμένο μοντέλο, αποκαλύπτοντας τη συμβολή τους στη απόφαση του μοντέλου. Αυτή η διατριβή αναπτύσσει μια νέα προσέγγιση στο πρόβλημα, σχεδιάζοντας κριτήρια που σχετίζονται με την απόκρυψη της πληροφορίας. Πρώτον, σχεδιάζουμε ένα αλγόριθμο για την απόκρυψη της πληροφορίας από ολόκληρη την εικόνα. Μεταφράζοντας τα κριτήρια σε συναρτήσεις κόστους, ο αλγόριθμος εντοπίζει τα πιο επιδραστικά σημεία που οδηγούν στη μείωση της βεβαιότητας του μοντέλου και τα χρησιμοποιεί για να ορίσει μια μέθοδο Ανάθεσης. Στη συνέχεια, για την απόκρυψη πληροφορίας από μέρη της εικόνας, τα κριτήρια πρέπει να επεκταθούν για να περιέχουν τις αλληλεπιδράσεις χαρακτηριστικών. Ένας αλγόριθμος βελτιστοποίησης εκπαιδεύεται για να πληροί αυτά τα κριτήρια, ενώ χρησιμοποιεί γεννητικά μοντέλα για την ανακατασκευή των απόκρυμμένων τμημάτων με ¨φυσική' συμπλήρωση. Η μέθοδος δοκιμάζεται με βάση πολλές μετρικές, εμφανίζοντας ισχυρή απόδοση έναντι άλλων τεχνικών. Στη συνέχεια, μπορεί να χρησιμοποιηθεί από διάφορες μεθόδους Ανάθεσης και μετρικές Αξιολόγησης που βασίζονται στην απόκρυψη πληροφορίας, προσφέροντας καλύτερα αποτελέσματα.

Συνολικά, αυτή η διατριβή προσπαθεί να αναπτύξει μια μεθοδολογία για αξιόπιστες τεχνικές συμπλήρωσης για τη Μηδενική Πληροφορία. Δεν παρέχει οριστικές απαντήσεις στο πρόβλημα, δεδομένη την πολυπλοκότητά του. Αντίθετα, ανοίγει το δρόμο για την ανάπτυξη καλύτερων κριτηρίων ως μια μελλοντική κατεύθυνση, προσφέροντας απαντήσεις σε θεμελιώδεις ερωτήσεις της ΤΝ.

Λέξεις-Κλειδιά

Επεξηγηματική Τεχνητή Νοημοσύνη, μέθοδοι Απόδοσης, μέθοδοι Απόκρυψης, μετρικές Αξιολόγησης, μαθηματικά Αξιώματα και Κριτήρια, Τεχνικές συμπλήρωσης, Μηδενική Πληροφορία Σημείου, Μηδενική Πληροφορία Κομματιών

Abstract

The rapid evolution of complex deep learning models has yielded remarkable achievements in various applications, underlining their proficiency in recognizing vital data patterns. However, this success came at the cost of transparency and interpretability. These intricate architectures are considered as black boxes, since they perform numerous low-level, non-linear calculations. Much like the human brain, we understand individual neuron interactions but we struggle to comprehend how the model combines information to form higher concepts. This gap of knowledge gave rise to Explainable AI (XAI).

This particular field of AI is in its developing stage and a sequence of questions to comprehend the model's functioning is yet to be formed. However, the exploration began with a fundamental question: *"what factors influence a model's decision for a given input"*, leading to the development of **attribution methods**. These methods are dedicated to attributing a model's decision to specific input features, by assigning importance scores to each feature. To achieve this, they leverage different mathematical tools, closely related to the notion of importance. The proliferation of such methods necessitated the creation of **evaluation metrics**, to gauge their effectiveness. Yet, such metrics exhibited limitations, leading some researchers to the development of sets of *axioms and criteria* that were considered essential for robust methods to satisfy. While facing similar limitations, they provided a more conceptually sound approach for robust attributions.

In response to these challenges, this thesis explores the concept of **Zero Information** as it may offer valuable insights. This concept aims to conceal all information contained in parts of an image for a particular model, revealing their contribution to the model's score. This thesis develops a new approach to the problem, by designing criteria related to information concealment. First, we design an algorithm for hiding information from the whole image. By translating criteria into loss functions, the algorithm finds the most influential points that lead to the drop of the model's confidence and uses them to define an attribution method. Then, for hiding information from parts of the image, criteria need to be extended to capture feature interactions. An optimization algorithm is trained to meet these criteria, while leveraging generative models for reconstructing the hidden parts with *natural* fill. The method is tested across multiple metrics, showing strong performance against other techniques. It can then be exploited by different attribution methods and evaluation metrics based on information concealment, yielding better results.

Overall, this thesis attempts to develop a methodology towards robust filling techniques for Zero Information. It does not give definite answers to the problem, since it is constrained by its complexity. Instead, it paves the way for better criteria to be developed in the future, to answer a fundamental question of XAI and unlock the power of different methods and techniques.

Key-words

Explainable Artificial Intelligence, Attribution methods, Occlusion methods, Evaluation metrics, Axioms and Criteria, Filling techniques, Zero Information Points, Zero Information Parts

Acknowledgements

This research thesis would not be possible without the brilliant and insightful guidance of my professor, dr. Yanis Avrithis. I am deeply grateful for the many hours we spent discussing various concepts in XAI and debating the design of new techniques. I acknowledge that his time was limited, and I sincerely appreciate the dedication he devoted to this thesis. I wish him the best of luck for the future.

I would also like to express my gratitude to my supervisor, S. Kollias. I appreciate his kindness and trust.

Furthermore, I would like to thank my family and friends that supported and believed in me throughout this challenging journey. This work might be the one I'm mostly proud of, and I hope it finds its place in the field of XAI, inspiring others to further advance these ideas.

Contents

Part I

INTRO

Chapter 1

Introduction

> *In the realm of artificial minds, Explainable AI (XAI) emerges as the eloquent bard, weaving narratives of transparency and insight. It bestows upon the computational enigma the gift of lucid verse, unveiling the intricate dance of reason in its decisions, a harmonious symphony for human trust and understanding to thrive.* – ChatGPT 3.5, when asked to define XAI in a poetic manner.

About this thesis

I welcome the reader to my research thesis on the topic of Explainable Artificial Intelligence (XAI). This is an emerging field within the realm of AI that aims to shed light on the decision-making processes of machine learning models. Unlike more established disciplines such as Linear Algebra or Functional Analysis, XAI is still in its formative stages, lacking definitive mathematical foundations. As a result, it presents a set of challenges without universally accepted solutions. Instead, researchers have charted various trajectories, each with its unique advantages and limitations.

The primary objective of this thesis is to provide a conceptual understanding and a comprehensive overview of the challenges in XAI. Throughout this journey, we examine several Attribution methods from a mathematical perspective, and critically assess different metrics and criteria, developed to prove the effectiveness of the aforementioned techniques. Ultimately, we focus our attention to a fundamental issue in XAI, that is to find a manner to properly conceal information from a model-image pair. For this problem, that has yet to be addressed properly, we design algorithms leading to new Attribution methods and techniques that advance the field of XAI further. This will be a lengthy and intricate journey through many concepts and ideas.

1.1 Machine Learning theory and application

Recent advances in Machine Learning techniques, especially the development of deep learning models, have triggered a remarkable expansion of the field [58, 8]. These cutting-edge models have demonstrated their prowess across diverse domains, ranging from medicine [28], disease diagnosis [74] and drug discovery [46], to education [17], agriculture [1], marketing [46], and finance [68]. As a result, deep learning models have found practical applications in automating a wide array of labor-intensive tasks. The following paragraphs aim to introduce the concepts of machine learning and deep learning, catering to readers who may be new to this field.

Overview of ML

Machine Learning. Machine Learning (ML) is a subset of artificial intelligence (AI) that focuses on the development of statistical models that enable computer systems to improve their performance on tasks that were considered *hard* for computers to tackle. Recognising objects in images is a task that humans excel without much difficulty, but a computer struggles. ML manages to tackle such tasks, by designing models that are based on experience or data. Instead of relying on explicit programming, ML systems learn patterns and make predictions or decisions based on input data. This process involves training a model on a labeled dataset, where the model learns to recognize patterns and relationships within the data. Once trained, the model can generalize its knowledge to make predictions or classifications on new, unseen data.

Deep Learning. Deep Learning is a subfield of Machine Learning that focuses on neural networks with many layers, known as deep neural networks (DNNs). These networks are inspired by the structure of the human brain and are designed to automatically learn and extract hierarchical features from data. Deep Learning has gained significant attention and success in recent years due to its remarkable performance in tasks such as image and speech recognition, natural language understanding, and autonomous driving. Deep Learning models, particularly deep neural networks called **Convolutional Neural Networks** (CNNs) for images and **Recurrent Neural Networks** (RNNs) for sequences, have the ability to capture complex patterns and representations, making them highly effective for tasks that involve large datasets and high-dimensional data. In recent years, a very powerful architecture was designed, namely the **Transformers** which excelled in many different applications.

Computer Vision. Computer Vision (CV) is among the most prominent applications of machine learning. In CV tasks, datasets primarily consist of images, and the common objective is to identify objects within these images and determine their positions. The field witnessed significant progress with the development of the first Convolutional Neural Networks (CNNs) [53] designed specifically for CV. Subsequently, more complex architectures emerged, boasting a substantial number of parameters [34, 51, 84, 93]. Notable among these is the ResNeXt architecture**ResNeXt** [59] , a recent achievement that delivers accuracy on par with state-of-the-art CV models. More recently, there has been a shift towards using Transformers**Transformers** [100], originally designed for natural language processing, in the realm of CV, giving rise to Vision Transformers **Vision Transformers** [20] that rival the accuracy of ResNeXt.

1.2 The hidden decision making process of DNNs

The black-box nature of DNNs

However, deep learning models are often regarded as 'black boxes' [104], primarily due to their intricate architectures, which involve numerous low-level calculations within nested, non-linear functions. These functions are grouped into layers, creating latent representation spaces with structures that remain largely unknown. Data is progressively mapped to lower-dimensional spaces within these structures, eventually leading to the model outputs. This inherent complexity somewhat somewhat parallels the operation of the human brain, where higher-level concepts emerge from the interactions of individual

neurons. Yet, much like our understanding of the brain, we still struggle to precisely explain how these low-level computations combine to form higher-level concepts, including thoughts, calculations, or memories. This challenge is also pronounced in the realm of artificial intelligence.

Statistically associating cause to effect

Despite the enigmatic nature of a model's decision-making process, one might argue that we can enhance trust in a model that is highly accurate. This argument has been pivotal for the deployment of such architectures in real-world applications. However, does this argument hold water? At a high level a model learns by statistically associating cause-and-effect relationships. For many applications, the number of possible feature combinations grows exponentially, resulting in multiple potential causes being associated with the same effect. Consequently, a model's strong performance might suggest a deep understanding of such relationships, and no *misalignment* between the model and human cause-and-effect associations. On the other hand, it might also suggest that we have not yet discovered in which examples this misalignment exists.

To illustrate this, consider the Imagenet dataset, one of the most popular in the field of AI. Within a specific class of fishes, the majority of images depict a fisherman holding the fish, as demonstrated in Figure 1. This scenario can perplex the model, causing it to learn that the concept of "tench" includes both the fish and the fisherman that holds it. This misalignment may not necessarily be the model's fault, it can rather be attributed to the design of the dataset itself. Since there is a redundancy in information within an image, different causes might statistically prevail those we would like them to be. Thus, in this particular example, we need to ensure that a model considers the fish as the cause of its decision and neglect the human behind it. Then, we can safely generalise such models to other crucial applications.

Figure 1: Images from the class "tench" of the Imagenet dataset. They were collected from the first 25 images of the class and show a person holding a fish.

The following section delves deeper into the necessity of Explainable AI and its applications across various real-world scenarios.

1.3 Unveiling model transparency: A quest for clarity

Understanding the inner workings of a Deep Neural Network (DNN) may not be inherently intuitive for the human mind which operates at a high level of abstraction. However, the benefits of such understanding are significant. Several compelling reasons underscore the importance of explainability, addressing distinct aspects of model performance and pattern recognition.

Enhancing safety

1. Addressing Safety Concerns. Safety concerns regarding a model's training process are not without foundation. The potential for a model to associate an effect with incorrect causes raises critical questions. Researchers have posited that widely-adopted and successful models may have learned patterns divergent from those desired. This phenomenon has been vividly described as the "Clever Hans" effect [52]. An illustrative example of this was observed with the Fisher Vector Classifier which won the PASCAL VOC competition. Rather than accurately identifying the object of interest, it often relied on unrelated contextual information, such as identifying boats based on the presence of water around them or airplanes by the blue sky in the background. In some exceptional instances, it classified images containing horses as "horse" based solely on the presence of label-text in the bottom left corner, which was a common feature in all images containing horses.

In domains as critical as medicine and self-driving vehicles, such erroneous decisions have the potential to imperil lives. Therefore, the imperative for models to provide explanations for their decisions, enhancing safety and security, cannot be overstated.

Ensure model's robustness

2. Model's Robustness. The revelation of adversarial attacks [94] prompted a reassessment of the robustness of DNN architectures. These attacks involve slight alterations to a model's input, imperceptible to the human eye, yet capable of fundamentally influencing the model's decisions. Real-world experiments [21] further underscored this vulnerability, demonstrating how simple physical alterations, such as affixing stickers to a stop sign, could render an AI system incapable of recognizing the sign. How do these subtle modifications affect the feature associations within the model, leading to divergent judgments? Explainable AI (XAI) potentially holds the key to unraveling this enigma, shedding light on the resilience and accuracy disparities among different models.

Insights into data patterns

3. New insights. Attributing an effect to its underlying causes serves as a valuable tool for interdisciplinarycollaboration and knowledge discovery. Biologists, for instance, seek to unravel the intricate interactions between proteins responsible for the emergence of specific traits. In the realms of biology and pharmacology, understanding which precise reactions triggered the curative effects of a drug is of paramount importance. Economists attempt to discover which factors triggered an economic crisis and historians try to find what drove a revolution at a particular historic moment. Assuming the abundance of data, a model can associate the *correct* causes to its effects. Thus, the understanding of the model's functionality holds the potential to shed light on such associations and lead to new discoveries in science.

With a clear understanding of why XAI is indispensable, we embark on a journey to delve deeper into its evolution. In the following chapter, we briefly discuss the evolution of XAI, including Attribution methods, Evaluation metrics, and other pivotal aspects that have shaped this burgeoning field.

1.4 The evolution of XAI

How to understand the model

A fundamental question we must address is: "What does it truly mean to comprehend a model's inner workings and make it transparent". Is it about understanding how its parameters interact? Perhaps it relates to uncovering what concepts are internally generated. Or does it revolve around comprehending the sequential nature of its computations? Even today, these questions lack definitive answers. In a noteworthy exploration of this subject [56], the author notes:

> *the term interpretability is ill-defined, and thus claims regarding interpretability of various models may exhibit a quasi-scientific character.*

In subsequent sections, the author attempts to define **desiredata for Interpretability**, pointing to **trust, causality, transferability, informativeless and fair & ethical Decision Making**. Later, he defines the notion of **post-hoc** explanations, that are meta-explanations for the model's decision making process. A post-hoc explanation could be a natural language explanation or visualizations of latent activations.

Attribution methods

This pioneering work lays a structured foundation for clarifying a model's transparency. Researchers then turned their attention to **Local Explanations**, constructing maps of importance for each particular input feature of a *given input example*. These maps are called **attribution maps** and the methods that produce them are defined as **Attribution methods**. The primary objective of such methods is to unmask the features that led to a particular effect, without answering *how* they cooperated to achieve this effect, rather only answering *what* caused it. Such an explanation could be important for different ML applications –an example in Computer Vision can be seen in Figure 2. Such a method points to pixels that are possibly related to a particular concept.

This shift in focus is well-founded and justified. When considering the application of DNN models in medical imaging, the primary goal is to ensure that the model makes accurate decisions in relation to a specific cause (the presence of cancer) and its effect (labeling an image as cancerous). One straightforward approach is to identify the crucial parts of the original image that the model deemed significant for its decision. If these regions correspond to cancerous areas, it suggests that the model has captured the *true* correlations between the cause and the effect.

Designing an Attribution method

Up to this point, researchers have pursued various avenues to design Attribution methods that score features according to some notion of importance for the model's decision-making. These methods have harnessed different tools from the toolbox of mathematics, drawing from diverse domains like mathematical analysis and gradient theory, linear algebra, probability theory, game theory, and thermodynamics. However, as the model's decision is deterministic and, thus, rooted in unique causal factors, the question of which Attribution method provides the correct explanation remains unanswered.

(a) Image (b) Attribution map overlay

Figure 2: The attribution map of GradCAM for this particular image of a dog. It clearly points the head of the dog, giving high importance scores to pixels that correspond to the object of interest.

Evaluation metrics

Researchers have endeavored to compare these diverse methods in order to deduce which of those is better. This led to the development of different **Evaluation metrics**, analogous to assessing different model architectures based on prediction accuracy and F-score. Yet, the design of such metrics is not evident, since the *true* attribution is elusive, in contrast to a model's classification score where there the expected labels are known beforehand. One approach might involve devising quantifiable measures that exhibit some degree of correlation with the concept of explainability. For instance, we could gauge the impact on the model's decision when concealing the critical elements highlighted by a reliable attribution map for a specific input, anticipating a change in the model's output due to the absence of the causal factors. Nonetheless, many of these metrics faced criticism regarding their validity. In this work, we not only challenge the effectiveness of some widely-used metrics but also introduce more robust methodologies.

Mathematical Criteria

At present, defining a reliable criterion for measuring the effectiveness of an Attribution method remains a challenge. To address this issue, researchers have introduced a set of mathematical Properties or Criteria that Attribution methods should satisfy. These Criteria are derived from intuitive notions about how robust attributions should operate. For instance, a desirable property is **determinism**, ensuring consistent outcomes for a given model-input pair. The introduction of these Criteria has demonstrated that many existing methods do not meet mathematical correctness standards, paving the way for the development of new Attribution methods that adhere to these essential Properties and Criteria. In this thesis, we will conduct an in-depth examination of these Criteria to assess their effectiveness.

Nevertheless, many of those Axioms and Criteria introduce biases to the model, driven by *our understanding of the notion of explainability*. They reflect general human concepts that do not respect the model's functioning. We will carefully and conceptually examine the Criteria designed so far and point to their strengths and weaknesses. This examination will lead our research towards more robust evaluation techniques.

Our Contribution

The primary contribution of this research thesis lies in the conceptual exploration of various facets of Explainability. Upon a comprehensive examination of diverse topics, our focus narrows to the pivotal concept of **Zero Information** within the realm of Explainability. The main challenge that this subfield of XAI tries to tackle is the discovery of a set of values for a subset of features -might contain the whole set of features as well-, that zero any information the features might contain. Our research detects significant biases that heuristic techniques introduce to the model, and formulates more theoretically sound Criteria for concealing the hidden elements. In particular, we propose desired properties for a robust image concealment and translate them to loss functions, that take into consideration the functioning of the model for the particular input. Then, we take a two-fold approach, based on these Criteria: firstly, we devise an algorithm for concealing images and extract an Attribution method from this process. Secondly, we establish a pipeline that tackles the more intricate challenge of concealing specific portions of images. These methods are evaluated through both visual assessments and the application of newly devised Evaluation metrics. It is essential to note that our methodology does not provide definitive solutions to the highly complex issue at hand. Instead, it sets the stage for a robust approach to tackle the challenges posed by **Zero Information**.

To this day, the question of what a model deems important for its decision-making process remains a compelling enigma, driving the curiosity and dedication of countless researchers in their quest for answers.

1.5 Thesis Structure

The introductory chapter has set the stage for our exploration. It's at this point that mathematics plays a pivotal role, providing a foundation for the subsequent chapters. Chapter 2 delves into the mathematical definition of Attribution methods and offers insight through various examples to facilitate a deeper understanding. In chapter 3 we comprehensively examine the most noteworthy Attribution methods developed to date. Chapter 4, discusses the different Evaluation metrics employed to assess these Attribution methods, while chapter 5 briefly touches on the concept of Criteria in the context of Attribution methods.

The subsequent chapters lay the theoretical groundwork for our research. Chapter 6 Chapter 6 introduces the concept of Zero Information, a pivotal element in our exploration, whereas chapter 7 further scrutinizes the challenge of out-of-distribution data and explores methodologies to address it. The following chapters delve into the aforementioned problems, designing algorithms for whole images (chapter 8) and image parts (9 and 10). Lastly, chapter chapter 11 outlines the experimental setup and the Evaluation metrics employed. The performance of the algorithms to these metrics are found in Chapter 12. The final chapter, Chapter 13, provides a comprehensive summation of our findings, explores its limitations and opens the door to further discussions and future work.

Chapter 2

Attribution

Attribution methods aim to unveil key mathematical properties of the model that are closely related to feature importance. These properties encompass various tools such as gradients, linear combinations, eigenvalues/eigenvectors and many more. The following section will present a mathematical definition of attribution, highlighting its nature as a *meta-explanation*, since inherently includes the *notion of importance*. To offer readers a more intuitive understanding of Attribution methods, a subsequent chapter will demonstrate how attributions can be calculated in simple examples.

2.1 Mathematical definition

This section expresses mathematically the definition of an Attribution method. The definition might be simple enough to formulate, yet, it does not provide any insights about its nature. Thus a simple definition is that the attribution is the measure of importance of each feature to the model's decision. Yet, it consists of a *meta-explanation*, since a mathematical definition of *feature importance* cannot be possibly formed. It is the particular definition of an Attribution method that defines it.

Mathematical formulation

Mathematically stated, such an attribution can be defined as a function of

$$R : \mathbb{F} \times \mathbb{R}^r \to \mathbb{R}^r,\ R(f, x) = R_f(x), \tag{1}$$

where $f : \mathbb{R}^r \to \mathbb{R}^n$ is an instance of all DNN models $\mathbb{F}$, and $r, n \in \mathbb{N}$. r is the number of dimensions of the input space and n is the number of classes. R assigns a value of importance of each feature of an input $\mathbf{x} \in \mathbb{R}^m$ with respect to f, depicting the importance score of that feature to the decision $f(x)$ of the model.

A point we should consider with care is that an attribution map resulting from an Attribution method ultimately produces a map of importance values, one to each feature. This does not mean that features have been decomposed to independent parts and the attributions express values only to those features. These values are all *codependent* on one another.

computer Vision

Throughout this thesis, the application of interest is Computer Vision. Thus, the input space is the image space $\mathcal{X}$ and the model f I have chosen to use is the ResNet architecture [34], which is a popular model for image classification. This selection is based on its relatively lower memory and CPU requirements compared to other models, without compromising performance. Additionally, I assume that after the application of ResNet, the output passes

through a softmax layer, which transforms logits into probabilities. I consider this layer to be an integral part of the model.

2.2 Attribution games

This section aims to address what the attribution of simple models should be, with the purpose of piquing the reader's interest, while also highlight the inherent complexities of this question.

Example 1. A simple linear model.

Consider a simple linear model with two variables x_1, x_2 and their corresponding weights w_1, w_2. The model is defined as:

$$f(x_1, x_2) = 10x_1 + 100x_2. \tag{2}$$

The weight of the second variable is much greater than that of the first. Does that mean that it has a higher contribution to the model's score? A first, naive approach would be to select

$$R(x_i) = w_i, \tag{3}$$

thus

$$R(x_1, x_2) = (10, 100).$$

We notice that this is identical to the ***Gradients*** of the variables.

However, what if, for a specific input $\mathbf{x}$, variable x_2 takes a much smaller value than x_1? Let's consider $\mathbf{x} = (1, \frac{1}{10})$. In this case, each of the terms $w_i x_i$ adds up to 10. The model then adds up the two values, with each having an equal contribution to the result. Hence, we realize that gradients alone are insufficient. Another choice could be to define attributions as:

$$R(x_i) = w_i x_i. \tag{4}$$

Notice that this is identical to considering *Gradients* x *Input*.

The multiplication of the weight by the corresponding input value captures the combined effect of both factors. While other functions may serve a similar purpose, this multiplication method is among the simplest. However, it has a notable drawback—it consistently attributes a zero value when the input is zero. In the context of images, this means that a black object of interest would be entirely neglected.

Example 2. A simple quadratic model with symmetrical variables.

Consider the following function $f : \mathbb{R}^2 \to \mathbb{R}$, where:

$$f(x_1, x_2) = x_1 + x_2 + 2x_1x_2. \tag{5}$$

In this example, the two variables have a completely symmetrical contribution to the output of the function. Thus, when their input values match, **their attribution should be equal**. If $x_1 = x_2 = c$, where $c \in \mathbb{R}$, then $R(x_1) = R(x_2) = r$.

Since this value r is unknown, it is logical to assume that r **is a function of** c. How could we discover the formula of this function? We could argue that the formula could not possibly be linear, since the function itself is quadratic. It is reasonable to assume that the formula should probably be quadratic as well. A criterion we will stumble upon in the upcoming chapters, states that the summation of the contributions should add up to the model's prediction, thus

$$\sum_{i=1}^{N} R(x)_i = f(x). \tag{6}$$

This logical assumption creates a very strong link between r and $c = f(r)$, because $f(r)$ *obeys the behavior of* r.

Note 1. For the method *Gradients* $\times$ *Input* (GI),

$$GI(x_i) = x_i \frac{\partial f}{\partial x_i} = x_i \ (1 + 2x_j) = x_i + 2 \cdot x_i x_j$$

where $i, j \in \{1, 2\}, i \neq j$. Could this be a feasible attribution? According to the aforementioned criterion, the addition $GI(x_1) + GI(x_2) = x_1 + x_2 + 4 \cdot x_1 x_2$, exceeds the model's prediction. The attribution this criterion suggests is $R(x_i) = x_i + 2x_i x_j$, which has the same form as GI. Nonetheless, in even more complex functions, GI might break completely.

Note 2. How should R behave in cases where $x_1 \neq x_2$? In such case, our intuition could not lead us to any conclusion about the model's feature attribution.

Now, let's consider a scenario inspired from game theory.

Example 3. Game theoretic.

In a scenario inspired from game theory, two teams A and B compete against each other to achieve a higher overall score. Each team comprises of two players: A_1, A_2 and B_1, B_2 respectively. In this scenario, the strategies of the players and the resulting scores are completely symmetrical. Team A comes up with the following strategy; player A_1 is devoted on increasing the score of his/her team, while A_2's aim is to minimize the score of the other. Their score is a_1. On the other hand, team B follows a different approach. The two players coordinate to maximize their score. They have a combined contribution b equally distributed, but because A_2 attacked them, the score of the B team is $b - a_2$. Let's assume that $a_1 > b - a_2$ and that team A wins.

How could we attribute the victory of team A to its players? We could argue that $R_A = (a_1, 0)$, since only the first player contributed to the team's score. Or, we could find a complex relationship that also includes the contribution of A_2 to the drop in score of the B team. What about team B?

Its score cannot only be attributed to its players ($R_B = (b/2, b/2)$). We should also consider A_2 to have a *negative contribution* to its score. Thus, an attribution that explains the outcome of the model should give positive and negative values to *all* the players of the game; $R(A) = ((a_1, 0), (0, 0)), R(B) = ((0, -a_2), (b/2, b/2))$. In such case, does the winning of the A team also owe to the players of B that did not play optimally?

A followup example combines and generalises non-linear models along with game-theory. This combination might possibly lead to the design of a DNN model. After all, a DNN is a non-linear model, for which in optimization the parameters are adjusted in such a way that not only a particular class is favored (a team wins), but also the other classes get zeroed (the other teams lose).

2.3 Towards a robust Attribution

The examples presented earlier were intended to provide readers with an initial grasp of the issue. Even for basic functions and scenarios, addressing these straightforward questions is immensely challenging. Robust methodologies have been devised and will be explored further in the following sections. The pursuit of comprehending various methodologies, their resilience, efficacy, and the creation of novel techniques for Explainable AI continues. The quest to identify the ideal attribution method remains an ongoing inquiry to this day.

Part II

BACKGROUND

Chapter 3

Attribution methods

Various approaches have been explored to design methods for attributing the importance score of a decision back to input features. These methods can be grouped into three main categories: **Gradient** methods, **Occlusion** methods and **Class Activation Mapping** methods.

Gradient-Based methods leverage gradients as a guide to feature importance. Gradients point to the most sensitive directions of features that significantly influence the model's decision. **Relevance Propagation methods** share similarities with gradient methods but apply distinct rules between layers to guide the backpropagation of information -usually resulting from gradients, yet other tools could be used as well-, backpropagating class information to the input features. On the other hand, **CAM** methods exploit the saliency maps of CNNs, by applying a linear combination of these maps. In contrast, **Perturbation** methods pivot on hiding parts of an image to unveil their importance. Lastly, **Local Approximation** methods fall between Gradient-based and Occlusion-based methods, trying to locally approximate the model using simpler, more interpretable functions.

We will briefly examine each category, focusing on the most mathematically robust methods of each. Before delving into these methods, it's essential to heed the words of authors [78] regarding the attribution of a model's decision:

> *"For attribution, no ground truth exists. If an attribution heatmap highlights subjectively irrelevant areas, this might correctly reflect the network's unexpected way of processing the data, or the heatmap might be inaccurate. Given an image of a railway locomotive, the attribution map might highlight the train tracks instead of the train itself. Current Attribution methods cannot guarantee that the network is ignoring the low-scored locomotive for the prediction".*

Thus, authors argue that no assumption should be made about the attribution of the model, its shape and appearance. Otherwise, authors who overlook this observation are subject to ***confirmation bias*** as defined in [23]:

> *"A severe limitation of these approaches (i.e. Attribution methods) is that they are subject to a confirmation bias: while they appear to offer useful explanations to a human experimenter, they may produce incorrect explanations. In other words, just because the explanations make sense to humans does not mean that they actually convey what is actually happening within the model."*

3.1 Gradient methods

Gradient methods perform single or multiple backward passes through the model, gathering information from gradients to generate an attribution map. The underlying concept behind these techniques is that gradients are closely related to *sensitivity*. They point to the directions in which a function experiences the most rapid changes. Consequently, they highlight *important features* that, when perturbed, lead to the most significant reduction in the model's confidence. In this sense, gradient methods share similarities with Occlusion Methods.

Gradient methods are often grouped with **Rule-Based methods** which define rules for propagating activations backward through the model's layers until they reach the input layer. Together, they form the **Backward methods**. These two methodologies are introduced in the following subsections.

3.1.1 Review of legacy gradient methods

The first and most straightforward Attribution method, known as **Saliency Maps** dates back to the analysis of the earliest CNNs [85]. The authors posited that feature importance corresponds to the absolute value of the feature gradients concerning the most highly activated class. In mathematical terms, this is expressed as:

$$R(x)_i = \left| \frac{\partial f_c(x)}{\partial x_i} \right|. \tag{7}$$

Here i represents a specific feature and c denotes the highest activated class. However, it was observed in [63] showed that this method can only provide a *local* that this method provides only a local explanation of the model's prediction. This implies that it can interpret only a small portion of the prediction, explaining just a fraction of the overall score, while most of it remains unexplained.

Authors of [82] proposed an alteration by removing the absolute value and multiplying the gradients with the input values of the features. Their method, commonly known as **Input** $\times$ **Gradient**, resulted in sharpening the resulting attribution map.

$$R(x)_i = x_i \, \frac{\partial f_c(x)}{\partial x_i}. \tag{8}$$

In **DeConvNet** [110], the authors outlined a rule for CNNs with Rectified Linear Unit (ReLU) activation, aiming to propagate the model's decision from feature maps in lower spaces to feature maps in higher spaces. The rule is defined as:

$$R(x)_i^l = (R(x)_i^{l+1} > 0) \; R(x)_i^{l+1}, \tag{9}$$

where $R(x)_i^l$ corresponds to the attribution of the i-th feature of layer l for input x. However, in **Guided Backpropagation** [88] the authors redefined the rule by introducing an additional term, yielding:

$$R(x)_i^l = (f(x)_i^l > 0) \; (R(x)_i^{l+1} > 0) \; R(x)_i^{l+1}. \tag{10}$$

3.1.2 Integrated Gradients

Authors of [92] design a gradient-based method, named **Integrated Gradients** (IG), which aims to redistribute the model's decision back to the input features, capturing information about feature importance through the model's gradients. IG follows a path from the input data to a predefined baseline point, which is typically considered as containing no information. This journey gathers information about the contributions of each feature along the way. The mathematical formula is provided below:

$$R(x)_i = (x_i - x_{0_i}) \int_{\alpha=0}^{1} \frac{\partial f(x_0 + \alpha(x - x_0))}{\partial x_i} d\alpha. \tag{11}$$

Here x_0 represents the *baseline* point. A notable strength of this method is its ability to trace back the model's activation to the input features. This is achieved because the Fundamental Theorem of Calculus ensures that:

$$\sum_i R(\mathbf{x})_i = f(\mathbf{x}) - f(\mathbf{x}_0). \tag{12}$$

This is not the only property IG satisfies. It has been proven to satisfy multiple properties, crucial for evaluating a method's robustness and closely related to feature importance. However, one notable limitation of IG is its dependence on the selection of an appropriate baseline point. While the original authors provide some initial guidance on this matter, other works [91] have conducted experiments with various baseline choices for the method. Yet, these experiments have not yielded a clear advantage for any specific baseline. This issue will be examined in more detail in section 8.

3.2 Rule-based methods

Rule-based methods backpropagate the model's output back to the input features, by applying different rules in order to let the information flow between the layers of the model. In this section, three such methods will be described, namely the **Layer-Wise Relevance Propagation** (LRP), **Deep Taylor Decomposition** (DTD) and **DeepLIFT**. The first two share many similarities and, thus, will be presented together.

3.2.1 $\epsilon-$LRP & DTD

Deep Taylor Decomposition [64] and **ϵ-LRP** [10] both developed by the same authors, stand out as two of the most robust Attribution methods designed to date, since they satisfy many desired properties related to importance. Overall, These methods employ very similar rules, with coincide after specifying the sets of hyperparameters.

DTD relies on Taylor Decomposition, which allows the local approximation of a function f around a specific point $\mathbf{x}$, using a *reference point* $\mathbf{x}_0$[1]. This technique facilitates the deconstruction of a model's decision into its individ-

[1] A root point is a point $\mathbf{x}_0$ with the property $f(\mathbf{x}_0) = 0$. It is the same as the *baseline point*, as authors of [92] refer

ual components, each assigned a score based on its contribution to the final decision. In mathematical terms:

$$f(\mathbf{x}) = f(\mathbf{x}_0) + \Big(\frac{\partial x}{\partial \mathbf{x}}\Big|_{\mathbf{x}=\mathbf{x}_0}\Big)^T (\mathbf{x} - \mathbf{x}_0) + \epsilon. \tag{13}$$

This can be expressed as:

$$f(\mathbf{x}) = 0 + \sum_i \frac{\partial f}{\partial \mathbf{x}_i}\Big|_{\mathbf{x}=\mathbf{x}_0} (\mathbf{x}_i - \mathbf{x'}_i) + \epsilon. \tag{14}$$

Thus, an attribution could be

$$R(\mathbf{x}) = \frac{\partial f}{\partial \mathbf{x}}\Big|_{\mathbf{x}=\mathbf{x'}} (\mathbf{x} - \mathbf{x'}). \tag{15}$$

However, determining a suitable $\mathbf{x}_0$ often proves to be an intricate task. Authors argue that the complexity of $f(\mathbf{x})$, can make this problem expensive or even unsolvable. Instead, authors apply **DTD** to each function of each layer in a DNN with *ReLU* activation, where finding appropriate x_0 points is more feasible. They then combine the information and apply rules to backpropagate the decision. These rules are further elucidated below, when **LRP** is explained.

In ϵ**-LRP**, authors introduce the quantity R_i^l signifying the *relevance* of unit i in layer l. Beginning at the output layer L, the algorithm redistributes relevance R_i^l of unit i to units in the preceding layers based on their contributions to R_i^l. One commonly employed rule for this purpose is the $\epsilon - LRP$ rule. For two neurons i, j in two consecutive layers $l, l+1$, along with b_j denoting the bias of neuron j, authors define

$$z_{ji} = w_{ji}^{l+1,l}$$

as the weighted activation of neuron i to j, and

$$Z_j = \sum_{i'} (z_{ji'}) + b_j$$

as the sum of these z-activations plus the bias. Then:

$$R_i^l = \sum_j \frac{z_{ji}}{Z_j + \epsilon \cdot sg(Z_j)} \cdot R_j^{l+1}. \tag{16}$$

Here sg represents the sign function, and ϵ serves as a stabilization term to prevent division by zero.

This method effectively backpropagates information through layers, ensuring that the sum of attributions remains constant for all layers, a property referred to as **Conservation**. Authors explored various rules for backpropagating relevance (another popular rule is the $\alpha\beta-$rule).

Despite their effectiveness in numerous experiments and visual examinations, both methods come with their limitations. Firstly, they disregard non-linearities, with DTD only considering first-order terms (due to its requirement), while LRP rules incorporate non-linear unit activations within x_is. Additionally, they exclusively consider non-negative relevance scores,

making them unable to identify negative attributions that certain features may have on the resulting score, as demonstrated by [6]. Lastly, for linear models, the attribution produced by these methods aligns with the weights of the variables, as noted by authors of [26] mention. Researchers [6] have raised further concerns regarding $\epsilon-$LRP.

3.2.2 DeepLIFT

DeepLIFT is another method that performs rule-based, backward propagation of the model's score, akin to LRP. This method assigns a score to each unit, representing the impact on the overall score when the value of a unit is set using a *reference* input, denoted as x_0. Specifically, for the last layer L:

$$R_i^L = \begin{cases} f_i(x) - f_i(x_0) & \text{if } i = c \\ 0 & \text{otherwise} \end{cases} . \tag{17}$$

Then, the reference values $\overline{z}_{ji}$ for all hidden units are determined by running a forward pass $f(x_0)$ of the baseline input x_0 and monitoring the activation of each unit:

$$z'_{ij} = w_{ji}^{(}l+1,l)x_0 i^l. \tag{18}$$

The rule applied, referred to as the *Rescale* rule, is as follows:

$$R_i^l = \sum_j \frac{z_{ji} - \overline{z}_{ji}}{\sum'_i \overline{z}_{ji'}} R_j^{(l-1)} \tag{19}$$

DeepLIFT appears to combine elements from both LRP and Integrated Gradients. The choice of the baseline is often set as the zero point, which is considered one of the method's limitations.

3.2.3 A comparison of Gradient and Rule-Based methods

In a study by the authors of [6], various Backwards methods were compared, and intriguingly, their findings suggest that these methods may not be as distinct as initially perceived. They reformulated ϵ-LRP and DeepLIFT (with the *Rescale* rule) using gradients, resulting in mathematical expressions that closely resemble the aforementioned gradient-based methods. This convergence in mathematical expressions becomes especially evident when the activation function used in the model is ReLU or Tanh. Specifically, four methods -$\epsilon-$LRP, DeepLIFT, IG and Gradient*Input- yield nearly identical attributions in this context.

This phenomenon occurs because ϵ-LRP effectively aligns with Gradient*Input when ReLU is employed, and aligns with DeepLIFTin the case of a network with no additive biases and $f(\mathbf{0}) = 0$. Additionally, they discovered a high correlation between DeepLIFT and IG. The authors elucidate this further:

> *"While Integrated Gradients computes the average partial derivative of each feature as the input varies from a baseline to its final value, DeepLIFT approximates this quantity in a single step by replacing the gradient at each nonlinearity with its average gradient. Although the chain rule does not hold in general for average gradients, we show empirically ... that DeepLIFT is most often a good approximation of*

Integrated Gradients. However, we found that DeepLIFT diverges from Integrated Gradients and fails to produce meaningful results when applied to Recurrent Neural Networks (RNNs) with multiplicative interactions (eg. gates in LSTM units ...)."

Considering these findings, it suggests that IG may be the most robust method among them. Therefore, this research thesis further explores the attributes of Integrated Gradients, attempting to address some of the challenges it presents in Section 8.

3.3 Class Activation Mapping methods

First introduced in [113], the authors described the initial **Class Activation Mapping** (CAM) as a method for explaining the decisions made by CNN models. CAM leverages **feature maps** which are two-dimensional maps resulting from the application of convolutions in CNN architectures.

CAM relies on a technique known as **Global Average Pooling**. Instead of flattening the last Convolutional layer's output into a vector, GAP computes the average of each feature map, reducing its the spatial dimensions to 1x1. These values are considered as weights for the corresponding feature maps, to form a weighted sum. For a class of interest c and feature maps A^k

$$R_c(\mathbf{x}) = \sum_k w_k^c A^k(x), \tag{20}$$

where

$$w_k^c(x) = \sum_{i,j} A_{i,j}^k \tag{21}$$

In the equations above, f_i represents the $i-$th feature map of the last layer, which is a two-dimensional matrix (comprising variables i and j). R_c is the resulting attribution map, which is upsampled to match the dimensions of the input image.

Essentially, CAM computes a weighted sum of feature maps where the weights are determined by the model's classification weights. Other methods select other values for the weights w_i^c, which they consider to yield more robust attribution maps. Until today, there exists a large number of such methods [19, 45, 102, 103, 111, 65] Two of the most popular CAM methods, namely **Grad-CAM, GradCAM++**, while **X-GradCAM** is also included, since it is designed in such way that it satisfies important criteria.

3.3.1 GradCAM & GradCAM++

GradCAM and GradCAM++ are discussed together because they both determine the weights for the weighted sum in a similar manner. **GradCAM** computes the weight of a feature map by considering the first-order derivative of the output score y^c with respect to each of its components. These values are then summed up and normalized by a factor Z. Expressed in mathematical terms:

$$w_i^c = \frac{1}{Z} \sum_{i,j} \frac{\partial y^c}{\partial A_{i,j}^k}. \tag{22}$$

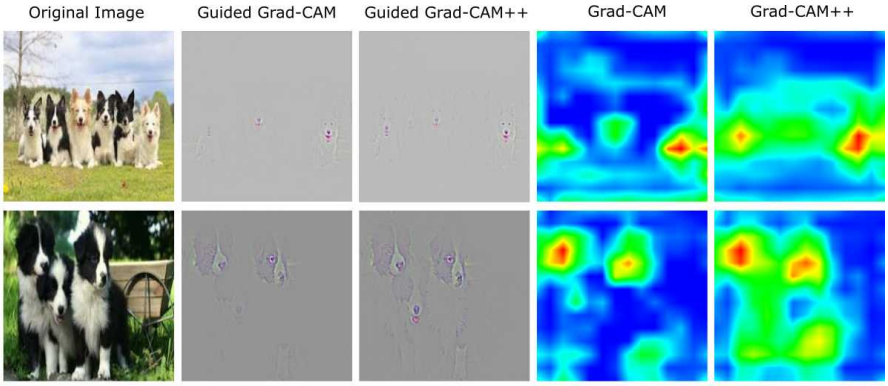


Figure 3: GradCAM and GradCAM++ in action.

On the other hand, **GradCAM++** aims to address certain inefficiencies of GradCAM, especially in scenarios where multiple objects or regions exist in an image. It leverages second-order gradients and applies a ReLU function to the gradients, which helps to better focus on the objects of interest. As a result, it provides more detailed and accurate visualizations, aligning more closely with human judgment regarding the important regions in an image. An illustration of the application of both methods can be found in Figure 3.

3.3.2 Explanation Grad-CAM

In the Explanation Grad-CAM approach, the authors of XGrad-CAM [27] aim to determine CAM weights in a robust manner. While their method is rooted in CAM techniques, it draws upon ideas from Occlusion methods, which will be discussed in the following section. The central concept is to select weights not based on intuition but in a way that meets critical mathematical properties. They define various mathematical criteria that they argue should be satisfied and translate these criteria into a set of equations, the solution to which yields the appropriate weights. These properties are discussed in more detail in Chapter 5, but are briefly outlined below.

1. Sensitivity.

The sensitivity criterion ensures that any reduction in the prediction score due to the occlusion of a specific feature map can be entirely attributed to the activation of that map when it is activated. Specifically, if c represents the class with the highest activation and f^{lk} is the activation of the $k-$th feature map in layer l, the weights w_k^c should satisfy the following equation:

$$f_c(x) - f_c(x \setminus \{k\}) = \sum_{i,j} w_k^c A_{ij}^k(x) \tag{23}$$

where K is the number of feature maps at layer l and $x \setminus \{k\}$ denotes the prediction for class c, when the k-th feature map in the target layer has been replaced by zero. As Equation 23 should hold for any k, it results in a set of K equations.

2. Conservation.

The principle of Conservation posits that the linear combination of the activations of all feature maps should match the model's prediction. Using the same notation as before, this criterion is translated as:

$$f_c(\mathbf{x}) = \sum_{k=1}^{K} \left(\sum_{i,j} w_c^k f^{lk}(i,j) \right). \tag{24}$$

The authors find an approximate solution to the $K+1$ equations mentioned above, thereby defining a CAM attribution. While their method may exhibit certain weaknesses, the direction of their approach is intriguing. They establish mathematical criteria that an Attribution method should satisfy, being a solution to the losses incurred by these criteria. In this thesis, we will also employ this concept in Chapter 9.

3.4 Perturbation & Occlusion methods

> *"When an image classifier makes a prediction, which parts of the image are relevant and why? We can rephrase this question to ask: which parts of the image, if they were not seen by the classifier, would most change its decision?"* - Refered in [13]

An intuitive way to evaluate importance is as follows: a feature is considered important for the model's prediction of a particular class if its presence leads the model to be confident about that class, while its removal results in a significant decrease in the model's confidence in that class.

Methods that utilize this criterion are called ***Occlusion*** or ***Perturbation Methods***. They apply alterations to various features of the input and measure the corresponding impact on the output. These changes to the input can either be small perturbations, such as adding slight random noise, or more substantial interventions in the form of occlusion methods. Such methods play a significant role in this particular thesis. Among the various occlusion methods, the most noteworthy are summarized below.

3.4.1 Occlusion-x

Occlusion-x, introduced in [110] represents one of the earliest Attribution methods. It operates on the principle of occluding parts of an image and measuring the resulting drop in the model's prediction score. In this method, the image features that are going to be hidden are grouped together, forming a box of size x. This box is then slid across the image. As it slides, it conceals the pixels they overlap by replacing them with a gray color. This method, though rudimentary, served as a foundational concept and laid the groundwork for more robust Occlusion methods. Figure 4 provides a visual representation of the sliding window operation in Occlusion-x.

Figure 4: In Occlusion-x a gray window slides across the image, monitoring the drop in model's confidence.

3.4.2 LIME

LIME [73] stands for "Local Interpretable Model-agnostic Explanations" and is a prominent Attribution method in Explainable AI (XAI). It relies on Perturbation to locally approximate the model's decision boundaries. LIME works by perturbing input data points and observing how the model's predictions change, enabling it to approximate the model's functionality around a particular point. Figure 5 offers a visualization of the method.

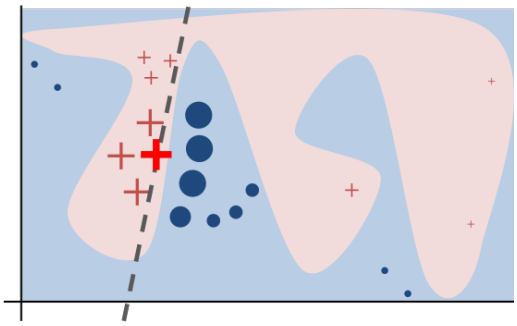


Figure 5: LIME samples points near an input point and locally approximates the model at hand.

In essence, LIME creates a simplified, interpretable surrogate model for a given instance by sampling and perturbing the input data points around it. It then trains this surrogate model to approximate the behavior of the complex model in the local neighborhood of the instance of interest. It achieves this by minimizing the following loss

$$\mathcal{L}(f, g, \pi_x) = \sum_{z, z' \in \mathcal{Z}} \pi_x(z)(f(z) - g(z'))^2. \tag{25}$$

Here f represents the model being approximated by a linear model g. Points are sampled around x, according to π_x and are denoted as z. These points are then mapped to the space of g as z'. The loss function ensures that the linear model effectively approximates the behavior of the model in the vicinity of x.

LIME's strength lies in its model-agnostic nature, making it applicable to various machine learning algorithms without requiring knowledge of their internal workings. This quality renders it a valuable tool for understanding and debugging machine learning models, fostering trust in AI systems.

3.4.3 RISE

RISE [69] stands for "Randomized Input Sampling for Explanation," and it works by generating a set of random masks that are applied to the input image. These masks are binary matrices that indicate which portions of the image are visible and which are hidden. By repeatedly applying these masks and observing how they affect the model's output, RISE collects information about the importance of different image regions for making a prediction. To be more precise, RISE attributes to a feature i a value

$$R(x)_i = \mathbf{E}_M[f(x \odot M)|M(i) = 1] \tag{26}$$

meaning that for a mask $M : \mathcal{I} \Rightarrow \{0,1\}^{|I|}$, feature i is visible. Thus, the method gathers all masked images where feature i appears, calculates the expected value of the predictions of f in those images and attributes this value as a score to i. Due to the computational infeasibility of constructing all possible masked images and calculating the score, the authors employ the

3.4.4 DeepSHAP

DeepSHAP, introduced in [60] is a novel approach to model explainability that draws inspiration from Shapley values in game theory [55, 101]. The fundamental idea behind Shapley values is to determine a fair attribution for the participants on a winning team, based on their contributions to the outcome.

DeepSHAP exhibits similarities to the RISE method and also incorporates elements from LIME. The method divides each image into batches and trains a linear model, akin to LIME, treating each batch as a variable. The weights assigned to these variables are calculated similarly to RISE. However, instead of measuring the model's score in all scenarios where the feature is included, DeepSHAP measures the difference in score when the feature is excluded. In mathematical terms, if F is the set of features and x_S the input value where only the features in $S \subseteq F$ are not hidden, the weight of the $i-$th feature is:

$$w(x)_i = \sum_{S \subseteq F \setminus \{i\}} \frac{|S!|(|F| - |S| - 1)!}{|F|!}[f_{S\cup\{i\}}(x_{S\cup\{i\}}) - f_S(x_S)]. \tag{27}$$

The SHAP method has proven to provide robust attribution and is mathematically established as the best choice among linear methods (ie LIME, $\epsilon-$LRP, Occlusion), based on certain critical criteria. These criteria will be explored in moire detail in Chapter 5.

3.4.5 Review

Numerous other methods incorporate perturbation or occlusion at their core [38, 22, 67, 18, 87, 24, 25]. We could also argue that methods such as Integrated Gradients [92] and DeepLIFT [83] are also related to this category as

they require the existence of a baseline input entirely devoid of information.

However, Occlusion methods come with their deficiencies. Authors of [86] revealed that LIME and SHAP are susceptible to adversarial attacks by models that imitate their behavior in the regions of interest but operate randomly elsewhere. These attacks stem from the models' ability to discern the origin of the data points (perturbed or not) because they follow distinct distributions. Another limitation of these methods is their inefficiency in addressing the concept of information concealment. Most of these approaches rely on zeroing the values of different features to mask them. Yet, as demonstrated in Chapter 6 this approach does not furnish a robust selection. Therefore, a different path must be pursued, guided by rigorous mathematical Criteria 9, 10. This challenge forms the focal point of the second part of this thesis.

3.5 A discussion on Attribution methods

This chapter has only covered a limited subset of Attribution methods developed thus far, focusing on those chosen based on their popularity or intriguing mathematical concepts. However, many other methods have been developed to address inefficiencies in the aforementioned methods or introduce entirely new designs. Some of these methods are briefly mentioned below.

IBA is a recent method introduced in [79] that introduces random noise to intermediate feature maps to calculate their contribution. It shares similarities with X-GradCAM and produces highly effective visualizations for attribution maps. **FullGrad** [89] is an advancement over gradient methods, considering both the neuron activations and input gradients. A different technique is applied by the authors of [49] which aims to measure the importance of a concept to a specific decision. It accomplishes this by collecting latent representations of a particular concept (e.g., the stripes of a zebra) and other random concepts, defining a linear classifier to distinguish them, and generating a scalar value that quantifies the concept's influence on the model's decision, for the particular input. Their method is named **TCAV**.

Some methods have attempted to blend attribution and explainability with interpretable rules and techniques. Certain methods train models to explain their decisions using natural language [TODO] while others translate explainability into first-order logic rules [TODO]. Additional approaches seek to link XAI with Information theory [16].

One of the most promising recent directions that researchers have explored involves Explainability in Transformers [100]. Transformers have gained considerable attention due to their strong performance and their innate capacity for explaining decisions, through the Multi-Head Attention [12]. Nevertheless, concerns about the effectiveness of such methods have arisen, and ongoing research continues to address these issues [42, 105]. Some approaches have endeavored to combine these techniques with traditional Attribution methods [15, 4].

In summary, there is currently a wealth of algorithms for computing attribution maps. The natural question that arises is which of these methods is the most suitable. Evaluation metrics, covered in Section [4] and mathematical Criteria [5] aim to provide answers to this question.

Chapter 4

Evaluation

The need for Evaluation metrics

In section 3 we explored various Attribution methods, each generating a distinct Attribution map, highlighting different features as crucial to the model's decisions. Some of these methods might present a theoretical similarity (as we saw in Subsection 3.2.3), but in general, they produce vastly different maps, exhibiting diverse shapes and minimal overlaps, which is particularly evident in the different CAM methods 3.3 as illustrated in Figure 6. Consequently, they do not converge to a unified decision. Determining which method outperforms the others is the fundamental quest that Evaluation metrics aim to address.

4.1 The nature of Evaluation metrics

Evaluation metrics serve as a means to assess the effectiveness of an Attribution method. In contrast to model training, where the correct predictions are known in advance, in the field of Explainable AI the ground truth is not predefined. Consequently, Evaluation metrics in Explainable AI significantly differ from the conventional scoring systems used to evaluate a model's performance. The true attributions of features are unknown, and yet Evaluation metrics attempt to measure the effectiveness of an Attribution method in some way.

Attribution methods and Evaluation metrics

Defining a score for Attribution method effectiveness in the absence of true attributions is not straightforward. This score should ideally be linked to the concept of feature importance, which is the primary objective of various Attribution methods. Conceptually, Attribution methods and Evaluation metrics are based on the same foundations and it is not uncommon for the development of an Attribution method to be guided by a novel Evaluation metric. The most notable instance is that of Occlusion 3.4, with other similar cases found [37]. However, the central question persists: How do we design such metrics?

To address this question, researchers have either constrained the attribution to specific local image regions or measured the satisfaction of Criteria associated with the concept of importance. As a result, different Evaluation metrics can be categorized into two primary groups; *Localization-biased* and *Importance-based* metrics. Yet, they also exist other *fundamental* metrics, to ensure that Attribution methods satisfy important properties. These are the *Sanity Checks* described below.

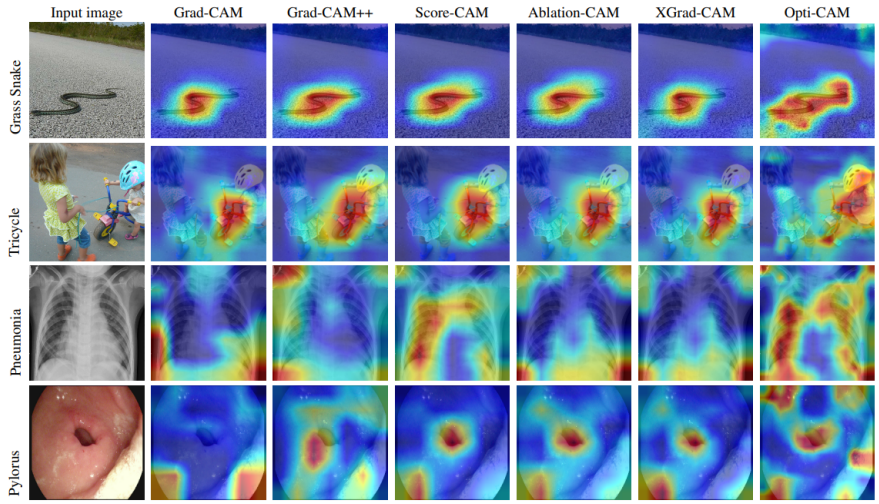


Figure 6: Attribution maps created using various CAM methods for four different examples from the Imagenet dataset. The first column displays the original images, while each subsequent column corresponds to a specific CAM Attribution map generated using the method referred in that column

4.1.1 Sanity Checks

Sanity Checks do not aim to measure importance scores or introduce localization bias to assess the effectiveness of Attribution methods. Instead, they encompass standard tests that any Attribution method should satisfy. An analogy might be that such checks does not measure how fast a human walks, they only confirm that it can stand.

Authors of [2] have devised two randomization tests to assess whether an Attribution method meets fundamental Criteria.

Model Randomization Test

The **Model Randomization Test** focuses on monitoring the dependence of an Attribution method on a model's parameters. It involves randomizing the weights of a model and then comparing the resulting Attribution map generated by the Attribution method. This randomization can be performed in layers, either sequentially or randomly. The underlying idea is based on the expectation that, as the model's decision-making process changes due to random weight perturbations, the resulting Attribution map should differ significantly from the initial attribution, essentially resembling a random attribution. The test revealed that certain methods, such as Guided Backpropagation [88] and GradCAM [80] were invariant to changes in higher-layer weights.

Similarly, authors of [96] have conducted multiple tests to examine the sensitivity of various Attribution methods to the underlying model used. However, these tests remain a subject of dispute, as it has been demonstrated that randomly initialized CNNs can act as powerful classifiers [77, 98]. To address this, authors of [108] created a controlled environment in which they asserted that random models could not interpret images. To do so, they generated synthetic images featuring multiple objects.

Data

The **Data Randomization Test** involves randomly altering the labels of

the images and then retraining a model. In this scenario, no predictive model should perform better than one that guesses randomly. The test calculates the Attribution map for both models (trained on correct and altered labels, respectively) using an Attribution method and measures the difference between the two maps.

4.2 Localization metrics

This section introduces a category of Evaluation metrics that aims to constraint the regions of the image to which an Attribution method should direct its focus. This can be accomplished by either introducing human bias into the regions considered vital or devising custom environments that exclude specific image areas from contributing to the model's decisions.

4.2.1 Pointing Game - Localization

The Pointing Game metric, originally introduced by the authors of [112] has since become a foundational Evaluation metric [81, 69], with similar criteria developed in subsequent works, such as the localization criterion in [14, 69]). This metric involves a human annotator for each image-label pair $(\mathbf{x}, y)$ who marks the regions within the image where objects are located. The Attribution method is then utilized to identify the most important pixel and check whether it falls within the annotated object regions. If it does, the image is classified as a *hit*. The *localization accuracy* score is calculated as follows:

$$Acc = \frac{\#Hits}{\#Hits + \#Misses} \tag{28}$$

Authors argue that the higher the localization accuracy of an Attribution method, the more effective the method is.

Nevertheless, such a technique introduces a human bias to the model. As authors of [69] state:

> *"Such evaluations not only require a lot of human effort but, importantly, are unfit for evaluating whether the explanation is the true cause of the model's decision. They only capture how well the explanations imitate the human-annotated importance of the image regions. But an AI system could behave differently from a human and learn to use cues from the background (e.g., using grass to detect cows) or other cues that are non-intuitive to humans. Thus, a human-dependent metric cannot evaluate the correctness of an explanation that aims to extract the underlying decision process from the network."*

4.2.2 DiFull

Authors of [72] employ a unique approach by creating composite images from four images in the dataset. Specifically, they shrink the four images and arrange them in a 2×2 grid, the same size as the original image. The top-left image in this grid contains the object of interest. The authors propose that an Attribution method, when backpropagating the decision for this specific object, should only highlight regions within this image. Any attribution

pointing to areas outside the top-left image is considered incorrect. A visualization of their concept is presented in 7.

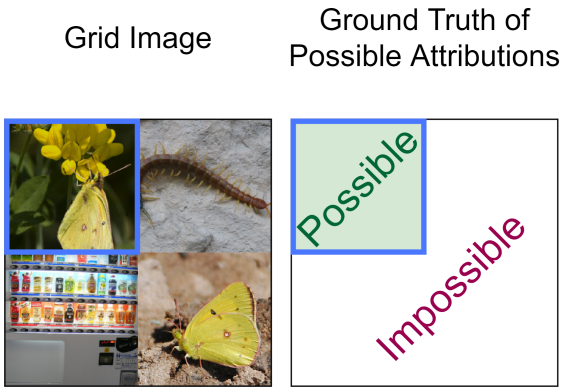


Figure 7: The 2×2 grid for a particular set of images. As the annotator image on the right shows, only the image that contains the object(s) of a particular class can have a possible attribution.

However, the conceptual basis of this approach might be questionable. The model is not trained on 2×2 grid images that contain multiple objects. It is capable of identifying patterns related to the object of interest in other regions as well. The model does not inherently understand the concept of a 2×2 grid or respect its borders. It can extend its focus beyond the specified region without any restrictions, resulting in both possible and impossible regions, making the approach challenging to justify.

4.2.3 Optimization tests

The methodology presented by the authors of [48] involves the application of optimization techniques to create a controlled experimental environment where the contribution of features is either permitted or restricted. Their approach bears resemblance to that discussed in **DiFull** 4.2.2, but it involves the application of loss functions to differentiate between possible and impossible image regions. Their experiments are summarized below:

- **Null Feature Experiment**. In this experiment, two images denoted as m and n are inserted in a vast area with random noise. The optimization process ensures that each maximizes a different class a and b respectively, while at the same time constraints to second-*null* image to have a zero contribution to a. If $x_{m,n}$ represents the composite image containing these two elements, the losses can be expressed as:

$$\min_{m} f_a(x_{m,n}) \tag{29}$$

$$\min_{n} f_b(x_{m,n}) + (f_a(x_{m,n}) - f_a(x_m))^2 + (f_a(x_n) - f_a(x))^2 \tag{30}$$

  where in x_m, only the m image is added to image x (x_n respectively) and no image is added in x. The two last terms of equation 30 ensure that image n does not contribute to any coalition of features. Thus, an Attribution method should only while the where the optimization algorithm performed on one of the two images, alters in in such a way that it does not affect the model's decision, in any cooperation it forms

with other players-images. An Attribution method should neglect this image, and point only to the first one.

- **Single/Double Feature Scenario** In these experiments, optimization was performed to allocate features to different classes (or, in the case of a single feature, to only one class). This resulted in Attribution maps that differed significantly between classes.

The authors used statistical tools to assess the performance of various Attribution methods in these scenarios. Their findings identified GradCAM [81], Extremal Perturbations [24] and IBA [79] as the best-performing methods. However, it is important to note that these methods may lack robustness due to the use of backgrounds with random noise (see Section 6) Additionally, optimization algorithms may not consistently produce the expected results.

4.3 Importance-based Evaluation metrics

Other Evaluation metrics are based on the concept of a model's importance and are closely aligned with Attribution methods.

4.3.1 Average Drop

The **Average Drop** (AD) criterion, introduced by [14] is rooted in a simple concept: hiding the most important image parts, as determined by an Attribution method, should significantly affect the model's confidence in its decision. In essence, a more effective Attribution method is one that results in a larger drop in the model's confidence.

Specifically, for a model f, an image x, f the model's decision favors class c, thus

$$f(x)_c = \max_j f(x)_j,$$

then, for an Attribution method R, AD normalizes the attribution $R_c(x)$ to values between zero and one. The normalized attribution is then used to calculate the Hadamard product $x' = x \odot \text{norm}(a_c(x))$. This process helps define AD as:

$$AD = \sum_{i=1}^{N} \frac{max(0, f(x)_c - f(x')_c)}{f(x)_c} \tag{31}$$

A lower AD indicates that the Attribution map effectively retains the most important regions, keeping the score $f(x')_c$ high. Thus, a better AD is one that is larger.

The authors also introduced a complementary metric known as the ***Increase in Confidence***. This metric counts the examples where $f(x')_c$ exceeds $f(x)_c$, and normalizes this value according to the dataset's size.

By applying these metrics, the authors were able to quantitatively evaluate the effectiveness of various Attribution methods. The detailed results can be found in their paper. It is important to note that implementing these metrics may introduce significant challenges for the model, as explained in 6. Hiding image information is a non-trivial problem, that can introduce bias to the

model. It can also generate images that are Out-of-Distribution (OoD) for the given data distribution.

4.3.2 Top-*K* Ablation

Authors of [33] define a similar metric to that of *AD* 4.3.1, where instead of calculating the hadamard product, they select the top-K features and drop the others, to calculate the $F1$-score of the model to the newly devised test set. They fill the values of the missing parts by selecting different *baseline* values according to heuristic methods for filling image parts (explained in 6.2). Thus, their method is more oriented towards evaluating different filling techniques.

4.3.3 Sensitivity-*N*

Sensitivity-N [6] is another metric that relies on feature concealment. It randomly masks a subset of the model's features and quantifies the correlation between the drop in the classifier's score and the attribution associated with the masked parts. Given a set T_N containing N randomly selected features, Sensitivity-n calculates the Pearson correlation coefficient as follows:

$$Sensitivity - N(x) = PCC(\sum_{i \in T_N} R_i(x), f_c(x) - f_c(x \setminus T_N)) \tag{32}$$

where $x \setminus T_N$ denotes that the features of x corresponding to T_N are set to zero.

4.3.4 Remove and Retrain

This metric, introduced by [36] assesses the impact of hiding the most important image parts, but does so only after retraining the model. The rationale behind retraining the model stems from the recognition that introducing black masks to the image can lead to data being outside the original distribution used for training. Consequently, the model may struggle to make informative decisions in regions it was not trained on. After retraining the model, the authors measure the drop in the model's confidence for the original image and its masked version. For Attribution methods that effectively highlight the most important image features, the expected drop in confidence should be greater compared to weaker methods. Surprisingly, the results showed that only a limited number of Attribution methods outperformed random masking.

However, this idea was later challenged by the authors of [75], who suggested that there might be information leakage from the mask to the model. They argued that the shape of the black masks' boundaries, hints at the identity of the object. This allows the model to identify patterns between the concealed objects and the classes to which they belong. Consider a simple example: a model trained to detect whether an apple is on the left or right side of an image. The background of the image is consistently grey. If a high-performing Attribution method points to the apple, attempting to hide the apple with a black mask and retraining the model may still lead it to

accurately identify the object's location, only now it recognizes a black object instead of an apple.

4.3.5 Insertion - Deletion

Another criterion for evaluating the effectiveness of an Attribution method is the *Insertion-Deletion* criterion, inspired by [26] and introduced in [69]. In this approach, an Attribution method is used to create an Attribution map, and features are sorted according to their attribution values. The *Deletion* criterion involves deleting feature values one by one until all features are removed, measuring the model's prediction at each step for the original image's predicted class. This process forms a curve, as depicted in Figure 8. The score calculated based on the area under the curve (AUC) represents the performance of the Deletion criterion. In contrast, the *Insertion* criterion operates in the opposite direction, starting from an empty image (with all features removed) and gradually adding features, maintaining the same order. A higher AUC in both Deletion and Insertion indicates better performance.

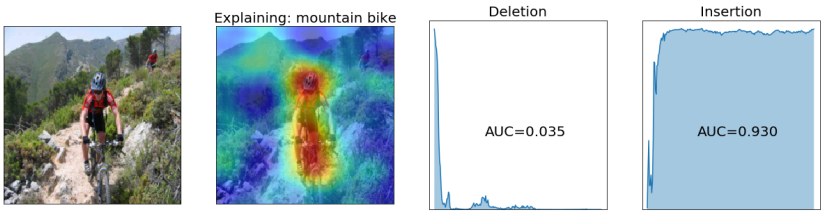


Figure 8: The curves of Deletion and Insertion when hiding an image according to an Attribution map.

However, authors of [29] have shown that these metrics lack robustness, as they do not account for the intensity of feature attribution. Furthermore, these methods may result in cases that deviate from the initial data distribution, resulting in OoD images.

Employing simplistic, heuristic techniques to hide input features can introduce significant bias to the model, potentially rendering the resulting score ineffective. It is also essential to consider that these methods assume independence between features, whereas the concealment of one feature may significantly affect the contribution of others, making independent measurement challenging. More on this will be discussed in Section6.

4.3.6 DC-AC

The *Deletion Correlation* (DC) and *Insertion Correlation* (IC) Criteria [29] are an alternative version of the *Insertion-Deletion* metrics. Instead of calculating the *AUC* of scores, these Criteria compute the linear correlation of class score variations and saliency scores at each step. This helps measure how much each feature contributes to the model's decision. However, it is worth noting that these metrics share the same limitations as *Deletion-Insertion* metrics.

4.3.7 Average DCC

The *ADCC* metric, introduced in [70] specifically for evaluating various *CAM* methods 3.3, calculates a score by combining elements of the Average Drop 4.3.1, the $L1-$norm of a *CAM*, and its ***Coherency***. Coherency measures the relationship between the attribution of a CAM to an image x and its attribution after being hidden according to $CAM(x)$. For the maximally activated class c for an image x,

$$Coherency(x) = PCC(CAM(x),\ CAM(x \odot CAM(x))), \tag{33}$$

where *PCC* stands for the Pearson's Correlation Coefficient. The three scores are combined using the harmonic mean to produce the *ADCC* metric. The Coherency criterion is interesting as it suggests that the image's attribution should not be significantly altered after removing less important parts. Despite its potential, Coherency faces similar challenges as the *AD* metric, which will be explored further.

4.3.8 Robustness-S

The *Robustness-S* criterion [37] takes a different approach compared to the aforementioned methods. It acknowledges that hiding information may introduce bias to the model and measures the robustness of an explanation by attempting to *break* it. This criterion quantifies the effort required to succeed in altering the model's prediction. For the general problem defined as:

$$\epsilon^*_{x_T}(f, x, T) = \min_{\delta}\{\|\delta\|_p, f(x+\delta) \neq c \wedge \delta_{\overline{T}} = 0\}, \tag{34}$$

where $\delta_{\overline{T}} = 0$ indicates that features not found in T are zeroed. The aim is to discover

$$\epsilon^*_{x_{T_R}} \text{ and } \epsilon^*_{x_{\overline{T_R}}}. \tag{35}$$

Here, T_R are considered to be the most important features according to an Attribution method R. The goal is thus to find a minimal perturbation of the important features to change the model's prediction. For the different Attribution methods, a small $\epsilon^*_{x_{T_R}}$ is considered superior (indicating less effort required), while the opposite holds for $\epsilon^*_{x_{\overline{T_R}}}$.

While finding the minimum value of a set through optimization is impossible [47], the perturbed features may indeed introduce bias to the model. It is important to note that changes in the model's prediction may not solely result from perturbing the non-fixed features, but also the other features as well. As we will prove in this thesis, optimization algorithms might work unexpectedly.

4.4 Challenges

Precise measurement

In the context of Evaluation metrics, one can identify potential challenges that might hinder their performance in *precisely* measuring the effectiveness of different Attribution methods. A more comprehensive examination of the most popular Evaluation metrics can be found in [96]. In many cases, the

challenges stem from the way features are removed.

Occlusion is well founded

Still, Evaluation metrics do indeed calculate the effectiveness of Attribution methods, but rather approximately. The idea of *Occlusion* is theoretically well founded: when concealing unimportant image parts that contain an object of interest, we expect a mild drop in the model's confidence in the corresponding class. However, the extent of this drop, which can only be explained via an Attribution method is yet to be precisely defined.Our general expectation is that random masking would lead to a more substantial drop due to the concealment of vital information. To gain further insights into this matter, an experimental study could be conducted to measure the drop in score when random masking is applied, and monitor how the drop in model confidence is distributed among different classes, ideally with no favoritism towards any specific class.

Most popular Evaluation metrics

Among the most popular Evaluation metrics found in the bibliography are Average Drop, Sensitivity-N, and Insertion/Deletion. The first two involve the removal of information in a one-step process, while the latter necessitates multiple steps in pixel removal. The latter approach introduces additional biases and errors to the implementation. For example, at each step, the Attribution map may change due to the interdependencies of the values of feature maps, as discussed in Section 2, resulting in a new sequence of values. Since the first two methods do not sort features based on their attribution scores, they are not susceptible to this particular issue. Consequently, this thesis aims to mitigate some of their deficiencies.

While various issues in the design of Evaluation metrics were briefly discussed earlier, a more detailed and in-depth exploration will be conducted in Sections 6, 7, 8, 9. These sections will focus on designing an algorithm for effectively concealing information from image parts. This development can potentially enhance different metrics based on Occlusion, ultimately creating a more robust tool for evaluation. The pursuit of designing a robust Evaluation metric remains an ongoing endeavor.

Chapter 5

Axioms and Criteria

The need for Criteria

In this chapter, we delve into the development of axioms and Criteria for the evaluation of attribution methods. These foundational principles aim to establish a theoretical foundation for assessing the robustness and effectiveness of different methods. While traditional evaluation techniques offer rough estimations of importance, they lack a definitive yardstick for ranking attribution methods – determining whether higher or lower scores indicate better performance. Given the intricacies of how models operate, making such judgments is inherently complex.

The lack of theoretical robustness of the Evaluation metrics has been acknowledged by researchers in XAI. Authors of [92] state the following:

> *"Roughly, we found that every empirical evaluation technique we could think of could not differentiate between artifacts that stem from perturbing the data, a misbehaving model, and a misbehaving attribution method. This was why* ***we turned to an axiomatic approach in designing a good attribution method****"*

The axiomatic approach involves defining Axioms and Criteria that inherently align with the concept of feature importance.A method's effectiveness is determined by its adherence to these Criteria. This provided a new way for evaluating and comparing different methods. This approach introduces a new methodology for evaluating and comparing different attribution methods: he evaluation is not based on a score, but rather a "Yes/No" answer tied to the satisfaction of essential Criteria.

Questions regarding Criteria

However, this approach also raises a series of questions and challenges. Are some Criteria more critical than others? Is there a hierarchy of importance among them? How many Criteria are necessary to ensure the effectiveness of a method? And, potentially, could the design of these Criteria be influenced by human bias?

5.1 Criteria-based Evaluation

While these questions remain unanswered and indicate the need for further research, the XAI community started engaging with Criteria-based evaluation approaches, as they offer a more robust methodology for assessing attribution methods. In the following sections, we will present various Criteria found in the existing literature.

5.1.1 Sensitivity

This criterion, as introduced by [92] consists of two key statements. **Sensitivity-a** posits that when two inputs and baselines differ in a single feature but produce distinct model predictions, the divergent feature should be assigned a non-zero attribution. **Sensitivity-b**[1] on the other hand, declares that when a model is independent of a specific variable, the attribution to that variable must always be zero.

At first glance, both statements appear logical since they establish a relationship between cause and effect. If a modification of a feature's value fails to produce an effect, it should not be considered a causative factor. While these two definitions complement each other, Sensitivity-a is more practical from an evaluation standpoint. It is relatively straightforward to assess by zeroing the values of a feature that influences the model's decision and verifying that an attribution method assigns an importance score to that feature. Conversely, identifying the variables upon which a complex model depends is a far more intricate task, requiring experimentation.

The authors illustrate this criterion with a simple example that demonstrates the apparent violation of **Gradients**. They extend the same reasoning to **DeConvNets** 9 and **Guided Backpropagation** 10, both relying on ReLU, which renders them susceptible to problems with Gradients.

Concerns

Trying to conceptually understand this criterion, one might express concerns about its validity. Features in DNNs have a combined effect to the model's prediction, thus an alteration of the value of a feature could lead to new interactions between features. A rule that restricts the feature of interest to attribute more or less than before does not hold water.

Lemma 1 (Sensitivity)**.** *A change to a model's prediction, caused by the perturbation of a particular feature's value should lead to a differing attribution of the contributing features.*

In practice, this criterion may not yield worthy results. Any change in a DNN's input feature is bound to produce a change to the output. Also, features are inherently entangled within the DNN, and modifying one feature may affect the entire feature set. In such cases, this criterion essentially asserts that if a change in a feature's value leads to a different output, the resulting attribution should also be different (which will be, for an attribution method that respects the input values).

5.1.2 Conservation

This criterion postulates that the sum of importance scores within an Attribution map should equate to the model's decision. In mathematical terms:

$$\forall \mathbf{x} : f(\mathbf{x}) = \sum_i R(x)_i. \tag{36}$$

While initially introduced in [10], this criterion can also be found in various other research works. It is referred as *Completeness* in Integrated Gradients

[1] authors of [60] refer to this criterion as *"Missingness"*.

[92], *Conservation* in X-Grad Cam [27] (for the needs of feature maps) and aligns with *local accuracy* in SHAP [60]. The underlying rationale for this criterion is grounded in the belief that the model's activation should be completely explained by the attributions of features. Failing to do so implies that something essential remains unexplained and unattributed, leaving part of the model's decision unaccounted for. Similarly, an attribution method should not attribute higher accuracies to features than the model's output score, as this overstates the significance of these features in the decision-making process.

The application and interpretation of this criterion may differ among methods. Integrated Gradients and SHAP use it as a *meta*-criterion that their methods satisfy to prove their robustness. Conversely, $\epsilon-$LRP and X-Grad-CAM are *engineered* according to it, guaranteeing its satisfaction. To achieve this, in X-Grad-CAM this criterion is transformed into a loss function, which is subsequently optimized while in the case of $\epsilon-$LRP, it serves as a guiding principle for backpropagating the model's output score to the attributions of individual features. In practice, evaluating this criterion can be challenging, primarily because a model's output is not scalar but a vector of class scores. Determining how attribution methods should calculate the attribution of each class and ensure that these sum up to the model's output score remains a subject of exploration.

Is this criterion theoretically rational, and is it intricately linked to the notion of importance? Some may argue that it might not be of great consequence if one were to scale the values within an attribution map by an arbitrary factor. What may be more important is the *relative relationship* between different attribution values. Additionally, between the model's input and output, numerous complex nonlinear functions come into play. It is certain, though, that there exists a correlation between the attribution scores and the model's decision, given the cause-and-effect phenomenon. Yet, it is not evident what this relationship should be.

5.1.3 Positivity

Introduced by $\epsilon-$LRP, [10], this criterion asserts that an attribution method is considered "positive" if the values within the attribution maps it generates are always greater than or equal to zero. In mathematical terms:

$$\forall \mathbf{x}, i : R(x)_i \geq 0 \tag{37}$$

This criterion enforces non-negativity in attributions, implying that a feature's effect on the model's decision can only be positive or neutral. This criterion serves to theoretically fortify the LRP method.

However, it raises concerns regarding the constraints it places on attribution methods, which may not necessarily align with the model's functionality and training process. During training, the objective is not solely to maximize the activation of a specific neuron but also to suppress the activation of *competing* neurons. Thus, the model's parameters may be adjusted to ensure a *negative effect* on specific neurons' activation when combined with the input. The question arises: can this constraint extend to the particular neuron of

interest? It is conceivable that a feature affects negatively the neuron of interest, by deactivating important neurons along the way. As such, the positivity criterion remains a topic of concern and debate.

5.1.4 Implementation Invariance

Defined in IG [92], the criterion states that for two **functionally equivalent** models, the attribution method should produce identical explanations. In mathematical terms,

$$\forall f_1, f_2 : \mathbb{R}^m \to \mathbb{R}^n \text{ for which } \forall x \in \mathbb{R}^m \text{ it holds that } R_{f_1}(x) = R_{f_2}(x). \quad (38)$$

This criterion might resemble the Randomization Tests 4.1.1 in a way that it comprises a basic property an attribution method should satisfy, more related to *general characteristics* of it, and not to the evaluation of a method. Yet, DeepLIFT and LRP fail to satisfy this criterion, which comprises a serious disadvantage for these methods.

5.1.5 Consistency

This criterion is applicable primarily in scenarios akin to SHAP, where a complex DNN is approximated by a linear explanation model. An input x for a model f is translated to z' for the approximate linear model through a function h_x (which usually corresponds to "hyper-variables" of f, such as hyper-pixels), it states the following:

Given $f(z') = f(h(z'))$ and $z' \setminus i$ denote the action of setting $z'_i = 0$, the criterion posits that for any two models f and f', if $\forall z' \in \{0, 1\}$ it holds that

$$f'(z') - f'(z' \setminus i) \geq f(z') - f(z' \setminus i),$$

then

$$\phi_i(f', x) \geq \phi_i(f, x). \quad (39)$$

Here ϕ_i represents the $i-$th weight associated with the $i-$th element of the linear model. The essence of this criterion is to establish a direct link between the relative contributions of i in f than f' where the exclusion of this feature results in a more substantial drop in the model's prediction. This heightened drop should correspond directly to the weight ϕ_i of i in the models; the weight for i in f *needs* to be larger than that of f'. Yet, this criterion suffers from the same limitations as Sensitivity

5.1.6 Weak Dependence

Introduced in [89], authors define the notion of weak dependence on inputs. They consider piece-wise linear models defined on open connected sets, thus

$$f(x) = \begin{cases} w_0 \cdot x + b_0, x \in \mathcal{U}_0 \\ ... \\ w_n \cdot x + b_0, x \in \mathcal{U}_n \end{cases} \quad (40)$$

where all $\mathcal{U}_i$ are open connected sets. Authors suggest that for this function, the Attribution map $R(x)$ restricted to a set $\mathcal{U}_i$ is independent of x, and

depends only on the parameters w_i, b_i.

They attempt to generalize the attribution in simple linear models, yet, such a selection for the attribution might be poor for those models (as seen in Example 2). By the use of this criterion they try to challenge Integrated Gradients, by defining the following piece-wise linear model:

$$f(x) = \begin{cases} 3x_1 + x_2, & \text{if } x_1, x_2 \geq 1 \\ x_1 + 3x_2, & \text{if } x_1, x_2 < 1 \\ 0, & \text{otherwise} \end{cases} \tag{41}$$

and consider a baseline $x' = (0, 0)$ three points (2, 2), (4, 4), (1.5, 1.5), all of which satisfy $x_1, x_2 > 1$ and thus are subject to the same linear function of

$$f(x_1, x_2) = 3x_1 + x_2.$$

However, depending on the point considered, IG yields different *relative* importances among the input features.

- for $\mathbf{x}$ = (4, 4), $R(\mathbf{x}) = (10, 6)$,
- for $\mathbf{x}$ = (1.5, 1.5), $R(\mathbf{x}) = (2.5, 3.5)$,
- for $\mathbf{x}$ = (2, 2), $R(\mathbf{x}) = (4, 4)$,

thus, for symmetrical points of the same function, the attributions vary, pointing to different features as the most important each time.

The problem with this approach is the selection of the baseline point. If for example, $x' = (1, 1)$, the problem is not present. In complex functions, the baseline might not just be the zero point. We conduct an extensive research in Chapter 8.

5.1.7 Continuity

Defined in [27], this axiom states that for two nearly identical inputs that lead to identical model activations, then the corresponding explanations should also be nearly identical. For $x_1, x_2 \in \mathbb{R}$ for which $|x_1 - x_2| \leq \delta_1$ and for a model f for which $|f(x_1) - f(x_2)| \leq \delta_2$, then

$$|R(x_1) - R(x_2)| \leq \epsilon, \tag{42}$$

for $\delta_1, \delta_2 > 0$, and $\epsilon = \epsilon(\delta_1, \delta_2) > 0$.
This criterion mainly applies to gradient methods, suffering from the problem of the shattered gradient [7]. Simple gradient methods, as well as CAM gradient methods suffer from it, yet methods that do not only use the local gradients (such as IG and DeepLIFT) are immune.

5.1.8 Combining different Criteria

The aforementioned criteria should not be viewed in isolation when assessing an attribution method's robustness. Each of these criteria, on its own, can be susceptible to manipulation by simple or handcrafted examples. For instance, the Conservation criterion could be misled by an attribution $R(x)_i = 1/n$, where n represents the output size, satisfying not only the Conservation

criterion but also Continuity and Positivity.

It is the interplay and combination of different criteria that enhance the robustness of an attribution method. While other criteria have been proposed (such as rotation [26], Symmetry [92], Model Saturation [83]), this thesis primarily focuses on the more established and widely recognized criteria, as they hold greater significance. A comprehensive and universally accepted set of criteria has yet to be formulated, with each criteria-based method defining its own criteria based on their design and mathematical framework.

5.2 The power of Criteria

Given the intense scrutiny directed at the entire spectrum of criteria in this thesis, and the assertion that their individual utility remains somewhat limited, it is natural for readers to ponder their effectiveness. To address this, let's delve into the work of SHAP [61] as an example. SHAP incorporates three criteria, as mentioned earlier, to assert and prove the superiority of their method among similar alternatives. Let's examine this method more closely.

Initially, SHAP introduces the notion of the *explanation model*, which treats a model's decision explanation as a distinct model, akin to the LIME approach. SHAP defines this explanation model to be linear, inherently making it more interpretable. Mathematically, this is represented as:

$$g(z') = \phi_0 + \sum_{i=1}^{M} \phi_i z'_i. \tag{43}$$

Different methods

Various methods, including DeepLIFT, LRP, LIME, and Shapley Value Estimation techniques, select different coefficient values for this model. It's this choice of coefficients that distinguishes these methods. SHAP then demonstrates that their method is the sole approach that satisfies the three criteria they have devised.

In essence, SHAP groups methods that share the same mathematical expression and concurrently defines criteria linked to the explainability properties of these methods. They subsequently identify coefficient values that meet these criteria, showcasing their method's superiority over others. This pathway holds promise for achieving model explainability when more refined and resilient criteria are established.

5.3 Further discussion

Not all criteria-based methods operate under the same principles. Integrated Gradients and SHAP first establish an attribution method and subsequently substantiate its effectiveness through the formulation of criteria tied to these methods' properties. In contrast, LRP and XGradCAM have attribution methods that are guided and directly derived from criteria. This distinction is noteworthy.

From where do these criteria originate and why did the authors select them instead of others? One might argue that these are logical criteria that an explanation method should fulfill, or it would be deemed inconsistent. However, in the case of SHAP, other authors employ different criteria for their methods. For instance, DeepLIFT employs the model saturation criterion to demonstrate its efficiency, which SHAP does not satisfy. Therefore, it's possible that there are additional properties related to explanation methods that methods like DeepLIFT, LRP, and LIME fulfill but that SHAP does not.

Set of Criteria

In any case, a comprehensive and well-rounded set of criteria that guarantees the effectiveness of an explanation method remains an uncharted territory, with much more exploration and research needed in this domain.

Part III

ZERO INFORMATION THEORY

Chapter 6

The concept of Zero Information in XAI

As previously mentioned in section 3 and 4, the notion of Information Concealment is fundamentally rooted to the field of Explainable AI. Different attribution methods, evaluation metrics and mathematical criteria are founded on it. Yet, research on a mathematically robust algorithm for information concealment still lacks. Most of the aforementioned methods and metrics make simple assumptions about information concealment, without diving deeply into the mathematical details. This in turn adds a serious bias to the model. This topic is further developed in the following two chapters.

6.1 Zero values

Zeroing: A standard method

Conventionally, the standard procedure for occluding a set of features within an input involves setting their values to a baseline. Zeiler and Fergus [110], introduced this idea and applied it to RGB images, selecting the grey color as the baseline value. Later on, the baseline evolved to the zero input (black value), recognized as a more suitable candidate for 'zero information'. This is in accordance with Information Theory, where the zero vector (or any constant vector) corresponds to a message with zero Entropy. As a result, this message contains the minimal possible quantum of information. For a discrete random variable X which takes values in $\mathbb{X}$ and is distributed according to $p : \mathbb{X} \to [0, 1]$ The entropy is defined as

$$H(X) = -\sum_{x \in \mathbb{X}} p(x) log_b p(x) = E[-log_b p(X)] \tag{44}$$

Thus, the entropy of a degenerate distribution[1] is equal to zero.

Patterns in black images

However, Deep Learning exhibits limited correlation with Information Theory. Blackening may not inherently equate to zeroing information for a DNN. Some evidence comes from the black image itself: by feeding a black image to different models, the outputs are non-zero. This phenomenon arises from the existence of positive biases in deep layers that force certain neurons to activate, triggering computations in the subsequent layers. Consequently, a particular neuron of the last layer will get a higher score (which will be amplified by the *softmax* function), raising the following question: *Does the black image contain any information?* Evidently, DNNs think it does.

[1] A degenerate distribution is a distribution with a basic characteristic: there exists some value x, for which $P(X = x) = 1$

Conceptually, we could generalize this idea to any black segments, in different backgrounds -not only in black, and arrive at the conclusion that blackening parts cannot hide information entirely.

IG and the black baseline

The authors of Integrated Gradients propose the use of *an image with no signal* as the baseline reference point, which might be a point with a small activation. They acknowledge the fact that images containing information might satisfy this criterion (such as adversarial examples) and so they advise the use of a black image. We quote:

> *"So we would additionally like the baseline to convey a complete absence of signal, so that the features that are apparent from the attributions are properties only of the input, and not of the baseline. For instance, in an object recognition network, a black image signifies the absence of objects. The black image isn't unique in this sense—an image consisting of noise has the same property. However, using black as a baseline may result in cleaner visualizations of "edge" features."*

In the case of the black image, the mathematical formula (Equation 11) implies that the attribution for the black parts of the input image will always be zero. That is because the value of those features matches the baseline value, leading to the nullification of the first multiplication term within the formula. This constraint poses a limitation to the method's applicability. It is not always true that a black segment of an image has zero contribution to the model's decision, as we will see in section 6.3.

6.2 Heuristic techniques

Alternative strategies

Researchers rapidly recognized that employing a black overlay was an inadequate strategy for information concealment. Consequently, they embarked on investigating alternative approaches in order to hide information effectively. Different filling methods were examined, which also inherited a human-perceived notion of zero information. Some of the most common techniques are the following:

- **Blurring** of the hidden part.
- **Addition of random noise** to the hidden part.
- **Replacement** of the hidden part **with random noise**.
- Replacement of the hidden part with the application of **max distance.**
- **Averaging** through different points for **max distance**.

The bias persists

Nonetheless, despite the diverse range of methods employed, an inherent bias persisted In a study conducted by the authors of [31], 10,000 images were generated, each filled with random noise. Surprisingly, when these images were fed into a *maxout network* [30] equipped with a *softmax* layer, it was found that for over 98% of the images, the model assigned a probability of more than 50% to some class. This observation suggests that a significant amount of information existed within random noise. Soon thereafter, authors of [66] extended these findings to images containing structures composed of various mathematical shapes. It became evident that heuristic techniques

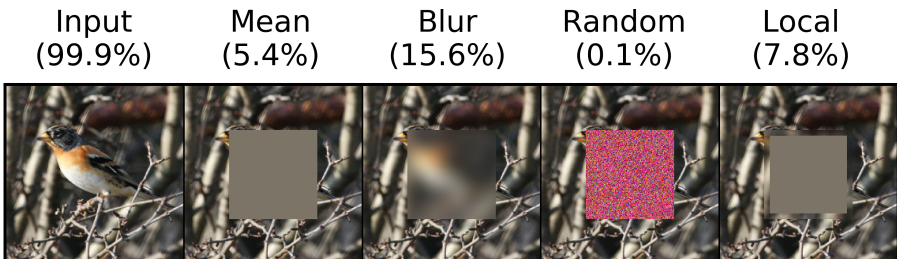


Figure 9: An example image from [13], featuring a sparrow to be concealed. Different heuristics for obscuring image details yield varying model responses. Despite human consensus that none of these methods introduce information to the image, the model's probabilities (shown above each image) diverge, indicating distinct findings in each image.

failed to effectively conceal information from the model.

Different rule-based techniques lead to different results

A comprehensive overview of the various techniques employed to conceal image parts or select a baseline point can be found in [33], where the authors carefully categorize each technique based on its characteristics. Through an array of experiments, the authors of this study deduce that no single method consistently outperforms the others on a large scale. Additionally, they observe that the choice of method significantly impacts the model's decision. This conclusion is in accordance with the findings in [90], where authors perform large scale experiments on Integrated Gradients [92]. To better illustrate the substantial impact of different filling methods on the resulting scores Alipour et al. [5] provided an insightful visualization. It can be seen in 9.

What are the fundamental problems that those methods blindly inherit to the images when occluding information from them? It is of major significance to identify them, in order to construct later a robust method that addresses those challenges.

6.3 The Added Bias problem

The added bias problem has been pointed out in different works [90, 37, 32, 23]. Authors of [37] state the following: for attribution methods and evaluation metrics that are based on Occlusion, hiding a part by setting its values to a reference value

> "*would favor feature values that are far way from the baseline value (since this corresponds to a large perturbation, and hence is likely to lead to a function value difference), causing an intrinsic bias for these methods and evaluations. For example, if we set the feature value to black in RGB images, this introduces a bias favoring bright pixels: explanations that optimize such evaluations often omit important dark objects such as a dark-colored dog*".

Example

Let's examine the aforementioned example in more depth. In Figure 10, we consider the part to be concealed to correspond to the body of a black dog. Hiding this segment by blackening it, would result in a small alteration of the model's judgement and prediction, given that the input was not substantially modified. On the other hand, selecting a bright color as a reference value

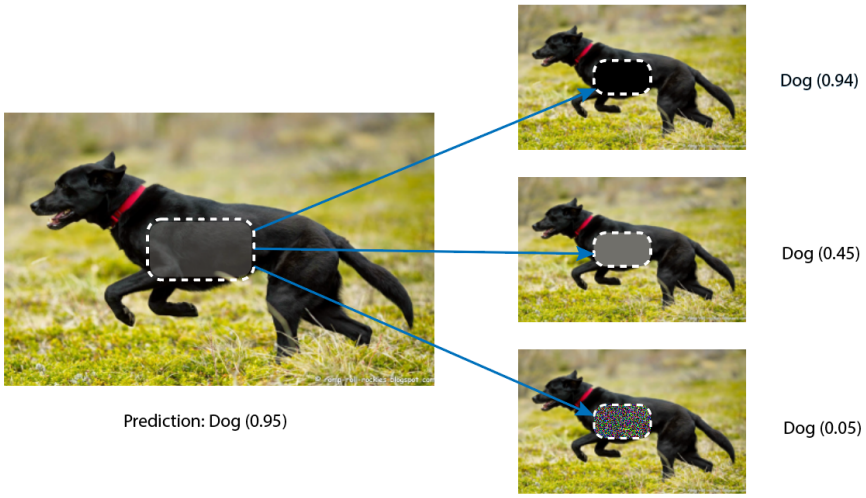


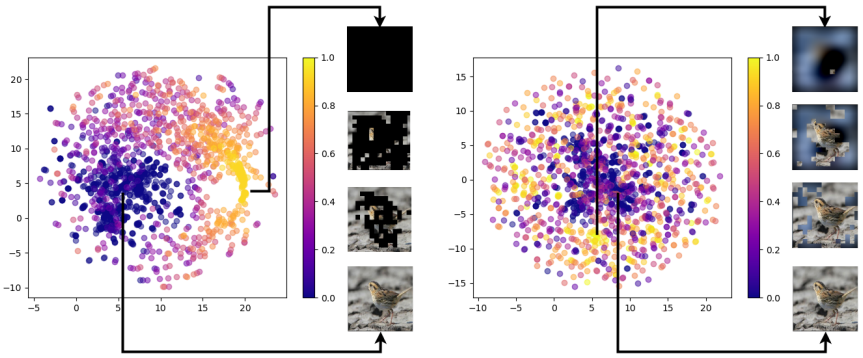
Figure 10: Filling techniques for hiding image parts result in diverging predictions for the ResNet50 model.

introduces a large perturbation of the original image, disturbing the shape of the dog and leading to a greater drop in the model's prediction score. This implies that the black filling retained information contained in this part, while the brighter color probably reduced it. A challenging question that arises is which is the best value of erasing any information and how can we evaluate it?

6.4 The Out-of-Distribution problem

The Out-of-Distribution problem (OOD) arises when there is a significant dissimilarity between the data distributions of the training and testing datasets. Consequently, a DNN trained on discovering patterns in a particular data distribution may struggle to identify different correlations in other distributions, resulting in a notable decline in the model's accuracy. This issue also extends to Occlusion methods in XAI, where new, out-of-distribution image parts are introduced to the model. In such cases, precisely attributing changes in the model's score becomes exceptionally challenging, as the observed drop may be attributed to the OOD problem rather than the occlusion of the object of interest.

In their work [29], the authors aimed to visually represent this distribution dissimilarity for the blackening and blurring filling methods, particularly concerning the Insertion/Deletion metrics 4.3.5. They achieved this by projecting the input data, along with their masked versions created through gradual occlusion, onto lower-dimensional representation spaces, as required by these metrics. The results, depicted in Figure 11 clearly showcase the evolving trajectory of gradual blackening or blurring, which progressively diverges from the original data distribution. This phenomenon extends to various other heuristic filling methods, when occluding image parts.



(a) (b)

Figure 11: The UMAP projection of representations, obtained while computing Deletion (a) and Insertion (b) on 100 images with representations from 500 points of the test set, to visualize the training distribution (in blue). By gradually masking the image, the representations converge towards a point (in yellow) that is distant from the points corresponding to unmasked images. Similarly, blurring the image causes the representation to move away from the training distribution.

6.5 The Attribution Shift problem

Considering the model as a set of nested calculations, the act of altering the values of a subset of features holds the potential to forge new associations between the features of the input in the computational graph. This in turn, would result in a change of the attribution of each feature; some features might now be able to attribute more -if they were suppressed by the zeroed features-, others could possibly attribute less -if they had a joined attribution with the zeroed features-. The DNN of the Figure 12 illustrates this phenomenon.

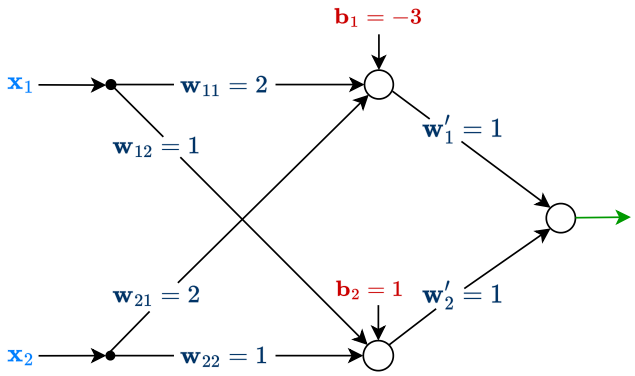


Figure 12: The *Attribution shift* problem is noticeable in this neural network architecture with one hidden layer and ReLU activation functions. If $x_1 = x_2 = 1$, both neurons of the hidden layer are activated, and the two inputs contribute to the output from both the upper and lower path. But, if $x_1 = 1, x_2 = 0$, the upper neuron of the hidden layer is deactivated, thus input x_1 cannot contribute from the upper path. It now contributes less to the model decision, leading to a drop in its attribution score.

Hence, the alteration of specific input components might create an avalanche *False measures*

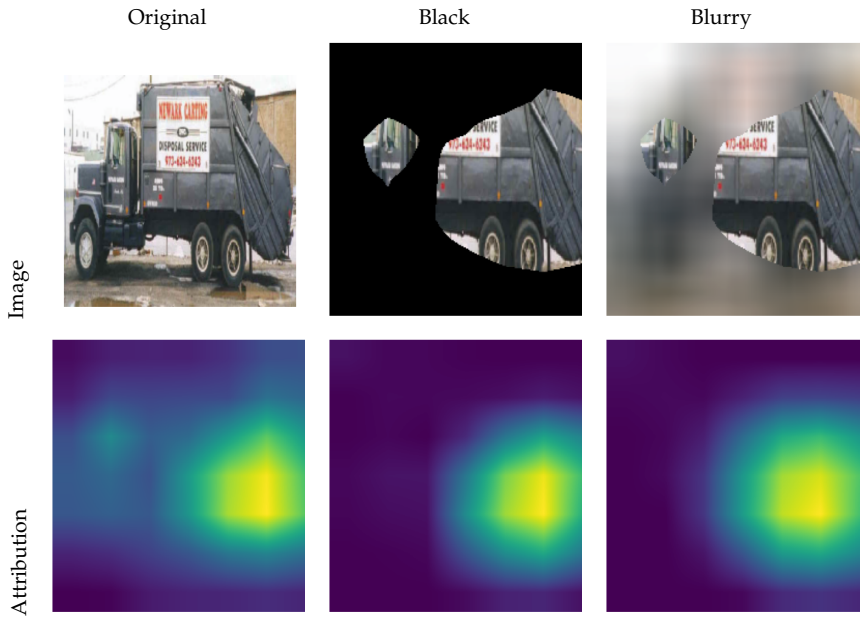


Figure 13: Heuristic methods for hiding image parts induce an *attribution shift* of the visible parts. (Top) The image is masked by thresholding the attribution map of GradCAM; (bottom) a new attribution map is obtained for the masked image. Masking by black or blurry overlays result in the smaller left segment of the object disappearing from the new attribution map.

phenomenon, capable of altering the attributions of all input features. In such cases, the change in accuracy cannot be directly and completely attributed to the concealed parts, since the attribution map has also been altered for the visible parts of the image. A real-world example can be seen in Figure 13.

6.5.1 Towards a robust filling method

These findings have steered many researchers away from employing occlusion methods altogether and work on parallel routes, directed towards adversarial examples [11, 37, 54]. Nonetheless, it is insightful to better study this problem in depth, in order to tackle it effectively and shed more light on the model's hidden structure.

Some methods have explored alternative rule-based techniques, such as neighborhood search [75] that fills the hidden part by using colours from neighboring pixels etc. A fundamental idea of those methods is that the reconstructed part should take into consideration the visible parts of the image. The following lemma summarises these conclusions.

Lemma 2. *Considering a model and an input, the function of hiding information from a subset of features for that particular input is dependent on the model and the image itself.*

Our novelty: An optimization algorithm

A fundamental idea in this project is not to impose different notions of 'zero information' to the model, but rather let the model show us what it considers to be zero information for a particular image. We design ***optimization***

algorithms to the hidden parts that are driven from the model itself, towards the direction of zero information. The next section introduces the reader to this idea.

We start by tackling the challenge of concealing the entire image—a task we find to be comparatively more tractable than the endeavor of constructing zero information segments. As we proceed, we will harness a suite of tools we have developed, enabling us to effectively tackle the latter question.

Chapter 7

Filling methods for OoD data

Conventional machine learning paradigms operate under the assumption that both the training and test datasets derive from the same distribution. This statistical consistency is formally denoted as *Independent and Identically Distributed* (i.i.d.). Nevertheless the existence of distributional shifts in real-world scenarios disrupts this assumption, leading to a significant degradation in model performance. This phenomenon is known as the '*Out-of-Distribution*' problem (OoD) [57].

Described mathematically from the authors, let $\mathcal{X}$ be the feature space and $\mathcal{Y}$ the label space. A parametric model is defined as $f_\theta : \mathcal{X} \to \mathcal{Y}$. Given a set of n training samples of the form $\{(x_1, y_1), ..., (x_n, y_n)\}$, drawn from training distribution $P_{tr}(X, Y)$, a supervised learning problem is to find an optimal model f_θ^* which can generalize best on data drawn from test distribution $P_{te}(X, Y)$. In real-world scenarios, the test distribution upon which a model is deployed may diverge from the training distribution;

$$P_{tr}(X, Y) \neq P_{te}(X, Y)$$

leading to the OoD problem.

An intuitive way to understand this phenomenon is to associate it with the problem of social misalignment, as defined in []. Authors of [32] state:

> Explanations are socially misaligned when people expect them to communicate one kind of information, and instead they communicate a different kind of information. For example, if we expected an explanation to be the information that a model relied on in order to reach a decision, but the explanation was actually information selected after a decision was already made, then we would say that the explanations are socially misaligned.

The generality of this problem makes it appear in different applications of AI. In particular, it has been noticed in Computer Vision[26, 5, 37, 44], Natural Language Processing [50, 99, 76], Audio Classification [40] and many others. Different methodologies have emerged in order to tackle it. Before we explore them, we need to establish the relationship between OoD and XAI.

7.1 Link between OoD & Explainable AI

How does this challenge relate to the field of XAI? As mentioned in section 6.4, occlusion methods, involving the application of masks to input images, can introduce foreign or unfamiliar elements into the image, thereby diverting it from its original distribution. We refer to these elements as ***artefacts***, as

defined in [32]. This phenomenon has been extensively reported and studied in literature [33, 86, 36, 13, 76, 107, 50, 26, 71, 3, 44, 37].

This problem is particularly pronounced in heuristic methods like Mean and Random filling, as illustrated in Figure 9. Furthermore, path-based approaches, as discussed in section 8.4, are also susceptible to OoD issues. The paths leading to a Zero Information Point may traverse regions entirely outside the expected distribution.

Lastly, even evaluation metrics and mathematical criteria based on occlusion are not immune to this challenge. To be more precise, metrics such as Average Drop 4.3.1 and top-K ablation 4.3.2 are vulnerable to OoD-related concerns. However, the extent of their impact on results remains uncertain, as the *correct* filling method has yet to be determined. Mathematical criteria such as Conservation 5.1.2 and Consistency 5.1.5.

7.2 Addressing the OoD challenge

In order to mitigate the challenge of OoD, a wide range of techniques is being tested and deployed.

Model Retraining

A first approach is to retrain the model to data with artefacts. Authors of [36] retrain a DNN model to the input dataset being masked according to different attribution methods and calculate the drop in the model's accuracy. Arguably, is this way, they manage to calculate the OoD-ness of the data, by calculating the difference in prediction of the two models. Another interesting method is introduced by authors of [32], who expose the model to artifacts during the training phase of the model. In that way, when introducing artefacts to the model in the test phase, they do not appear to be OoD. Nevertheless, the approach of retraining the model suffers from an important disadvantage. As shown in [75] artefacts might add information to the model and contribute to its score for a particular input and class.

7.2.1 Marginalizing OoD data

Another technique that was explored by different researchers, was to indentify and marginalize Out-of-Distribution data. Authors of [71] develop an algorithm that improves methods such as LIME, RISE and OCCLUSION, that are based on the construction of multiple masked versions/perturbations of the input. They calculate an *Inlier Score* of each perturbation, which is related to the probability of the model to produce such sample. This score functions as a weight for the perturbation, before the aforementioned attribution algorithms attribute the scores back to image regions.

7.2.2 Selecting artefacts near distribution

In Natural Language Processing, authors of [76] modify the algorithm of Integrated Gradients [92] in such a way that the points it interpolates are not selected blindly in a strait line, thus gathering information from OoD points,

rather collecting data from words in a non-straight path that are more close to the model's distribution. Starting from the initial word to be concealed, the algorithm at each point-embedding searches for the word which embedding is the closest and, at the same time, conserves the monotonicity of the path towards the zero information word. This might require a small perturbation of some features of a word embedding, although the point it stands remains in close distance to an original word of the dataset. Instead of applying heuristic techniques for finding points near distribution, other methods *generate* the artefacts to be in distribution.

7.2.3 Filling the hidden features

A different approach that many researchers are exploring is to fill the hidden parts of the image with different values, that might correspond to the notion of no information, while alleviating the problem of OoD-ness.

Heuristic Methods

Another approach is to attempt to fill the hidden variables with a value that results in an *in-Distribution* (ID) data point. Authors of [43] select a reference value –to use as a zero information point– and define it to be the the expectation $\mathbb{E}[f(\mathbf{X})]$ over the activations of f to the data of the underlined distribution $\mathbf{X}$. For the hidden features $\mathbf{H}$, they select to fill them with the expectation

$$\mathbb{E}[f(x_V, \mathbf{X}_H)], \tag{45}$$

where x_V is the value of the visible features. Researchers in [114] follow a similar approach, calculating the expected value over a neighbourhood of the missing pixels, while authors of [114] use the *Gaussian distribution* to fill the hidden regions. In a different approach, authors of [26] opt to find the optimal mask, considering the added artefacts to be in-Distribution if they form a simple, regular structure. They achieve to do so by regularizing the mask in total-variation (TV) norm and blur the minimal masked part. This term is then added to the loss function.

Generative Models

Authors of [13] translate the criteria of *Smallest Deletion Region* (SDR) and *Smallest Supportive Region* (SSR) into an objective function, that its maximization satisfies them both. In order to fill the hidden features, they deploy different generative models, such as Variational Autoencoders [39] and Contextual Attention GAN [109]. The later seems to reconstruct the hidden parts of the image with a very natural fill, as can be seen in Figure 14. On the other hand, researchers in [107] argue that their technique lacks robustness, since the hidden parts *are not sampled*, rather filled deterministically. Instead, they use a neural network, which they refer to as PatchSampler. Lastly, authors of [3] use a *Generative Image Impainting* model [109] to fill the hidden parts, using the same criterion as in [13], improving the results of different attribution methods, relying on occlusion.

Statistical methods performed poorly when applied for image filling, while generative models have yet to show their power.

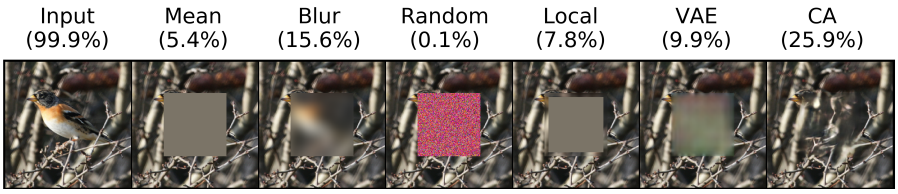


Figure 14: Different filling techniques for image parts, along with the resulting model activation of the class of interest. Generative models (VAE, CA), reconstruct the images with a more friendly and natural fill.

7.3 Generative models for robust Filling

The aforementioned filling methods might alleviate the OoD problem to some extent, but they fail to tackle the problems of added bias 6.3 and attribution shift 6.5. For that reason, we think that a solution should combine a generative model as long as it is combined with optimization criteria. We describe a complete pipeline in section 10. This pipeline uses as a generative model a **Masked Autoencoder** (MAE) [35]. The standard architecture is described below.

7.3.1 Masked Autoencoders

The architecture of MAE is an *asymmetric* encoder-decoder architecture that can reconstruct hidden image parts effectively, while offering scalability. The encoder constructs a hidden representation of the visible parts and the decoder learns to reconstruct the image based on those latent representations. The main idea of the model is that images, in contrast to language, are natural signals with heavy spatial redundancy. For that reason, in order to reduce redundancy and achieve a holistic understanding of the data distribution beyond low-level image statistics, a large portion of the image is hidden (75%). In order to enhance this objective, this architecture is applied to patches of the image, not to the features of it. The architecture can be seen in Figure 15, while the functioning of the model is described below.

Each image is considered as a set of 16x16 patches. Each patch is being mapped to a lower space. A number of patches is being randomly masked.

1. **Encoder**. The visible patches pass through an encoder that maps the input to a lower-dimensional representation. It produces a reduced-dimensional representation of the input data that captures important features of the visible parts.

2. **Decoder**. The encoded data is then passed through a decoder, which aims to reconstruct the original input from the reduced-dimensional representation along with the masked tokens. Its objective is to fill in the missing information caused by the masking in the encoder.

3. **Loss Function**. The quality of the reconstruction is evaluated using a loss function, such as mean squared error or binary cross-entropy, which measures the difference between the original input and the re-

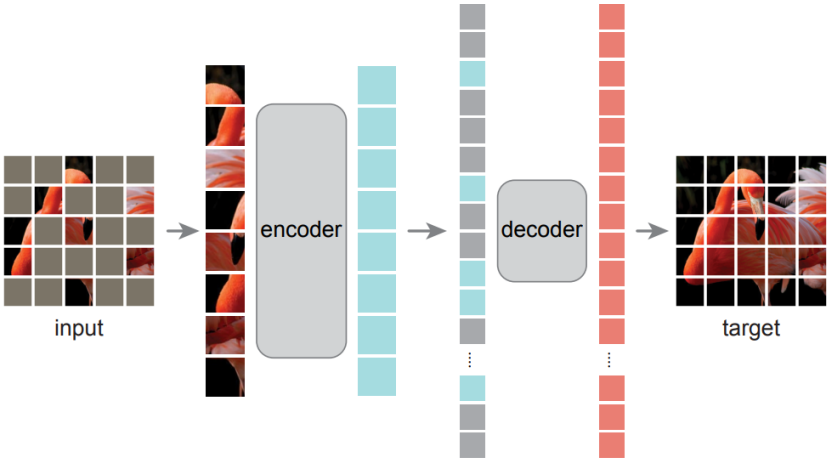


Figure 15: The architecture of MAE autoencoders and the asymmetric encoder-decoder sequence.

constructed output.

4. **Training**. The network is trained to minimize the loss function by adjusting its parameters, including the masking pattern in the encoder. The goal is to learn an efficient encoding that captures the essential information in the data.

The architecture is considered to be asymmetric, since the encoder is being applied only on the visible parts of the image, while the decoder functions in both the visible and hidden image tokens. The model has proven to be useful for other applications as well. Authors of [97] used *MAE* to detect adversarial attacks. They managed to do so by leveraging *MAE* losses to build a Kolmogorov-Smirnov test [62] that detects adversarial samples. Furthermore, they use the *MAE* losses to calculate input reversal vectors to repair adversarial samples. Also, authors of [106] used *MAE* for image augmentation. The masked autoencoders were used to generate the distorted view of the input images, and thus, enlarge the training dataset and increase its diversity with complex examples. They have proven that models trained on the augmented datasets excelled in many vision tasks.

Part IV

ZERO INFORMATION METHODS

Chapter 8

Zero Information Points

The assumption of a baseline point.

As previously discussed, Integrated Gradients [92], DeepLIFT [82, 83] and Shapley values [61] are fundamentally rooted in the existence and utilization of a **baseline point**, which they assert to contain no information. The strategies they employ to discover such points, derive from simplistic, heuristic techniques like adopting random noise or using a black image. Yet, as previous research suggests, even the black and random images may contain valuable information [31].

Nonetheless, it has been demonstrated that none of these choices consistently outperform the others in all scenarios, as underscored by the findings in [91], leaving uncertainty about which choice to make when confronted with an input image. Some of their results are presented in Figure 16.

Zero Information Points

We prefer to refer to such points as **Zero Information Points** (ZIPO), a term that provides a more profound insight into their inherent characteristics. Based on those findings, authors of [41] attempt to tackle the problem of finding a ZIPO for the DeepSHAP method [61] in a different manner. Their main idea is that a technique for finding such a point should respect the model's parameters. Thus, a ZIPO should not be defined, rather be *unearthed* through the application of an algorithm that takes into account the model's parameters. With this in mind, we can deduce the following lemma:

Lemma 3. ***Zero Information Points*** *emerge from the intrinsic operations of a model itself.*

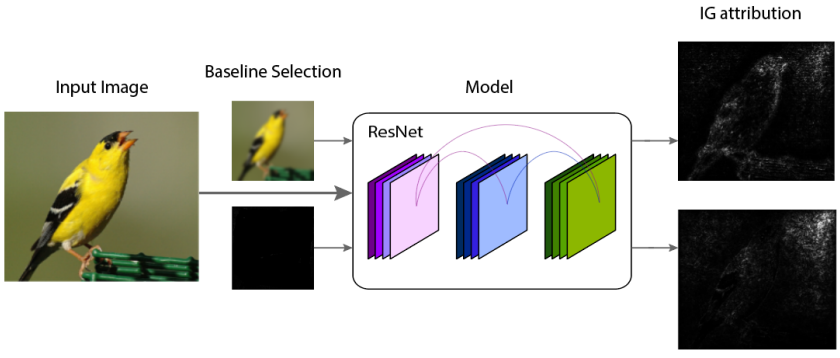


Figure 16: The resulting attribution maps after selecting different baseline values for the initial image.

This lemma implies that the concept of global points unsupportable. This conclusion aligns with a high-level, conceptual intuition. Since each model functions in a unique way, a point that -for some reason- is characteristic to the model, should somehow get correlated with its specific operations and parameters. To assume the existence of a global ZIPO would be akin to asserting that there exists a single root point for all distinct functions.

A ZIPO is not global

In the following subsections, we delve into the fundamental concept of ZIPOs and develop an algorithm to discover them.

8.1 What are Zero Information Points

We inquire "what is a Zero Information Point within the context of a particular model f? How can it be translated to the functioning of the model f? Intuitively, a point x_0 for which f cannot find any information prompts the model to a state of confusion and indecisiveness, given the absence of discriminating cues. This is translated to a uniform activation of the neurons within the last layer. None of the neurons is activated more than the others; rather, they all converge to a dormant state where they leave the model indecisive. Consequently, our challenge lies in the pursuit of a point x_0 for which $f(x_0)$ is gives no advantage to any class. In other words, it is a point that *maximizes the entropy of the model's decision*.

A ZIPO makes a model indecisive

Definition 1. *We define a point x_0 to contain **zero information** if and only if*

$$\boldsymbol{x}_0 = \underset{\boldsymbol{x}}{argmax}(H(f(\boldsymbol{x}))). \tag{46}$$

In a DNN with a *Softmax* layer, this results in a uniform distribution of activations that sum to one. Researchers in Machine Learning have recognized the existence of such points for nearly a decade now. In [31] authors refer to these points as **Rubbish Points** or **Degenerate Inputs**[1] and consider a robust model to be one that satisfies an important property; after the model's computation for such an input

Uniform Distribution

> *"we want all classes output near zero probability of the class being present, and in the case of a multinoulli distribution over only the positive classes, we would prefer that the classifier output a high-entropy (nearly uniform) distribution over the classes."*

Thus, our research stumbles upon the same phenomenon, arguing that such points might be useful for model explainability. Authors of [41] use the same definition, although expressed in a slightly different way:

ZIPO's activation at the decision boundary

> *"the value α is neutral if the decision maker's choice is determined by the value of $f(x)$ (the model) being either below or above α."*

meaning that a model can make no judgement, if its activation is α. Thus, such a point **lies at the decision boundary of the model**.

Authors then refer to the problem of OoD data 7.1, and develop an algorithm for finding such a point at a SLP. The algorithm starts by tracking the lowest output value back to the input values, and slightly increases those values,

An algorithm for a Shapley ZIPO

[1] They are also defined as **fooling images in [66]**

considering that the model is monotonic. At one point, the algorithm will change its decision. This means that it has passed the decision boundary. The ZIPO lies in the interval of the last two steps. It can continue the search in this smaller interval. The algorithm resembles root-finding algorithms in arithmetic analysis. It is then generalized to MLPs, by breaking them down to sequences of SLPs and applying the aforementioned algorithm many times, for each of the SLP components.

Information exists internally

Remark. *Assuming that the DNN architecture applies a softmax function within its final layer, the model's indecisiveness is translated to a uniform activation of the softmax layer. Consequently, this does not exclude the presence of information within the deep layers of the model, however it assures that it cancels out when softmax is applied. Arguably, due to architecture of DNNs,* ***there exists no point for which all neurons of the model are deactivated****, unless we make strong assumptions about the model itself. Thus, it is reasonable to consider that zero information only exists at the model's output.*

Theorem 1. *There exist DNN architectures, for which there exists no input that deactivates all neurons of their layers.*

Proof. Let's consider a FFNN f, comprising L layers and employing the Rectified Linear Unit (*ReLU*) as its activation function. This model contains some neurons that have a positive bias. Let's assume the existence of a point x for which $f^l(x) = \mathbf{0}$, meaning that all neurons of a specific deep layer l are deactivated. Then, for the subsequent layer $l+1$, solely the biases of the neurons contribute to the computation. To prevent neuron activation, we have to strictly impose the condition $\mathbf{b}^{l+1} \leq \mathbf{0}$ to the $l+1$ layer. Since the choice of l was random, a deactivation of all neurons f would imply that $\mathbf{b}^l \leq \mathbf{0}, \forall l \in \{1, 2, ..., L\}$. This contradicts our assumption about the model architecture. □

While the algorithm introduced by [41] is intriguing, we have developed an alternative technique that bridges the concepts of model importance and zero information. The forthcoming subsection will delve into the specifics of this new algorithm.

8.2 An algorithm for Zero Information Points

Optimization Algorithm

As previously noted, a methodology for the discovery of a ZIPO necessitates the guidance of the model. It is natural thus to consider an optimization algorithm for this task. The algorithm we present starts from an initial point, and charts a path towards a local optimum, optimizing entropy while leveraging gradient information along the trajectory. In contrast to Integrated Gradients, this path is not necessarily linear; rather, it is **data** and **model driven**. At each juncture, the algorithm queries the model: *'How would you modify the present input to induce greater confusion?"*.

Intuition behind the algorithm's design

This process constitutes another attribution method, aimed at approaching a *notion of importance* for a particular pair of model-input. As the model adjusts features that wield the most influence over its decision, it effectively conceals

the pivotal parts of the input. Upon reaching a local optimum, the juxtaposition of the original input and the optimal point yields an attribution map.

For an input x, model f, learning rate ϵ and epochs N, The algorithm is outlined in 1.

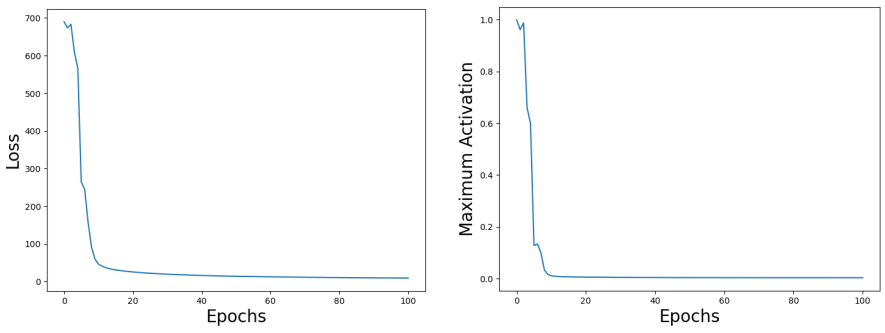
Algorithm 1 The ZIPO algorithm

Require: f, x, N, ϵ
  $x' \leftarrow x$
  for $i \leftarrow 1, ..., N$ **do**
    $y \leftarrow f(x')$
    $l \leftarrow H(y)$
    $x' \leftarrow x' - \epsilon \nabla l$
  end for
  $x_0 \leftarrow x - x'$
  return x_0

8.3 Performance

We perform several tests to our method, to ensure its functionality and stability. First and foremost, we need to secure that the optimization algorithm works and manages to find a trajectory towards a Zero Information Point. We plot the loss graph, as well as the evolution of the maximum value, in order to secure that the algorithm converges to a point with maximum entropy. The two graphs appear in Figure 17. We can see that the two functions behave the same. The application of the algorithm to a single image can be found in Figure 18. Others can be found at the end of the chapter, in Figure 22.

The algorithm succeeds in optimizing the image and erasing any information. The maximum class gets a score less than 0.004 in almost all cases -remember, we want it to be 0.001, but this might be infeasible to do. The best value we found was 0.0019, and in most cases, this value was around 0.0025-0.0034. In some examples the algorithm converged after only 15 epochs, while in others, it needed at least 50 epochs for the loss to start dropping towards zero. In order to reassure that the algorithm succeeds, we increased the gradients of



(a) (b)

Figure 17: The evolution of the loss (a) and maximum activation (b) functions.

(a) Original Image (b) ZIPO Attribution

Figure 18: The application of the zipo algorithm to an image.

the loss by multiplying it with a *weight* of 100, which can be considered as a fixed hyperparameter.

8.4 A discussion on the algorithm

Disadvantages of the algorithm

While the ZIPO algorithm offers an improved perspective compared to Integrated Gradients by considering the model's reasoning and alleviating the problem of defining a ZIPO beforehand, it does come with its own set of limitations.

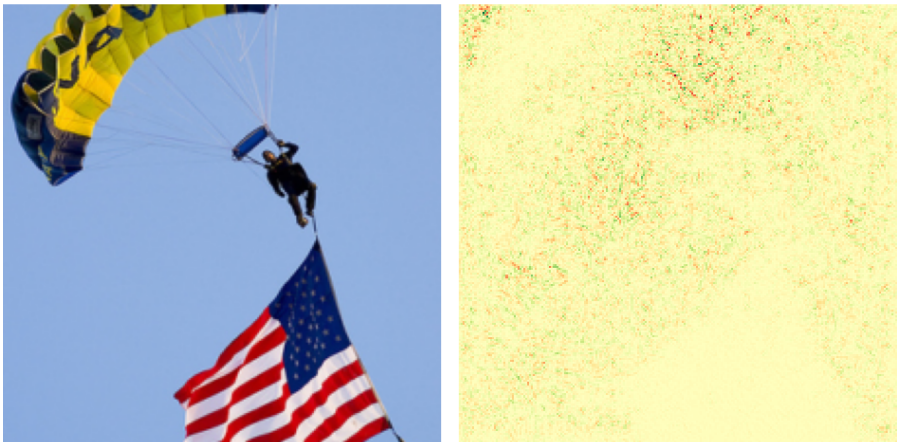
1. It departs from the linear path

Firstly, the ZIPO algorithm departs from the principle of linearity, as the path it treads is no longer constrained to be linear. Consequently, the algorithm does not adhere to the criterion of ***Symmetry Preservation*** as defined in [92]. In a two-step process of the algorithm, a change in direction may lead to alterations of feature contributions. The information gathered during the first step may now be irrelevant. Significant changes in direction, ideally owing to the complex shape of the function, but less desirably due to the introduction of bias and OoD elements into the problem, may occur. We should acknowledge the fact that at each step, the model makes a decision based on a different image, deviating from the original one, potentially leading to OoD regions (to be fair, this is a problem that IG also suffers from). Nevertheless, conducting experiments (Figure 19) with different steps and generating resulting attribution maps showcases consistency and confidence in the algorithm's direction. This suggests that the bias and OoD issues may not significantly influence the algorithm's decisions.

Relaxing the constraint of linearity may not always yield negative outcomes. This could steer the model toward more in-distribution data, allowing the algorithm to rectify its decisions along the way. In a noteworthy example, as presented in [76], authors construct a non-linear yet monotonic path. At each step, this path approaches the ZIPO, which, in the case of NLP -relevant to their application— it corresponds to a fixed token. During each step, the algorithm explores the space of word embeddings to find the nearest

(a) Initial Image (b) Step 5 (c) Step 10 (d) Step 20 (e) Step 50

Figure 19: The evolution of the attribution map for the initial image (a), as the number of steps increases. The behavior of the attribution remains predictable, yet the random noise increases as the number of steps grows.



(a) Image (b) ZIPO Attribution

Figure 20: For the original image (a) the ZIPO attribution is calculated (b). The attribution attacks to the shapes of the objects, leaving the objects themselves intact. The attributions of the image parts that correspond to the flag and the parachute are zeroed.

embedding of a word in the vocabulary in the direction of the ZIPO. Subsequently, it might need to slightly adjust the embedding values to maintain monotonicity. Overall, the algorithm was a success, demonstrating improved results compared to the standard IG.

2. It is susceptible to Adversarial Examples

Secondly, it's not always the case that moving towards the most indecisive direction inherently neglects the most important parts of the image. Within the vast data space, a phenomenon akin to adversarial attacks [95] might be possible to occur.

This suggests that this method might not perfectly align with the concept of *importance*; rather, focusing on introducing confusion and distrubing the model's decision-making process. We observed such a phenomenon in different cases where the optimization algorithm *attacked the shape of an object*. In order for the algorithm to find a way to conceal the information that a complex object carries, it might not hide the object, rather *perturb the surrounding pixels to form a completely different shape* and thus confuse the model. This phenomenon is better illustrated in Figure 20.

(a) Image (b) ZIPO Attribution

Figure 21: For the original image (a) the ZIPO attribution is calculated (b). The attribution attacks not only to the chainsaw (class of interest), but other parts and objects they might be related to other classes, such as the bottle of beer.

3. It zeroes all the Information

The algorithm's core design revolves around the complete erasure of all information contained within an image. Consequently, after eliminating information associated with the highest activated classes, the algorithm may disturb other parts that could be associated with other classes. This could possibly explain a phenomenon noticeable across different images; In the loss curve, the loss drops quickly and then increases again, before dropping near to zero. During these initial steps, the object of the leading class may be successfully concealed, potentially allowing the emergence of other classes. Such an example can be seen in Figure 21.

There remains room for the development of more sophisticated algorithms. These could be crafted to selectively erasing information correlated to the highest activated class, while leaving any other information unspoiled and showing robustness against adversarial attacks. Yet, this raises a new question; how to effectively conceal information from a part of an image, without altering the other. This is the question of the following chapter.

8.5 Conclusion

Despite the inefficiencies of the proposed algorithm, it may still be intertwined with the concept of model importance and serve as a viable approximation of the *accurate* attribution map.

Figure 22: The application of the ZIPO algorithm in different images.

Chapter 9

Zero Information Parts

As explained in Section 6.2 heuristic techniques have the potential to introduce substantial biases into models, in contrast to the *model-based approaches* discussed in section 8. Specifically, when hiding specific parts of an image, heuristic techniques may significantly distort the associations among visible features, as exemplified in Image 9.

The focus of this section is to develop a model-based algorithm tailored to situations where only a subset of features requires occlusion. This chapter opens the discussion of the Zero Information Parts. It does not provide a definite answer to the problem, it rather introduces some problematics and maps a road towards answering the question.

9.1 Problem formulation

In section 8.2, the criterion for identifying a zero information point centered on maximizing the model's entropy. However, in the case when we exclude only a subset of features, what criteria should guide this process? We start our study by outlining the problem at a high level of abstraction.

Definition 2 (Problem Formulation)**.** *Consider a model $f : \mathbb{R}^r \to \mathbb{R}^n$, and a masking function $M : \mathbb{R}^r \to \mathbb{R}^r$. For an input $\boldsymbol{x}$, fill the hidden parts of $M(\boldsymbol{x})$ in such a way that there is no addition/removal of information to/from the visible parts of $M(x)$ with respect to the function f.*

The problem can be translated to decoupling any associations between the visible and the hidden parts, while ensuring that the latter has a zero contribution to the model's decision making. Whether this is possible to achieve, might be a matter of debate. Nevertheless, a tentative to satisfy the aforementioned propertied can be attempted and evaluated later on its success. The criteria are summarized below.

Lemma 4 (Criteria)**.** *The following criteria are deemed necessary for a robust masking of a subset of features.*

- *The filling must not alter the contributions of the visible features.*
- *The filling should introduce no additional information to the model.*

The question at hand is if and whether it is possible to satisfy them both, or if they are in all cases mutually exclusive, thus the quest for an algorithm would be doomed to fail.

9.2 Feasibility of Criteria satisfaction

To illustrate the concept, let's consider Figure 12, depicted again in Figure 23. In this specific architecture, any value chosen for x_2 that fails to activate both neurons in the hidden layer results in a decrease in the attribution of x_1. However, to ensure the activation of the hidden layer x_2 must also *add information* to the model, just as x_1 does. Is there an optimal point for x_2 that can fulfill both these objectives effectively?

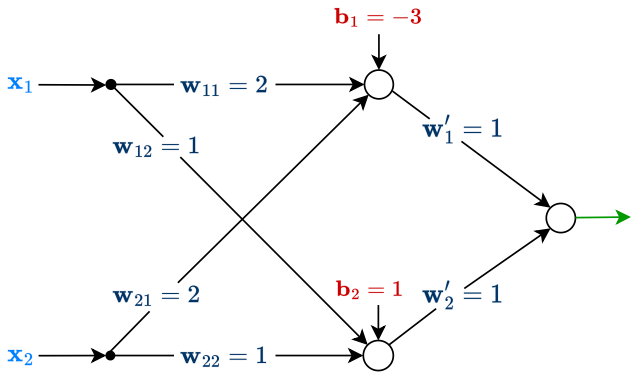


Figure 23: In the architecture presented, if $x_1 = 1, x_2 = 0$, the upper neuron in the hidden layer becomes deactivated, preventing input x_1 from contributing through the upper path. To satisfy both fundamental criteria, the values of x_2 need to be determined in a way that ensures their dual purpose.

In the context of a simple model, like the one illustrated in the example above, an optimal point might not approximate well the solution , as it may not satisfy both criteria adequately. However, in the case of complex models, there might be potential to discover approximate solutions that better meet the criteria.

9.3 A first approach to the problem

To transform the two criteria mentioned earlier into actionable mathematical expressions grounded in the model's parameters and the input image, the design should result it a set of equations. Finding a solution to this set might be infeasible, or computationally expensive, thus the design should focus on establishing a loss function to be optimized.

Consider the input image vector as $\mathbf{x} \in \mathcal{X}$ and a masking function as $M : \mathcal{X} \to \{0,1\}^r$. The masking function splits the image in two parts:

$$\mathbf{x} = [\mathbf{x}_v; \mathbf{x}_h], \tag{47}$$

where

$$\begin{cases} \mathbf{x}_v = \{i \in cX | M(\mathbf{x})_i = 1\} \\ \mathbf{x}_h = \{i \in cX | M(\mathbf{x})_i = 0\} \end{cases} \tag{48}$$

We introduce a vector variable $\mathbf{V}$, the same shape as x_h and define the zero image parts as:

$$\mathbf{x}_0 = [\mathbf{o}; \mathbf{V}], \tag{49}$$

where $\mathbf{0}, \mathbf{V}$ replace the visible and hidden parts $\mathbf{x}_v, \mathbf{x}_h$ of $\mathbf{x}$ respectively, where the former is a constant zero vector, the same size as $\mathbf{x}_v$. An intuitive idea for Zero Image Parts is to determine a set of values for the hidden pixels that correspond to the zero element of addition:

$$\begin{cases} f(\mathbf{x} + \mathbf{x}_0) = f(\mathbf{x}) \\ f(\mathbf{x}_0) = \mathbf{0}. \end{cases} \tag{50}$$

This implies that $\mathbf{x}_0$ serves as the zero element of addition for the model f. Adding it to the original image should leave the classification score unchanged. Importantly, only the non-zero values of $\mathbf{x}_0$ should be altered, preserving the visible parts of $\mathbf{x}$ when $\mathbf{x}_0$ is added. Upon identifying the *neutral element*, the final step of the algorithm involves returning a new vector, $\mathbf{x}'$, which can be constructed from $\mathbf{x}$ and $\mathbf{x}_0$ as:

$$\mathbf{x}' = [\mathbf{x}_v; \mathbf{V}]. \tag{51}$$

Conceptually, for two different points that equally activate a model, their attributions will inevitably be distinct. That is because, as we mentioned earlier, a robust attribution should take the input data values into account, rather than blindly pointing to some "default" regions. After having said that, we also mention that since the model is a mapping from higher to lower-dimensional spaces, there are different points that activate the model in a similar fashion for various reasons.

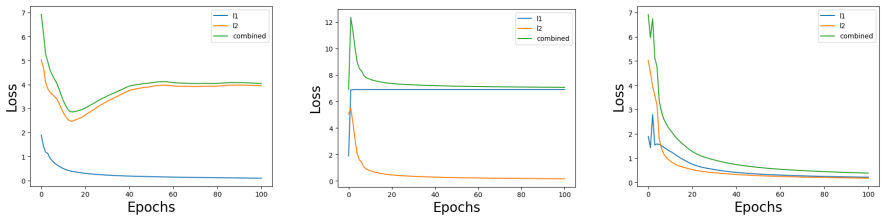
The goal of Equation 50(1) is to discover a new point (which is identical to the original only on the visible parts) is to discover a new point that activates the model in a similar way (though not for the same reasons). Such points might be abundant. We need to select the one that adds the least amount of information to the original image, when replacing the hidden part. This is what 50(2) tries to achieve.

9.4 The ZIP algorithm

Our algorithm is named **ZIP** and we outline it below. We consider f to represent the model, x as the input image and M as a mask with values in $\{0, 1\}$ (0 is "hide"). Additionally, N is the number of steps and ϵ the learning rate:

Algorithm 2 The ZIP algorithm

Require: f, x, M, N, ϵ
  $x' \leftarrow rand(size(x))$
  for $i \leftarrow 1, ..., N$ **do**
    $y \leftarrow f(x)$
    $y' \leftarrow f(x + (1 - M)x')$
    $y_0 = f((1 - M)x')$
    $l_1 \leftarrow H(y - y')$
    $l_2 \leftarrow H(y_0)$
    $l \leftarrow l_1 + l_2$
    $x' \leftarrow x' - \epsilon \nabla l$
  end for
  $x_0 \leftarrow Mx + (1 - M)x'$
  return x_0



(a) L_1 Loss (b) L_2 Loss (c) L Loss

Figure 24: The evolution of the two losses, when considered alone for l_1 and l_2 (a) and b) respectively) and together (c). For plots a) and b) where the loss is applied to only one of l_1, l_2, the other loss is also plotted at the same graph, in order to study the effect of the one on another. It seems as l_1 leads to the drop of l_2 as well, but the opposite does not hold. In any case, when both losses are considered, the overall loss drops smoothly towards zero.

9.5 Visual examples and performance

In this subsection, we assess the performance of our algorithm by visually inspecting the loss functions and the reconstructed images. The first and most crucial test involves evaluating the evolution of losses. In all the images we visually inspected, the losses behaved as illustrated in Figure 24c. To further examine their behavior, we considered the losses independently to determine if their combination led to a significant increase in both of them. This phenomenon indeed occurred, indicating that the losses are not independent and may compete with each other, although this effect is limited in scale.

Based on multiple experiments with different images, we can conclude the following:

- the l_1 loss alone diminishes to a point where the maximum difference of probabilities between the two images falls within the range of [0.0015, 0.003];
- the l_2 loss alone diminishes to a point where the probability of the maximum class falls within the range of [0.0025, 0.004];
- When both losses are considered, these scores fall within the range of [0.0025, 0.004] for the first loss and [0.003, 0.0045] for the second.

Next, we examine the resulting reconstructed image and its appearance. Similar to many optimization algorithms, the hidden parts are filled with random noise, which can lead to out-of-distribution (OoD) results. An example of this can be seen in Figure 25.

9.6 A discussion on the algorithm

This represents an initial attempt to address the challenge of zeroing image parts, aligning partially with the arguments and lemmas discussed earlier. However, its robustness may be called into question. That is because, the optimized tensor uses black for the visible part, while the reconstructed image is Out-of-Distribution. They are further developed below.

(a) Initial Image (b) Masked Image (c) ZIP Image

Figure 25: The results of the ZIP algorithm. The initial image (a) gets masked (b) and then reconstructed according to the Zero Parts criteria (c). It is clear that the reconstructed image is OoD.

1. Zeroing Information in the Added Part

The added image is optimized only for the part that matches the hidden image. The other part is deliberately selected to be black because, when added to the original image, these parts do not affect the visible portions. Nevertheless, the second criterion states that the added image should contain no information, maximizing the model's entropy. As previous work suggests, black images may contain rich information.

2. Out-of-Distribution Data

This method fails to satisfy the OoD criterion, resulting in the creation of images that appear *unnatural* to the model. In such cases, the model's predictions cannot be trusted.

To address the later issue, a more suitable solution can be found, as described analytically in sections 6.4 and 7. We also outline an alternative solution in the following section, which we believe addresses the former problem.

9.7 Towards a robust Occlusion

The challenge of optimizing only the hidden parts while using a baseline for the visible parts of $\mathbf{x}_0$ (considered to be zero) reveals a significant limitation of $\mathbf{x}_0$. To address this, it is essential to include the visible parts in the optimization process. Let's consider $\mathbf{x} = [\mathbf{x}_v; \mathbf{x}_h]$ and $\mathbf{x}_0 = [\mathbf{x}_{0_v}; \mathbf{x}_{0_h}]$, where the indices v, h symbolize the visible and hidden parts of an image, respectively, according to the mask M. The ";" symbol denotes the vector concatenation operation. We propose the following set of optimization criteria:

$$\begin{cases} f([\mathbf{x}_v; \mathbf{x}_{0_h}]) + f([\mathbf{x}_{0_v}; \mathbf{x}_h]) = f(\mathbf{x}) \\ f(\mathbf{x}_0) = \mathbf{0}. \end{cases} \tag{52}$$

In this case, the image is split into two parts: the visible and the hidden parts. Each part is complemented by the respective hidden and visible part of the zero image to construct two images of the same size as the original one. Now x_0 operates in a manner that encompasses both its visible and hidden parts and combines them to achieve two objectives:

1. The combination of these parts with x decomposes its visible and hidden components without removing any information from it (as indicated in Equation 52(1))

2. It adds the least amount of information to the model (as expressed in Equation 52(2))

Consequently, the image with its hidden parts optimally concealed will be:

$$\mathbf{x}' = [\mathbf{x}_v; \mathbf{x}_{0_h}]. \tag{53}$$

Chapter 10

The MAE-ZIP algorithm

We have now reached the final stage of our methodology. The accumulated evidence from our previous observations and conclusions suggests that the solution to the problem of concealing image information resides at the intersection of criteria-based optimization algorithms and generative models. The challenge is to determine how to effectively combine them into a method that can accomplish our task. This chapter sets out to address this question.

10.1 The pipeline

In this section, we introduce our innovative approach, the MAE-ZIP algorithm. It effectively tackles both the issue of introduced bias and out-of-distribution (OoD) artifacts by associating key attributes and criteria (as outlined in Section 9) with a Generative Model, specifically a Masked Autoencoder [106]. Before delving into the algorithm, we must first redefine the criteria to align them with our revised objectives.

Lemma 5 (Criteria)**.** *For robust masking of a subset of features, the following criteria are indispensable:*

- ***Foreground stability.*** *The filling process must not alter the contributions of the visible features.*
- ***Background neutrality.*** *The filling should introduce no additional information to the model.*
- ***Minimal impact.*** *The filling must fulfill the two preceding criteria while remaining in-distribution and having minimal impact.*

To initiate this process, we consider a masking function denoted as $M : \mathcal{X} \to \{0,1\}^r$, where $\mathcal{X}$ represents the image space and r denotes its dimensions. Consequently, the masked image, designated as $x \odot M(x)$ is derived from the Hadamard product of the original image and the mask M.

The Masked Autoencoder (MAE) comprises both an encoder

$$g^e : \mathcal{X} \times \{0,1\}^r \to \mathcal{X}^{r^{g-h} \times l^m}, \tag{54}$$

and a decoder

$$g^d : \mathcal{X}^{r^{g-h} \times l^g} \times \mathcal{X}^{r^h \times l^g} \to \mathcal{X}. \tag{55}$$

Here, r^g corresponds to the spatial resolution, while l^g refers to the dimension of the latent tensor representation.

The encoder receives an input x and a random masking function M. This function hides parts of x, leading to a loss of $r^h \times l^g$ spacial dimensions. A learnable mask token $m_h \in \mathbb{R}^{r^h \times l^g}$ is added for the decoder $g^d(g^e, m) = g^d_m(g^e)$. We note that m is originally shaped in $\mathbb{R}^{l^g}$ and it is then augmented to match the size of the hidden parts. It represents the model's knowledge of the image space.

Our *MAE-ZIP* algorithm extends MAE by incorporating a tensor $z \in \mathcal{R}^{r^m \times l^g}$ in the latent space and optimizes z at test time to accommodate for the Zero Parts properties, as defined in 9. When given input $x \in \mathcal{X}$ and mask $M \in \{0,1\}^r$, we modify MAE by perturbing the mask token m by z, to generate the *reconstructed image*, which is defined as:

$$x' = g^d(g^e(x, M), m_h + z). \tag{56}$$

Additionally, without involving the encoder, we add the hidden part of z to the mask token m_h and decode into the *zero image*, designated as:

$$x^0 = g^d(m_{g-h}, m_h + z). \tag{57}$$

Here m_{g-h} represents the mask token being populated to the size of $\mathbb{R}^{r^{g-h} \times l^g}$.

Both x' and x^0 pass through the classifier f. The output $f(x')$ is used, along with $f(x)$, to account for the first property, while $f(x^0)$ is used to address the second property. This is achieved by applying appropriate loss functions to the two outputs. The optimization of variable z involves back-propagating the loss through the classifier f and the decoder g^d, taking into account the gradients from both paths through x' (56) and x^0 (57).

Each of the loss terms is multiplied by a hypermarameter to calibrate them and ensure equal consideration. This detail is not included in the following description of the losses.

1. **Foreground stability.** According to the first property, the foreground parts of the original image x and reconstructed image x' should have the same contribution to the prediction of f, as indicated by their attribution. This is accomplished using the information from the deep activations of x and x'. In particular, we decompose function f as $f^c \circ f^e$, where $f^e : \mathcal{X} \to \mathbb{R}^{r^l \times d^l}$ is an encoder and $f^c : \mathbb{R}^{r^l \times d^l} \to \mathbb{R}^k$ is a classifier. Here, r^l is the spatial resolution and d^l the dimension of the intermediate tensor representation. With these definitions in place, the *foreground loss*
$$L_F = \|f^e(x') - M^{lat}(f^e(x))\|^2, \tag{58}$$
brings the foreground features of x and x' close to each other, where $M^{lat} \in \{0,1\}^{r^l}$ augments M to fit the spatial resolution r^l.

2. **Background neutrality.** According to the second property, when decoded, the generated background part should not contribute to the prediction of f. as reflected by having zero attribution. Since the ground truth class is unknown, the *background loss*
$$L_B = -H(f(x^0)). \tag{59}$$

maximizes the entropy of $f(x^0)$ such that the model is completely indecisive between classes for the zero image x^0.

3. **Minimal impact.** The last property constraints variable z to serve a minimum perturbation on the mask token m as needed for the two aforementioned properties, thus having minimal impact on the MAE reconstruction. We thus **initialize** z **as zero** and define

$$L_z = \|z\|^2, \tag{60}$$

In 11, the values of the hyperparameters, specifically the weight of the losses, are fixed, ensuring that they all contribute to the result and yield highly accurate outcomes based on various metrics.

Chapter 11

Experimental setup

This chapter provides comprehensive details regarding the experiments conducted throughout this research thesis. For all the experiments, we consistently employed the same models and datasets. However, for MAE-ZIP, we extended the experimental setup to encompass multiple Attribution methods and specially designed Evaluation tools, which will be explained in detail in this chapter. We commence with an explanation of the model architectures and specific insights into our customized models. Subsequently, we delve into the dataset utilized and elucidate the different metrics developed to evaluate our results.

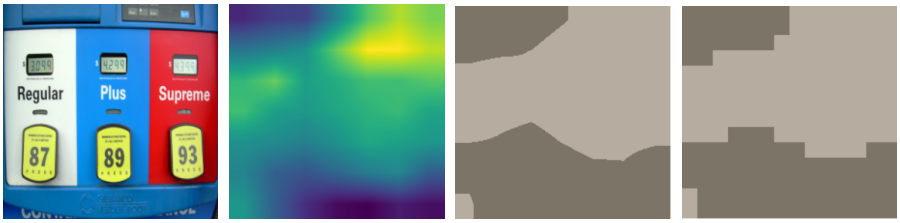
11.1 Model architecture Implementation details

The architecture employed in our experiments comprises three distinct components, collectively forming a pipeline. Given an input image, we first apply a mask. The visible section of this image then proceeds through the generative model, responsible for reconstructing both the hidden image and the zero image. Following this, we deploy a baseline model designed for pattern recognition within the two images. The final component involves the application of loss functions, which enable the backpropagation of losses from the classifier to the generative model, ultimately influencing the added variables. Below, we provide an overview of the two models utilized in this setup. The losses are described in Section 10.

11.1.1 Generative Model

As previously mentioned, our generative model is based on Masked Autoencoders. This architecture is originally intended to randomly obscure a portion of image patches and efficiently reconstruct them. However, our application of MAE-ZIP necessitated a modification of its design to serve our specific objectives.

Masking. In contrast to MAE, where image patches are randomly hidden, MAE-ZIP demands customized masks for constructing Zero Parts tailored to any given image and mask. To achieve this, we had to adapt MAE to allow for custom masking. In MAE, masking occurs in batches of size 16x16, resulting in 14x14 = 196 patches. To define the mask within this space, we downsample it to 14x14 using a mean kernel of size 16x16 and a stride of 16. This step computes the mean value for each patch, thus "patchifying" the mask. Subsequently, we apply a threshold $t \in [0, 1]$ to binarize the mask, hiding each patch if $t \times 16 \times 16$ of its pixels were hidden, making it visible otherwise. This operation's outcome is illustrated in Figure 26.



(a) Initial Image (b) Image attribution (c) Mask (d) Latent Mask

Figure 26: The process of downsampling the mask. For an initial image (a), we calculate its attribution (b) according to a particular Attribution method, and derive a mask from it (c) by keeping a percentage of the most important features. The mask is then "patched" and downsampled according to MAE. It is then upsampled to form the latent mask (d).

The selection of the mask can now be arbitrary. However, this technique also allows for evaluating various attribution methods. In such cases, the choice of the masking function M can be selected as follows:

$$M(x) = x \odot \mathbb{1}[R(x) \geq \overline{(R(x))}]. \tag{61}$$

Here $\odot$ represents the Hadamart product, $\mathbb{1}$ the indicator function and $R : \mathcal{X} \to \mathcal{X}$ a particular attribution method. Different functions could be applied instead of the mean function, in order for the mask M to distinguish between the important and unimportant features that a suggests. Although, a function that chooses whole image parts and not sparse pixels might be required.

Decoder. As detailed in the MAE-ZIP method, we introduced variables with the *mask embedding* and incorporated them as hidden variables for the Decoder to reconstruct. The loss functions play a crucial role in ensuring that these added variables gradually converge toward Zero Information. In scenarios where the second loss requires these added variables to be entirely devoid of any information, no input from the encoder is available. Therefore, we designed a slightly customized decoder that reintroduces the mask token with repeated dimensions, matching the size of the encoded visible parts.

11.1.2 Baseline Model

To serve as the baseline classifier in our pipeline, we adopted a Residual Neural Network (ResNet) [34].This model constitutes the final component of our pipeline, responsible for making decisions based on input images and backpropagating losses to the latent features. ResNets represent a class of deep neural network architectures developed in response to the vanishing gradient problem [9] often encountered when training very deep neural networks [93, 84]. Introduced in 2015, ResNets have since emerged as a fundamental building block in numerous state-of-the-art deep learning models.

The central idea behind ResNets involves the incorporation of residual blocks, which feature *shortcut connections* (also known as *skip connections*) allowing for the bypassing of one or more layers within the network. These shortcut connections facilitate the flow of gradients during backpropagation, effectively mitigating the vanishing gradient issue. In a residual block, the input

to a layer is combined with the output of one or more subsequent layers, thereby reintroducing the information of previous blocks unspoiled. This architectural innovation simplifies the network's learning process and enables the fine-tuning of the desired mapping.

By stacking multiple residual blocks, ResNets have the capacity to train extremely deep networks, often exceeding 100 layers, without suffering from performance degradation. The introduction of shortcut connections has significantly advanced deep learning architectures, addressing a fundamental challenge in artificial intelligence. This architectural approach has also been widely adopted in **Transformer** architectures [100], facilitating the flow of gradients during training. Deep networks were able to train efficiently, leading to improved performance on a wide range of tasks in computer vision, natural language processing, and other domains.

11.2 Dataset

The dataset we selected is the all-famous ImageNet dataset. It is a widely used benchmark dataset in the field of artificial intelligence, specifically in the domain of computer vision. It was created by researchers at Stanford University and contains a vast collection of labeled images, organized into thousands of categories. The dataset was introduced in 2009 and has since played a crucial role in the development and evaluation of various computer vision algorithms, especially for tasks like image classification, object detection, and image segmentation.

ImageNet consists of over a million images, with each image belonging to one of approximately 20,000 categories. The dataset covers a wide range of objects, scenes, and concepts, making it a comprehensive resource for training and testing computer vision models. ImageNet's scale and diversity have made it a standard benchmark for assessing the performance of image recognition systems. The annual ImageNet Large Scale Visual Recognition Challenge (ILSVRC) has further popularized the dataset by setting competitions to encourage the development of more accurate image classification algorithms. As a result, many state-of-the-art deep learning models, including CNNs, have been trained and evaluated on the ImageNet dataset, leading to significant advancements in the field of computer vision.

When conducting experiments on ImageNet, a common practice is to use the *Evaluation* dataset, which comprises 50.000 images. The ResNet model is trained on 1.000 of those classes. Nevertheless, due to limited access to computational resources, the experiments were performed on a randomly selected portion of the dataset, containing five images from each class.

11.3 Zero Information metrics

This section introduces the evaluation metrics we developed to assess the effectiveness of our approach for concealing information while constructing an image that conforms to the target distribution. The results of MAE-ZIP based on these metrics will be presented in Chapter 12.

11.3.1 Attribution Mask

We introduce the **AttMask** metric, which aids in comparing our method with other filling techniques. For a given attribution method R, it computes mean value of the difference in the attribution of the reconstructed image x_{rec} and the masked attribution of the image $M(R(x))$:

$$\textbf{AttMask} = \|(M(R(x)) - R(x_{rec}))\|_2^2 \tag{62}$$

This score provides a rough estimate of the effectiveness of our algorithm. Our primary objective is to fill the hidden portions of the image in a way that preserves the attribution of the foreground while reducing that of the background to zero. The metric accurately measures the deviation from the correct attribution R. However, as this true attribution is not known, we approximate it using various attribution methods. The methods we employ are **GradCAM**[80], **GradCAM++** [14], **XGradCAM** [27] and **LayerCAM** [45].

11.3.2 Accuracy Preservation

The purpose of this score is to measure how closely the reconstructed images align with the model's data distribution. To achieve this, we fill the hidden parts of the images using the aforementioned methods and measure the drop in the model's accuracy. For the dataset D, we define:

$$\textbf{AP} = \frac{\sum_{d \in D} \mathbb{1}[\hat{y}_x = \hat{y}_{x_{rec}}]}{|D|}. \tag{63}$$

We define $\hat{y}_x = argmax\{f(x) = c\}$ as the class with the highest probability for f and input x. The intuition behind this approach, is that if the images are OoD for the model, the model's performance will significantly decrease. While a drop in accuracy can be attributed to the concealment of information, this metric provides a rough estimate of how closely a filling method aligns with the data distribution when applied to large datasets.

11.4 Baseline methods

We compare the ZIP algorithm with different heuristic techniques, as mentioned in 6.2. More specifically, we define as:

- **black**; filling the background with black color.
- **blur_x**; filling the background with blurry versions of the input image where **x** is the size of the blurry filter.
- **rand_noise**; replacing the background with random noise.

Chapter 12

Results

This chapter presents the results of the MAE-ZIP method, according to the Evaluation metrics developed in Section 11.3. Yet, we first need to ensure that the algorithm functions as expected, meaning that the optimization leads to a decrease in all the losses. As mentioned in Chapter 10, each loss is derived from a different mathematical formula and accompanied by a weighting term, the range of values may vary. It is essential to calibrate these weights to make the total loss consider them all equally important.

12.1 Losses

For this purpose, a set of experiments is conducted to select a particular weight setup for the losses. The aim is to find values that balance all three losses. It is discovered that a setup leading to the decrease of all values is $\{w_1 = 1 \times 10^1, w_2 = 1 \times 10^{-2}, w_3 = 1 \times 10^{-4}\}$ and this set is referred to as *standard*. For this particular set, the losses are plotted in Figure 27. In what follows, the first two weights are considered fixed, since they adjust the first two losses in a way that lead to a combined drop of both. The third weight is treated as a hyperparamenter, and an ablation study is conducted to find its optimal values in the following section.

Loss

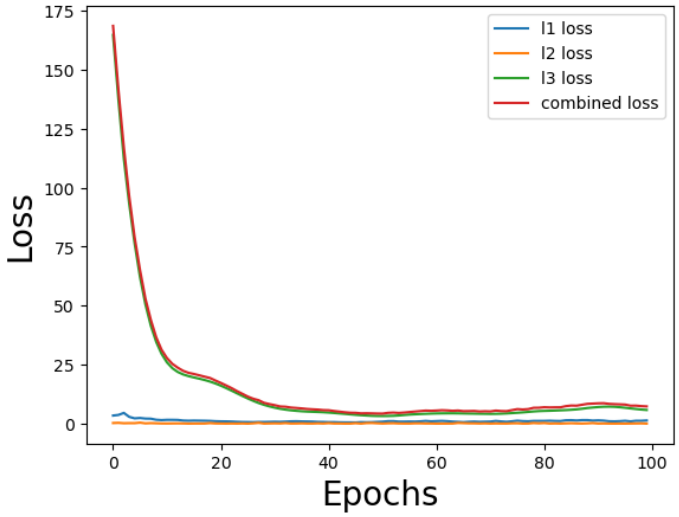


Figure 27: The plot of the losses when all combined together, for the *standard* set of hyperparameters.

Figure 28: displays the original image along with the reconstructed images for two different setups in the first row, namely the standard and one with a highly biased to the third loss, along with their corresponding GradCAM attributions in the second row.

Additionally, visual examples for the MAE-ZIP method are provided in Figure 28. It can be observed that the MAE-ZIP image introduces various artifacts for a large norm of the optimized values. By increasing the weight of the third loss, these artifacts disappear, and the reconstructed image more closely resembles the custom MAE reconstruction. Further fine-tuning of the weights for the losses is expected. However, it is anticipated that the reconstructed image may appear slightly "patchier" in areas of the hidden part, where MAE introduces some cues of information. In the example in Figure 28, the algorithm erases the pants of the person and perturbs the tail of the fish to obscure the information they introduced.

12.2 Metrics

In this section, an ablation study is performed for different values of the third variable, which is considered as a hyperparameter in this context. The study aims to compare the method with baseline filling techniques, as described in Section 11.4. To assess the different setups and methods, the evaluation metrics designed in Section 11.3, are used with various weight configurations for w_3.

The results of the metric can be found in Table 1. The ablation study reveals that ZIP-MAE performs well across different hyperparameter configurations. However, the setup that yields the best scores is selected, which consists of $w_1 = 1 \times 10^1, w_2 = 1 \times 10^0, w_3 = 1 \times 10^{-4}$. As mentioned before, this is considered the standart setup.

Following the evaluation of different weight configurations and identifying the best among them, a comparison is made with heuristic techniques under

Method	w_3	GradCAM		GradCAM++		XGradCAM		LayerCAM	
		AM↓	AP↑	AM↓	AP↑	AM↓	AP↑	AM↓	AP↑
MAE		**0.117**	0.714	**0.120**	0.717	**0.116**	0.717	0.120	0.717
MAE-ZIP	0	0.119	**0.755**	0.121	**0.763**	0.119	0.756	0.121	**0.761**
MAE-ZIP	1×10^{-4}	0.119	**0.755**	0.121	0.760	0.121	**0.763**	**0.119**	0.754
MAE-ZIP	1×10^{-2}	0.119	0.745	0.121	0.756	0.119	0.739	0.121	0.756
MAE-ZIP	1×10^{0}	0.119	0.743	0.121	0.752	**0.118**	0.738	0.121	0.753
MAE-ZIP	1×10^{2}	0.119	0.743	0.121	0.754	0.119	0.733	0.121	0.753

Table 1: The effect of loss weights on MAE-ZIP performance, on 5K images on Imagenet validation set. AM: Attribution Mask(62); AM: Accuracy Preservation (63). The 95% confidence interval is 0.004 in general.

the same metrics. The results are presented in Table 2 with some key observations:

Method	GradCAM		GradCAM++		XGradCAM		LayerCAM	
	AM↓	AP↑	AM↓	AP↑	AM↓	AP↑	AM↓	AP↑
Black	0.231	0.049	0.206	0.053	0.230	0.047	-	0.057
Blurry (20)	0.160	0.150	0.155	0.155	0.160	0.149	-	0.157
Blurry (75)	0.195	0.085	0.180	0.090	0.195	0.083	-	0.090
Random noise	0.162	0.001	0.091	0.012	0.162	0.001	-	0.000
MAE	**0.117**	0.714	**0.120**	0.717	**0.116**	0.717	0.120	0.717
MAE-ZIP	0.119	**0.755**	0.121	**0.760**	0.121	**0.763**	**0.119**	**0.754**

Table 2: The results for the application of **AttMask** (AM) and **AD** for the different filling methods, compared to the best MAE-ZIP configuration $w_3 = 1 \times 10^{-4}$. The samples we used for calculating the table were 5K images from the validation set of Imagenet. Whenever w_1 and w_2 are not defined, they are assumed to be equal to 1. For the different experiments at the **AttMask** score, the length of the 95% confidence interval for each method ranges at around 0.002-0.005.

- Heuristic techniques lead to OoD data, as evidenced by the decline in accuracy preservation for these methods.
- Small blurring is a potential candidate among heuristic techniques for filling an image part with no information, although the Attribution Preservation (AP) score is relatively low.
- MAE, when used as a standalone method, proves to be an excellent choice for information concealment. It outperforms heuristic techniques and provides stable results.
- MAE-ZIP, compared to MAE, does not significantly improve Attribution Mask (AM) but increases the Attribution Accuracy of MAE.

We suggest that the reason MAE seems to perform slightly better than different MAE-ZIP configurations may be related to the choice of Class Activation Mapping (CAM) methods. CAM methods might not capture fine-grained details that MAE-ZIP reconstructs. However, when the Attribution Accuracy metric is combined with the AP criterion, the optimization's effectiveness becomes clearer. With an increase in AP, the reconstructed image not only remains in distribution but also constructs a valuable reconstruction for the model. The algorithm might be able to isolate background information from the foreground and zero the contribution of the former, enabling better recognition of the object of interest by the model. Additionally, it is suggested that the algorithm does not add relevant information to the image, as this would

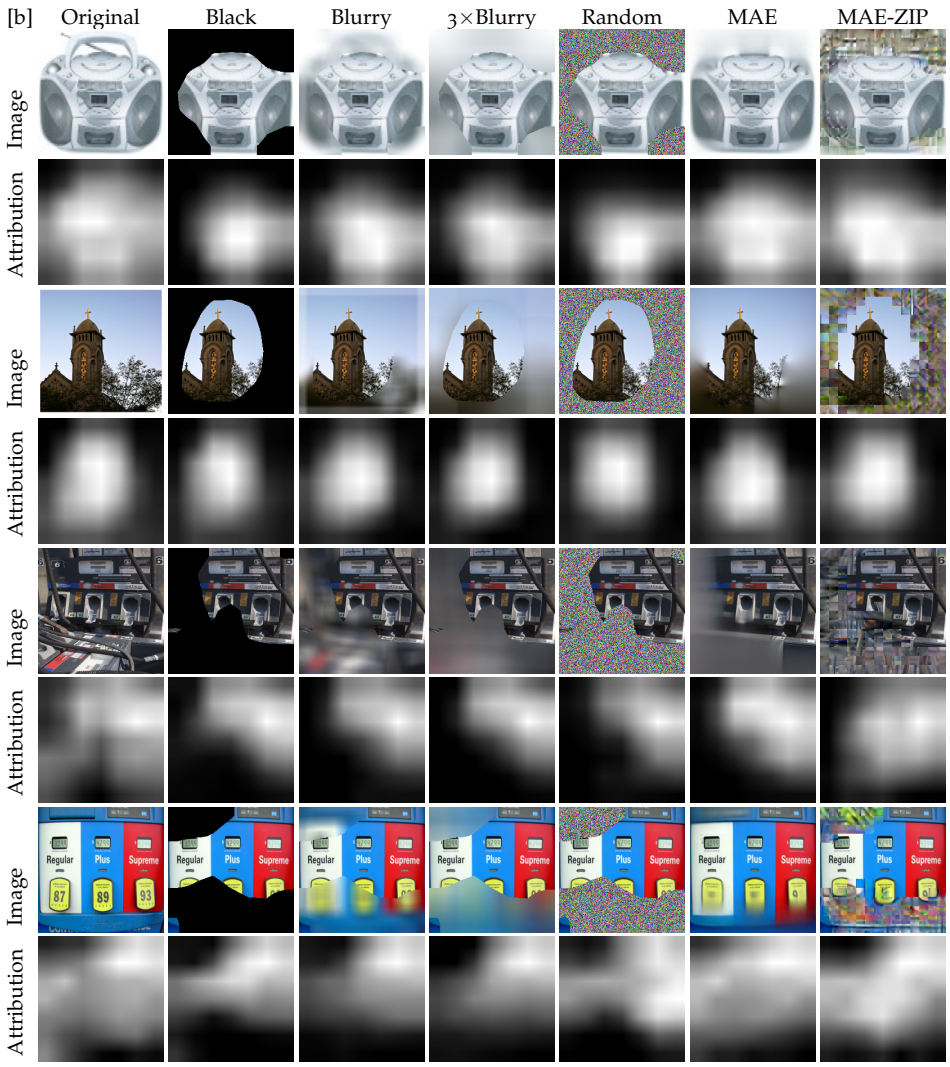


Figure 29: A visual comparison of the examined filling techniques.

also increase the value of AM significantly, which is not observed.

The presence of bias in different methods may affect the results, especially in CAM methods that work with saliency maps. Yet, overall, the algorithm is deemed to be an effective first step towards hiding information from image parts.

12.3 Visual comparison

For the standard set of hyperparameters, we provide multiple visual examples for different heuristic filling techniques, as compared to MAE and MAE-ZIP. The results can be seen in Figure 29.

Part V

FINALE

Chapter 13

Conclusion

As this research thesis concludes, it has explored numerous concepts and ideas, providing a direction for addressing the challenge of Zero Information. It remains open to further evaluation and exploration. More robust criteria designs are likely to emerge as different paths are explored. Yet, this research has pushed the boundaries of our understanding of model functioning and model attribution. The following three sections present the findings, limitations, and potential future work arising from this thesis.

13.1 High-level findings

This section summarises briefly the most significant high-level findings of this thesis.

Interconnectedness of Feature Contribution and Feature Attribution

The research has highlighted a strong connection between the concepts of feature contribution and feature attribution. Disrupting various pathways that lead to a model's output affects all features that contribute to that output through those pathways. This observation led to the **Attribution Shift** problem, which suggests that altering a portion of the image might result in changes to the attribution of all features. Consequently, all Attribution methods, Evaluation metrics, and Criteria based on Occlusion suffer from this issue. The extent of this problem cannot be directly measured since the initial attributions of features are unknown, but it can be indirectly estimated through approximation methods using different measures. Yet, due to the complex architecture of deep neural networks and the intricate interdependencies between features across layers, this issue should not be dismissed as negligible.

Zero Information Properties and Heuristic Techniques

The research argued that points with Zero Information properties, including regions or entire images devoid of information, cannot be achieved through heuristic techniques alone. Instead, they can be revealed through the careful consideration of the model's parameters and input features, guided by well-defined criteria that enforce the absence of information. These criteria should be translated into loss functions, considering the model's activations at the last or intermediate layers. The design and formulation of these criteria may not be immediately evident. Nevertheless, the criteria applied in the algorithms have proven effective -through visual inspection- in concealing

information. In the case of the ZIPO algorithm, it succeeded in hiding the target object in many instances (although it was susceptible to attacks). In MAE-ZIP, the algorithm subtly altered any information introduced by the Masked Autoencoder.

Susceptibility of Generative Models to Perturbations

The research has shown that even generative models are vulnerable to specific types of noise tailored to their characteristics. In the case of Masked Autoencoders, rather than Gaussian noise, the output appears to be more distinct, yet the resulting image may exhibit a "patchy" quality. Although individual patches may have clear shapes, their combination may not appear natural. This suggests that images with Zero Information properties may need to slightly relax the In-Distribution constraint in order for the optimization to succeed.

13.2 Limitations

This section expresses the different limitations of the algorithms developed in this thesis.

Algorithmic Biases and Deviation from Original Goals

It is evident that the proposed loss functions may introduce unintended biases to the algorithms and, in some cases, result in a slight deviation from the original objective. For instance, the ZIP algorithm aims to conceal elements in the image associated with different categories. However, the Zero addition element might not function as a local Zero Information Part, perturbing only those features with irrelevant information. It may not behave as expected when considered independently, without the hidden part of the original image. Addressing this issue, the criteria introduced in Section 9.7 attempt to mitigate these biases. However, these criteria cannot be readily employed alongside the Masked Autoencoder (MAE), unless the generative model undergoes further modification. This is because MAE operates on the premise that only one image part is visible, leaving the other part unconsidered. Concatenating the two reconstructed parts for the zero element might appear unnatural and out-of-distribution (OoD).

Application of Mask in Patched Image Space

Owing to the architectural design of MAE, which initially decomposes the image into patches and maps each to a lower-dimensional representation space, the mask cannot be directly applied to the original image. Instead, it necessitates downsampling the mask to the latent space of the patches. This means that a slightly different mask is applied to the problem, resembling the original but appearing more "patchified," characterized by straight lines rather than curves.

Bias Susceptibility in Effectiveness Criteria

The criteria proposed for assessing the effectiveness of different filling techniques may be susceptible to biases. For the **Attribution Mask**, the utilization

of an Attribution method to calculate two Attribution maps for comparison deviates from the original goal. As discussed in Chapter 12, Class Activation Mapping (CAM) methods may introduce strong biases towards the hidden parts. On the other hand, when considered in isolation, **Accuracy Preservasion** is primarily linked to the OoD criterion. In other contexts, it needs to be combined with another metric to ensure the fulfillment of the Zero Information criterion.

13.3 Future work

Building upon the identified limitations, we offer the following suggestions for future work:

1. **Enhanced Loss Design**. There is a need to refine the loss functions to align optimization results more closely with the desired criteria. Optimization algorithms often introduce misalignment due to constraints limiting their freedom. For the ZIPO algorithm, criteria should focus solely on the class of interest while preserving information related to other classes. Additionally, it would be valuable to test the ZIPO algorithm against Integrated Gradients. In the case of ZIP, we have proposed a more robust direction, but realizing better practical results and achieving alignment remain challenges, especially in integration with MAE.

2. **Exploration of Alternative Generative Models**. Instead of MAE, exploring the use of different generative models that can reconstruct hidden parts based on diverse criteria is an avenue for further investigation. Diffusion models, for instance, could be suitable candidates as they can reconstruct hidden objects of various sizes without decomposing the image into patches. This approach would enable the resulting mask to more accurately match the true mask, rather than "normalizing" it to suit the needs of MAE.

3. **Development of Robust Evaluation Metrics**. It is essential to design improved metrics for evaluating the effectiveness of Zero Information algorithms. As mentioned earlier, the criteria we proposed may be susceptible to biases and may not be sufficiently robust. Creating alternative metrics that are less prone to such biases is a crucial area of future research.

4. **Application and Expansive Usage**. After optimizing the algorithms and addressing the criteria, MAE-ZIP can be applied to numerous practical applications. This approach facilitates understanding the individual contribution of different image parts to the model's predictions. By applying MAE-ZIP to conceal specific parts while leaving others unchanged, we can quantify the impact of each part on the model's decisions. While this may not answer the question of what caused the model to activate in a particular manner, it does provide insights into the extent to which a specific part contributes to the model's decisions. For more comprehensive insights into causality, combining the algorithm with other Occlusion methods is necessary. Furthermore, MAE-ZIP can effectively complement Evaluation metrics and Criteria based on Occlusion, broadening its potential applications.

13.4 Conclusion

In conclusion, this work does not present a definitive solution to the concept of Zero Information. It is our hope that this thesis will contribute to the field of Explainable Artificial Intelligence, prompting further exploration of this concept by other researchers.

A code repository will be made available soon, complete with clear usage instructions. We invite researchers to explore our methods, share their findings with us, and develop new ideas.

Bibliography

[1] Understanding the potential applications of artificial intelligence in agriculture sector. Advanced Agrochem, 2(1):15–30, 2023. ISSN 2773-2371. doi: https://doi.org/10.1016/j.aac.2022.10.001. URL https://www.sciencedirect.com/science/article/pii/S277323712200020X.

[2] Julius Adebayo, Justin Gilmer, Michael Muelly, Ian Goodfellow, Moritz Hardt, and Been Kim. Sanity checks for saliency maps, 2020.

[3] Chirag Agarwal and Anh Nguyen. Explaining image classifiers by removing input features using generative models, 2020.

[4] Ameen Ali, Thomas Schnake, Oliver Eberle, Grégoire Montavon, Klaus-Robert Müller, and Lior Wolf. XAI for transformers: Better explanations through conservative propagation. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato, editors, Proceedings of the 39th International Conference on Machine Learning, volume 162 of Proceedings of Machine Learning Research, pages 435–451. PMLR, 17–23 Jul 2022. URL https://proceedings.mlr.press/v162/ali22a.html.

[5] Kamran Alipour, Aditya Lahiri, Ehsan Adeli, Babak Salimi, and Michael Pazzani. Explaining image classifiers using contrastive counterfactuals in generative latent spaces, 2022.

[6] Marco Ancona, Enea Ceolini, Cengiz Öztireli, and Markus Gross. Towards better understanding of gradient-based attribution methods for deep neural networks, 2018.

[7] David Balduzzi, Marcus Frean, Lennox Leary, JP Lewis, Kurt Wan-Duo Ma, and Brian McWilliams. The shattered gradients problem: If resnets are the answer, then what is the question?, 2018.

[8] Stefano Baruffaldi, Brigitte van Beuzekom, Hélène Dernis, Dietmar Harhoff, Nandan Rao, David Rosenfeld, and Mariagrazia Squicciarini. Identifying and measuring developments in artificial intelligence. 2020. doi: https://doi.org/https://doi.org/10.1787/5f65ff7e-en. URL https://www.oecd-ilibrary.org/content/paper/5f65ff7e-en.

[9] Sunitha Basodi, Chunyan Ji, Haiping Zhang, and Yi Pan. Gradient amplification: An efficient way to train deep neural networks. Big Data Mining and Analytics, 3(3):196–207, 2020. doi: 10.26599/BDMA.2020.9020004.

[10] Alexander Binder, Grégoire Montavon, Sebastian Lapuschkin, Klaus-Robert Müller, and Wojciech Samek. Layer-wise relevance propagation for neural networks with local renormalization layers. 04 2016. ISBN 978-3-319-44780-3. doi: 10.1007/978-3-319-44781-0_8.

[11] Akhilan Boopathy, Sijia Liu, Gaoyuan Zhang, Cynthia Liu, Pin-Yu Chen, Shiyu Chang, and Luca Daniel. Proper network interpretability helps adversarial robustness in classification, 2020.

[12] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers, 2021.

[13] Chun-Hao Chang, Elliot Creager, Anna Goldenberg, and David Duvenaud. Explaining image classifiers by counterfactual generation, 2019.

[14] Aditya Chattopadhay, Anirban Sarkar, Prantik Howlader, and Vineeth N Balasubramanian. Grad-CAM: Generalized gradient-based visual explanations for deep convolutional networks. In 2018 IEEE Winter Conference on Applications of Computer Vision (WACV). IEEE, mar 2018. doi: 10.1109/wacv.2018.00097.

[15] Hila Chefer, Shir Gur, and Lior Wolf. Transformer interpretability beyond attention visualization, 2021.

[16] Jianbo Chen, Le Song, Martin J. Wainwright, and Michael I. Jordan. Learning to explain: An information-theoretic perspective on model interpretation, 2018.

[17] Lijia Chen, Pingping Chen, and Zhijian Lin. Artificial intelligence in education: A review. IEEE Access, 8:75264–75278, 2020. doi: 10.1109/ACCESS.2020.2988510.

[18] Piotr Dabkowski and Yarin Gal. Real time image saliency for black box classifiers, 2017.

[19] Saurabh Desai and Harish G. Ramaswamy. Ablation-cam: Visual explanations for deep convolutional network via gradient-free localization. In 2020 IEEE Winter Conference on Applications of Computer Vision (WACV), pages 972–980, 2020. doi: 10.1109/WACV45572.2020.9093360.

[20] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale, 2021.

[21] Kevin Eykholt, Ivan Evtimov, Earlence Fernandes, Bo Li, Amir Rahmati, Chaowei Xiao, Atul Prakash, Tadayoshi Kohno, and Dawn Song. Robust physical-world attacks on deep learning visual classification. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 1625–1634, 2018.

[22] 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, 2021.

[23] 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, 2023.

[24] Ruth Fong, Mandela Patrick, and Andrea Vedaldi. Understanding deep networks via extremal perturbations and smooth masks, 2019.

[25] Ruth C. Fong and Andrea Vedaldi. Interpretable explanations of black boxes by meaningful perturbation. In 2017 IEEE International Conference on Computer Vision (ICCV). doi: 10.1109/iccv.2017.371.

[26] Ruth C. Fong and Andrea Vedaldi. Interpretable explanations of black boxes by meaningful perturbation. In 2017 IEEE International Conference on Computer Vision (ICCV). IEEE, oct 2017. doi: 10.1109/iccv.2017.371. URL https://doi.org/10.1109%2Ficcv.2017.371.

[27] Ruigang Fu, Qingyong Hu, Xiaohu Dong, Yulan Guo, Yinghui Gao, and Biao Li. Axiom-based grad-cam: Towards accurate visualization and explanation of cnns, 2020.

[28] Rezida Maratovna Galimova, Igor Vyacheslavovich Buzaev, Kireev Aydar Ramilevich, Lev Khadyevich Yuldybaev, Aigul Fazirovna Shaykhulova, and Jing-Ling Bao. Artificial intelligence-developments in medicine in the last two years. Chronic Diseases and Translational Medicine, 05(01):64–68, 2019. doi: 10.1016/j.cdtm.2018.11.004.

[29] Tristan Gomez, Thomas Fréour, and Harold Mouchère. Metrics for saliency map evaluation of deep learning explanation methods, 2022.

[30] Ian J. Goodfellow, David Warde-Farley, Mehdi Mirza, Aaron Courville, and Yoshua Bengio. Maxout networks, 2013.

[31] Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples, 2015.

[32] Peter Hase, Harry Xie, and Mohit Bansal. The out-of-distribution problem in explainability and search methods for feature importance explanations, 2021.

[33] Johannes Haug, Stefan Zürn, Peter El-Jiz, and Gjergji Kasneci. On baselines for local feature attributions, 2021.

[34] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition, 2015.

[35] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pages 15979–15988, 2022. doi: 10.1109/CVPR52688.2022.01553.

[36] Sara Hooker, Dumitru Erhan, Pieter-Jan Kindermans, and Been Kim. A benchmark for interpretability methods in deep neural networks, 2019.

[37] 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, 2021.

[38] Alexey Ignatiev, Nina Narodytska, and Joao Marques-Silva. Abduction-based explanations for machine learning models, 2018.

[39] Satoshi Iizuka, Edgar Simo-Serra, and Hiroshi Ishikawa. Globally and locally consistent image completion. 36(4), 2017. ISSN 0730-0301. doi: 10.1145/3072959.3073659. URL https://doi.org/10.1145/3072959.3073659.

[40] Turab Iqbal, Yin Cao, Qiuqiang Kong, Mark D. Plumbley, and Wenwu Wang. Learning with out-of-distribution data for audio classification, 2020.

[41] Cosimo Izzo, Aldo Lipani, Ramin Okhrati, and Francesca Medda. A baseline for shapley values in MLPs: from missingness to neutrality. In ESANN 2021 proceedings. Ciaco - i6doc.com, 2021. doi: 10.14428/esann/2021.es2021-18. URL https://doi.org/10.14428%2Fesann%2F2021.es2021-18.

[42] Sarthak Jain and Byron C. Wallace. Attention is not explanation, 2019.

[43] Dominik Janzing, Lenon Minorics, and Patrick Blöbaum. Feature relevance quantification in explainable ai: A causal problem, 2019.

[44] Neil Jethani, Mukund Sudarshan, Yindalon Aphinyanaphongs, and Rajesh Ranganath. Have we learned to explain?: How interpretability methods can learn to encode predictions in their interpretations, 2021.

[45] Peng-Tao Jiang, Chang-Bin Zhang, Qibin Hou, Ming-Ming Cheng, and Yunchao Wei. Layercam: Exploring hierarchical class activation maps for localization. IEEE Transactions on Image Processing, 30:5875–5888, 2021. URL https://api.semanticscholar.org/CorpusID:235610245.

[46] José Jiménez-Luna, Francesca Grisoni, Nils Weskamp, and Gisbert Schneider. Artificial intelligence in drug discovery: recent advances and future perspectives. Expert Opinion on Drug Discovery, 16(9): 949–959, 2021. doi: 10.1080/17460441.2021.1909567. URL https://doi.org/10.1080/17460441.2021.1909567. PMID: 33779453.

[47] Guy Katz, Clark Barrett, David Dill, Kyle Julian, and Mykel Kochenderfer. Reluplex: An efficient smt solver for verifying deep neural networks, 2017.

[48] Ashkan Khakzar, Pedram Khorsandi, Rozhin Nobahari, and Nassir Navab. Do explanations explain? model knows best, 2022.

[49] Been Kim, Martin Wattenberg, Justin Gilmer, Carrie Cai, James Wexler, Fernanda Viegas, and Rory Sayres. Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (tcav), 2018.

[50] Siwon Kim, Jihun Yi, Eunji Kim, and Sungroh Yoon. Interpretation of nlp models through input marginalization, 2020.

[51] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. Imagenet classification with deep convolutional neural networks. Commun. ACM, 60(6):84–90, may 2017. ISSN 0001-0782. doi: 10.1145/3065386. URL https://doi.org/10.1145/3065386.

[52] Sebastian Lapuschkin, Stephan Wäldchen, Alexander Binder, Grégoire Montavon, Wojciech Samek, and Klaus-Robert Müller. Unmasking clever hans predictors and assessing what machines really learn. Nature communications, 10(1):1–8, 2019.

[53] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel. Backpropagation Applied to Handwritten Zip Code Recognition. Neural Computation, 1(4):541–551, 12 1989. ISSN 0899-7667. doi: 10.1162/neco.1989.1.4.541. URL https://doi.org/10.1162/neco.1989.1.4.541.

[54] 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, 2019.

[55] Stan Lipovetsky and Michael Conklin. Analysis of regression in game theory approach. Applied Stochastic Models in Business and Industry, 17:319 – 330, 10 2001. doi: 10.1002/asmb.446.

[56] Zachary C. Lipton. The mythos of model interpretability, 2017.

[57] Jiashuo Liu, Zheyan Shen, Yue He, Xingxuan Zhang, Renzhe Xu, Han Yu, and Peng Cui. Towards out-of-distribution generalization: A survey, 2023.

[58] Jiaying Liu, Xiangjie Kong, Feng Xia, Xiaomei Bai, Lei Wang, Qing Qing, and Ivan Lee. Artificial intelligence in the 21st century. IEEE Access, 6:34403–34421, 2018. doi: 10.1109/ACCESS.2018.2819688.

[59] Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. A convnet for the 2020s, 2022.

[60] Scott Lundberg and Su-In Lee. A unified approach to interpreting model predictions, 2017.

[61] Scott Lundberg and Su-In Lee. A unified approach to interpreting model predictions, 2017. URL https://arxiv.org/abs/1705.07874.

[62] Frank J. Massey. The kolmogorov-smirnov test for goodness of fit. Journal of the American Statistical Association, 46(253):68–78, 1951. ISSN 01621459. URL http://www.jstor.org/stable/2280095.

[63] Grégoire Montavon, Sebastian Lapuschkin, Alexander Binder, Wojciech Samek, and Klaus-Robert Müller. Explaining nonlinear classification decisions with deep taylor decomposition. Pattern Recognition, 65: 211–222, may 2017. doi: 10.1016/j.patcog.2016.11.008. URL https://doi.org/10.1016%2Fj.patcog.2016.11.008.

[64] Grégoire Montavon, Sebastian Lapuschkin, Alexander Binder, Wojciech Samek, and Klaus-Robert Müller. Explaining nonlinear classification decisions with deep taylor decomposition. Pattern Recognition, 65: 211–222, 2017. ISSN 0031-3203. doi: https://doi.org/10.1016/j.patcog.2016.11.008.

[65] Mohammed Bany Muhammad and Mohammed Yeasin. Eigen-CAM: Class activation map using principal components. In 2020 International Joint Conference on Neural Networks (IJCNN). IEEE, jul 2020. doi: 10.1109/ijcnn48605.2020.9206626. URL https://doi.org/10.1109%2Fijcnn48605.2020.9206626.

[66] Anh Nguyen, Jason Yosinski, and Jeff Clune. Deep neural networks are easily fooled: High confidence predictions for unrecognizable images, 2015.

[67] Paul Novello, Thomas Fel, and David Vigouroux. Making sense of dependence: Efficient black-box explanations using dependence measure, 2022.

[68] Harikumar Pallathadka, Edwin Hernan Ramirez-Asis, Telmo Pablo Loli-Poma, Karthikeyan Kaliyaperumal, Randy Joy Magno Ventayen, and Mohd Naved. Applications of artificial intelligence in business management, e-commerce and finance. Materials Today: Proceedings, 80:2610–2613, 2023. ISSN 2214-7853. doi: https://doi.org/10.1016/j.matpr.2021.06.419. URL https://www.sciencedirect.com/science/article/pii/S2214785321048136. SI:5 NANO 2021.

[69] Vitali Petsiuk, Abir Das, and Kate Saenko. Rise: Randomized input sampling for explanation of black-box models, 2018.

[70] Samuele Poppi, Marcella Cornia, Lorenzo Baraldi, and Rita Cucchiara. Revisiting the evaluation of class activation mapping for explainability: A novel metric and experimental analysis, 2021.

[71] Luyu Qiu, Yi Yang, Caleb Chen Cao, Jing Liu, Yueyuan Zheng, Hilary Hei Ting Ngai, Janet Hsiao, and Lei Chen. Resisting out-of-distribution data problem in perturbation of xai, 2021.

[72] Sukrut Rao, Moritz Böhle, and Bernt Schiele. Towards better understanding attribution methods, 2022.

[73] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. "why should i trust you?": Explaining the predictions of any classifier, 2016.

[74] Guoguang Rong, Arnaldo Mendez, Elie Bou Assi, Bo Zhao, and Mohamad Sawan. Artificial intelligence in healthcare: Review and prediction case studies. Engineering, 6(3):291–301, 2020. ISSN 2095-8099. doi: https://doi.org/10.1016/j.eng.2019.08.015. URL https://www.sciencedirect.com/science/article/pii/S2095809919301535.

[75] Yao Rong, Tobias Leemann, Vadim Borisov, Gjergji Kasneci, and Enkelejda Kasneci. A consistent and efficient evaluation strategy for attribution methods, 2022.

[76] Soumya Sanyal and Xiang Ren. Discretized integrated gradients for explaining language models, 2021.

[77] Andrew M. Saxe, Pang Wei Koh, Zhenghao Chen, Maneesh Bhand, Bipin Suresh, and A. Ng. On random weights and unsupervised feature learning. In International Conference on Machine Learning, 2011. URL https://api.semanticscholar.org/CorpusID:8907667.

[78] Karl Schulz, Leon Sixt, Federico Tombari, and Tim Landgraf. Restricting the flow: Information bottlenecks for attribution, 2020. URL https://arxiv.org/abs/2001.00396.

[79] Karl Schulz, Leon Sixt, Federico Tombari, and Tim Landgraf. Restricting the flow: Information bottlenecks for attribution, 2020.

[80] Ramprasaath R Selvaraju, Abhishek Das, Ramakrishna Vedantam, Michael Cogswell, Devi Parikh, and Dhruv Batra. Grad-cam: Why did you say that?, 2017.

[81] 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. International Journal of Computer Vision, 128(2):336–359, oct

2019. doi: 10.1007/s11263-019-01228-7. URL https://doi.org/10.1007%2Fs11263-019-01228-7.

[82] Avanti Shrikumar, Peyton Greenside, Anna Shcherbina, and Anshul Kundaje. Not just a black box: Learning important features through propagating activation differences, 2017.

[83] Avanti Shrikumar, Peyton Greenside, and Anshul Kundaje. Learning important features through propagating activation differences, 2019.

[84] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition, 2015.

[85] Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps, 2014.

[86] Dylan Slack, Sophie Hilgard, Emily Jia, Sameer Singh, and Himabindu Lakkaraju. Fooling lime and shap: Adversarial attacks on post hoc explanation methods, 2020.

[87] Dylan Slack, Sophie Hilgard, Sameer Singh, and Himabindu Lakkaraju. Reliable post hoc explanations: Modeling uncertainty in explainability, 2021.

[88] Jost Tobias Springenberg, Alexey Dosovitskiy, Thomas Brox, and Martin Riedmiller. Striving for simplicity: The all convolutional net, 2015.

[89] Suraj Srinivas and Francois Fleuret. Full-gradient representation for neural network visualization, 2019.

[90] Pascal Sturmfels, Scott Lundberg, and Su-In Lee. Visualizing the impact of feature attribution baselines. Distill, 5, 01 2020. doi: 10.23915/distill.00022.

[91] Pascal Sturmfels, Scott Lundberg, and Su-In Lee. Visualizing the impact of feature attribution baselines. Distill, 2020. doi: 10.23915/distill.00022. https://distill.pub/2020/attribution-baselines.

[92] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks, 2017.

[93] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions, 2014.

[94] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. In International Conference on Learning Representations, 2014. URL http://arxiv.org/abs/1312.6199.

[95] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks, 2014.

[96] Richard Tomsett, Dan Harborne, Supriyo Chakraborty, Prudhvi Gurram, and Alun Preece. Sanity checks for saliency metrics, 2019.

[97] Yun-Yun Tsai, Ju-Chin Chao, Albert Wen, Zhaoyuan Yang, Chengzhi Mao, Tapan Shah, and Junfeng Yang. Test-time detection and repair of adversarial samples via masked autoencoder, 2023.

[98] Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky. Deep image prior. International Journal of Computer Vision, 128(7):1867–1888, mar 2020. doi: 10.1007/s11263-020-01303-4. URL https://doi.org/10.1007%2Fs11263-020-01303-4.

[99] Keyon Vafa, Yuntian Deng, David M. Blei, and Alexander M. Rush. Rationales for sequential predictions, 2021.

[100] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need, 2017.

[101] Erik Štrumbelj and Igor Kononenko. Explaining prediction models and individual predictions with feature contributions. Knowl. Inf. Syst., 41 (3):647–665, dec 2014. ISSN 0219-1377. doi: 10.1007/s10115-013-0679-x. URL https://doi.org/10.1007/s10115-013-0679-x.

[102] Haofan Wang, Rakshit Naidu, Joy Michael, and Soumya Snigdha Kundu. Ss-cam: Smoothed score-cam for sharper visual feature localization, 2020.

[103] Haofan Wang, Zifan Wang, Mengnan Du, Fan Yang, Zijian Zhang, Sirui Ding, Piotr Mardziel, and Xia Hu. Score-cam: Score-weighted visual explanations for convolutional neural networks, 2020.

[104] Lukas-Valentin; Wanner, Jonas; Herm and Christian Janiesch. How much is the black box? the value of explainability in machine learning models. 2020. URL https://aisel.aisnet.org/ecis2020_rip/85.

[105] Sarah Wiegreffe and Yuval Pinter. Attention is not not explanation, 2019.

[106] Haohang Xu, Shuangrui Ding, Xiaopeng Zhang, Hongkai Xiong, and Qi Tian. Masked autoencoders are robust data augmentors, 2022.

[107] Jihun Yi, Eunji Kim, Siwon Kim, and Sungroh Yoon. Information-theoretic visual explanation for black-box classifiers, 2021.

[108] Gal Yona and Daniel Greenfeld. Revisiting sanity checks for saliency maps, 2021.

[109] Jiahui Yu, Zhe Lin, Jimei Yang, Xiaohui Shen, Xin Lu, and Thomas S. Huang. Generative image inpainting with contextual attention, 2018.

[110] Matthew D Zeiler and Rob Fergus. Visualizing and understanding convolutional networks, 2013.

[111] Hanwei Zhang, Felipe Torres, Ronan Sicre, Yannis Avrithis, and Stephane Ayache. Opti-cam: Optimizing saliency maps for interpretability, 2023.

[112] Jianming Zhang, Zhe Lin, Jonathan Brandt, Xiaohui Shen, and Stan Sclaroff. Top-down neural attention by excitation backprop, 2016.

[113] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization, 2015.

[114] Luisa M Zintgraf, Taco S Cohen, Tameem Adel, and Max Welling. Visualizing deep neural network decisions: Prediction difference analysis, 2017.