POLITECNICO DI TORINO

Master's Degree in Management Engineering

Master's Degree Thesis

Design and validation of a comparison methodology for Explainable AI techniques

Supervisors

Prof. Tania CERQUITELLI

PhD Salvatore GRECO

Candidate

Francesca VANNI

March 2022

Abstract

Nowadays, Artificial Intelligence (AI) has expanded everywhere and people have become accustomed to the fact that AI can make decisions for us in our daily lives, ranging from product recommendations on Amazon and films on Netflix, to suggestions of friends on Facebook or Instagram, or even advertisements tailored to who is browsing web pages provided by Google. However, in decisions that can really make a difference, such as diagnosing a disease, it is important to know the motivation behind such a risky decision. Explainable Artificial Intelligence (XAI) systems are a potential solution towards accountable AI, making it trustworthy by explaining decision processes and AI logic to end users. In particular, an explanation of the algorithms allows for control in the event of unintended or undesirable outcomes, e.g. cases of social or racial discrimination.

This thesis aims to make a general state of the art on the subject, dealing with what Artificial Intelligence is and the importance of explanations, and their usefulness in today's world. A general classification of the main characteristics of the most common explanation techniques is made, after which the most common ones will be listed, explaining for each one in a general way how they work and an example of how they have been validated. Finally, a global overview of the surveys in the literature comparing explanation methods is proposed, along with a general and comprehensive methodology for comparing the different explanation techniques and their testing. In this last part we have focused on explanation techniques that support textual data and we included both objective and human-based metrics and criteria.

Acknowledgements

Grazie a Tania Cerquitelli, per essere stata una guida in questo ultimo percorso di tirocinio e tesi. La ringrazio infinitamente per avermi trasmesso la passione per questo ambito lavorativo. Grazie a Salvatore, per avermi supportato ed avermi aiutato nelle difficoltà in questo percorso di tesi.

Grazie a mamma e papà, per avermi spronato e appoggiato in tutti questi anni di percorso universitario. A vostro modo, siete stati dei perfetti modelli di vita. Grazie a nonna Silvana, grazie alla tua passione e i tuoi interessi sempre accesi. Ogni singolo esame e ogni giorno di scrittura di questa tesi voi l'avete vissuto con me, sostenendomi e incoraggiandomi. Vi ringrazio, e vi voglio un bene immenso.

Grazie a tutti i miei parenti, zii e cugini. Grazie per avermi vista crescere e, soprattutto, avermi aiutata a farlo.

Grazie a mia sorella Angelica per tutti i momenti che abbiamo condiviso. Grazie di avermi sostenuto per tutte le mie difficoltà, semplicemente sedendoti accanto a loro, e facendomi segno di prendere un bel respiro.

Grazie ad Alberto, per essere stato il mio complice in tutto, il mio migliore amico, la mia anima gemella. Grazie per non avermi mai lasciata sola, e per rendermi la vita ogni giorno più dolce.

Grazie a Chiara G., per avermi regalato un'anima degna di questo nome. Grazie a Chiara B. e Laura, per essermi sempre state accanto. Grazie per gli ultimi anni passati insieme, i primi e quelli che verranno.

Grazie a Norman, Carlo, Stefano M., Rita e Ginevra, perchè siete stati la mia luce in fondo al tunnel. Grazie anche a Elena, Stefano B., Alessia, Francesca, Lorenzo, Federica, Andrea e Ilaria, per le risate e i momenti passati insieme.

Grazie a Stefano S., a chi mi ha accompagnata sui banchi in questo percorso

e l'ha reso molto più piacevole, Donatella, Irma, Valerio, Paola, Carlo e Fabio.

Grazie ai miei insostituibili colleghi di tirocinio, Luca, Edona, Stefania, Stefano, Edoardo e Jacopo.

"Endings are never easy. I always build them up so much in my head, they can't possibly live up to my expectations, and I just end up disappointed. I'm not even sure why it matters to me so much how things end here. I guess it's because we all want to believe what we do is very important, that people hang on to our every word, that they care what we think. The truth is, you should consider yourself luck if you even occasionally get to make someone, anyone, feel a little better."
John Dorian

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

The discipline of Artificial Intelligence (AI) powered by Machine Learning and Deep Learning has experienced incredible changes over the last few years. Even though its first launch was only academic and research-oriented, its domain expanded over different industry field, such as technology, health care, banking, insurance, retail and many more. Therefore, the aim of AI and machine learning has shifted from academic purpose to solving real-world society and industry problems over the last decade, making our life simpler and way better.
However, explaining the reasons behind a model to the business is typically very challenging, hence the model performances often are sacrificed to obtain a better interpretability.
Moreover, with the development of increasingly high-performance AI systems and with applications in the most diverse fields, there is a growing need to explain the real decision-making behind artificial intelligence models. In this regard, the branch that has been developing rapidly in recent years, namely eXplainable Artificial Intelligence (XAI), is introduced.

1.1 What does "black-box" mean?

Data scientists and practitioners work create models and solutions for the business. Nevertheless, domain-user and common-user, when approaching a black-box model, always ask the same questions, such as “How does the model make its decisions?” or “Why should I trust this model?” [1]. One may possibly argue that if a model is performing well, there should be no reason to question how it is working, however there are many real-world scenarios where inaccurate model predictions might have devastating negative effects. Some of the scenarios in which the prediction interpretability is the first goal could be potential terrorism, fraud revealing, loan scoring, risk scoring of court judgments and so on.

This can also impact more generally on accountability [2], on safety [3], and on industrial liability [4]. As reported in [5], in fact, companies increasingly release market services and products by embedding data mining and machine-learning components, often in safety-critical industries such as self-driving cars or and personalized medicine and healthcare. Another inherent risk of these components is that many of these models may present bias against specific groups of people (e.g. racism), generally because the data used to train them were also biased.
Other tasks, instead, can be much more ordinary, such as in voice-based interaction with virtual assistant technology (e.g. Echo Dot from Amazon) or recommended movies on streaming services based on what the user has watched previously (e.g. Netflix streaming algorithm).
Some might say, if an AI system has sufficiently high accuracy, there should not be a need for explanations. Quoting the article [6], we can pose the following hypothetical scenario. Suppose we have a serious medical ailment and there are two treatments available. The first will cure patients 95% of the time, but won't be able to explain its process. The second will cure patients 90% of the time, but will be able to explain its process. Which one will we choose? The expectation that the first one will be chosen is used to suggest accuracy is what really matters. In fact, the higher the impact, the more likely there is a need for explanations. Nevertheless, the impact from decisions can vary greatly: choosing a drug treatment, denying a promotion, or suggesting a sentencing can have tremendous life consequences for the individuals involved, directly and indirectly. In contrast, decisions regarding what advertisement to show, what news story to recommend, or what movie to watch next are usually not life-changing decisions for the individuals involved.

Regardless of the unlimited potential of AI, the reasons why such algorithms make decisions still remains a secret. This is where we introduce the concept of "black-box" algorithms. A black-box system runs with an input (e.g. dataset, features...) and gives back an output, with absolutely no clue of what happened in the inside and how the model worked.
One step towards reducing people's suspicion in these algorithms is to explain the black-box's decision-making from an artificial intelligence, and that's where eXplainable Artificial Intelligence (XAI) becomes relevant. From a certain result of a model, an explainable algorithm offers an explanation of why that particular output is proposed. These algorithms operate by showing the end-user some details of the insides or presenting which inputs were most relevant.

Figure 1.1: Cartoon on the operation of a black-box model [7]

1.2 Why push harder on studying the XAI?

If the risk of misunderstanding or lack of interpretability in black-box models is high, some might ask why the demand of AI systems is increasing so rapidly.
Explainable AI (XAI) methods deal with this challenge because they give human-interpretable explanations for black-box models, so the end-users can understand, fully trust and operate with AI outputs.
It is known, in fact, that an AI system can easily surpass human performance. As shown in [8], XAI methods could help users and practitioners further evaluate their models beyond standard performance metrics (e.g., accuracy metric) by examining and analysing individual predictions from the information given by their explanations [1]. Furthermore, these methods could possibly bring out biases in the trained dataset, classes, multiple labels, and other mistaken correlations learned by a model [9]. Additionally, more understandings could be obtained, in cases, e.g., that a model overcomes human performance; it may have incorporated scientific knowledge that can be extracted via an XAI method providing insights to the domain experts and scientific community [10].

1.3 Thesis goal and next chapters

The goal of this thesis is to make a general and exhaustive comparison of the main eXplainable Artificial Intelligence (XAI) techniques.

In Chapter 2, we will introduce the usefulness of eXplainable Artificial Intelligence (XAI), and then go on to understand what an explanation actually is and explain the meaning of the concept of "interpretability". Then, a general classification will be made on the methods of XAI present in the literature, how they are distinguished from each other, the characteristics that they have in common and the types of data that they can support.

In Chapter 3, a general overview of the main XAI techniques will be given, ranging from the most common to the most recent. These will be briefly introduced and particular attention will be paid to their operation and validation.

In Chapter 4, a summary of the experimental and theoretical surveys in the literature will be introduced, in which a comparison is made of some of the techniques mentioned in Chapter 3. It will also introduce the desiderata or evaluation measures required by an XAI technique and the actual metrics with which to test them.

In Chapter 5 the proposed methodology will be illustrated, together with the actual experimentation which has focused on textual data, made in particular on three XAI methods: LIME [1], SHAP [11] and T-EBAnO [12].

In Chapter 6, the final conclusions of the thesis are drawn and possible future developments are proposed.

Chapter 2

State of the art

This chapter introduces an overview of the various concepts of Artificial Intelligence and what an explanation is. It then focuses on the meaning of interpretability of an explanation, and illustrates the different categorisations of the topic and the main terminologies used.

2.1 What is the AI?

Artificial Intelligence (AI) describes the creation of an intelligent hardware or software that can match behaviour reminiscent of humans, like learning and problem-solving. AI is a broad discipline of computer science that concentrates on a machine's ability to produce rational behaviour. The aim of AI is to implement systems or models that can execute tasks that can be partly or fully replaced by human intelligence.
Nowadays, AI and machine learning applications have turn out to be prevalent: Big Tech such as Apple, Amazon, Google or Facebook have collected so much data from the world's population that they can entirely shape each person's interests and preferences. However, the past negative interference of social media bots, for example in political elections [13], [14] has been a negative sign of how influential our lives are to the mishandling of AI and big data [15]. For these reasons, those who rely on AI applications, such as the Big Tech, increasingly need predictable and accountable AI systems. Transparency and interpretability of algorithms is a crucial point for products related to AI, big-data and information communication.
AI has now expanded everywhere, and we have become accustomed to the fact that AI can make decisions for us in our daily lives, ranging from product recommendations on Amazon and films on Netflix, to suggestions of friends on Facebook or Instagram, or even advertisements tailored to who is browsing web pages provided by Google.

However, in decisions that can really make a difference, such as diagnosing a disease, it is important to know the motivation behind such a risky decision.
As the impact of powerful black-box machine learning models in the big-data era has reached huge significance, the interpretability of such models has therefore been studied in various research contexts.
For example, in 2018 the legal right to explanations in machine learning systems was established, formally mentioned in the European Union's General Data Protection Regulation (GDPR) commission. As the regulations now mainly focus on user data protection and privacy, in the future they are expected to cover more requirements for transparency of algorithms and explanations from AI systems [16].
In today's world, algorithms examine user data and have an impact on the decision-making of millions of people on a variety of issues such as employment, insurance rates, and even criminal justice [17]. However, such algorithms have crucial roles in many industries and have their own drawbacks, that can result in discrimination [18], [19], and unfair decisions [15]. For example, recently, news feeds and targeted advertising algorithms in social media have drawn much interest for exacerbating the absence of information variety in social media [20]. An important aspect of this issue is certainly reflected in the fact that algorithms in decision-making systems do not allow end users to choose between recommended options, but only between the most relevant options, chosen by the algorithm itself.
To deal with this issue, Bellotti and Edwards [21] suggest that context-aware intelligent systems should not act on behalf of the end-user, instead opting for user control over the system as a principle to support accountability of a system and its users.
The main benefits of transparency and interpretability of black-box systems are a general awareness and accountability of the end-user, the possible detection of bias or discrimination of any kind [22], and a behaviour of such systems that can also be interpreted by humans [23].

Machine learning (ML), as reported in [24], is "the study of computer algorithms that can improve automatically through experience and by the use of data".
Machine Learning (ML) systems are increasingly used in various disciplines and applications and are becoming more and more efficient in various tasks ranging from everyday life problems (e.g. smart health) to decision making for high-risk domains (e.g. clinical decision support). Examples include simple tasks in everyday life such as object recognition in images or translation of words or speech between different languages, or more complex tasks such as autonomous car driving, automatic drone flight, or have proved very useful in environmental and healthcare changes.
Unfortunately, even though they seem very powerful in terms of output and predictive decisions, AI algorithms lack in transparency, so much so that it is almost impossible to obtain a complete view of their inner workings, particularly Machine

Learning (ML) algorithms. This fact aggravates the issue even more, because assigning vital decisions to a system that cannot be transparent presents obvious problems and potential risks.
With the aim of solving this problem, eXplainable Artificial Intelligence (XAI) offers a version of AI that is much more transparent and interpretable, providing a group of methods that provide more accountable models, while still preserving high levels of performance.

2.2 What is the XAI?

The eXplainable Artificial Intelligence (XAI) is, by definition, "a research discipline in which the results of the solution can be understood by humans" [25]. The term was first coined in 2004 by Van Lent et al. [26], to describe the way in which their system explained the behaviour of AI-controlled entities in simulation games. Although the term is rather new, the dilemma of explainability was devised in the mid-1970s, when researchers studied explanation for expert systems [27]. Until now, progress and research on the subject has mainly focused on the implementation of models and algorithms that highlights predictive power, while the ability to explain the reasons behind decision processes has received less attention.
However, nowadays there is still no universally accepted standard definition of Explainable AI. In fact, the term XAI refers purely to the study, research and efforts made to make AI methods more transparent to users and trustworthy, rather than to a technical-formal concept. According to DARPA [28], XAI aims to “produce more explainable models, while maintaining a high level of learning performance (prediction accuracy); and enable human users to understand appropriately, trust, and effectively manage the emerging generation of artificially intelligent partners". The goal of enabling explainability in ML, as stated by FAT* [29], is to "ensure that algorithmic decisions as well as any data driving those decisions can be explained to end-users and other stakeholders in non-technical terms". FICO [30], the organizer of xML Challenge, see XAI as "an innovation towards opening up the black-box of ML" and as "a challenge to create models and techniques that both accurate and provide good trustworthy explanation that will satisfies customers’ needs".
eXplainable Artificial Intelligence (XAI) systems are a potential solution towards accountable AI, making it trustworthy by explaining decision processes and AI logic to end users [28]. In particular, an explanation of the algorithms allows for control in the event of unintended or undesirable outcomes, e.g. cases of social or racial discrimination.

An XAI system can be defined as a self-explanatory intelligent system that describes the reasoning behind its decisions and predictions [31].

The XAI system, as reported in [31] and illustrated in Figure 2.1, is able to generate explanations and describes the reasoning behind machine learning decisions and predictions. In this figure, we shall see the user interacting with the explainable interface, sending queries to the interpretable machine learning and receiving model prediction and explanations. Ex post, such explanations allow users to understand how data is processed and aim to bring to light possible biases and malfunctions of the system. On the other hand, we shall see the interpretable model interacting with the data and generating explanations or new predictions for the user query.

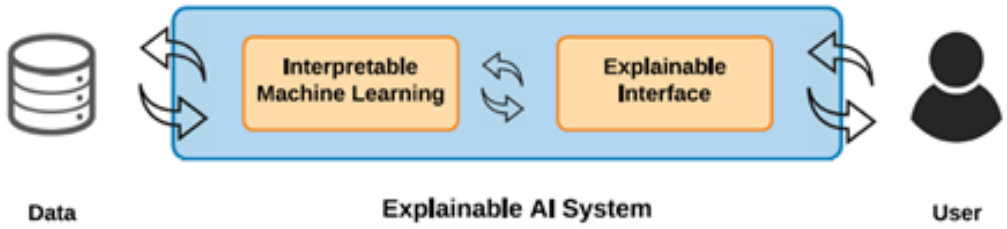


Figure 2.1: Outline of an XAI system [31].

Resuming the Big Tech discourse of the previous paragraph, a study was carried out by Rader et al. [32] to measure users' knowledge of the most popular social media algorithms. In this crowdsourced study, it was analysed how different types of explanations influence users' opinions on the algorithmic transparency of feeds on various social platforms. In this way, the awareness, correctness, and responsibility of users could be measured to assess their algorithmic transparency, and it was found that users became much more aware of the behaviour of the system thanks to the explanations provided.

2.2.1 Why do we need to explain?

Now we propose an overview of the four most important aspects of explainability, as reported in [33]:

Explain to justify. When we refer to an explanation of a prediction, we usually imply the need to understand the reasoning used or a justification for that answer, rather than a detailed description of the internal logic of the decision-making process. The use of XAI systems generates the information needed to justify outputs, especially when making unexpected predictions; it also demonstrates that algorithmic methodologies are reasonable and ethical, which generates trust in the user who interprets them. Likewise, as seen above, AI must always give explanations to comply with the current legislation. For example, mention is made of the "right to an explanation", a regulation included in the General Data Protection Regulation (GDPR) that comes into force across the EU on 25 May 2018 [34].

Explain to control. Explainability is not only essential to explain predictions, but can also prevent erroneous output. A greater understanding of the system's behaviour can provide better insight into the vulnerabilities and flaws in the model that are not yet known, allowing for better human control of the situation.

Explain to improve. Another purpose for using explainable AI models is the need to constantly enhance them, since a model that can be explained is also a model that can be enhanced more effortlessly. The user, in fact, knowing the reasons behind certain outputs of the system, will also be able to make that system sharper and faster.

Explain to discover. Obtaining explanations is a useful tool for discovering new paradigms, and thus acquiring more expertise. Compared to simple AI systems, XAI systems can have this extra usefulness, placing the human being to increase the spectrum of his skills and knowledge.

2.3 What is an explanation?

In contexts where crucial decisions need to be made based on the prediction of an Artificial Intelligence (AI) system, it is important that such a system can provide explanations that are interpretable and responsible. In fact, as also demonstrated in [35], a proper explanation can increase the trust that humans have towards the AI system, thus allowing a better collaboration between humans and AI.
The concept of explanation is most likely as outdated as the concept of human communication. Generally, as described in the article [6], "an explanation is a communication from one person (A) to another (B) that provides a justification for an action or decision taken by person A".
Academics normally tend to use «evidence» to formally offer explanations for their work. These evidences are in fact constructed using a logic and formalism common to all, so that anyone with knowledge in the discipline can prove their validity. Problematically, for the explanations of the context we are dealing with, there is, as seen above, no formal logic that unites everyone in final knowledge.
The purpose of providing explanations indirectly implies answers to questions such as "How does it work?" or "How can it go wrong?" or even "Why did it use this procedure and not another?". The concept behind an explanation is not the statement itself, but it is the interaction that follows what the learner/user needs, especially from the user's goals.
In Table 2.1 below, as presented in [36], "triggers" on the part of the user for a possible explanation provided by a system are set out. The AI system (whether algorithm or system) must be able to predict what the user will want to know

about the functioning of the system.

TRIGGERS	USER/LEARNER'S GOAL
How do I use it?	Achieve the primary ask goals
How does it work?	Feeling of satisfaction at having achieved an understanding of the system, in general (global understanding)
What did it just do?	Feeling of satisfaction at having achieved an understanding of how the system made a particular decision (local understanding)
What does it achieve?	Understanding of the system's functions and uses
What will it do next?	Feeling of trust based on the observability and predictability of the system
How much effort will this take?	Feeling of effectiveness and achievement of the primary task goals
What do I do if it gets it wrong?	Desire to avoid mistakes
How do I avoid the failure models?	Desire to mitigate errors
What would it have done if "x" were different?	Resolution of curiosity at having achieved an understanding of the system
Why didn't it do "z"?	Resolution of curiosity at having achieved an understanding of the local decision

Table 2.1: Triggers and goals in explanations [36].

Hence, an explanation can easily be an example of specific need that the user has for his goals or purposes.

2.3.1 What to explain

When users approach an XAI system, they may request different types of explanations and each explanation may need its own characteristic. Now, as reported in [31], we detail the following six common types of explanations required and used in XAI systems.

How-explanations. They show a holistic representation of the machine learning algorithm, and are for example visual representations, model graphs and decision boundaries.

Why-explanations. They describe why a decision is made for a particular input. Such explanations aim to show the features in the input data or the logic of the model that led to the decision made by the model then shown in the output.

Why-Not-explanations. Also called Contrastive explanations, they show the reasons why a certain result was not expected in the system output, outlining in particular

the differences from the output expected by the final user.

What-If-explanations. What-if scenarios are generated by the model or requested by the user, and are useful to demonstrate how certain changes in algorithms or data affect the output or parameters of the model.

How-to-explanations. They show the methodologies by which the model shows up with a given output, also possibly working interactively and evolving the system often through iterative testing.

What-Else-explanations. Taking data from the training dataset as input, they demonstrate the same or similar outputs of the final model. Such explanations are not always accurate because the training datasets often do not have a uniform distribution of data.

2.3.2 How to explain

The explanation methods can also be divided into three types, as stated in [37], [38].

Model-based. The explanations use a model to justify original task models, therefore either such task model itself is exploited as an explanation or many other understandable models are provided to justify the task model. Some of the quantitative metrics to evaluate the goodness of this type of explanations are: model size, runtime operation counts, interaction strength, main effect complexity, and level of (dis)agreement.

Attribution-based. The explanations assess the explanatory capacity of input features and exploit it to justify the task model (e.g. feature importance). Some of the quantitative metrics to evaluate the goodness of this type of explanations are: monotonicity, (non) sensitivity, effective complexity, remove and retrain, recall of important features, implementation invariance, selectivity, continuity, n-sensitivity, and mutual information.

Example-based. The explanations justify the task model by picking occurrences from the training dataset or the testing dataset, or else even creating new occurrences (e.g. creating counterfactual examples). Some of the quantitative metrics to evaluate the goodness of this type of explanations are non-representativeness and diversity.

2.4 What does "interpretability" mean?

As described in the previous sections, a black-box predictor is a locked model of machine learning, whose core mechanisms are either unclear to the user or are clear but not interpretable by the human. In [39], Doshi-Velez et al. define the concept of interpretability as "the ability to explain or provide meaning in terms understandable to a human".
It should be emphasised that the concept of interpretability is intertwined with the concept of explainability: in fact, interpretability may be granted in a model, but not explainability. Also, if explainability is required, the model must be interpretable as well. There are in fact some models for which an explanation is not necessarily required, e.g. if one wants to know whether a picture contains a certain object or not, this information is not «crucial», or at any rate there are no disastrous consequences should the model produce an incorrect output.

2.4.1 Dimensions of interpretability

The dimensions of the interpretability can be categorized on three aspects, as stated in [40].

Global and local interpretability. We talk about global interpretability if a model is fully interpretable, i.e. we can understand the logic and reasoning that then leads to all the outputs of the system. On the other hand, one speaks of local interpretability if a model is interpretable only for certain forecasting logics.

Time limitation. When an explanation is provided, it is always necessary to consider the time the end user will have to understand it. In fact, for cases where the scenario may be imminent (e.g. medical), the user will need a very easy and understandable explanation, while for cases where the scenario may be longer term (e.g. selection of a candidate for a job offer), the decision will not necessarily be a constraint, so a much more complex and studied explanation may be provided.

Nature of the user's competence. When an explanation is given, it is necessary to take into account the future interpretability that the user will be able to give to this explanation. Domain expert users will, for example, be able to understand a more complex and sophisticated model as opposed to a highly simplified one as a basic user might prefer, based on their knowledge.

2.4.2 Desiderata of an interpretable model

Since a model must necessarily generate an explanation in order to be interpretable, it is mandatory to list the main desiderata that must be present in it, given in [40], [39].

Interpretability. This measures the extent to which the decisions of the model are easily comprehensible to humans, so the term "interpretability" can easily be combined with the term "comprehensibility". It is, however, very difficult to know how to measure exactly the complexity and comprehensibility of a model, so we often refer to its dimensions.

Accuracy. Measures how well the model is able to make accurate decisions about non-visible instances. Accuracy can be measured by scores found in the literature.

Fidelity. This measures how well the model is able to faithfully reproduce the behaviour of a black box predictor. It is also possible to measure fidelity using scores found in the literature.

Fairness. It measures the extent to which the model is able to protect the final output against direct or indirect discrimination [41].

Privacy. It measures the extent to which the model favours and respects the privacy standards of the users, not disclosing sensitive information [42].

Usability. Measures how useful the information generated by the model is to users for their tasks, emphasizing the interactivity of a model and discouraging fixed explanations.

Reliability and robustness. Measures the ability of a model to remain performant despite small changes in certain parameters or differences in input data.

Causality. Measures the ability of the model to adapt to perturbing inputs that should theoretically change its general behaviour.

Generality. A measure of a model's ability to adapt to different sources of input data or differentiated inputs, thus discouraging training with certain constraints.

2.4.3 Data Types in an Interpretable Model

The data entered in a black-box model can be of different types, depending on the rank of interpretability established by the human user, as suggested in [40]. In

Table 2.2 the XAI techniques covered in this thesis are summarised according to the data they support. It should be noted that some methods have sub-methods that support other data types, or others that only theoretically support a data type but have not yet been tested.

Tabular data. This is the most common type of data, as algorithms are able to organise them into matrices and in this way manage them more easily without the need for further modification. However, they have the disadvantage of also having to represent meta-data, which make the user understand the meaning of the data in the various tables.

Images and text. This is the type most immediately understandable to the human and does not require the addition of any meta-data for the understanding of the meaning. However, the transformation required by the model is difficult for these data, as they are often transformed into vectors, so not all existing interpretable models can be adapted to this type of data.
We point out that this thesis focuses on this type of unstructured data, in particular text and images.

XAI technique	**Data support**
LIME [1]	Images, Text
SHAP [11]	Images, Text
LRP [43]	Images, Text
DeepLIFT [44]	Images, Text
Grad-CAM [45]	Images
T-EBAnO [12]	Text
IntGrad [46]	Images, Text
RISE [47]	Images
Anchors [48]	Images, Text
SmoothGRAD [49]	Images
SENN [50]	Images, Text
SITE [51]	Images, Text
VA-GAN [52]	Images
ICAM [53]	Images
Archipelago [54]	Images, Text
Mahè [55]	Images, Text
XRAI [56]	Images

Table 2.2: Data support for each XAI technique

2.5 Classification of Explainability Methods

We now propose a general classification of the different explainability methods. As reported in [33], the methods rely heavily on the concept of explainability and interpretability, which are linked to each other, as seen above. We can therefore classify the methods according to the following three criteria:

- The complexity of interpretability;
- The scoop of interpretability;
- The level of dependence on the machine learning model used.

In the next paragraphs, we will characterize each of these classes in detail.

2.5.1 Complexity related methods

It is assumed that the more complex a machine learning model is, the more difficult it is to interpret. With this assumption one can distinguish two sub-types of methods according to the methodology they have of interpretability:

- Intrinsic;
- Post-hoc.

Intrinsic

The first type of methods, the instrinsic ones, have the common characteristic of having an interpretability that is intrinsic and already contained in the model itself. Their interpretability is in fact already present in the model in nature, some examples being linear and parametric models, or tree based models. A negative aspect that hampers the usability of this type of models is finding a middle ground between their interpretability and their accuracy. As Breiman [57] argues, "accuracy generally requires more complex prediction methods (...) [and] simple and interpretable functions do not make the most accurate predictors".

Post-hoc

The second type, post-hoc, advances a different criterion, i.e. the interpretability of the model is done a posteriori, often outside the black-box itself. The basic intuition, in fact, is to train a black-box (e.g. neural networks) and verify the interpretability later (e.g. feature importance), thus making a sort of reverse engineering mechanism. The disadvantage of this typology is that it can often turn

out to be very expensive, however most of the works of XAI in recent years belong to this typology, also for reasons of better accuracy.

To summarise, depending on the decision task of the model, an intrinsic method will be chosen if the model is already accurate enough for the task and the complexity is not exorbitant; a post-hoc method will be chosen if the model is very accurate and the complexity is of another level. Table 2.3 summarises the XAI techniques discussed in this thesis according to whether they adopt intrinsic or post-hoc interpretability.

XAI technique	**Complexity of interpretability**
LIME [1]	Post-hoc
SHAP [11]	Post-hoc
LRP [43]	Post-hoc
DeepLIFT [44]	Post-hoc
Grad-CAM [45]	Post-hoc
T-EBAnO [12]	Post-hoc
IntGrad [46]	Post-hoc
RISE [47]	Post-hoc
Anchors [48]	Post-hoc
SmoothGRAD [49]	Post-hoc
SENN [50]	Post-hoc
SITE [51]	Post-hoc
VA-GAN [52]	Intrinsic
ICAM [53]	Intrinsic
Archipelago [54]	Post-hoc
Mahè [55]	Post-hoc
XRAI [56]	Post-hoc

Table 2.3: Complexity of interpretability for each XAI technique

2.5.2 Scoop related methods

A distinction must be made between the ways in which a model can be understood whether in its entirety or in part. Indeed, the literature differentiates between methods of explainability according to their interpretability, namely:

- Global interpretability;
- Local interpretability.

Global interpretability

A global interpretability, as also seen in the previous paragraphs, simplifies the comprehension of the global reasoning of the model. In this way, it is possible to observe the functioning in its entirety and all possible results derived from it.
This type of method is used if the result to be predicted is of vital importance, but the structure is usually well thought out and very specific, so they will be more comprehensible than predictable.
It should also be noted that global interpretability is less applicable than local interpretability: as pointed out in [33], similarly to humans, by concentrating only on one part of the model (local) the effort to understand it will be much less and the comprehensibility will be greater. In fact, in practice global interpretability is rather difficult to achieve, especially for models that have many binding parameters.

Local interpretability

A local interpretability exposes the logic behind a certain and often single model decision, thus acting locally on it. This type of interpretability tends to explain a certain instance of the entire prediction, e.g. why the model made that single decision rather than another.
There are many famous local explanation methods in the literature, of which we mention one of the most important, LIME [1], seen and studied in more detail below.

XAI technique	Scoop of interpretability
LIME [1]	Local
SHAP [11]	Global/Local
LRP [43]	Global/Local
DeepLIFT [44]	Local
Grad-CAM [45]	Local
T-EBAnO [12]	Global/Local
IntGrad [46]	Global/Local
RISE [47]	Local
Anchors [48]	Local
SmoothGRAD [49]	Global/Local
SENN [50]	Local
SITE [51]	Local
VA-GAN [52]	Global
ICAM [53]	Global
Archipelago [54]	Global/Local
Mahè [55]	Local
XRAI [56]	Local

Table 2.4: Scoop of interpretability for each XAI technique

2.5.3 Model related methods

A final methodology for classifying explanation methods is based on their applicability to machine learning algorithms and is also based on the differentiation seen above of intrinsic or post-hoc methods. The differentiation is as follows:

- Model-specific;
- Model-agnostic.

Model-specific

Model-specific methods are models that are based only on certain classes of models. The intrinsic methods seen in the above paragraph are, by definition, (as seen in Figure 2.2 [33]) model-specific methods. This type of method is not so common, nor is it immediate, since one is limited to the interpretation provided by the model itself or the multiple models that provide it, taking away space from models that are perhaps more representative of the situation one is working in.

Model-agnostic

The model-agnostic methods tend to keep the decision and interpretability on two distinct planes, in fact for this reason they are not fixed on any specific type of model. The post-hoc methods seen above are usually also model-agnostic methods and can potentially work on any machine-learning model.
Most of the techniques developed in recent years, with the aim of achieving better model interpretability, are in fact model-agnostic.

XAI technique	**Model of interpretability**
LIME [1]	Agnostic
SHAP [11]	Agnostic
LRP [43]	Agnostic
DeepLIFT [44]	Agnostic
Grad-CAM [45]	Agnostic
T-EBAnO [12]	Specific
IntGrad [46]	Agnostic
RISE [47]	Agnostic
Anchors [48]	Agnostic
SmoothGRAD [49]	Agnostic
SENN [50]	Agnostic
SITE [51]	Agnostic
VA-GAN [52]	Specific
ICAM [53]	Specific
Archipelago [54]	Agnostic
Mahè [55]	Agnostic
XRAI [56]	Agnostic

Table 2.5: Model of interpretability for each XAI technique

To summarize what has been said so far, Figure 2.2, shown by [33], is helpful. The diagram initially differentiates between methods that have immediate interpretability (intrinsic) and post-hoc methods, which were invented later to shed light on black-box models with much greater complexity. In addition to these methods, there are also model-specific methods, which are intrinsic by definition, and model-agnostic methods, which are linked to post-hoc methods, and which have greater comparability and are independent of the model.
Moreover, all these methodologies can be further distinguished according to whether they have a local interpretability, of a portion of the model so as to provide more confidence in it, or global, of the whole model so as to better understand its general mechanisms.

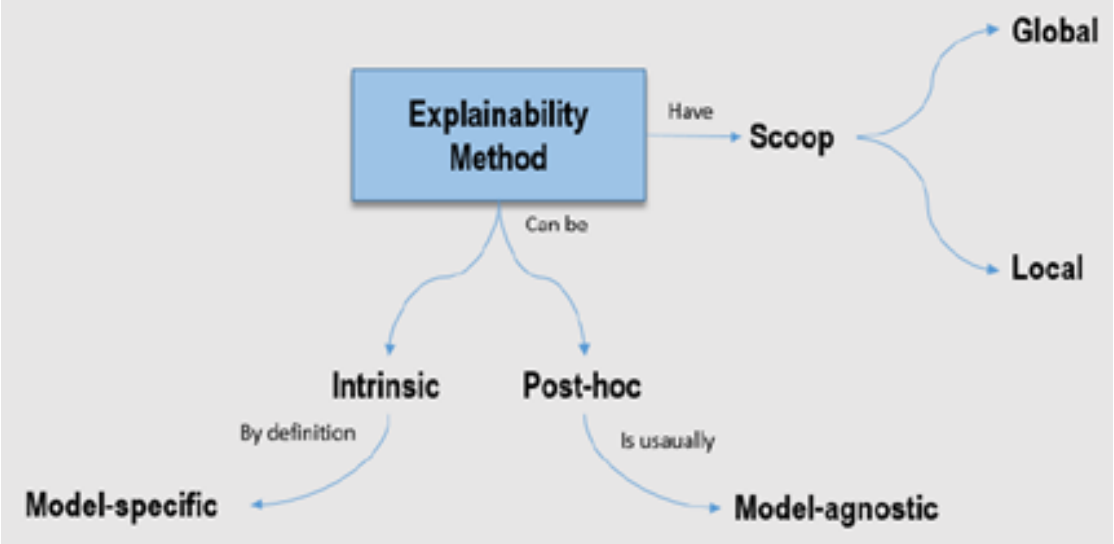


Figure 2.2: A pseudo-ontology of XAI methods taxonomy [33].

2.5.4 Perturbation-based methods

The perturbation-based explanation methods are based on a perturbation operation that depends on the response of the model, but remain independent of it, being always of the black-box type. These model methods construct explanations by studying the response of the model to local changes and act accordingly. For example, saliency maps are created by perturbing the input and checking what the effects are on the output.
Examples of perturbation methods are LIME [1] and SHAP [11] themselves, which, being both model-agnostic, perturb certain parts of images to generate explanations. Another perturbation-based method used subsequently for the experimental section is T-EBAnO [12].

2.5.5 Gradient-based methods

Gradient-based explanation methods instead directly calculate the gradients of the output class with respect to the input as an explanation. These methods, in fact, calculate the amount of the prediction gradient, also called classification score, based on the input features.
Compared to perturbation-based methods, these have in common that the explanations have the same size and rely on feature detections for explanation or classification. However, gradient-based methods prove to be computationally faster, mainly because they do consider the model. Indeed, referring to the previous examples, LIME [1] and SHAP [11], as far as image classification is concerned, would propose an output with higher computational costs.

Some common gradient-based methods in the literature are Grad-CAM [45], which analyses the single gradient assignment and based on whether it is positive or negative calculates the predicted class probability, and SmoothGrad [49], which is mainly an explanation methodology that can be applied to upgrade any gradient-based method.

2.5.6 Propagation-based methods

Propagation-based explanation methods, on the other hand, are dependent on the model they attempt to explain, in fact they involve the internal structure of the model itself in the explanation procedure. Compared to other techniques, such as gradient-based, propagation-based methods do not have typical problems, such as discontinuity of the explanation due to gradients.
A well-known method of explanation propagation-based is LRP [43], in fact its operation is based on the propagation of the explanation from output to input, via local redistribution rules.

XAI technique	**Operating principle**
LIME [1]	Perturbation-based
SHAP [11]	Perturbation-based
LRP [43]	Propagation-based
DeepLIFT [44]	Gradient-based
Grad-CAM [45]	Gradient-based
T-EBAnO [12]	Perturbation-based
IntGrad [46]	Gradient-based
RISE [47]	Gradient-based
Anchors [48]	Perturbation-based
SmoothGRAD [49]	Gradient-based
SENN [50]	Perturbation-based
SITE [51]	Perturbation-based
VA-GAN [52]	Perturbation-based
ICAM [53]	Perturbation-based
Archipelago [54]	Gradient-based
Mahè [55]	Perturbation-based
XRAI [56]	Gradient-based

Table 2.6: Operating principle for each XAI technique

2.5.7 Explanators and black-box classification

We then make a classification based on the different types of classificators and black-box that can be took in a model, as shown in [40].

Explanators

In this subsection, different types of interpretable explanator will be classified.

Decision Tree (DT) or Single Tree. This type of explanator is generally one of the most understandable and comprehensible models, for both global and local explanations.

Decision Rule (DR) or Rule Based Explanator. This type belongs to the human-understandable methods and is generally exploited to justify the model or to create a transparent design.

Features Importance (FI). This type of explanator gives the weight of the features as an explanation. It is a valid explanator and it works either as a global and a local explanation.

Saliency Mask (SM). This type is generally used for texts or images because it brings out the causes of the explanation, thanks to a "mask" that explains by sight the outcome.

Sensitivity Analysis (SA). This analysis weigths the uncertainty of the results keeping track of the uncertainty in the input data.

Partial Dependence Plot (PDP). This type facilitates the understanding of the results related with the input, thanks to a concentrated feature space.

Prototype Selection (PS). This type of explanator works with a propotype, an object that reviews the instances that are similar to the output. With this artifact, the PS shows a prototype similar to the outcome, so that the measures of the prediction becomes clearer.

Activation Maximization (AM). This type activates the most important neurons of the input that worked to obtain the output.

Black-box

In this subsection, different types of black-box will be classified following the example in [40].

Neural Network (NN). This black-box is created by a collection of neurons linked to each other. Each connection between the neurons transmits a kind of signal, which is sent to the adjacent neurons. The network of neurons is usually organised in

layers, which can be different from each other and perform certain transformations on the input data. Furthermore, the connections between the various neurons may have a certain weight, which changes depending on the learning of the network.

Tree Ensemble (TE). These methods link several learning algorithms together in order to improve predictive power as each is trained on a different subset of the input data. Examples of Ensemble Tree are Random Forest, Boosted Trees and Tree Bagging.

Support Vector Machine (SVM). SVMs have the speciality of using so-called support vectors, which are generally a subset of the training data, as a decision boundary.

Deep Neural Network (DNN). It is a NN (Neural Network) with a combination of non-linear relationships with multiple layers of hidden basic units, in fact, generally, the data in this network travels only in one direction, from input to output. They differ mainly because a DNN is more complex and deeper than an NN. Some important networks belonging to this category are the RNN (Recurrent Neural Networks), important for having as component the LSTM nodes (Long Short-Term Memory), or the CNN (Convutional Neural Networks), very important for the study of the images.

Non-Linear Model (NLM). The operation behind it is based on the non-linear combination of model parameters using one or more independent variables.

Chapter 3

State of the art explanation techniques

This chapter discusses the main eXplainable Artificial Intelligence (XAI) techniques in the literature. The most known in the domain, LIME [1], SHAP [11], LRP [43], Grad-CAM [45] and DeepLIFT [44], will be discussed, followed by a recent technique used in the experiments of the following chapters (T-EBAnO [12]) and other well-known techniques included in most of the well-known comparative surveys.
In presenting the various techniques, a general introduction on the main characteristics of the methodology will be made, after which the general functioning of the methodology will be summarised, ending with the validation of the technique contained in the reference paper.

3.1 LIME

Local Interpretable Model-Agnostic Explanations (LIME) is the first technique that we are going to analyse, and it is a local, post-hoc method that aims to explain the decisions of any classifier [1]. It generally supports images and text.
This methodology is based on the local analysis of the single model explanation, unlike many other methods that rely on global explanations, as the understanding logic it uses is to approximate the model locally with an interpretable linear model. This ensures minimal implementation work, yet at the same time LIME provides a simple and very interpretable model as it approximates a complex model locally.
LIME is a very popular method and extends in three different versions depending on the type of data that is provided as input (tables, text or images).

3.1.1 How does it operate?

The idea behind LIME is to achieve two main steps, once the prediction model and the input data set are received:

1. Through a random perturbation of the data set, LIME performs a sampling and generates a new perturbed data set.

2. Using the distance between the single perturbed sample and the previous one, LIME performs a feature selection on the perturbed data set in order to obtain the most relevant features.

We can see how the LIME method works in the Figure 3.1. You can see that the blue/pink background describes the function of the original model, which is why it is not linear, while the red cross represents the sample to be explained. The other crosses shown next to the red cross are the perturbed instances and are of different sizes in this case according to their weight. These perturbations, in fact, help to explain the last object in this figure, namely the dashed black line, which represents a very good approximation of the model with respect to the red cross just mentioned.

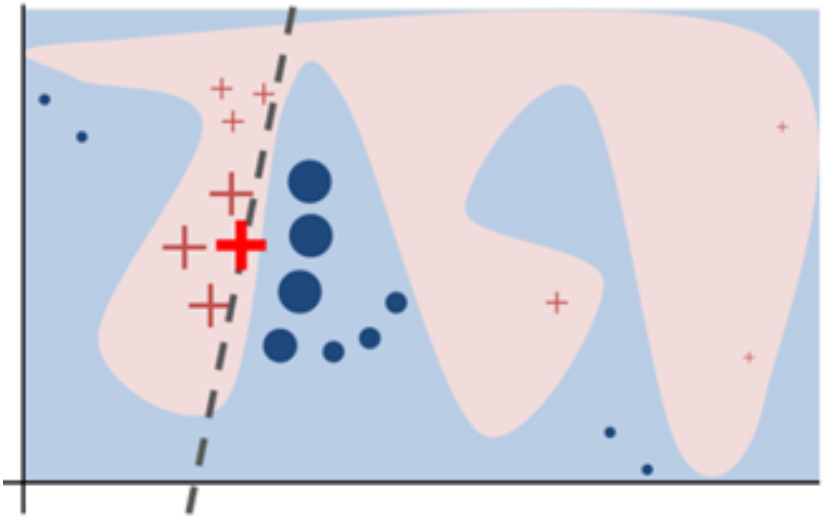


Figure 3.1: Toy example to show the intuition behind LIME [1].

LIME operates with the help of each of the following items [1]:

- g: an explanation is described with the model $g \in G$, the possible explainable model of the class G. That class can include many types of models and the domain is $\{0,1\}^{d'}$, i.e. the model g vary with the existence of certain interpretable features.

- $\Omega(g)$: it evaluates the density of the explanation $g \in G$. It is a distant value from the interpretability because it measures how much a model is complex. For example, if we consider a decision tree, $\Omega(g)$ could quantify the depth of the tree, not its interpretability.

- x: it denotes on d features the explanation of an instance, in fact $x \in R^d$.

- f: it denotes the explanation of a model, indicated by $f : R^d \rightarrow R$. Then, $f(x)$ is the probability that a specific class contains the instance x.

- $\pi_x(z)$: it describes the locality of x, in fact it is a proximity quantity of x with an instance z.

- $\mathcal{L}(f, g, \pi_x)$: it quantifies of how badly g is estimating f in the π_x proximity.

If we want a valuable local fidelity-interpretability trade-off, we must reduce the $\mathcal{L}(f, g, \pi_x)$, and $\Omega(g)$ should be decreased along with the tolerance of the human's interpretability [1].
Given those assumption, the formula that assesses LIME's explanation is the following:

$$\xi(x) = {}_{g \in G}\, \mathcal{L}(f, g, \pi_x) + \Omega(g) \tag{3.1}$$

3.1.2 Validation example with text classification

Take as an example two classes that are likely to be harder to differentiate, in fact in Figure 3.2 we can see that they have several words in common ("Christianity" and "Atheism"). In this example we train a random forest with 500 trees and achieve a particularly high test set accuracy of 92.4%.
In the figure below, we can notice an example in which the model performs a correct prediction, but for incorrect reasons: in the left image, for example, the word "posting" occurs in 21.6% of the instances of the training set and only twice in the class "Christianity"; the same situation happens in the test set, where it occurs with a little less percentage and always only twice in "Christianity". From this example we can see how useful the explanations are compared to having only the dirty input data in hand, and how interpretable they are. Indeed, the classifier, despite being complex in its own right, in the vicinity of the example approximates to a linear model. To take a numerical example, if one were to remove the words "NNTP" and "Host" from the experiment, the probability that the model predicts "Atheism" would become lower.

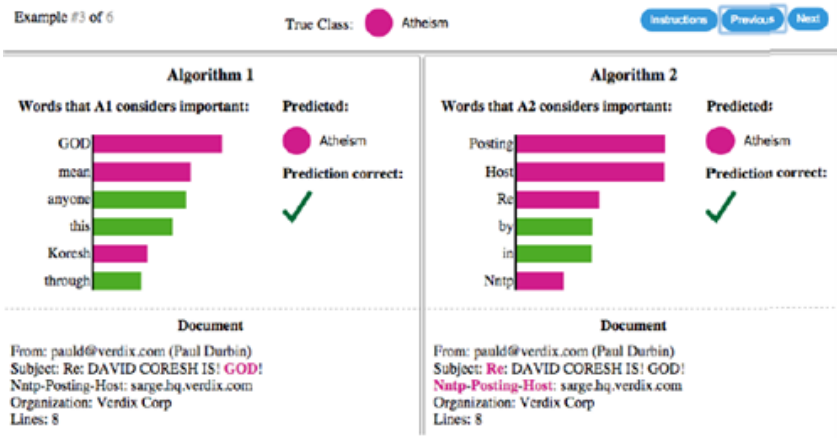


Figure 3.2: Explaining individual predictions of competing classifiers trying to determine if a document is about "Christianity" or "Atheism" [1].

3.1.3 Validation example with images

The figure 3.3 is an example taken from [1], in which an attempt is made to explain how Google's Inception neural network works on images. This example is in fact much more visual: the explanations are given by the clippings of the image which in this case were most positive for a particular class. From the original image, one can see the three explanations as: electric guitar, acoustic guitar and Labrador. The first explanation is misleading because the classifier manages to justify the error seen by the human, i.e. the part of the image that determines "electric guitar" only reveals the upper part of an acoustic guitar, the fret board, which could also be associated with an electric guitar.
The LIME code for experiments is available on https://github.com/marcotcr/lime.

(a) Original Image (b) Explaining *Electric guitar* (c) Explaining *Acoustic guitar* (d) Explaining *Labrador*

Figure 3.3: Explaining an image classification prediction made by Google's Inception neural network [1].

3.2 SHAP

SHapley Additive exPlanations (SHAP) is a post-hoc explanation technique that draws on game theory, in particular Shapley's optimal values [11] and it supports images and text. The method by which SHAP works is to, via the SHAP kernel estimator, explain individual predictions by estimating the weight of each feature in the final prediction.

3.2.1 Shapley's values

SHAP values can be seen as a quantitative evaluation of each of the feature weights, in fact they assign an importance to the features according to certain parameters, in order to simplify the incoming input. This is done because we want to quantify how much each input data actually contributes to the final output. The desirable parameters or properties that go into determining the attribution of additive features are, as reported in [11], as follows:

- Local accuracy: if we approximate the original model f according to a certain input x, then the explanation model will have to be at least equivalent to the output of f for the reduced input x.
- Missingness: if the reduced input x expresses the existence of features, then the missing features in the original input should not affect.
- Consistency: if the model is modified to make a reduced input raise or remain stationary in spite of the others, then the attribution of that input should not reduce.

3.2.2 How does it operate?

To explain how the correct search for feature importance works, reference is made to game theory. Let's imagine a D-players game in which every feature j $\in$ {1,..., D} corresponds to a player. What we do is evaluate each player's contribution. We can count 2^D hypothetical coalitions, each one (S) connected with its own characteristic function $v : 2^D \rightarrow \mathbb{R}$.
We can calculate the Shapley value [11] of each player j as:

$$\phi_j(v) = \sum_{S \subseteq \{1,\dots,d\}/\{j\}} \frac{|S|!(D - |S| - 1)!}{D!}[v(S \cup j) - v(S)] \tag{3.2}$$

The intuition is that if a player j performs better than others, then $v(S \cup \{j\})$ overcomes $v(S)$ and consequently $\phi_j(v) \gg 0$.
The ideology behind SHAP, and like most of the explanation methods in this text,

is to locally approximate the original input model with a new prediction model, which is more understandable and interpretable.
For example, if we want to clarify the explanation $f(x)$ given by model f for the instance x, we can use a simplified instance x' and a mapping function $x = h_x(x')$, so that if $x' \approx z'$, then $g(z') \approx f(h_x(z'))$ and $g(x') = f(h_x(x')) = f(x)$. The class of additive feature attribution tecniques is described by:

$$g(z') = \phi_0 + \sum_{j=1}^{M} \phi_j z'_j \tag{3.3}$$

with $z' \in \{0,1\}^M$, such that ϕ_j is the importance of feature j. This way, we can rewrite 3.3 as:

$$\phi_j = \sum_{S \subseteq F/\{i\}} \frac{|S|!(F - |S| - 1)!}{F!} [f_{S \cup i}(x_{S \cup i}) - f_S(x_S)] \tag{3.4}$$

with f_S as the model that is (re)trained on subset S of the attributes summed F. It is clear that this is very expensive, because we would have to train $2^{|F|}$ number of models.
Moving on, we then describe the model in the class of the additive feature attribution methods that meets the properties seen above, namely the following:

$$\phi_j(f, x) = \sum_{z' \subseteq x'} \frac{|z'|!(M - |z'| - 1)!}{M!} [f_x(z') - f_x(z'/j)] \tag{3.5}$$

with $|z'|$ as the number of non-zero items in z, and $z' \subseteq x'$ as the total of z' vectors in which the non-zero items are a subset of the non-zero items in x'.
By doing so, SHAP values are characterised as the Shapley values of a conditional expectation function of the original model, i.e. $f_x(z') = \mathbb{E}[f(z)|z_S]$. If we consider a linear model with the form $f(x) = \sum_{j=1}^{M} w_j x_j + b$, then SHAP values are represented with:

$$\phi_j(f, x) = w_j(x_j - \mathbb{E}[x_j]) \tag{3.6}$$

3.2.3 Validation example with sickness score

In this validation, SHAP is compared with LIME [1] and DeepLIFT [44], a technique explained in the next subsections. All these techniques calculate feature importance values differently to produce explanations. Thus, these three methods are compared in two setups with human explanations. In the first graph (A) in Figure 3.4, the feature attribution values are compared by means of a sickness score that evaluates higher when only one of the two symptoms (fever, cough) is present. In the second graph (B), a max allocation problem is used, with profit allocation between three

different men based on the different correct answers. In both graphs you can see that SHAP predicts human agreement much better than the other techniques.

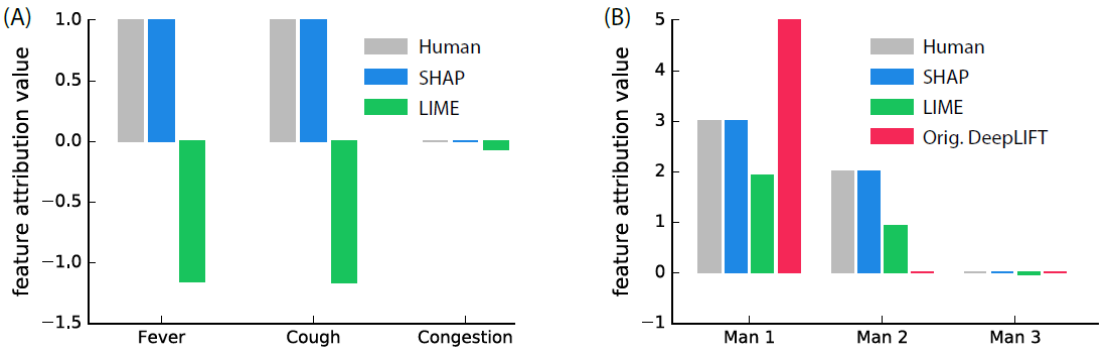


Figure 3.4: Validation example with attribution score, graph A and graph B [11].

3.2.4 Validation example with digit classification

The second validation of SHAP is proposed again by comparing it with LIME [1] and DeepLIFT [44] through the layers of an image. DeepLIFT is extended in two versions for a better approximation close to SHAP, for SHAP the Kernel SHAP approximation is used, and for LIME the output of the model is simply explained by single pixel segmentation. In Figure 3.5, in the image on the left (A) the red areas represent a high likelihood of the class, while the blue areas represent a low likelihood. In the graph on the right (B), instead, it represents the change in log-odds referred to the experiment, in which we notice that the best estimate given by the masking is of the SHAP values.
The SHAP code for experiments is available on https://github.com/slundberg/shap.

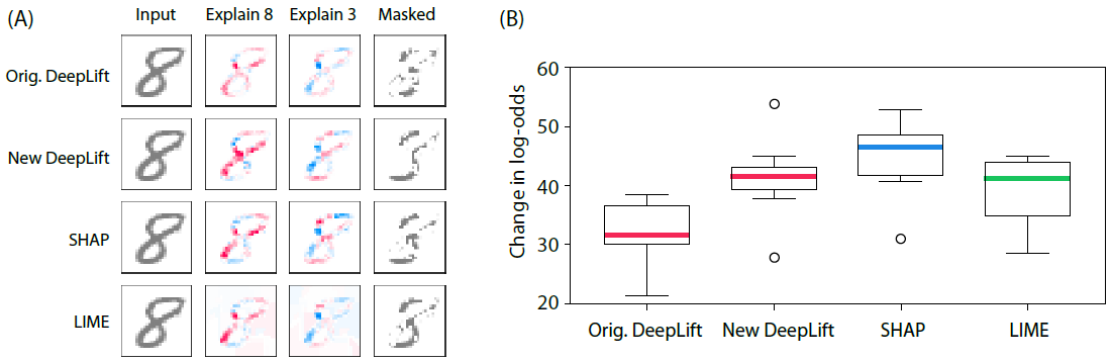


Figure 3.5: Validation example with image layers, image A and graph B [11].

3.3 LRP

Layer-wise Relevance Propagation (LRP) [43] is a post-hoc explanation technique that generally works with very complex neural networks and provides a high level of explainability, whose inputs can be tabular, images or text.
The general idea is, given a prediction on a sample of inputs, it computes on each input dimension with a relevance index, decomposing the prediction according to the sample test and propagating the prediction top-down to the neural network, using specially designed rules. Similarly to the conservation laws of Kirchoff in the circuits, the propagation that performs the LRP technique follows a conservation property, for which what is given to a neuron must be given back to the lower layer in equal quantity.

3.3.1 How does it operate?

If we place j and k as the neurons in two successive layers in the neural network, then, in a certain layer on the neurons referred to the lower one, the propagation of relevance scores $(R_k)_k$ will be given by:

$$R_j = \sum_k \frac{z_{jk}}{\sum_j z_{jk}} R_k \tag{3.7}$$

The amount z_{jk} determines the extent to which the neuron j impacts and makes relevant the neuron k. Furthermore, in the formula, the denominator has the function of exercising the so-called conservation property.
The propagation ends as soon as the input characteristics have been obtained.Using this rule on all neurons present in the neural network, one can then test the layer-level conservation property $\sum_j R_j = \sum_k R_k$, and therefore, also the global conservation property $\sum_i R_i = f(x)$, as specified in [43]. The overall LRP procedure is presented in Figure 3.6.

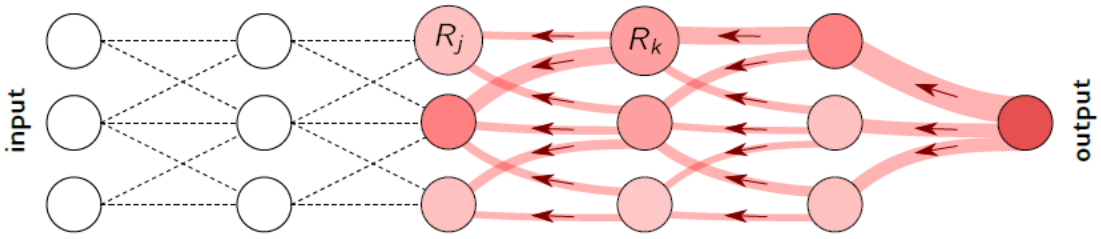


Figure 3.6: Illustration of the LRP procedure. Each neuron redistributes to the lower layer as much as it has received from the higher layer [43].

3.3.2 LRP Rules

We now discuss probably the most common LRP application today, to deep rectifier nonlinearities (ReLU) [43]. This approach also includes popular architectures for image recognition, e.g. VGG-16 deep neural networks [58].
The neurons that make up deep rectifying networks have the following structure:

$$a_k = max(0, \sum_{0,j} a_j w_{jk}) \tag{3.8}$$

The quantity $\sum_{0,j}$, moreover, is applicable on all other activations in the lower layers $(a_j)_j$, considering also an extra neuron acting as a bias.
The three propagation rules for the networks used by LRP are now described, analysing their several properties.

Basic Rule (LRP-0). This is a rule that reallocates in proportion to the contributions of all the various inputs to the activation of the neuron; it also respects the basic conditions, e.g. $(a_j = 0) \vee (w_j = 0) \Rightarrow R_j = 0$, which brings together notions like zero weight, deactivation and no connection.

Epsilon Rule (LRP-ϵ). The purpose of this rule is to obtain relevance in the case where the importances are weak in the activation of the neuron k, in fact, as ϵ increases, the weaker explanatory factors are eliminated and only the more salient ones remain. In this case we obtain explanations with less noise and less defined per input features.

Gamma Rule (LRP-γ). This rule is based on the operation of the parameter γ, which controls how many preferences there are for positive contributions over negative ones, so the greater the γ, the fewer the negative contributions. This effect is reflected in the detections in the propagation phase, resulting in much more robust explanations.

3.3.3 Validation example

The LRP technique is validated considering with the desideratas of comprehensibility and fidelity, i.e. asking this methodology for an explanation that specifically represents the output neuron and is as human-readable as possible.
Figure 3.7 shows a detailed comparison of the various LRP application rules and their explanations, applied to a VGG-16 «castle» image. The explanations visible in the figure are either generated by the uniform application of a single propagation rule to all layers, or by a composite strategy in which several rules are applied in several layers, as described in the final experiment in [43].

Differences in the various explanations can therefore be noticed. As far as Uniform LRP is concerned, LRP-0 does not stand out for either comprehensibility or fidelity. It tends to capture more local objects than the input, so the explanation is too elaborate and does not fully emphasise the main focus of the image, the castle. Then, LRP-ϵ reports an explanation which is not very comprehensible, but which nevertheless maintains an adequate fidelity: in fact, the noise has been opportunely eliminated and the resulting output underlines the figure of a castle in broad lines (unfortunately not enough). Finally, LRP-γ reports an explanation with a high comprehensibility but a poor fidelity; this is demonstrated by the visible skeleton of the castle, in which, however, the street lamp is also depicted, which distracts from the main image.
The composite LRP, on the other hand, proposes an explanation that depicts the main features of the image, and in particular only those of the castle, thus significantly surpassing the Uniform LRP explanations in both understandability and fidelity.
The LRP code for experiments is available on https://github.com/sebastian-lapuschkin/lrp_toolbox.

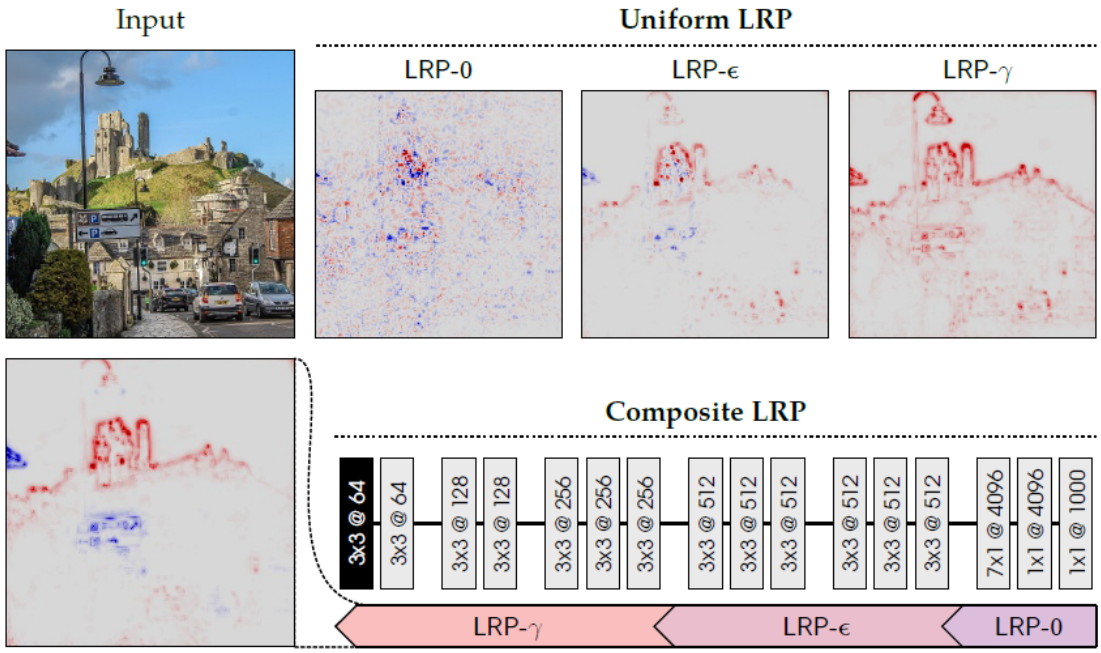


Figure 3.7: Input image and pixel-wise explanations of the output neuron 'castle' obtained with various LRP procedures. Parameters are $\epsilon = 0.25$ std and $\gamma = 0.25$. [43].

3.4 DeepLIFT

Deep Learning Important FeaTures (DeepLIFT) [44] is a post-hoc, gradient-based explanation technique and it usually supports images and text. DeepLIFT is a method that, taking a given input, divides the final decision of a neural network and reassigns all the weights of each neuron to each feature of the input itself. This approach is based, in fact, on a calculation of the scores obtained from the comparison of the differences between the actual output and a given output of reference. In this way, it is possible to transmit the information from one neuron to the other also when the gradient is equal to zero; moreover, it is possible to work separately going to consider positive and negative weights in single.

3.4.1 How does it operate?

The DeepLIFT technique attempts to describe the variation of the output from a given referenced output, based on the variation of the input from a given referenced input. This referenced input is a predefined input that is selected based on the problem being addressed.
In terms of formulas, we imagine that t is any output neuron of interest. Furthermore, we imagine that x_1, x_2, ..., x_n are neurons present in intermediate layers or even sets of layers useful to compute the output neuron t. Finally, we consider t^0 the reference activation again of the neuron t. Then, we can describe the difference with respect to the reference t - t^0 as the quantity Δ, i.e. $\Delta = t$ - t^0. DeepLIFT gives contribution scores $C_{\Delta x_i \Delta t}$ to Δx_i s.t.:

$$\sum_{i=1}^{n} C_{\Delta x_i \Delta t} = \Delta t \tag{3.9}$$

Equation 3.9 is more properly called "summation-to-delta". In fact, the factor $C_{\Delta x_i \Delta t}$ can also be seen as the quantity of difference-from-reference in t that is directly related to the difference-from-reference of x_i. Moreover, the factor $C_{\Delta x_i \Delta t}$ can be different from zero even when $\frac{\partial t}{\partial x_i}$ is zero.
In fact, as can be seen in Figure 3.8, DeepLIFT manages to overcome a major limitation typical of gradient-based techniques, in that a neuron can provide meaningful information despite the fact that its gradient may be zero. We can see in the figure a basic network that undergoes input signal saturation. In fact, when $i_1 = 1$ and $i_2 = 1$, a perturbation of one of the two factors will not necessarily change the input.

DeepLIFT succeeds in overcoming another disadvantage of gradient-based techniques, which can be seen in Figure 3.9. Gradients are usually discontinuous by themselves, which generates sudden skips in importance scores for infinitesimal changes in input. DeepLIFT, however, has a continuous reference difference, and

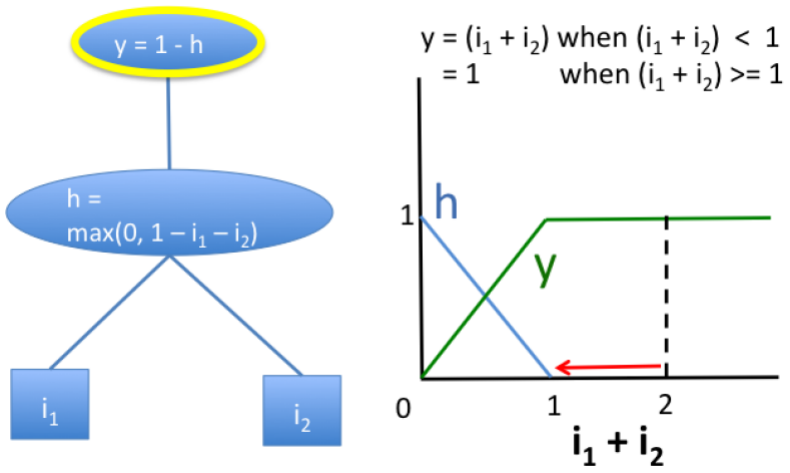


Figure 3.8: Perturbation-based approach and gradient-based approaches fail to model saturation [44].

this can be demonstrated by the example in the figure, which illustrates the response of a single linear unit rectified with a bias of -10. The gradient and the input one × both present a discontinuity at $x = 10$. At $x = 10 + \epsilon$, gradient × input gives a contribution of $10 + \epsilon$ to x and -10 to the bias, knowing that ϵ is positive number and it is very small. So, when $x < 10$, both the contributions on x and the bias are 0. On the other hand, on the top figure, the red arrow represents the difference-from-reference that allows a continuous rise in the contribution score.

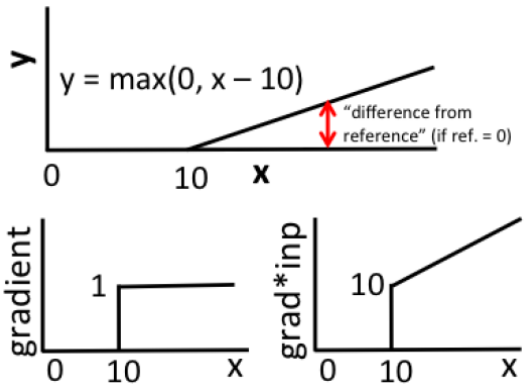


Figure 3.9: Discontinuous gradients can produce misleading importance scores [44].

3.4.2 Validation example

In this work, we show only the first validation experiment of DeepLIFT, i.e. via digit classification and a CNN. In the experiment, in fact, the starting basis is represented by two convolutional layers, succeeded by a fully connected layer and

then by the softmax output layer. Furthermore, DeepLIFT is compared mainly by different approximations of the integrated gradients technique [46]. Importance scores are calculated by identifying which pixels have to be eliminated in order to convert the image to a target class c_t, in this case about 20% of the image. In 3.10 the comparison, in which different scores are applied, is shown visually. An evaluation of the change in the log-odds score in [44] between the different classes was also made, and from this it emerges that DeepLIFT significantly outperforms all other techniques.
The DeepLIFT code for experiments is available on
https://github.com/kundajelab/deeplift.

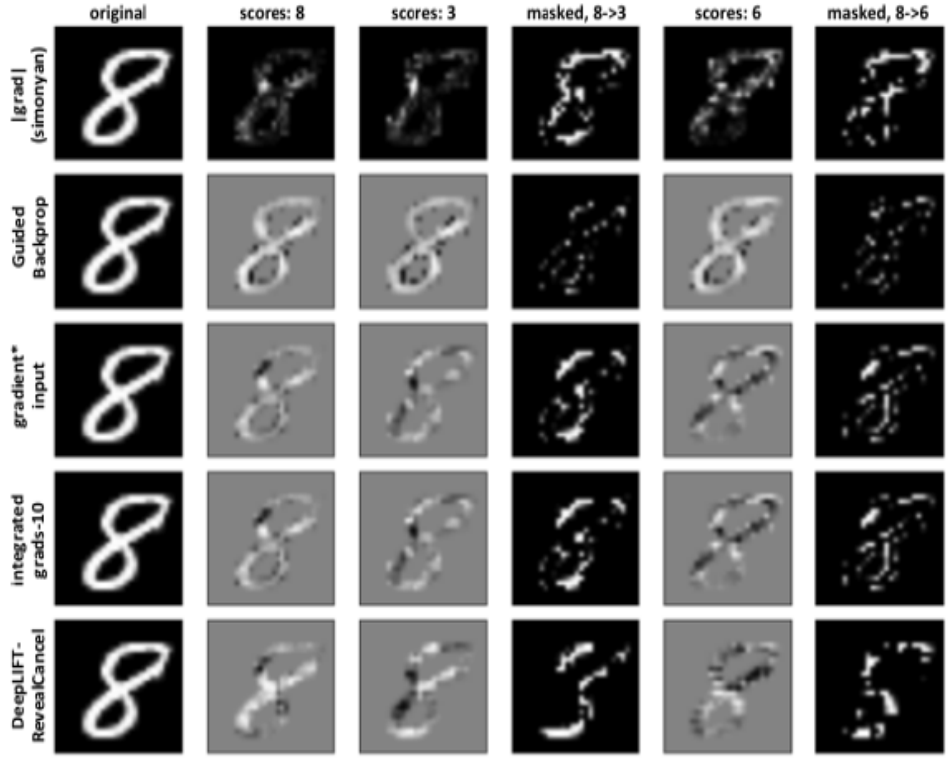


Figure 3.10: DeepLIFT validation example with digit classification [44].

3.5 Grad-CAM

Gradient-weighted Class Activation Mapping (Grad-CAM) is a post-hoc, gradient-based explanation technique and it generally supports images. Grad-CAM produces a broad location map using the gradients of the input data that end up in the convolutional layer, so as to show the visual reasons and make the prediction understood. This methodology is an extension of CAM [59], which reduced performance

for greater model transparency, but unlike this Grad-CAM method does not change the structure of the model, so it remains accurate.

3.5.1 How does it operate?

The Grad-CAM technique takes the gradient information passing through the last convutional layer of the neural network and associates weights with each neuron. Figure 3.11 represents the general operation of the Grad-CAM method. The basic intuition is that, starting with an input of an image and a class of interest, these are fed into the model and then a raw score is given for the category. A gradient of 1 is then only considered for the desired class, while all others are set to 0. Then, the rectified convection maps of the features of interest receive the signal and, as visible in the Grad-CAM box (blue heat map), combine to estimate the location and thus the decision. As a final step, the heat map is multiplied to the guided back-propagation, in order to provide higher resolution Guided Grad-CAM visualisation.

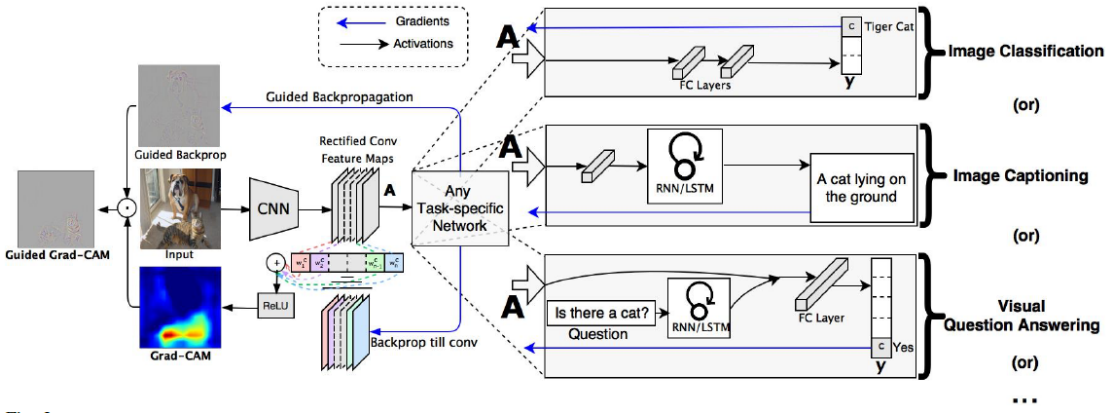


Figure 3.11: Grad-CAM overview [45].

We now describe the operations performed in the mechanism of Figure 3.11. Consider the discriminative location map of class Grad-CAM $L^c_{Grad-CAM}$ in $\mathbb{R}^{u \times v}$, which has a width u and a height v for any class c. To compute such a map, one must first look for the gradient of the score for the class c, y^c (before softmax), and then consider the feature map activations A^k of a convective layer, hence $\frac{\partial y^c}{\partial A^k}$. Furthermore, in order to find the importance weights of the neuron α^c_k, back-flowing gradients receive global average pooling in width (i) and height (j). This is described by the formula:

$$\alpha_k^c = \overbrace{\frac{1}{Z}\sum_i\sum_j}^{\text{global average pooling}} \underbrace{\frac{\partial y^c}{\partial A_{ij}^k}}_{\text{gradients via backprop}} \tag{3.10}$$

3.5.2 Validation example

The validation of the Grad-CAM method was done with particular emphasis on the relationship between interpretability and faithfulness metrics. A first validation of the method we are going to analyse was done to analyse failure modes of image classification CNNs, while the second was done to understand the effect of adversarial noise.
In this first example, first a sorting of all correct network classifications (in this case VGG-16) is done, and therefore Guided Grad-CAM is used to observe both the explanation, the correct prediction and the predicted class. In this case, in Figure X, it can be seen that the model fails to correctly predict some classes (e.g. a and d) without looking at the display of the predicted class, however Guided Grad-CAM appears to work with high resolution.

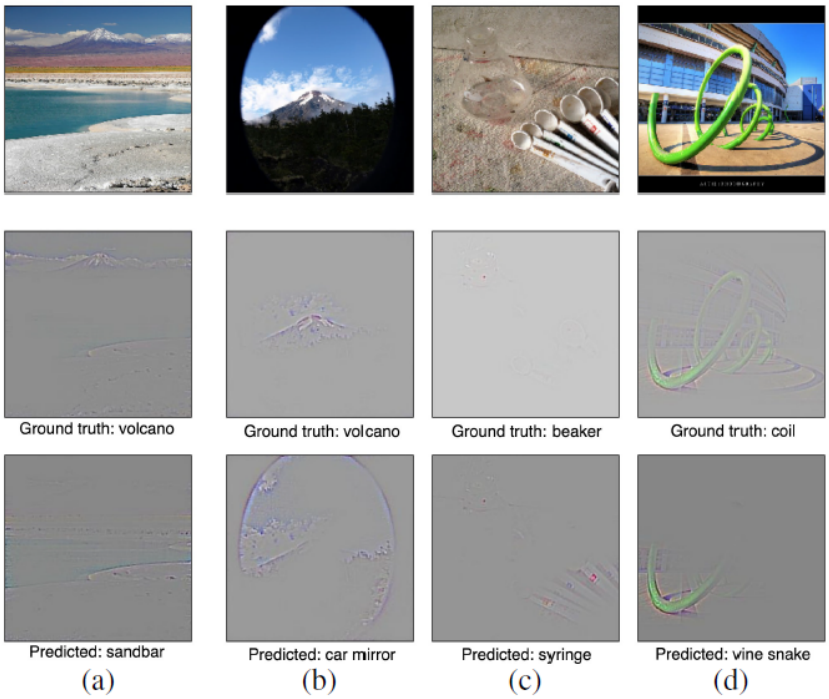


Figure 3.12: Analyzing failure modes for VGG-16 with Grad-CAM [60]

The second example focuses on solving the problem of deep networks with

adversarial examples, i.e. tiny input perturbations that trick the network into incorrectly categorising with high confidence. In this example, adversarial images are provided that have a very high probability (> 0.9999) for categories that are not even present and a low probability for categories that are present. This can be seen in Figure X. Grad-CAM is applied, which immediately identifies categories on which the model is uncertain. In fact, in figures (c) and (d), it can be seen that Grad-CAM manages to locate the actual categories very accurately, despite the model not having categorised them as necessarily present. This shows that Grad-CAM is a robust method to this issue.
The Grad-CAM code for experiments is available on
https://github.com/jacobgil/pytorch-grad-cam.

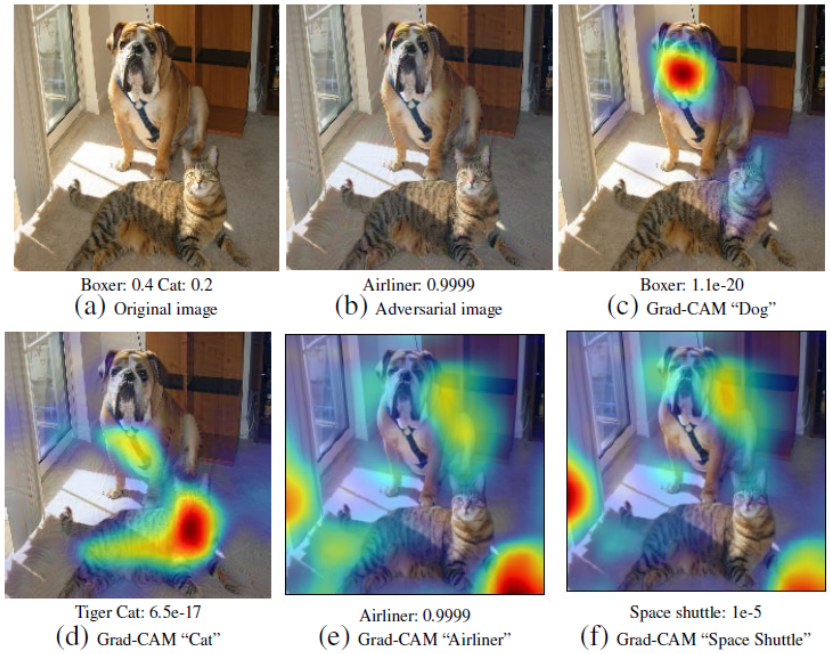


Figure 3.13: Grad-CAM resolution of effect of adversarial noise on VGG-16 [60]

3.6 T-EBAnO

Text-Explaining BlAckbox mOdels (T-EBAnO) [12] is a model-specific, perturbation-based explanation technique for local and global prediction that is based on Natural Language Processing (NLP). The general operation of this technique is that, given a deep NLP model and an input in text format, T-EBAnO derives interpretable features through model learning. It then uses the Perturbation Influence Relation (PIR) index to measure the weight of each feature in the model decision process.

3.6.1 How does it operate?

T-EBAnO justifies the inside reasoning of black-box models in NLP analytics tasks framework. The T-EBAnO local explanation process is shown in Figure 3.14 and it follows these steps:

1. An input textual document is given to the black-box model;
2. A class label is given as an ouput by the pre-trained model;
3. A combination of interpretable features is obtained by T-EBAnO;
4. It perturbs all interpretable features and sees if the model results work on the inputs. This perturbation can have several outcomes on the model, including:
   (a) The predicted probability increases: the features affected negatively the process;
   (b) The predicted probability decreases: the characteristics affected positively the process.
   (c) The predicted probability is not significantly altered: the input was not relevant to the process.

   The nPIR index also deals with quantifying the difference that occurs before the process and after the forecasting process, i.e. checking how much the effect of the perturbation has affected it.
5. The process ends with the local explanation report, which illustrates the outcomes of the perturbation.

If we then combine all the various local explanations produced by T-EBAnO, we can obtain model-global explanations that explain the behaviour of the model and its predictive process.

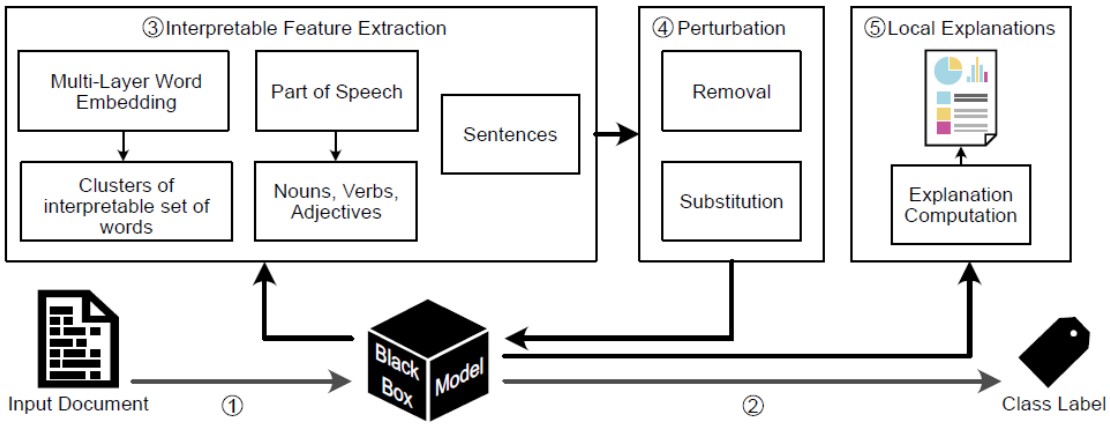


Figure 3.14: T-EBAnO local explanation process [12].

3.6.2 Validation example

The experimental results of T-EBAnO were applied in two different use cases of binary text classification, which show the great flexibility of this technique through two different NLP models, namely LSTM and BERT. The first use case consists of a binary toxic comment classification task, in which T-EBAnO explains whether a comment can turn out to be "clean" or "toxic". This experiment is done using the LSTM model and evaluated using the nPIR index. The second use case is a sentiment analysis that tries to predict whether the sentiment of a text given as input will be "positive" or "negative".
An example of a textual explanation of T-EBAnO, in this case customised with an LSTM model and labelled as "toxic" with a very high probability, is shown in 3.15. In (a) the original text is shown, while in the following experiments the more informative explanations are shown, i.e. with a combination of nouns and adjectives (b) and through a Multi-Layer Word Embedding extraction (c). It is possible to observe the most important features as they are highlighted in red.
The T-EBAnO code for experiments is available on https://github.com/EBAnO-Ecosystem/Text-EBAnO-Express.

```
Criticize a black man and the left calls you a racist.
Criticize a woman and you are a sexist.  Now I will
criticize you as a fool and you can call me intolerant.
```

(a) Original text

```
Criticize a black man and the left calls you a racist.
Criticize a woman and you are a sexist.  Now I will
criticize you as a fool and you can call me intolerant.
```

(b) `EXP1:` Adjective & Noun - POS feature extraction

```
Criticize a black man and the left calls you a racist.
Criticize a woman and you are a sexist.  Now I will
criticize you as a fool and you can call me intolerant.
```

(c) `EXP2:` Multi-layer word embedding feature extraction

Figure 3.15: T-EBAnO example of textual explanation for toxic classification task [12].

3.7 IntGrad

IntGrad is a post-hoc, gradient-based explanation tecnique and it supports text and images. IntGrad does not need any changes to the original network settings and is very easy to employ. The only thing it demands are some assessments to the standard gradient operator. IntGrad method unites the property of Implementation Invariance of Gradients along with the sensitivity of common methods, such as DeepLIFT or LRP, as well reported in [46].

3.7.1 How does it operate?

Let's imagine that we have a function $F : \mathbb{R}^n \rightarrow [0, 1]$ that stands for a deep network. Precisely, we name $x \in \mathbb{R}^n$ the available input and name $x' \in \mathbb{R}^n$ the baseline one. If we are dealing with text models, the zero embedding vector can be the baseline input, whereas for image networks it can be a black image.
Integrated gradients determine the path integral of the gradients beside the straight trajectory from the baseline input x' to x. We regard the straight trajectory in $\mathbb{R}^n$ from the baseline input x' to x. We then calculate each gradient of every point along this trajectory.
The integrated gradient for a baseline x' and an input x for the i^{th} dimension is described as:

$$IntGrad_i(x) = (x_i - x'_i) \times \int_{\alpha=0}^{1} \frac{\partial F(x' + \alpha \times (x - x'))}{\partial x_i} d\alpha \tag{3.11}$$

The gradient of *F(x)* for the i^{th} dimension is $\frac{\partial F(x)}{\partial x_i}$.

3.7.2 Validation example

This technique has been applied to some image models, some text models and a chemical model, in order to show its capacity to fix networks, to obtain rules from a network and to allow users a better understanding of the models. It should also be noted that the integrated gradient technique is relevant to a wide range of deep networks.
Sundararajan et al. [46] consider images of Diabetic Retinopathy (DR), a disease due to complications of diabetes that involves the eyes. Integrated gradients can be used to analyse the feature weight in this network, and the explanations are very important to retinal experts, who will gain confidence in the model's decisions for potential testing and screening. The baseline, as in the case of object recognition, is a black baseline image (as explained above).

Figure 3.16 illustrates an image of the integrated gradients for a retinal fundus view. The original image is displayed on the left and the overlayed gradients on a gray scale are displayed in the figure on the right. The integrated gradients on the colour channel are combined and covered on the original gray scale image on the red channel if the attribution is negative, and on the green channel if it is positive. In doing so, the integrated gradients are confined to certain pixels that may be retinal wounds. The interior of the lesions receives negative attribution while the periphery receives positive attribution indicating that the network focuses on the lesion boundary. In the original image, the lesions are visible to the naked eye, thus confirming that the attributions point correctly to them.

The IntGrad code for experiments is available on
https://github.com/ankurtaly/Integrated-Gradients/.

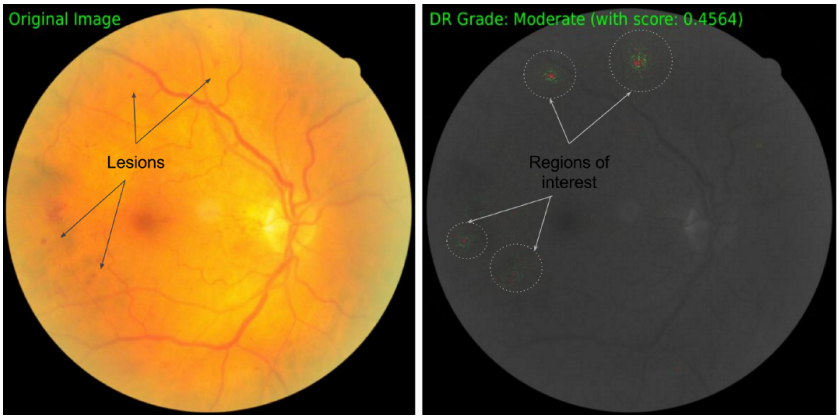


Figure 3.16: Attribution for Diabetic Retinopathy grade prediction from a retinal fundus image. [46]

3.8 RISE

Randomized Input Sampling for Explanation (RISE) [47] is a gradient-based explanation technique and it generally supports images. This method is applicable to any off-the-shelf image network, and is technically different from traditional (white-box) Grad-CAM [45] approaches, but is applicable to models for any architecture. This technique operates according to a black-box approach, and works by creating an importance map where each pixel is quantified by its weight. RISE works by empirically testing the model with different versions of the input image to obtain increasingly decipherable output. The input image is in fact sub-sampled using random masks, which are saved as output. The final saliency map is a linear combination of all the masked images in the output. A schematic of RISE operation can be seen in Figure 3.17.

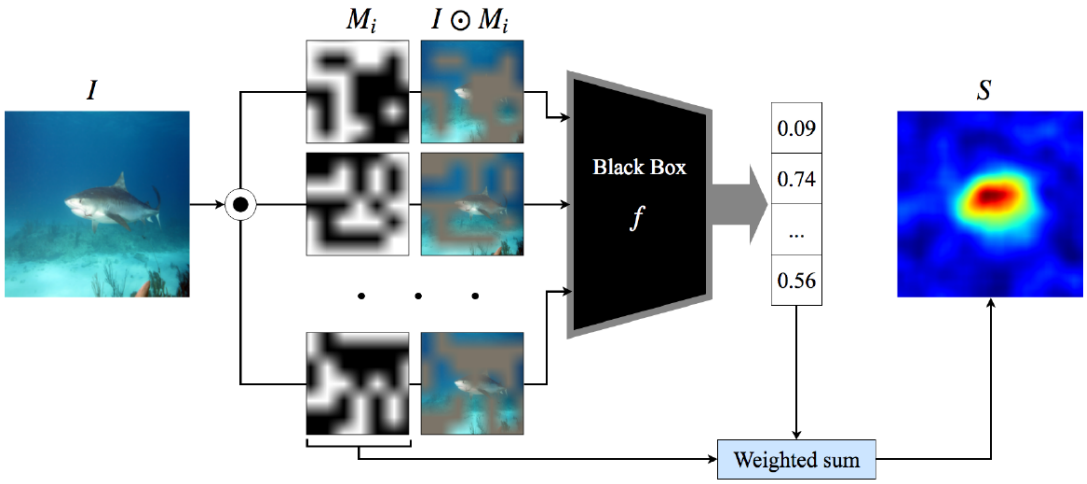


Figure 3.17: Summary of RISE: the input image I is sub-sampled using random masks M_i and the masked images are shown with the output [47].

3.8.1 How does it operate?

RISE, as seen before and reported in [47], creates saliency maps from the sub-sampling of random masks. The decisive saliency map is the weighted sum of the random masks, with the importances being the resultant output on the node of interests:

$$E_{RISE}(I, f)_c = \sum_i f_c(I \odot M_i)M_i \tag{3.12}$$

If $f : I \to \mathbb{R}$ is a black-box model and I is the input, then M_i is the random mask and $\odot$ is the element-wise product in spatial elements. The concept behind this is that if a mask maintains significant parts of the image, it receives a higher sum on the output, and thus a higher importance and more prevalent influence on the decisive saliency map.

3.8.2 Validation example

An initial comparison was also made by the authors by taking an input image and comparing the RISE method with the Grad-CAM and LIME methods, using the deletion metric. This metric measures the loss of probability of a class that important weights (in this case pixels) are removed: if the probability curve shows a small area, this indicates that the explanation was good. In this example, Figure 3.18, RISE gets more accurate saliency and performs the lowest deletion score.
Another metric examined by the authors is the insertion metric, which studies the weight of pixels based on their ability to sum up the image. This metric is

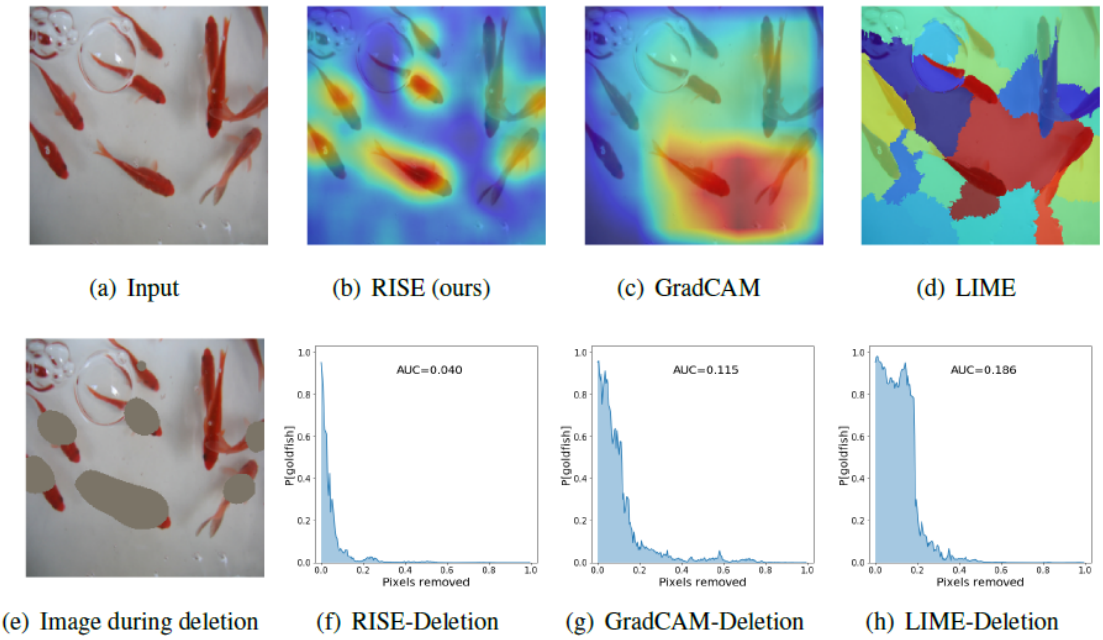


Figure 3.18: Comparison of RISE and other state of the art methods through deletion score (AUC) [47].

measured by increasing the probability of the class of interest as pixels are added to the saliency map.

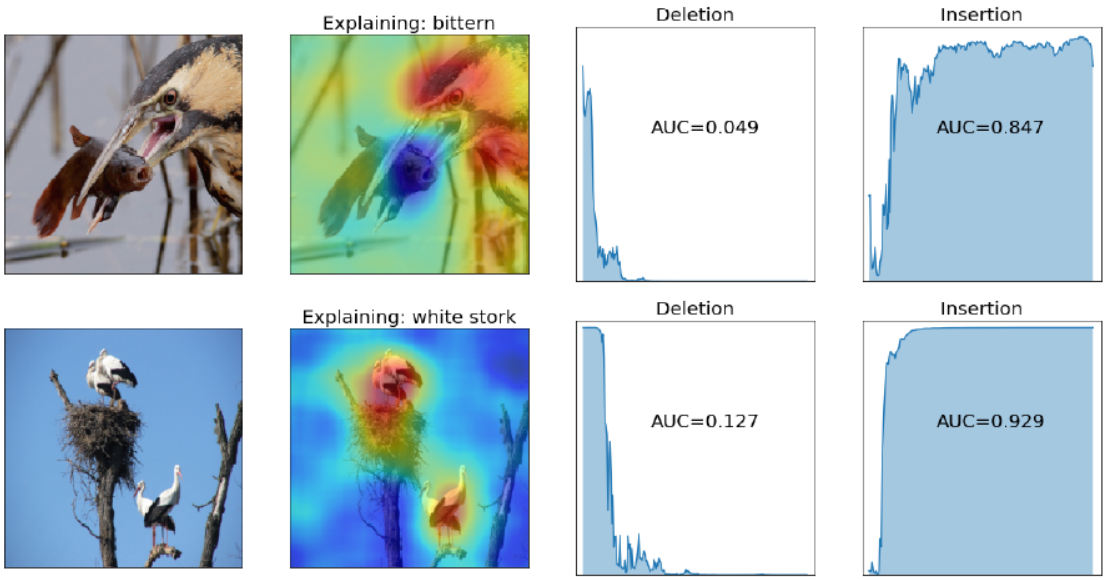


Figure 3.19: The input images (first column) are turned in saliency maps (second column) thanks to RISE tecnique with graphs of deletion (third column) and insertion (fourth column) [47].

Subsequently, the authors perform extensive experiments on several benchmark datasets, in which it is shown that RISE equals or exceeds the performance of other methods, including white-box approaches. In particular, they evaluated it on 3 object classification datasets and tested the saliency maps created by different explanation methods for a target object category of the images.
The RISE code for experiments is available on https://github.com/eclique/RISE.

3.9 Anchors

Anchors [48] is a post-hoc, perturbation-based explanation technique and it supports text and images. Anchor is an algorithm that efficiently computes explanations for any black-box model by means of precise rules named "anchors", based on the "if-then" model. These rules imply a firm local explanation, such that variations in the weight of the instance are not relevant. Figure 3.20 shows an example of sentiment prediction in order to immediately understand the difference with the LIME method [1], also developed by the same authors. The instances in question are "not good" and "not bad": while LIME explanations are weighted with scores and detached from positive and negative sentiment, in the Anchors method sentiment is predicted with "anchors", so they are clear and easy to understand.

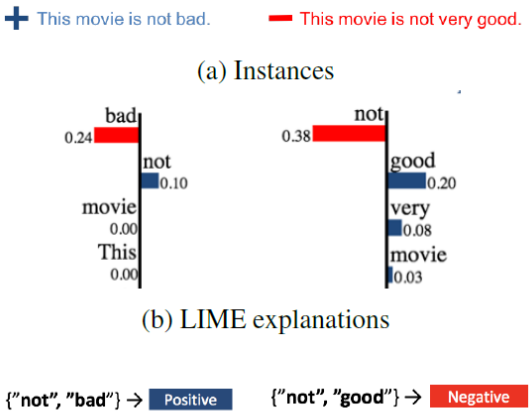


Figure 3.20: Sentiment prediction with LSTM of two sentences with LIME [1] and Anchors [48]

3.9.1 How does it operate?

Imagine that A is a rule (i.e., a group of predicates) working on an interpretable representation, for which it holds that if all feature predicates are true for instance x, then $A(x)$ returns 1. Ribeiro et al. [48] set the following example, let instance x = "This movie is not bad", so $f(x) = Positive$ with a sentiment prediction, and $A(x)$

= 1, where A = "not", "bad". Then, let D be the perturbation and $\mathcal{D}(\cdot|A)$ indicate the conditional distribution if rule A is employed. Rule A becomes an anchor if *A(x)* = 1 and A is a sufficient term for *f(x)* with high probability, if a model z of D(z|A) is calculated *Positive*, so *f(x)* = *f(z)*. Officially, A becomes an anchor if:

$$\mathbb{E}_{\mathcal{D}(z|A)}[\mathbb{1}_{f(x)=f(z)}] \geq \tau, A(x) = 1 \tag{3.13}$$

3.9.2 Validation example

The study of Anchor has allowed the authors to use it in several different models and tasks of machine learning, such as classification, text generation and prediction, applying it to several different domains, such as tables, text and images. The study carried out is user-based and shows that users are able to predict the behaviour of the model based on unseen instances, more easily and more accurately than other techniques seen previously. Tests were also carried out on simulated users, using tabular datasets, so that data is modelled from the training set and explanations are derived from instances in the validation set, then measured on instances in the test set. These studies have resulted in the fact that users find it easier to understand anchor explanations rather than linear explanations: this is because anchors are easier to apply and this is also demonstrated in user feedback and application times.
The Anchor code for experiments is available on
https://github.com/marcotcr/anchor-experiments.

3.10 SmoothGrad

SmoothGrad [49] is a gradient-based explanation technique and it generally supports images. SmoothGrad is a technique that sharpens sensitivity maps based on gradients for humans, lessening visual noise, and with the additional possibility of implementing other sensitivity map models. The basic intuition of SmoothGrad is to receive an input image, add noise to this image by sampling comparable images and finally, for every sampled image, calculate the average of the final sensitivity maps.

3.10.1 How does it operate?

Let's imagine a model that categorizes an image in one class from a group C. Let the input image be x, then the image classification networks calculate a class activation function S_c for every class $c \in C$, and the last classification *class(x)* is decided by the class that has the highest total:

$$class(x) = argmax_{c \in C} S_c(x) \tag{3.14}$$

So, we can create a sensitivity map $M_c(x)$ for every image, when of course the functions S_c are piecewise differentiable, only by differentiating M_c taking the input into account. This way, we describe:

$$M_c(x) = \frac{\partial S_c(x)}{\partial x} \tag{3.15}$$

in which ∂S_c is the derivative (gradient) of S_c and M_c denotes the small variation every pixel of x would get to the classification record for class c.
The work done by Smilkov et al. [49] is to generate superior sensitivity maps, in fact they build the visualization on a smoothing of ∂S_c with a Gaussian kernel instead of doing it simply on the gradient ∂S_c. They calculate a stochastic approximation, getting casual samples in a neighbourhood of the input x and averaging the subsequent sensitivity maps:

$$M_c(x) = \frac{1}{n} \sum_{1}^{n} M_c(x + \mathcal{N}(0, \sigma^2)) \tag{3.16}$$

in which n is the number of samples taken into account and $\mathcal{N}(0, \sigma^2)$ is the Gaussian noise.

3.10.2 Validation example

In order to validate the SmoothGrad method, various experiments were carried out with the help of a neural network suitable for image classification, measuring the level of noise added. Figure 3.21 shows an example of the effects of the sharpness of the sensitivity maps by adding Gaussian noise $\mathcal{N}(0, \sigma^2)$.

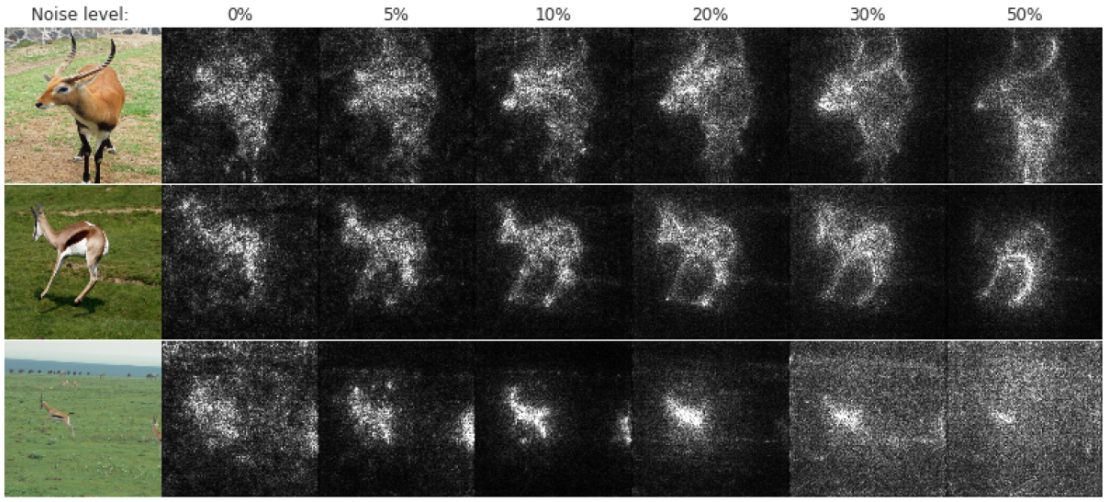


Figure 3.21: Effect of noise level on gazelle images [49].

All the experiments carried out by the authors lead to the conclusion that the gradient estimation M_c, a novelty introduced with this tecnique, returns sensitivity maps that are visually much more compact than the gradients previously used. The SmoothGrad code for experiments is available on https://github.com/hs2k/pytorch-smoothgrad.

3.11 SENN

Self-Explaining Neural Networks (SENN) [50] is a local explanation technique and it supports texts and images. SENN is interpretable through the underlying regularisation scheme, and is also similar to a linear model in terms of local behaviour. The SENN is composed by three parts, as reported in Figure 3.22:

- a concept encoder (the green part): it converts the input into a small group of interpretable basis features;
- an input-dependent parametrizer (the orange part): it creates relevance grades;
- an aggregation function: it unites to make a prediction.

The robustness deficit on the parametrizer ensure that the full model computes locally as a linear one on $h(x)$ with parameters $\theta(x)$.

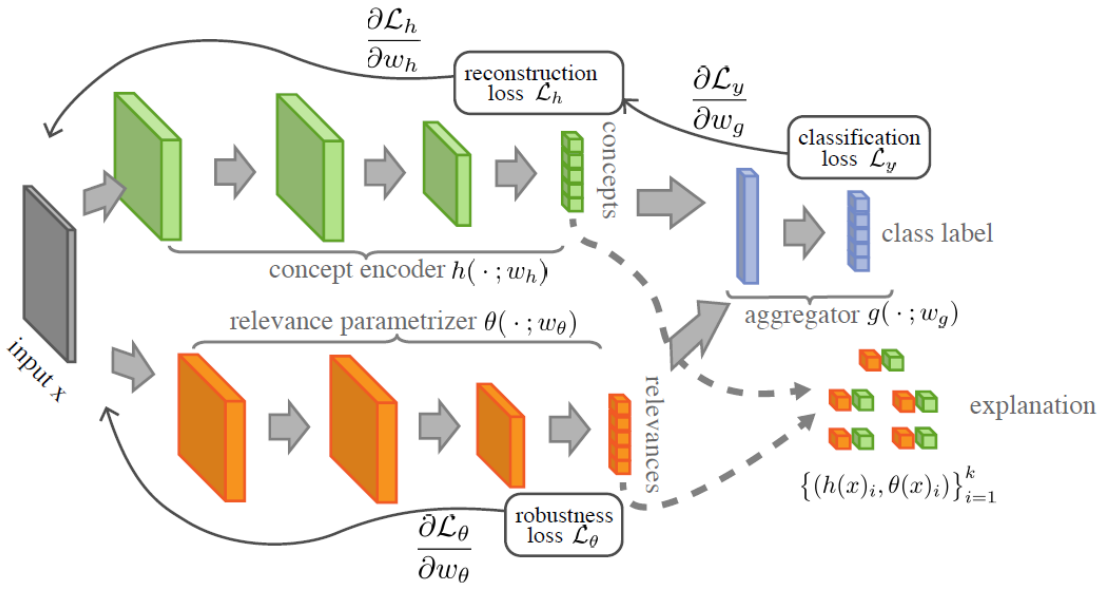


Figure 3.22: Overview of SENN [50].

Moreover, the coefficients of this model change slightly around all inputs (locally), effectively keeping the model to a linear model: all this is done to ensure the stability of the model. In this way, the resulting model is a very complex interpretable model, but one that preserves the desirable characteristics of normal linear models and does not lose performance.

3.11.1 How does it operate?

Consider $x \in \mathcal{X} \subseteq \mathbb{R}^n$ and $\mathcal{X} \subseteq \mathbb{R}^m$ input and output ranges. Then, let $f : \mathcal{X} \to \mathcal{Y}$ be a "self-explaining prediction model" with the following expression:

$$f(x) = g(\theta(x)_1 h(x)_1), ..., \theta(x)_k h(x)_k) \tag{3.17}$$

in which g is a monotone function and wholly additively separable, θ has a bound with local difference by h and $h(x)$ is an interpretable representation of x. This class of functions also contains linear predictors, e.g., nearest-neighbor classifiers and generalized linear models. By the way, the true strength of the models show in this formula occurs when $\theta(\cdot)$ is made by architectures with large modeling capacity; and when $\theta(\cdot)$ is made with a neural network, we rely to f as a Self-Explanaining Neural Network.

3.11.2 Validation example

This structure, in light of the experimental results obtained by the authors, allows to combine interpretability and complexity of the models. The validation was done for datasets whose models behave equally if not modular and not interpretable; moreover, the evaluation was done based on three fundamental criteria, namely explicitness and intelligibility, fidelity and stability. Figure 3.23 shows an excerpt from [50] in which the SENN method was compared to very popular methods for their interpretability, namely LIME [1], several gradient-based methods, e.g. IntGrad [46] and LRP [43], also seen previously.

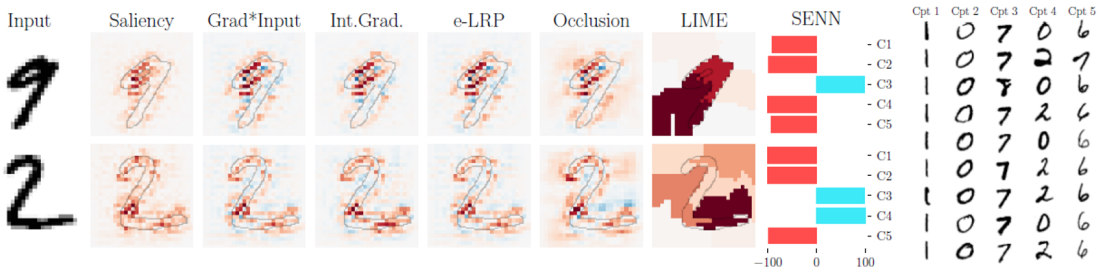


Figure 3.23: A comparison of SENN's concept-base tecnique and common input-based ones [50].

SENN's explanations differ from others in that they provide a characterization of the proposed input concepts in the form of prototype descriptions. The evolution brought by SENN is to go against the classical notions of interpretability provided by the proposed methods, in order to evolve future complex architectures and at the same time obtain desiderata already intrinsic to the model, i.e., those previously proposed in the experiments. Thus, the basic idea of the authors was to show that this type of architecture can generate very complex and interpretable models that provide equally powerful explanations.
The SENN code for experiments is available on https://github.com/dmelis/SENN.

3.12 SITE

Self-Interpretable model with Transformation Equivariant Interpretation (SITE) [51] is a post-hoc explanation technique and it supports text and images. SITE is a self-interpretable model, i.e. it is able to provide predictions and at the same time make them interpretable. In fact, with this technique we generate input-dependent prototypes for each class and make the prediction so that it is a kind of product between the features extracted from the model and each prototype; and by upsampling it is possible to see the various interpretations. SITE differs from other methodologies in that it provides understandable and interpretable interpretations while maintaining uncommon prediction power, and does not need extra domain

knowledge. Figure X shows how SITE works. The model can receive as input both the original image x and the transformed image $T(x)$. This is immediately sent to the feature extractor F_1, after which the model, via the generator G, formulates the prototypes. At this point, both the prediction and the interpretation are formulated via the product $\odot$ between each prototype and the hidden representation $F_1(T(x))$.

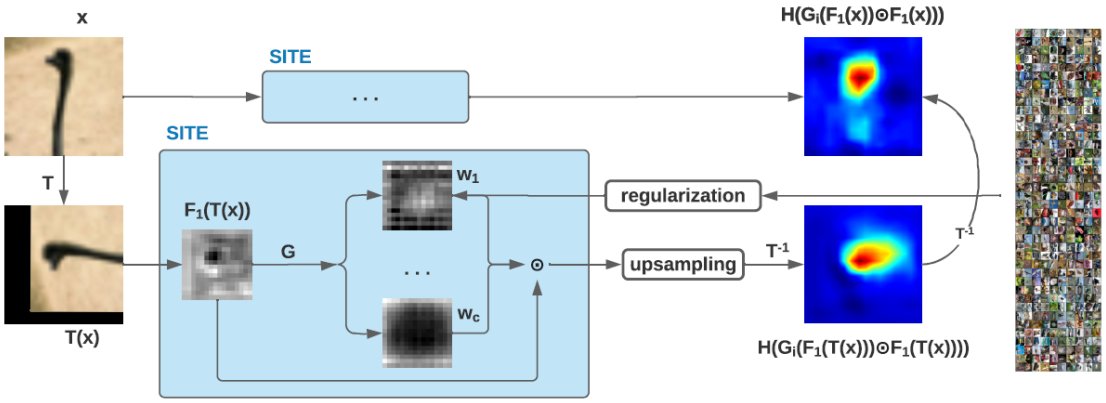


Figure 3.24: Overview of the model SITE [51].

3.12.1 How does it operate?

Here we show the example formula for image classification. Let $x \in \mathbb{R}^p$ represent the input image and let one-hot vector $y \in [0,1]^c$ represent the expected class probabilities. Then, consider p the result of the total channels, the width, and height of the image x, and c the total classes. Thus, a normal classifier $F : x \mapsto y$ can be split into $F = F_2 \circ F_1$ with the classifier F_2 and the feature extractor F_1, in which $F_1 : x \mapsto z$ and $F_2 : z \mapsto \hat{y}$. F_1 is typically made of convutional neural networks, and F_2 involves fully connected layers. Then, $z \in \mathbb{R}^d$ represents the extracted hidden representations of x and it is generally smaller because $d < p$. The aim of the model is to lessen the classification loss, so the formula is the following:

$$\min_{F=F_2 \circ F_1} \mathbb{E}_{x \in \mathcal{X}, y \in \mathcal{Y}} L_{ce}(F(x), y) \tag{3.18}$$

in which $\mathcal{X}$ is the input data set and $\mathcal{Y}$ is the target set, and L_{ce} represents the cross-entropy loss function.

3.12.2 Validation example

The validation of SITE is done through experiments in which its quality of interpretation and prediction is affirmed. The first experiment is carried out with three images (an aeroplane, a dog and a car), with which it is demonstrated by means of

appropriate transformations with the model that SITE highlights the main parts of the objects mentioned in each image very well, making them comprehensible also for the human. This is represented in Figure 3.25, where the first rows illustrate the input images and their arbitrary changed version, while the second rows illustrate the parallel interpretation heat maps.

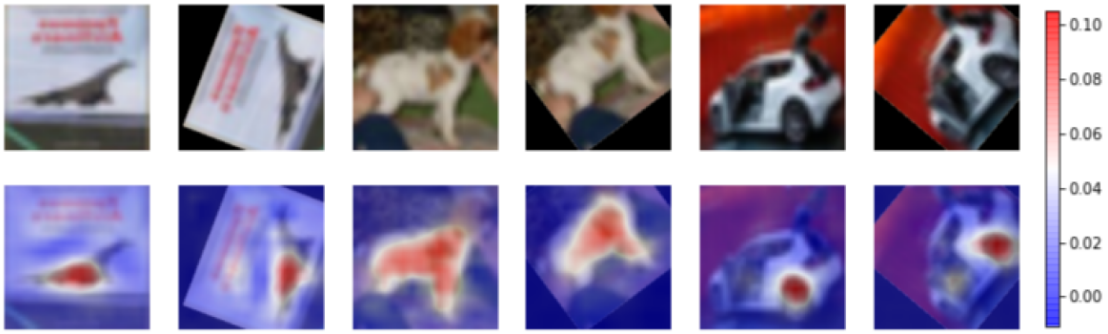


Figure 3.25: Interpretation example of SITE [51].

Afterwards, a comparative experiment was carried out with other common explanation techniques, such as techniques already discussed in this thesis (Grad-CAM [45], IntGrad [46] and RISE [47]). In this experiment, visible in Figure 3.26, it is possible to observe in the first and third row the interpretation of the various models on an input image, while in the second and fourth row the interpretation is made on the transformed image. It can be seen from this example that many models provide interpretations with noise, or do not work correctly; on the contrary, SITE maintains almost all the transformation and the interpretation is correct.

Figure 3.26: SITE comparison with other common explanation tecniques [51].

3.13 VA-GAN

Visual Attribution based on Generative Adversarial Networks (VA-GAN) [52] is an intrinsic explanation technique and it usually supports images. VA-GAN is a technique based on feature attribution and relies on Wasserstein Generative Adversarial Networks (WGAN) [61], which are particular because they lessen an approximation of the Wasserstein distance between the generated and real image distributions. The peculiarity of this method is that it does not rely on any classifier (or use one trained independently or by an expert), but works by means of a map that is totally different from the images of a base category when attached to the input image of a category. For this reason, this technique needs a base category, which is common for medical images: in fact, this technique works remarkably well

for synthetic data sets and real neuroimaging data.

3.13.1 How does it operate?

With the VA-GAN technique we estimate a map that tries to highlight areas in an image that are peculiar to it. We prepare the method with two classes $c \in 0, 1$: a base class and a class of interest. We represent an image with x and the distribution of images from the class $c = 0$ with $p_d(x|c = 0)$ and images from the class $c = 1$ with $p_d(x|c = 1)$. The formula of the method is to estimate of a map function $M(x)$ that generates an image, when added to an image x_i of category $c = 1$, thus:

$$y_i = x_i + M(x_i) \tag{3.19}$$

which is identical from the images sampled by $p_d(x|c = 0)$. So, the map $M(x_i)$ has all the features that characterize the input image x_i from the other category.

3.13.2 Validation example

This method is validated specifically for medical fields, as mentioned above, on synthetic 2D data and on 3D MRI data, for which we aim to find some more peculiar diseases, such as Alzheimer's disease (AD). With this validation, the technique is compared with common state-of-the-art methods discussed in this thesis, e.g. IntGrad and CAM (more specifically we focused on sub-techniques). Examples of the above-mentioned estimated disease effect maps can be seen in Figure X. For the back-propagation methods, it was found that the network compressed the least predictive features, thus focusing mainly on the edges rather than the whole object. The CAM method, on the other hand, has a reduced spatial resolution for its computation. We show, therefore, that the VA-GAN method produces much more localised effect maps for this disease, by being able to focus on whole squares and correctly target edges, accurately identifying both focal points representing the disease.

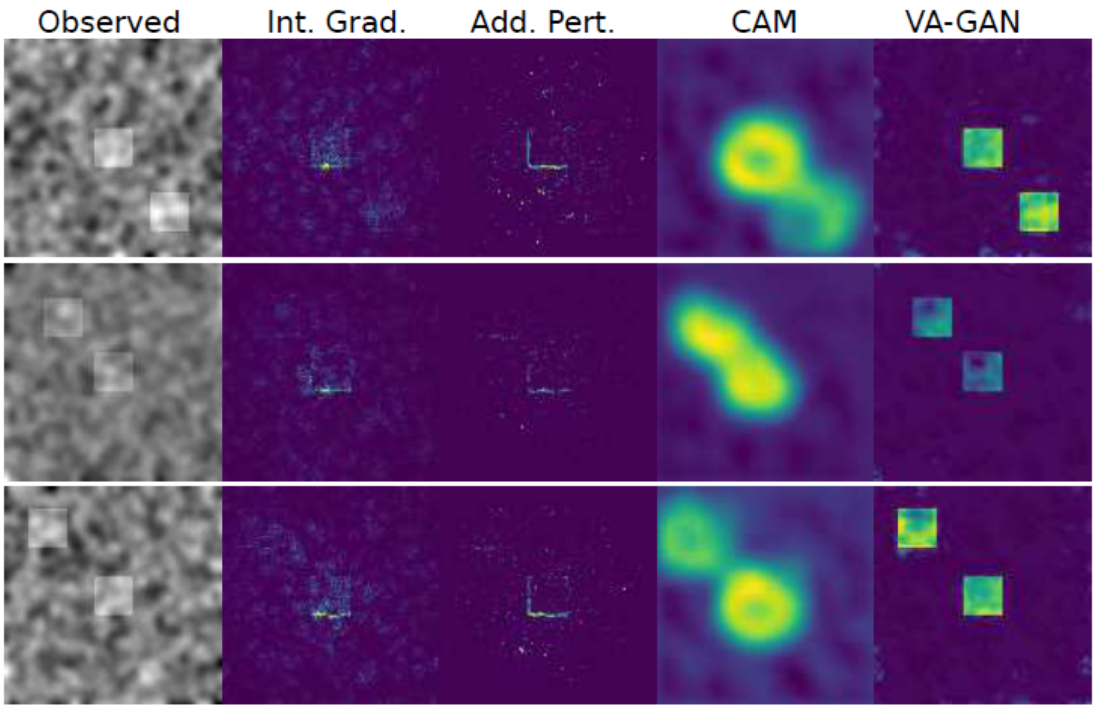


Figure 3.27: VA-GAN comparison with other common explanation tecniques [52].

The VA-GAN code for experiments is available on
https://github.com/baumgach/vagan-code.

3.14 ICAM

Interpretable Classification via disentangled representations and feature Attribution Mapping (ICAM) [53] is an intrinsic explanation technique and it usually supports images. ICAM is a method of feature assignment that is characterised by its interpretability and the presence of very efficient feature assignment maps. ICAM, in fact, is based on an image-to-image translation structure, so that it performs feature attribution by distinguishing attributes that are important to the class from those that are not. This explanation technique uses a classifier that converts the inputs of several classes into a discrete latent space, and a generator that combines the feature attribution maps with all the important features for the class. Figure 3.28 shows a schematic of ICAM's operation, in which it performs classification with attribute map generation for two images given as input x and y.

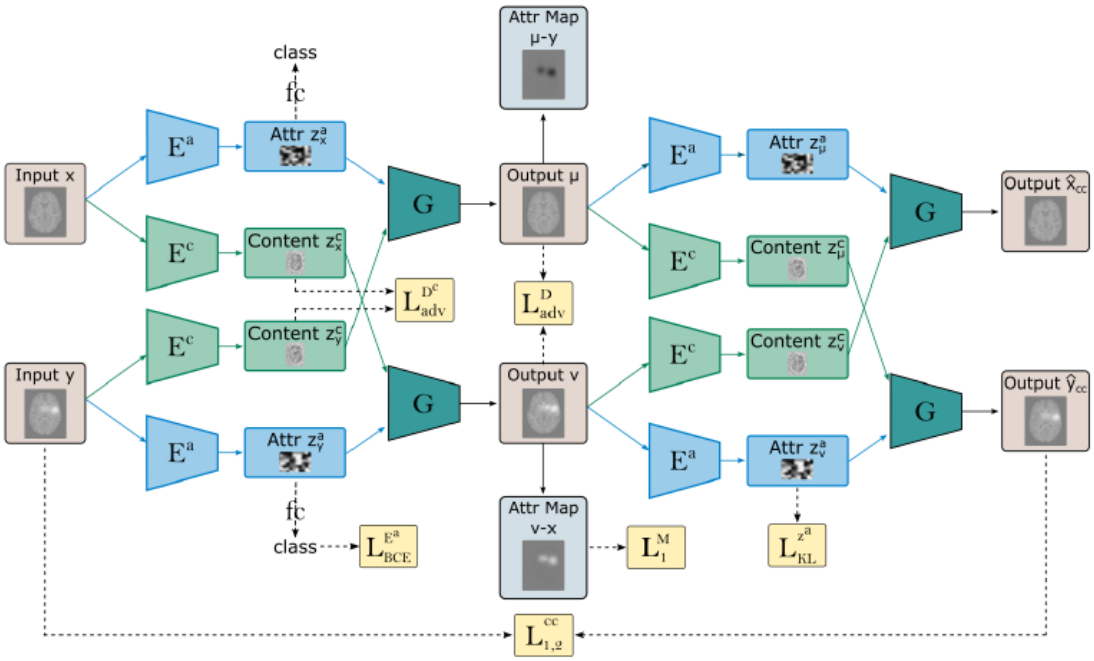


Figure 3.28: Overview of ICAM method [53].

3.14.1 How does it operate?

To explain the operations behind ICAM we consider a content encoder E^c that converts class-irrelevant information from a common content latent space $z_x^c, z_y^c \in C$, through the request of a content discriminator D^c, whose aim is to categorize the classes and domains. For input images$x$, y of classes c_x and c_y, respectively, the aim of the encoder E^c is to trick the discriminator to categorize an input improperly $(E^c : x \rightarrow z_x^c)$, $(E^c : y \rightarrow z_y^c)$. Let's consider then an attribute encoder E^a that acquires all pertinent class information and categorizes among domains $(E^a : x \rightarrow z_x^a \rightarrow c_x)$, $(E^a : y \rightarrow z_y^a \rightarrow c_y)$ working with an entirely connected/dense layer that is employed to the common attribute latent space $z_x^a, z_y^a \in A$. Then, we consider a generator G that combines an image trained on the content and on attribute latent spaces by switching the content latent space. By changing the domains we can see the distinctions between the two classes, thanks to the feature attribution map: $M_x = v$ - x, $M_y = \mu$ - y. Ultimately, the domain discriminator D differentiates the generated and the original images and categorizes the two domains.

3.14.2 Validation example

The validation of ICAM was done on several datasets that focused on medical studies, specifically ablation on 2D simulations, estimation of the accuracy of the created attribution maps, and study of the flexibility of the method to analyse

phenotypic variation. What emerged was that ICAM presents itself as the single technique that creates feature attribution maps from latent attribute (class-relevant) and content (class-irrelevant) spaces. Moreover, its strongly interpretable latent space grants full evaluation of phenotypic variability by analysing variance and mean feature attribution maps. An example of this result is shown in Figure X, in which ICAM is compared with VA-GAN [52], a method very similar in application and operation, but with differences in attribute classification and variance analysis. The image shows a modelling of healthy ageing, in which a transformation is made from an old to a young individual. It can be seen that ICAM has a better detection in the objects in question, namely in the cortex (green), in the ventricles (blue) and in the hippocampus (pink). It can also be seen that, while VA-GAN only makes small variations to the intensity of the pixels, ICAM is instead capable of adjusting the form of the various regions of the brain.
The ICAM code for experiments is available on
https://github.com/CherBass/ICAM.

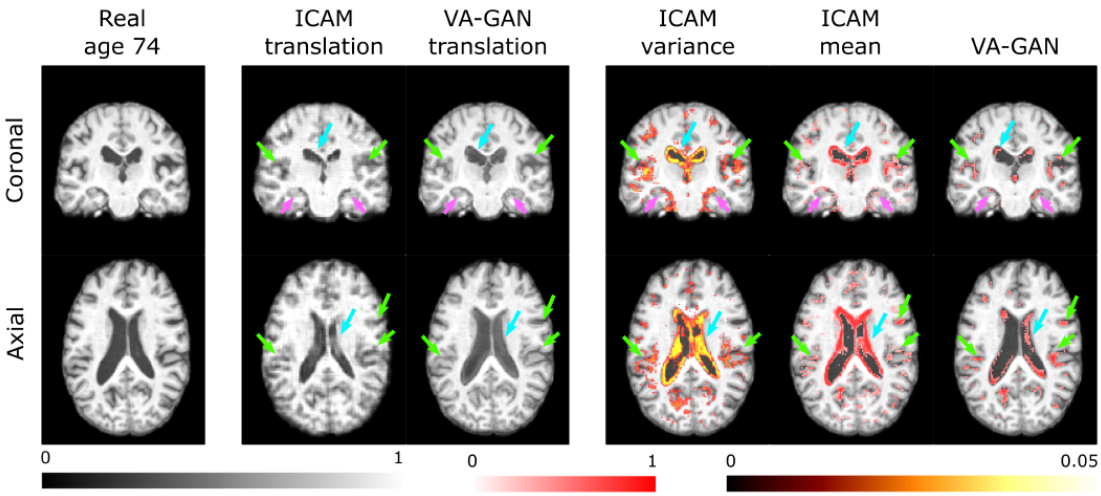


Figure 3.29: Comparison of ICAM and VA-GAN [53].

3.15 Archipelago

Archipelago [54] is a post-hoc explanation technique and it supports text and images. Archipelago is based on the attribution and identification of interactions, and it's named after its ability to give explanations by isolating feature interactions, or to be more clear, 'islands' of features.This technique differs from the others in that it better finds the most important interactions and is much more interpretable e.g. when used on annotation labels in sentiment analysis or even in image classification.

3.15.1 How does it operate?

Archipelago receives as input a black-box model f and a data set x^*. The output it will return is composed of a set of interactions and individual features $\mathcal{I}$ and for each set of features an attribution score $\phi(\mathcal{I})$ will be assigned. Archipelago is divided as mentioned into two methods: ArchAttribute, the method of attributing interactions, and ArchDetect, the corresponding method that detects interactions. To explain ArchAttribute, consider $\mathcal{I}$ as the set of feature indicators that match with a required attribution score. ArchAttribute is the anticipated attribution score, and it is represented by:

$$\phi(\mathcal{I}) = f(x^*_{\mathcal{I}} + x'_{\mathcal{I}}) - f(x') \tag{3.20}$$

Basically, it isolates the attribution of $x^*_{\mathcal{I}}$ from the adjacent baseline content $x'_{\mathcal{I}}$ and as reported above we name this isolation the "island effect".
ArchDetect, instead, works in pair with ArchAttribute, considering the interaction strength $\omega_{i,j}(x)$ between two features i and j for the context $x_{i,j}$. ArchDetect's work for the pairwise interaction detection is given by:

$$\bar{\omega}_{i,j} = \frac{1}{2}(\omega_{i,j}(x^*) + \omega_{i,j}(x')) \tag{3.21}$$

3.15.2 Validation example

The validation of Archipelago was carried out on text, specifically for a sentiment analysis, and on image classification. Based on the evaluations made by the authors, it can be shown that Archipelago proved to be highly interpretable in terms of explanations and that the model proves to be reliable and consistent.
The first experiment, shown in Figure 3.30, involves text input and is based on BERT visualisation with casual phrases with BERT tokenization. It can be seen that the arrows imply interactions and the colours denote attribution scores, while on the right hand side the sentiment classification ($f_c ls$) is represented. It is observable that interactions indicate relevant and sometimes long-range word sets, and colours look sensible.

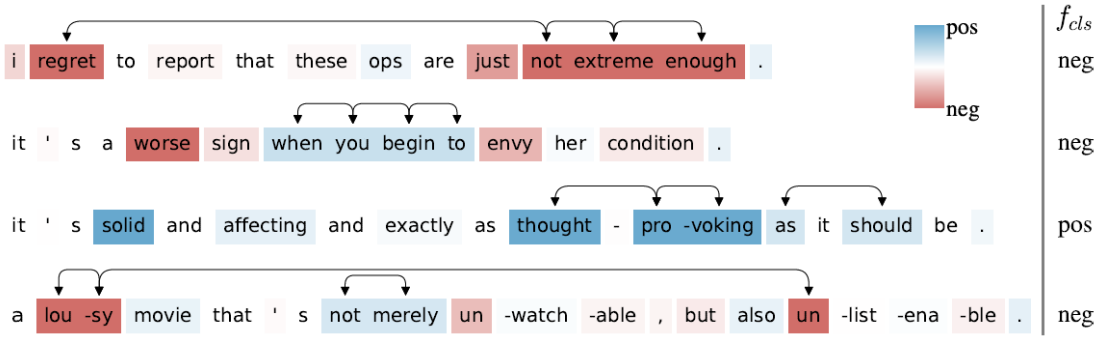


Figure 3.30: Archipelago's text sentences experiment [54].

The second experiment, shown in Figure 3.31, concerns an image input with a COVID-19 classifier working on casual lung X-rays classified as positive. It can be seen that the coloured contours show the feature sets identified with positive attribution and the interactions, on the other hand, almost all focus on the heart great vessel region delineated in green.
The Archipelago code for experiments is available on
https://github.com/mtsang/archipelago.

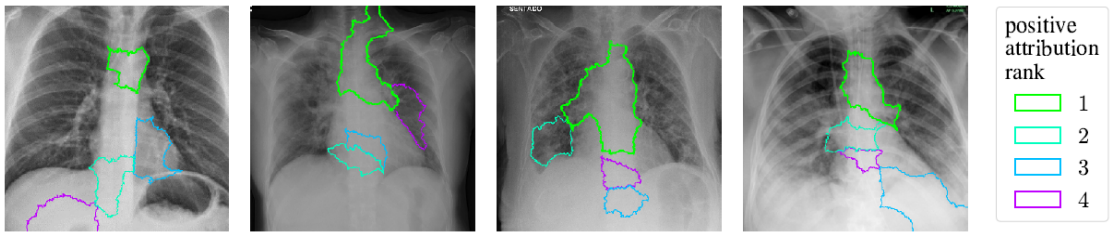


Figure 3.31: Archipelago's image classifier experiment [53].

3.16 Mahé

Model-agnostic hierarchical explanations (Mahé) [55] is a local, post-hoc explanation technique and it supports text and images. Mahé, as its name implies, offers model-independent but context-dependent hierarchical explanations; this is done through a local interpretation algorithm that easily processes all interactions of diverse order and acquires context-free explanations through the simplification of context-dependent interactions to predict global behaviour. Going deeper, Mahé receives as input a data instance and a model to be explained, and returns as output a hierarchical explanation, which succeeds in indicating the local group-variable relationships that are employed in the explanation. In the case of context-free explanations, on the other hand, Mahé receives as input a model and representative data corresponding to an interaction of interest and returns it as output if this

interaction is context-free. Thus, Mahé differs from methods in the literature in that it refines context-dependent explanations based on interaction detection, adaptation performance and model generalisation, and it is also the first technique to offer context-free explanations of interactions in deep learning models.
Figure 3.32 provides an overview of how context-dependent hierarchical explanation works. In the first step, a data instance is given as a input into the model (e.g. a classifier). In the second steps, the model locally perturbs the input to make the prediction. This diagram illustrates how, in contrast to LIME, Mahé uses a neural network to learn the nonlinear decision margin used to categorise the instance. Finally, the third step represents the attribution score of the data instance interactions.

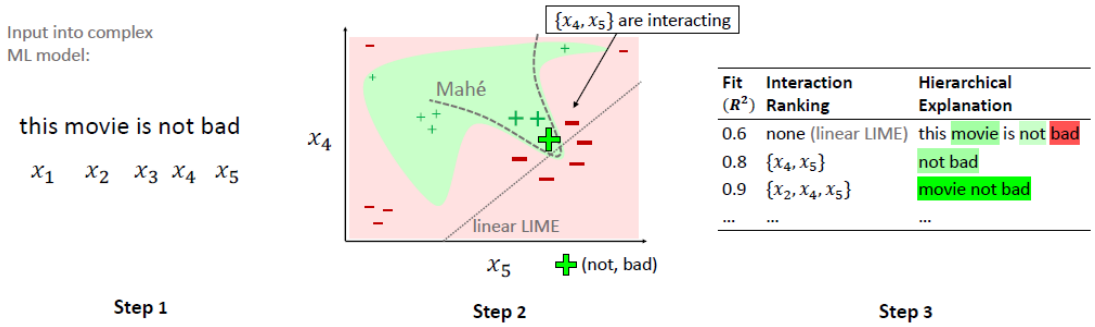


Figure 3.32: Overview of context-dependent hierarchical explanation [55].

3.16.1 How does it operate?

Mahé, as reported above, can offer context-dependent and context-free explanations of interactions. Consider $f(\cdot)$ is a target function of interest (e.g. a classifier), and $\phi(\cdot)$ a local approximation of f. Consider $g_i(\cdot)$ any function where attribution scores are provided by $g_i(x_i)$ for every feature i. Also, consider a data instance $x \in \mathbb{R}^p$ and a bias b. The generalization of interaction explanation can be given by:

$$\phi_K(x) = \sum_{i=1}^{p} g_i(x_i) + \sum_{i=1}^{K} g_i'(x_{\mathcal{I}}) + b \tag{3.22}$$

where $x_{\mathcal{I}} \in \mathbb{R}^{|\mathcal{I}|}$ are the interacting variables matching the variable indicators $\mathcal{I}$ and $\{\mathcal{I}_i\}_{i=1}^{K}$ is a set of K interactions.

3.16.2 Validation example

The validation of Mahé was carried out on synthetic and real data, demonstrating how this technique manages to match and even surpass other models in the literature explaining interactions especially context-free ones. Some examples of validation of

hierarchical explanations are given in the following illustrations. An example of a context-dependent explanation in hierarchical format is given in Figure 3.33. The colours in the super-pixels symbolize the attribution totals and their polarity. Parts coloured cyan influence positively the prediction, while parts coloured red influence negatively. The limits between corresponding interactions are united when the attribution polarities match.

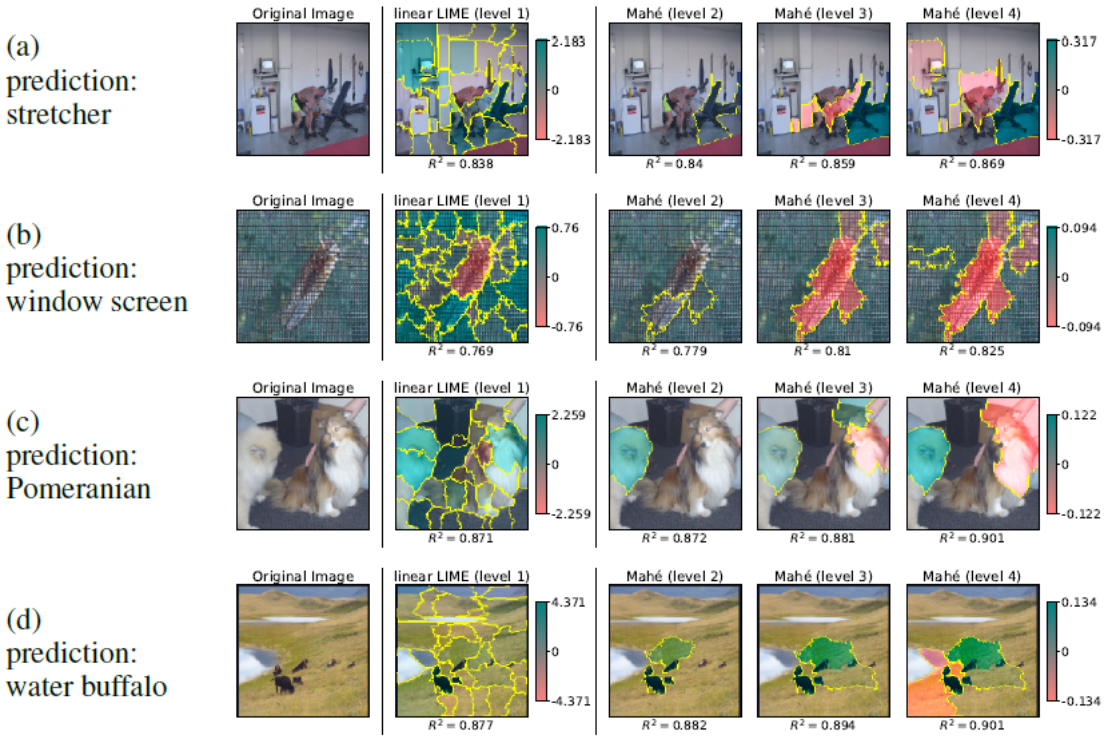


Figure 3.33: Example of context-dependent hierarchical explanation with images [55].

Another example of context-dependent hierarchical explanations is given in Figure 3.34, in which the interaction attribution is emphasised by different colours and shades. Green represents a positive contribution and red a negative contribution. The various attributions are then normalised and the scores are represented on the right, according to max attribution magnitudes.

Method	Level	Fit (R^2)	Hierarchical Explanation	Max magn.
linear LIME	1	0.621	the film is really not so much bad as bland	0.744
Mahé	2	0.751	not, bad	
Mahé	3	0.916	not, bad, bland	
Mahé	4	0.926	film, not, bad, bland	0.119
linear LIME	1	0.519	a very average science fiction film	0.708
Mahé	2	0.598	science, fiction	
Mahé	3	0.819	a, average	
Mahé	4	0.923	a, very, average	0.213
linear LIME	1	0.612	a charming romantic comedy that is by far the lightest dogme film and among the most enjoyable	0.612
Mahé	2	0.856	charming, enjoyable	
Mahé	3	0.923	charming, lightest, enjoyable	0.072

Figure 3.34: Example of context-dependent hierarchical explanation on sentiment analysis with LSTM [55].

3.17 XRAI

XRAI [56] is a gradient-based explanation technique and it usually supports images. XRAI is an integrated gradient-based attribution method and can be implemented with any DNN-based model if the input features can be segmented using similarity metrics. XRAI is a saliency method that gradually expands attribution regions, and can assure high quality, correctly delimited saliency regions that surpasses saliency techniques found in the literature. The diagram of how this technique works is shown in Figure 3.35. First, the image is given as input and the system over-segments it into many regions of different shapes, overlapping each other. After that, segments are gradually put according to their integrated gradient density. The region importance level can be retrieved from the trajectory. For example, in this particular case, the method restored the face present in the image, i.e. that of the leopard, thus providing an accurate classification, then put the body and the remaining part of the image. Finally, on the right is the diagram of the evaluation method for the image and a certain area threshold. In this section, the unfocused image and the mask are united to offer the saliency-focused image.

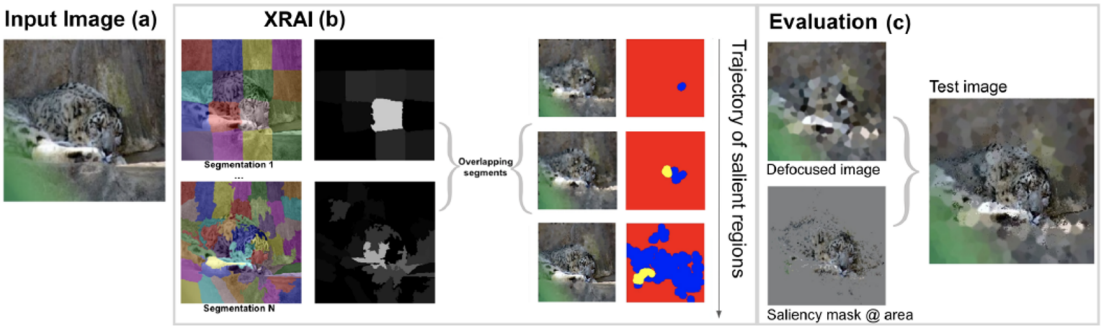


Figure 3.35: Overview of XRAI method [56].

3.17.1 How does it operate?

To work towards the attribution, XRAI employs Integrated Gradients with black and white baselines. Employing a black image as a baseline is helpful to decrease attribution of dark input pixels, thus an RGB number of (0, 0, 0) will get precisely 0 attribution. This becomes clear from the Integrated Gradients' formula:

$$IG_i(x) = (x_i - x_i') \times \int_{\alpha=0}^{1} \frac{\partial F(x' + \alpha \times (x - x'))}{\partial x_i} d\alpha \tag{3.23}$$

in which x_i is the input pixel i and $x\prime_i$ is the equivalent baseline pixel.
Furthermore, in order to choose regions, XRAI computes that given two regions, the one adding the most positive value is more relevant for the classifier. From that, XRAI begins with an empty mask, then adds the regions choosing from the one that returns the greatest increase in total attributions per area. The model goes on until it is out of regions or if the full image is obtained.

3.17.2 Validation example

The validation of XRAI has been done through empirical experiments that show that it offers very good results compared to other saliency methods in the literature. Two examples of comparisons of XRAI with other common saliency methods are given below.
In Figure 3.36, an example of the output of many common methods is shown for a fixed area threshold on an image, in which two dogs are visible, one larger and the other smaller. All variants of integrated gradients generally work well, however they do have some grainy regions. In addition, the edges often get more prominence than the two subjects within the image, in fact from this it can be understood that the edge method works best with only one (relatively large) subject within an image, because those of the main object stand out and there would be only those.

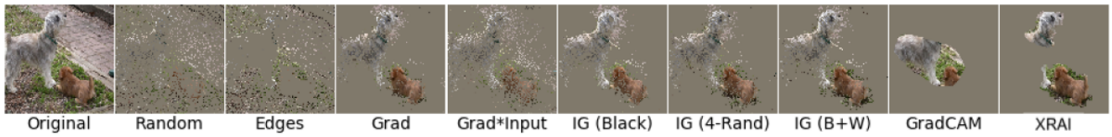


Figure 3.36: Example of XRAI and comparison with other common techniques [56].

Figure 3.37 shows a single comparison between XRAI and Grad-CAM [45], which is the method that comes closest in terms of performance. While Grad-CAM works by choosing one region and gradually enlarging it depending on the region, XRAI is able to work on several regions. For example, in the figure it can be seen that XRAI works by focusing mainly on the object of interest, whereas Grad-CAM covers circular areas between the objects.

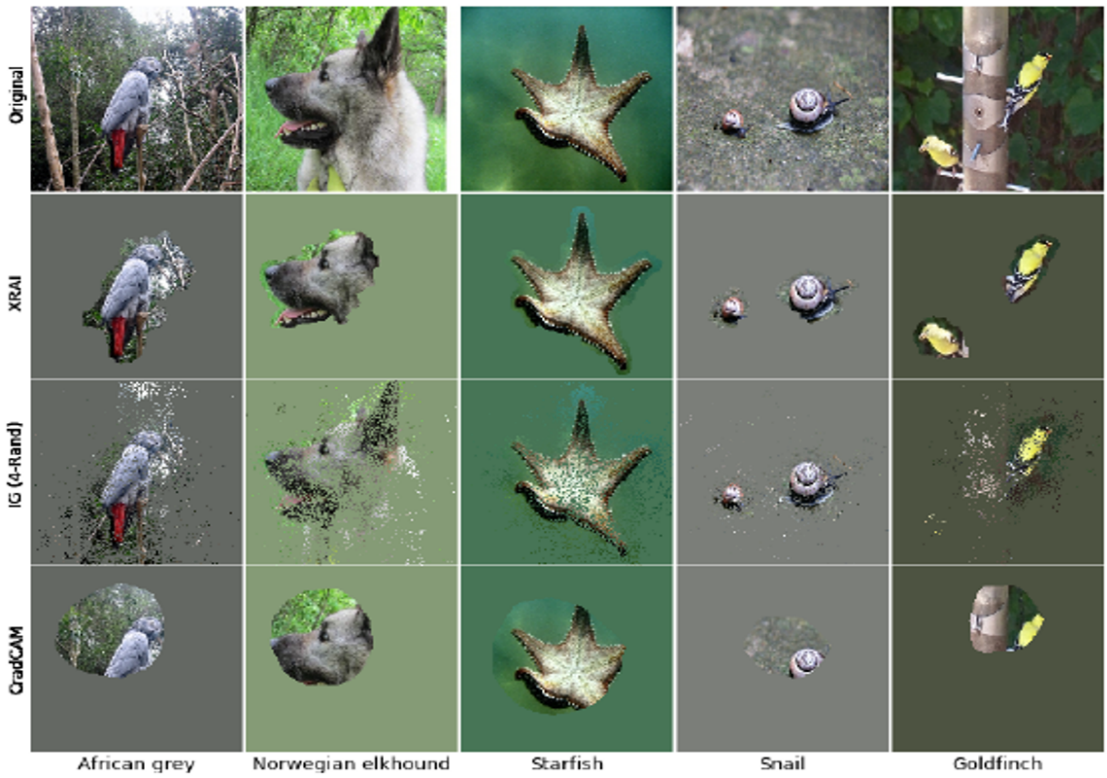


Figure 3.37: Comparison between XRAI and Grad-CAM [56].

The XRAI code for experiments is available on https://github.com/PAIR-code/saliency.

Chapter 4

State of the art comparative

In this chapter, we address the issue of comparing and testing various explanation techniques. How are they validated and with which methodologies are they compared?

4.1 Theoretical and experimental surveys

Many surveys exist in the state of the art that make purely theoretical comparisons. For example, in [31] a survey of existing literature was conducted for the full evaluation of XAI techniques, methods collected through the largest conferences and computer science journals in the field of HCI and machine learning. In [40], on the other hand, an in-depth categorisation of all the main components in machine learning was made, starting with the desiderata of an interpretable model, to categorise the types of black-box models and explanators. The paper [37] focuses on detailed descriptions of what it is and what types of explanations exist, how they are categorised, and also discusses objective and subjective metrics for how to evaluate them. In [33] you can find a thorough overview of all the most used terms in machine learning, as well as an important explanation of what interpretability is. In [62], on the other hand, a comprehensive list is made of all ethical principles within the XAI, the various definitions related to explanations and all the main categorisation features of the most common explanation techniques are summarised schematically. In [63], finally, an overview is given for the evaluation of explanation methods, and special attention is paid to perturbation-based methods and the inputs they can receive.

Other surveys are experimental, such as [64], in which three different tasks are performed to evaluate various methods: the survey results very comprehensive in terms of the various human-grounded evaluation tasks proposed and the number

and dimensions of explanation methods being evaluated. In the paper [65], on the other hand, a survey is proposed in which XAI methods are compared using a new verification methodology that incorporates the temporal dimensions. In [66], a quantitative experimental comparison is made on all dimensions used by the XAI methods, using specific metrics and desiderata, such as faithfulness, localisation and stability. In [67], the authors demonstrate through an experimental study which method performs better between LIME and SHAP, with different neural networks, according to some fundamental metrics, such as identity, stability, and separability. In [68] a number of XAI methods are compared using the new MDMC evaluation framework, and through this it is shown that several metrics in the various methods can be improved, such as accuracy, transparency and final prediction. On the other hand, in [69] a new method of quantitative comparison of explanation methods is proposed by means of a specific task, for which there are several explanations to take into account and the evaluation is done by users. As a last step, the authors in [70] proposed some tasks in human-grounded design to compare the different representations and effects of XAI methods, in order to increase the demand for more informed design decisions in XAI interfaces.

4.2 Evaluation metrics

In this section we are going to define the various metrics for evaluating both AI models and XAI techniques, proposed in many of the surveys mentioned above, to assess the goodness of the model and of the explanations. As defined above, experimental surveys are usually validated by certain metrics, which refer to desiderata to be achieved by a method of explanation. While in the paper [71] certain metrics (e.g. computational cost and recovering difference) are defined for a specific experiment, many other surveys in the literature, including those listed above, specify some criteria on which the evaluation of one or more explanation methods is based.
It is important to remember that a good evaluation metric must make a comprehensive quantification of the quality of the predictive output. Moreover, it should be applicable to different structures, so that they are easily comparable.

4.2.1 Interpretability

Interpretability is one of the fundamental and also most meaningful measures in the field of explanations. According to Guidotti et al. [40], interpretability can be defined as the ability to explain or provide meaning in a way that is understandable to humans, so a model or prediction will be interpretable if it can be understood by humans. The most heated discussion introduced by these authors is how to actually measure this. According to Doshi-Velez et al. [39], interpreting a model also means presenting it in conditions comprehensible to humans, and therefore

this article also speaks of interpretability as comprehensibility. Finally, as further proof, according to Mohseni et al. [31], interpretability is the ability to help the human being understand and comprehend the decision-making methodologies of the model process and its predictions.

4.2.2 Accuracy

Accuracy, along with interpretability, is one of the most important and most competitive measures to achieve among the various explanation techniques. The accuracy of the model, according to Guidotti et al. [40] measures how well the it is able to make accurate decisions about unseen instances. An explanation technique, on the other hand, measures the accuracy in which it describes how the model works. Accuracy can be measured by scores found in the literature, e.g. the F1-score [72] or the sickness score [11].

4.2.3 Fidelity

Fidelity is a very similar metric to accuracy (and can have the same metrics such as the F1-score) but takes into account the black-box outcome. In fact, according to Guidotti et al. [40], the fidelity of a model measures how well it is able to faithfully reproduce the behaviour of a black-box predictor and evaluates the goodness of the imitation of that. As for an explanation technique, on the other hand, this measures how reliable it is in reproducing and explaining the behaviour of the model, and especially, as we shall see in the experimental part, showing the user how it actually works.

4.2.4 Fairness

The fairness, as reported in [41], measures the extent to which an AI model is able to protect the final output against direct or indirect discrimination. Also according to Doshi-Velez et al. [39], the concept of fairness is associated with non-discrimination towards protected groups, whether implicit or explicit. In [31], it is also reported that the concept of fairness implies ethical analysis of the model and data used in the prediction and decision-making process. As explained by Pastor et al. [73], it is often the input data that is discriminatory or unfair, so it is necessary to know where it comes from and whether or not there are any potential discriminating factors in order for the entire model to meet fairness standards.

4.2.5 Usability

The usability of a model measures how useful the information generated is to users for their task, emphasizing the interactivity of a model and discouraging

fixed explanations. For example, as reported in [39], the usability of a model can give information to the user to complete a task, such as the aircraft collision avoidance system. In the context of explanation techniques, we talk about usability or «usefulness» when we want to try to understand how helpful an explanation has been to a user in comprehending how the model works.

4.2.6 Reliability

The reliability measures the ability of a model to remain performant despite small changes in certain parameters or differences in input data. By changing certain inputs, in fact, as stated in [39], a model must maintain a certain level of performance, which is why this concept is often associated with that of robustness. Also according to Mohseni et al. [31], reliability is useful to ascertain the confidence a user has in the model and to help them follow instructions to maintain high performance. A similar concept can be found in the reliability of an explanation, i.e. the ability of the explanation to show even the smallest variations in the model, or features that have no relevance to the final prediction.

4.2.7 Faithfulness

The faithfulness, mainly described by Li et al. [74], explains how important relevance scores are for the decisions made by the model. For example, this measure can be quantified by disrupting the model by removing or adding certain features. There are several metrics to measure this, for example iAUC is presented in [47]. The faithfulness of an explanation (also called trustworthiness) can be calculated in the same way, i.e. it assesses how faithful it is in emulating the behaviour of a model, to which possibly a feature perturbation has been made.

4.2.8 Stability

Stability measures how stable the model is to unanticipated changes. According to Li et al. [74], with examples of saliency maps, an explanation is considered stable when, by slightly perturbing an input, the prediction is very similar and has the same confidence distribution. For this reason, in [75], the authors compare the term stability with the term identity. Stability can be measured by various metrics, e.g. $SENS_{max}$.

Chapter 5

Proposed comparative

In the next sections we propose a comparison that provides a detailed example for the subjective and objective comparison of the various explainability techniques proposed so far.

5.1 Study and comparison using human-based metrics

The subjective comparison we propose is the construction of a survey based on experiments that evaluate both objectively and subjectively the XAI techniques through users.
The implementation of the survey required several steps, in which we reflected on the various human-based assessment metrics in the literature and those that could be newly implemented. We decided to base the survey on local explanations, as we tried to explain the reason why a model predicted a certain label.
The structure of the proposed questionnaire is divided in six sections, including the first introductory section. For the work of this thesis, the users of the survey could be both domain-experts and non-expert users, however we tried to prefer a more experienced user base in the AI field. The survey was distributed for about 7 days to users mainly from the Politecnico di Torino and the total turnout was 45 people.

To carry out this survey, we decided to train a BERT (version base) [76] model and use a binary type of dataset (IMDb dataset of 50k movie reviews [77]). Two models were used and trained to conduct the survey, one of which was deliberately trained to a high accuracy and the other, useful only for the purposes of section 2, deliberately trained to a lower accuracy. The "well-trained" model, which was used for all the tasks, was trained on the 25k samples of the balanced training set (12.5k labelled as "Positive" and 12.5k labelled as "Negative"). The model obtained

an accuracy on the other 25k samples of the test set of 93.7%. The "bad-trained" model, on the other hand, was only trained on 1000 samples (500 "Positive" and 500 "Negative"), significantly less, while 25k samples were always used for the testing set. The accuracy of this model was 71.6%. We trained both models for one epoch, i.e. using the whole training set once. The learning rate was 2e-5, the batch size was 8 and the optimiser was AdamW.

5.1.1 Introductory first section

In the first introductory section, an attempt was made to summarise the basic concepts of Artificial Intelligence and Explainable Artificial Intelligence, and then to explain how a predictive model works and the concept of explanation. This was done primarily to introduce the subject to non-expert users of the domain, in order to expand the compilation of the survey as much as possible. As a final point, the definition of a "good predictive model" and a "good explanation technique" was suggested to the user. The user was then asked two questions regarding their knowledge of machine learning and Explainable Artificial Intelligence, in order to map the users' responses to the questionnaire.

5.1.2 Second section: fidelity

The second section of the survey concerns the first actual question on metrics. In particular, this section evaluates the fidelity of the XAI technique. This section has been designed taking inspiration from experiments in [64].
A "well-trained" model was trained with 25.000 samples, and a "bad-trained" model was trained with 1.000 samples, so that the accuracy is significantly lower than the first. We nominate by convention the first model as "good model" and the second model as "bad model".
After that, explanations for both models are generated using T-EBAnO, LIME and SHAP techniques. The aim of this application is to evaluate the goodness of explanations in recognising whether a model is working well or not.
For the purposes of the survey, for each XAI technique, the user is provided with an input text and two explanations, one explaining the good model and the other explaining the bad model. The label predicted by the model is also provided, which is required to be the same for both models of the specific input text in order to design the experiment. Then, the user will be asked which of the two provided explanations he thinks best explains the predicted label. Indirectly, in fact, the user is being asked to recognise, thanks to the good functioning of the XAI technique used, which model has predicted better. Therefore, if the XAI technique performs well, the user will easily be able to recognise the explanation referring to the good model.

The user was given an example (Figure 5.1) to better understand the reasoning they would have to do in answering the following questions.

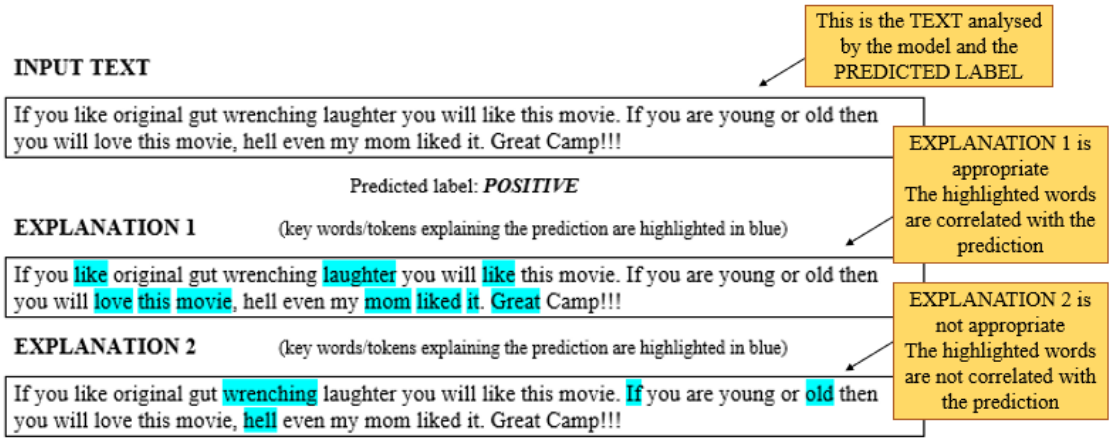


Figure 5.1: Example given to the user for section 2.

The scores assigned to this section were also based on the experiment carried out in [64], and range from +1 to -1, with 5 degrees of scoring according to the possible responses the user can give. In the case of this task, whose example of a question and response choice is shown in Figure 5.2, the scores were:

- ± 1 if "Explanation 1 is far more appropriate than Explanation 2" is correct/incorrect;
- ± 0.5 if "Explanation 1 is rather more appropriate than Explanation 2" is correct/incorrect;
- 0 if the user selects "Explanation 1 and Explanation 2 are equally appropriate".

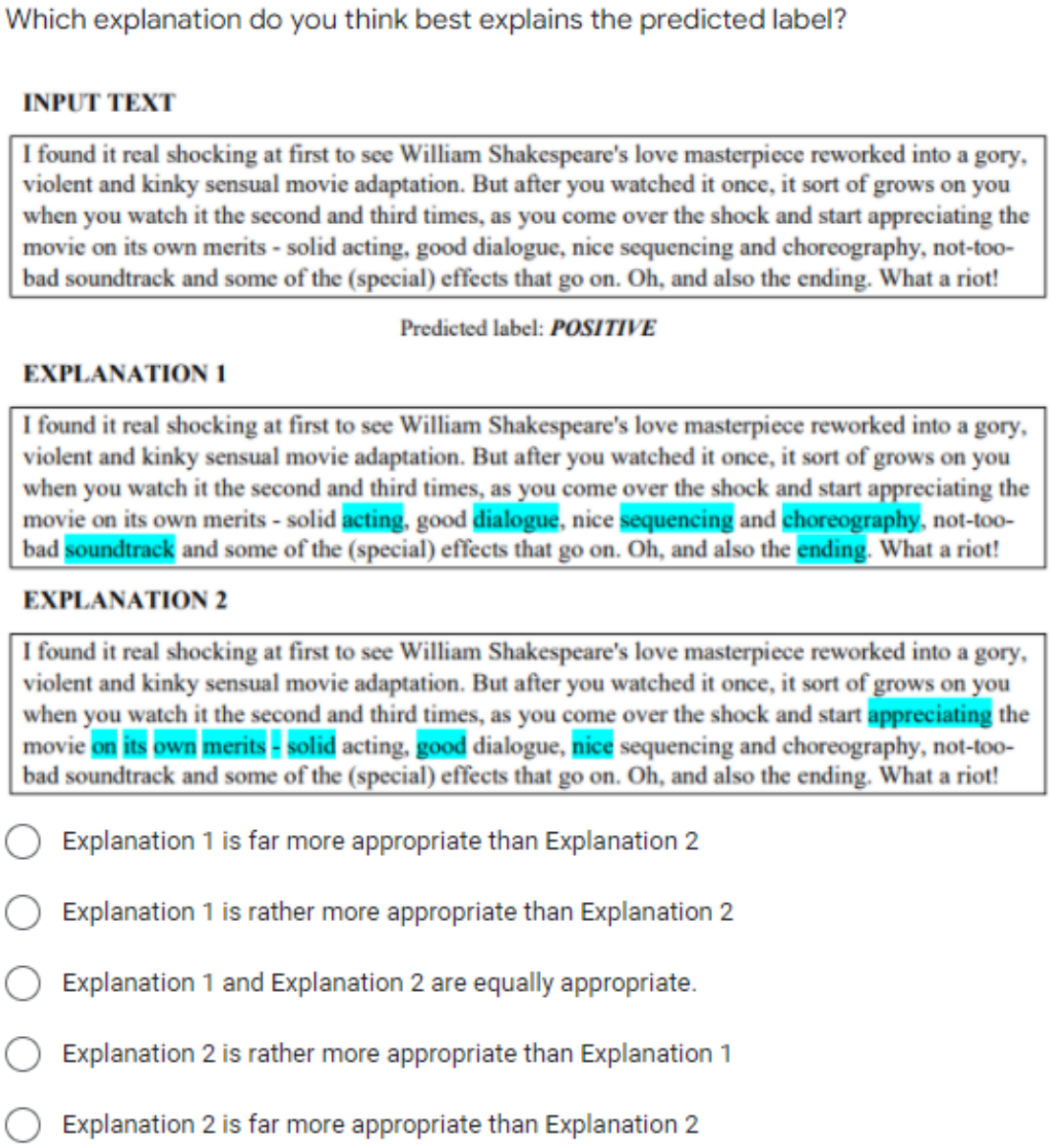


Figure 5.2: Example question from section 2.

5.1.3 Third section: accuracy

This section aims to evaluate the accuracy of the XAI technique. This section has been designed taking inspiration again from experiments in [64].
The "well-trained" model has been used and explanations for the model are generated using T-EBAnO, LIME and SHAP techniques. The purpose of this application is to assess the goodness of explanations in justifying the behaviour of the model.

Hence, for each technique, the user is provided with the explanation (most relevant words/tokens). Based only on the words of the explanation, the user is asked to guess which is the most likely predicted label referring to that explanation. Again, a different application is used to assess the goodness of the explanation technique. In fact, it must underline and justify the prediction of the model, so for a good explanation technique and a model that performs well, it should be easy for the user to recognise the label predicted by the explanation alone.
The user was given an example (Figure 5.3) to better understand the reasoning they would have to do in answering the following questions.

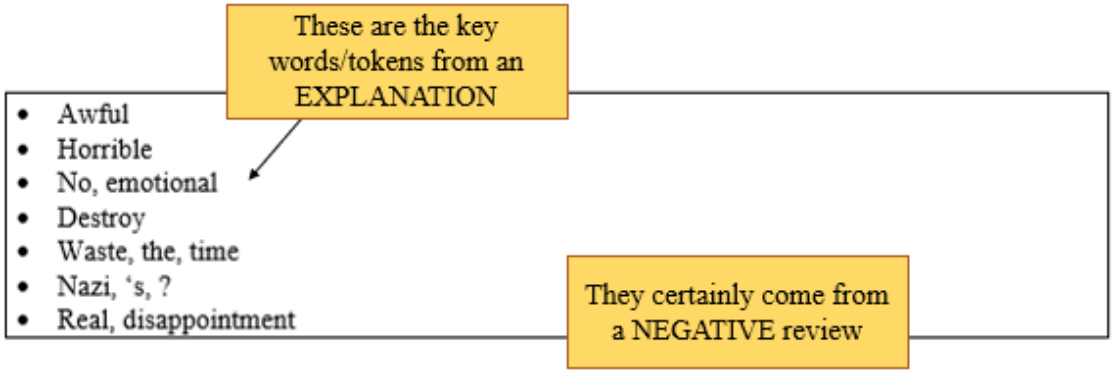


Figure 5.3: Example given to the user for section 3.

The scores assigned to this section were also based on the experiment carried out in [64], and range from +1 to -1, with 5 degrees of scoring according to the possible responses the user can give. In the case of this task, whose example of a question and response choice is shown in Figure 5.4, the scores were:

- +1 if the user selects "I'm certain that they are from a POSITIVE/NEGATIVE review" and the label is actually positive/negative;
- +0.5 if the user selects "I'm not certain but they are likely from a POSITIVE/NEGATIVE review" and the label is actually positive/negative;
- 0 if the user selects "I can't say if it's POSITIVE or NEGATIVE";
- -0.5 if the user selects "I'm not certain but they are likely from a POSITIVE/NEGATIVE review" and the label is actually negative/positive;
- -1 if the user selects "I'm certain that they are from a POSITIVE/NEGATIVE review" and the label is actually negative/positive.

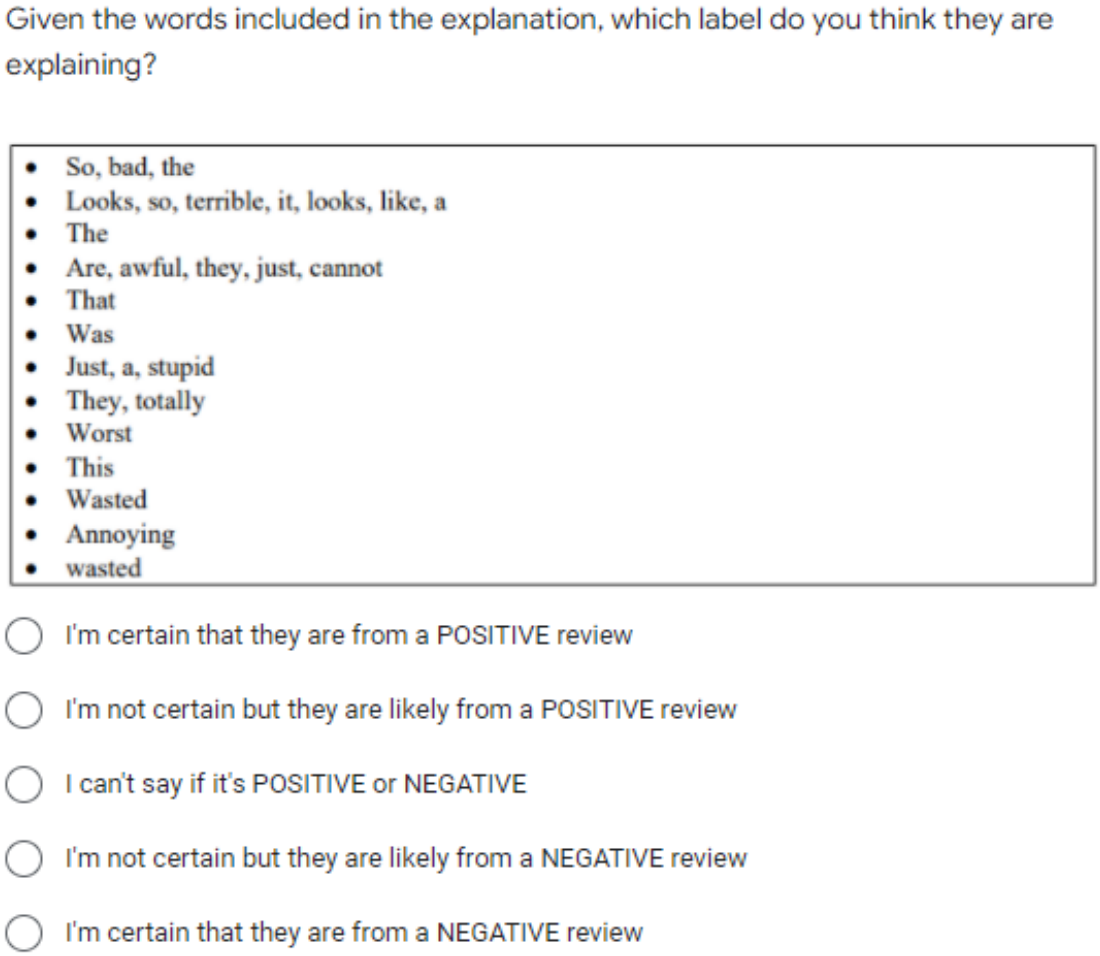


Figure 5.4: Example question from section 3.

5.1.4 Fourth section: reliability

In this section we evaluate the reliability of the XAI technique. This section has been designed taking inspiration again from experiments in [64].
This task assesses the goodness of explanations in helping humans investigate uncertain predictions of the model. In other words, if the model is working badly, or the probability of a predicted label is relatively low, is the explanation technique able to point this out to me? Can I tell why the model has been working badly by the relevant features that are highlighted?
The "well-trained" model has been used and explanations for the model are generated using T-EBAnO, LIME and SHAP techniques. By means of a query, the cases in which the prediction probabilities were around 50% and 60% were taken. According to this reasoning, then, the counter-evidence probabilities were their complementary (e.g.: if the prediction probability of the label "positive" was 55%, then that of the

counter-evidence "negative" was 45%). With these data, the explanations provided by the XAI technique under consideration were investigated, covering both case histories, the evidence and the counter-evidence. Thus, the user was offered the probability of both labels, noting which one was actually predicted by the model. The user was then shown the explanations for both labels to try to understand how the model had worked. The user was asked, based on these explanations, to state their opinion whether the label actually predicted by the model or the counter-evidence was more appropriate for the missing input text.
It is stressed that this experiment can only be done for cases in which the model is uncertain about the actual prediction, and not for high prediction probabilities.
The user was given an example (Figure 5.5) to better understand the reasoning they would have to do in answering the following questions.

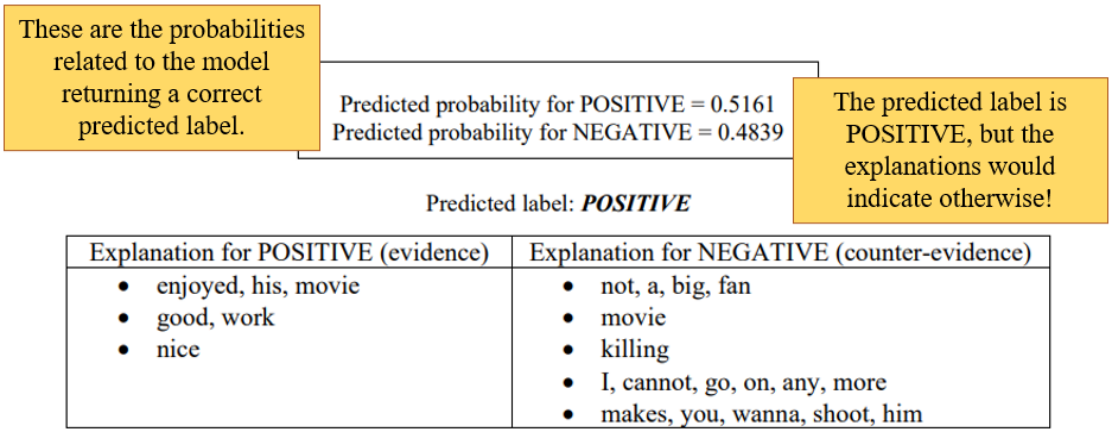
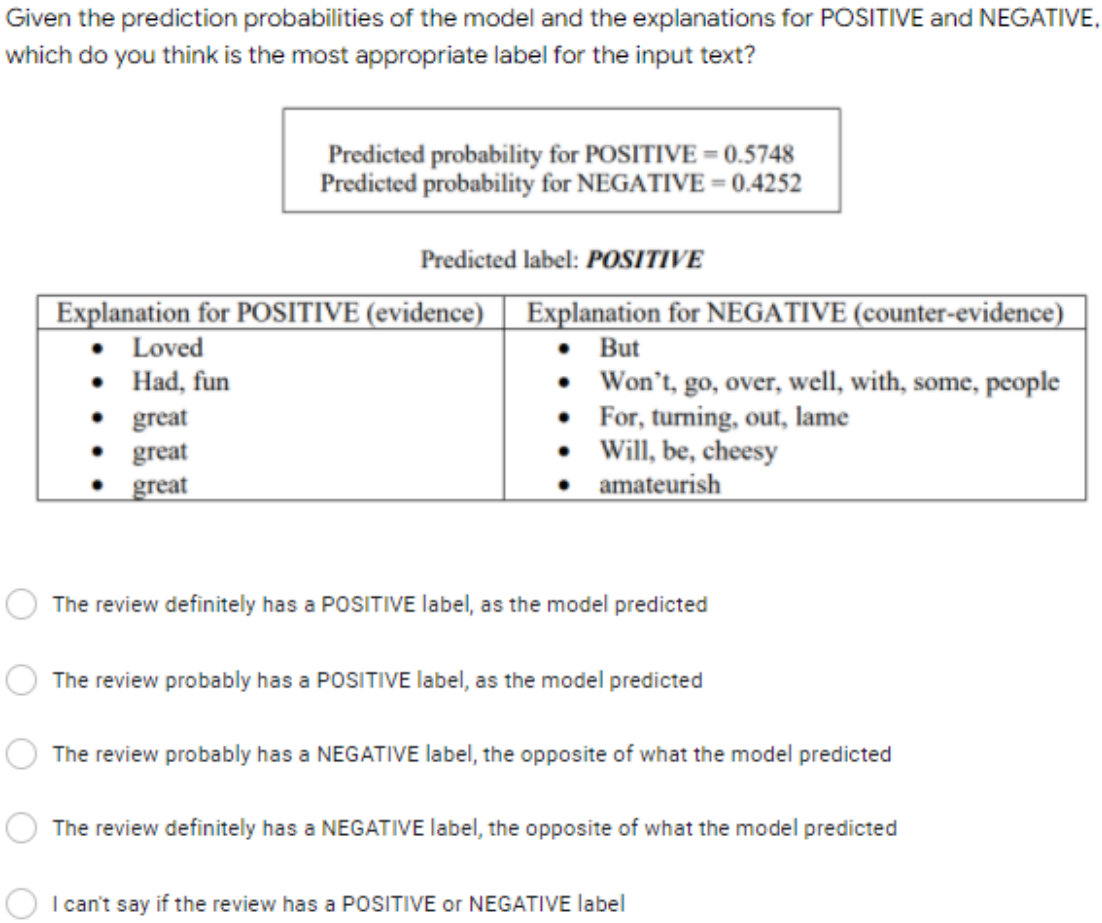


Figure 5.5: Example given to the user for section 4.

The scores assigned to this section were also based on the experiment carried out in [64], and range from +1 to -1, with 5 degrees of scoring according to the possible responses the user can give. In the case of this task, whose example of a question and response choice is shown in Figure 5.6, the scores were:

- ±1 if "The review definitely has a POSITIVE/NEGATIVE label, as the model predicted" is correct/incorrect;
- ±0.5 if "The review probably has a POSITIVE/NEGATIVE label, as the model predicted" is correct/incorrect;
- ±0.5 if "The review probably has a POSITIVE/NEGATIVE label, the opposite of what the model predicted" is correct/incorrect;
- ±1 if "The review definitely has a POSITIVE/NEGATIVE label, the opposite of what the model predicted" is correct/incorrect;

- 0 if the user selects "I can't say if the review has a POSITIVE or NEGATIVE label".


Given the prediction probabilities of the model and the explanations for POSITIVE and NEGATIVE, which do you think is the most appropriate label for the input text?

Predicted probability for POSITIVE = 0.5748
Predicted probability for NEGATIVE = 0.4252

Predicted label: ***POSITIVE***

Explanation for POSITIVE (evidence)	Explanation for NEGATIVE (counter-evidence)
• Loved • Had, fun • great • great • great	• But • Won't, go, over, well, with, some, people • For, turning, out, lame • Will, be, cheesy • amateurish

- The review definitely has a POSITIVE label, as the model predicted
- The review probably has a POSITIVE label, as the model predicted
- The review probably has a NEGATIVE label, the opposite of what the model predicted
- The review definitely has a NEGATIVE label, the opposite of what the model predicted
- I can't say if the review has a POSITIVE or NEGATIVE label

Figure 5.6: Example question from section 4.

5.1.5 Fifth section: comprehensibility, completeness and usefulness

In this section we decided to evaluate three relevant metrics of the XAI technique, respectively the comprehensibility, the completeness and the usefulness. These metrics, compared to the assessment made previously, are much more subjective, i.e. they depart slightly from the objective perception of the user and rely much more on the first impact and general understanding of the individual.
The "well-trained" model has been used and explanations for the model are generated using T-EBAnO, LIME and SHAP techniques. Again, an input text, the predicted label and highlighted explanations are provided, for the three different

XAI techniques. This task evaluates the explanation in regard of the three following criteria:

- Comprehensibility. This metric quantifies how comprehensible and human-readable the explanation is, regardless of how it reflects the behaviour of the model. An explanation is comprehensible when the human being can best interpret and understand it.
- Completeness. This metric quantifies how complete and thorough an explanation is with respect to the class predicted by the model. An explanation is complete if it includes all the features that can best explain the predicted label.
- Usefulness. This metric quantifies how the explanation helped to better understand the label. An explanation is useful if it helps the user to better understand the model's decisions in predicting the label.

The user was given an example (Figure 5.7) to better understand the reasoning they would have to do in answering the following questions.

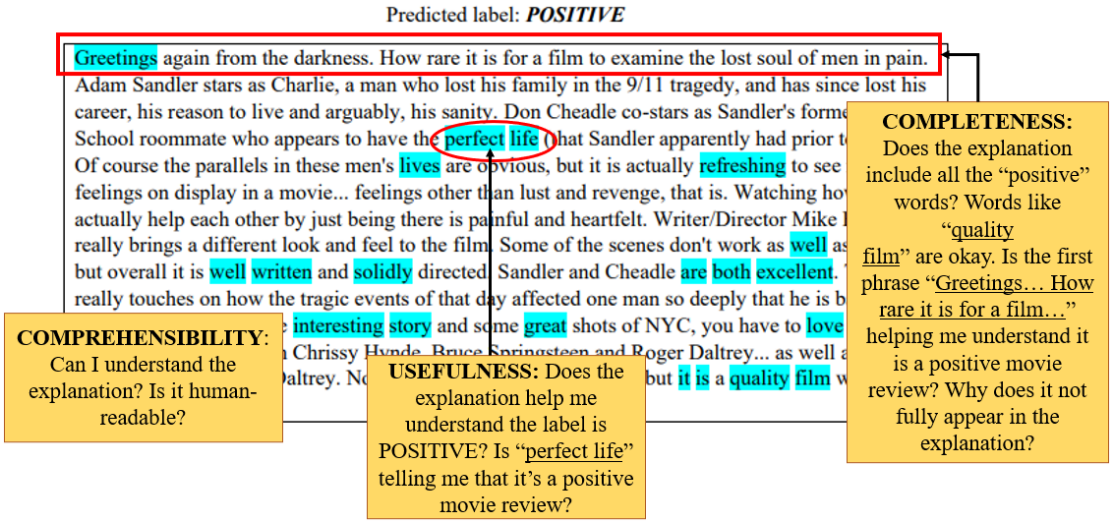


Figure 5.7: Example given to the user for section 5.

The scores assigned to this section were instead assigned in relation to the three previous tasks. Since this task is much more subjective and at the user's discretion, we decided to give a lower weight to the answers, ranging from a range of +0.5 to -0.5, with 5 grades of score according to the possible responses that the user can give. In the case of this task, whose example of a question and response choice is shown in Figure 5.8, the scores were:

- ±0.5 if the user selects "I totally agree/disagree";
- ±0.25 if the user selects "I agree/disagree";
- 0 if the user selects "I'm indifferent".

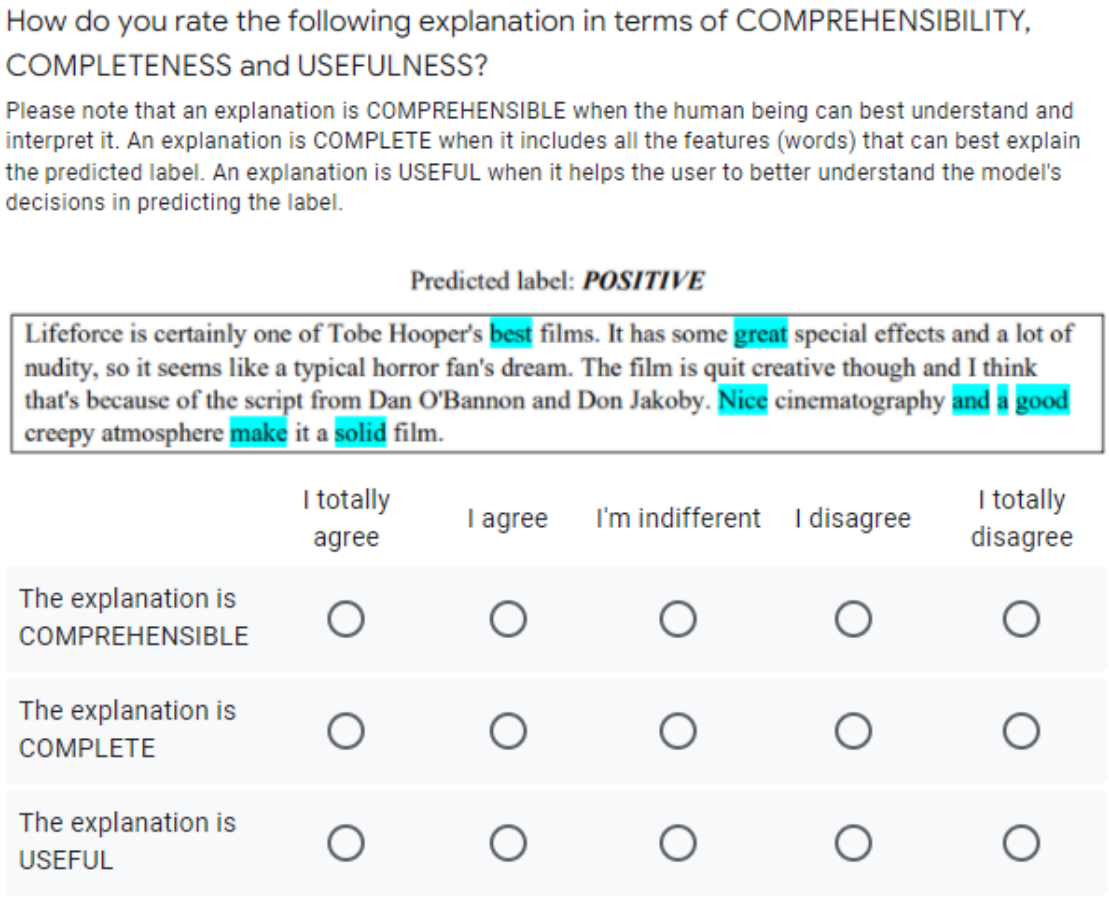


Figure 5.8: Example question from section 5.

5.1.6 Sixth section: visualisation

The last section we designed was related to the visualisation of the output of the XAI techniques. In fact, in the previous sections we decided to adopt the same explanation display for all XAI techniques, and it was done for two reasons. The first reason is that the audience was not just domain-expert users, but more diverse. The second reason was that our focus was on evaluating techniques through seemingly complex applications, so we did not want to distract from the task itself. The user did not have to focus in that case on understanding the visualisation of the explanations; moreover, by doing so, we placed the techniques on the same level

of readability. Hence, we provided the input text and the actual ways in which each XAI technique returns the output. The user was asked to quantify the degree of readability and understandability of the real visualisation of the single techniques.

This section, considered much more subjective than the first three tasks, also received a range of scores from +0.5 to -0.5, with 5 degrees of scoring according to the possible responses the user could give. In the case of this task, whose example of a question and response choice is shown in Figure 5.9, the scores were:

- +0.5 if the user selects "Very good";
- +0.25 if the user selects "Good";
- 0 if the user selects "Adequate";
- -0.25 if the user selects "Sufficient";
- -0.5 if the user selects "Scarce".

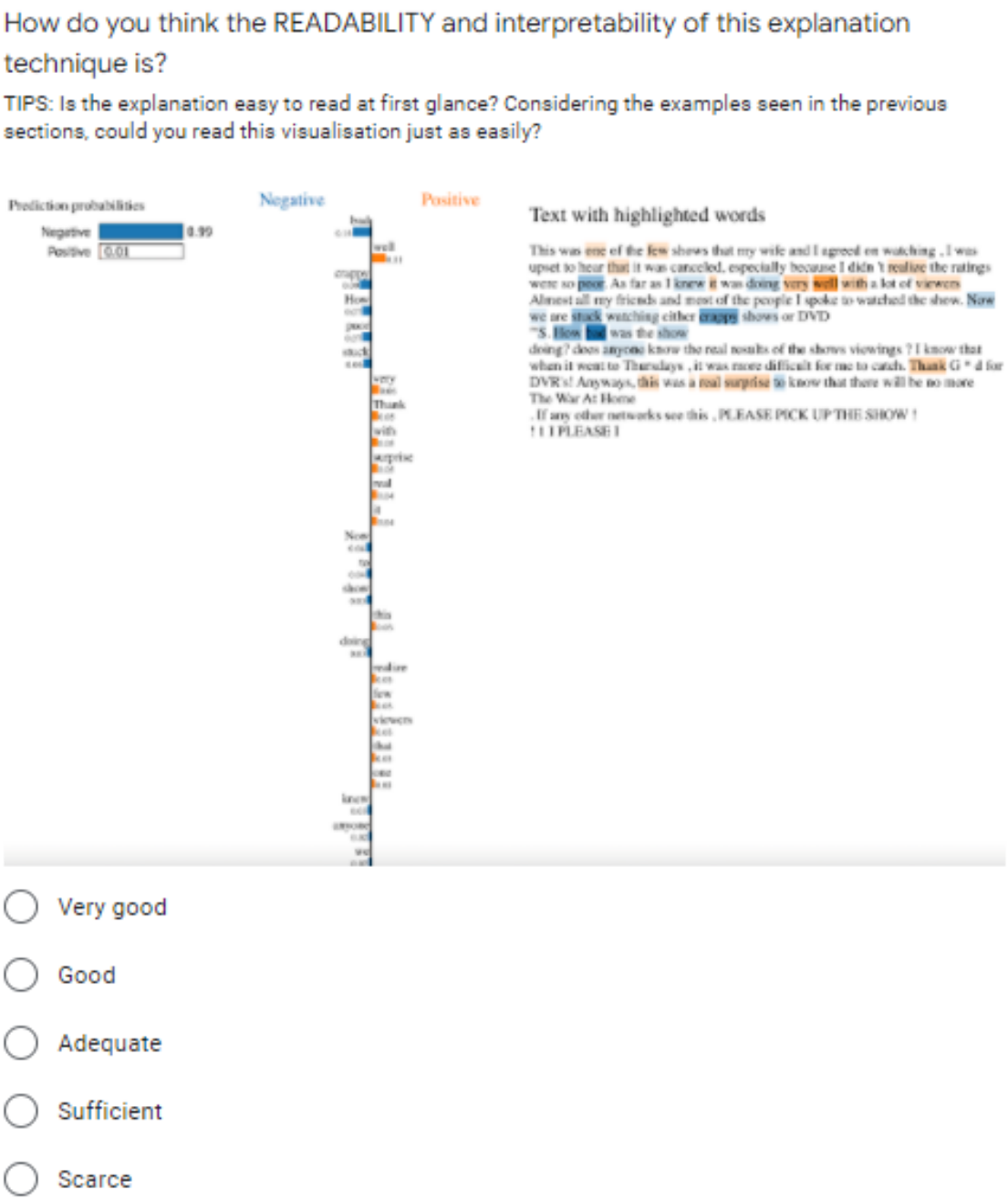


Figure 5.9: Example question from section 6.

5.2 Survey results

In this section we propose and comment on all the results obtained in the different subsections of the survey.

5.2.1 Introductory first section

In the first section we gave a general introduction to the survey context and in the results we propose the user mapping, created by two questions.
The first question, the results of which can be seen in Figure 5.10, asked the users whether they were familiar with the concept of Artificial Intelligence. It can be noticed that our user base registered an 80% of positive response. The second question, the results of which can be seen in Figure 5.11, went into more detail about the topics covered in the survey, asking users if they were already familiar with Explainable AI techniques. As can be observed, this concept was far from familiar among our users, with only 17.8% giving a positive response.

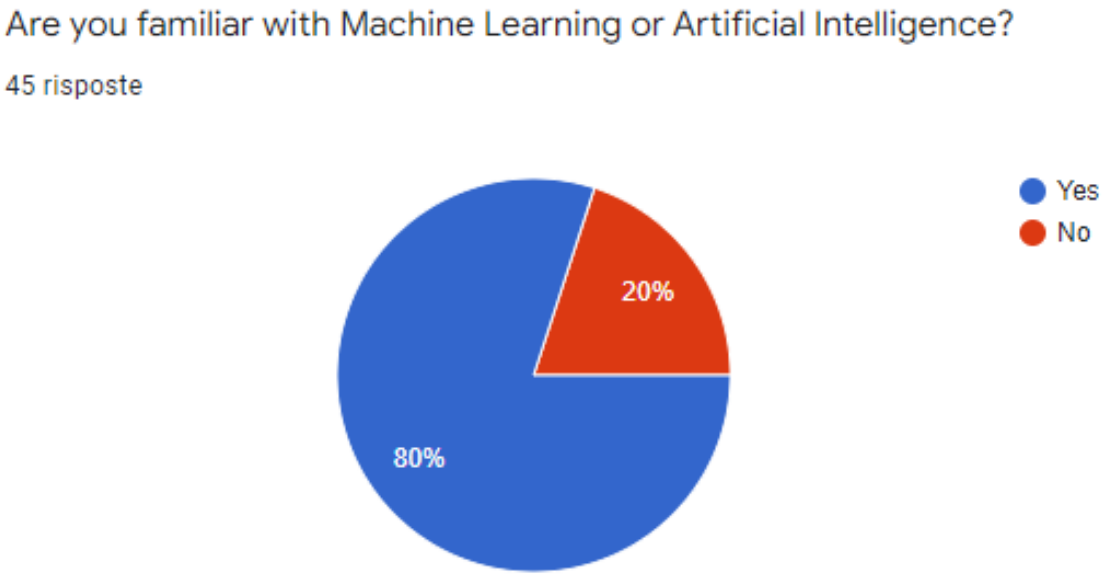


Figure 5.10: Response mapping for the question "Are you familiar with Machine Learning or Artificial Intelligence?".

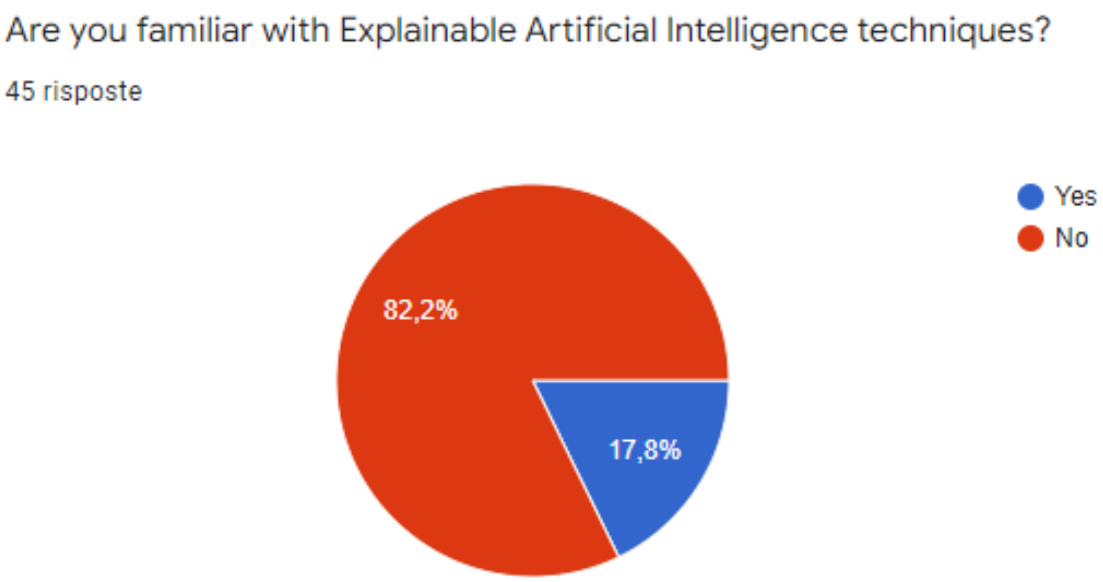


Figure 5.11: Response mapping for the question "Are you familiar with Explainable Artificial Intelligence techniques?".

5.2.2 Second section: fidelity

In this section, as mentioned, an attempt has been made to evaluate the goodness of explanations in recognising whether a model is working well or not.
Six questions were proposed for each explanation technique, making a total of 18 questions.

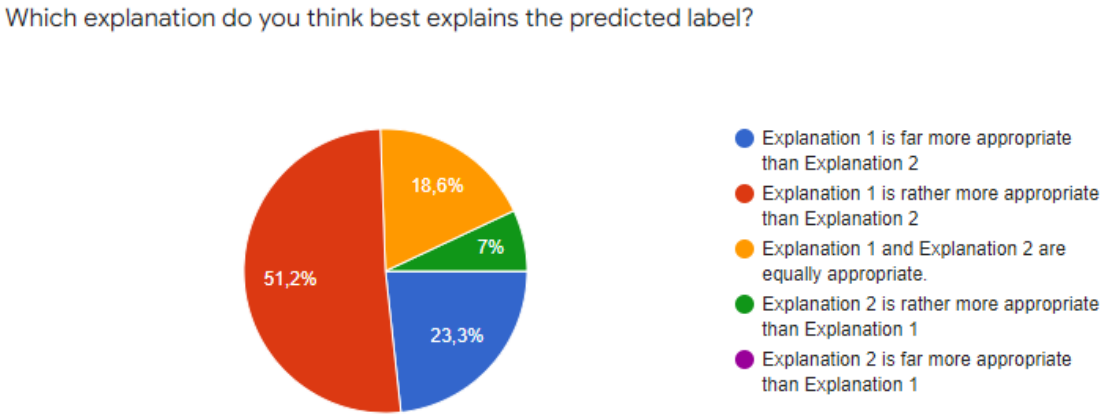


Figure 5.12: Example of response mapping for section 2.

Thanks to the scores setting, it was possible to map an overall result of the performance of the XAI techniques, shown in Table 5.1. It can be seen that T-EBAnO is the best performing technique, with an overall score of 187, followed

by LIME with a score of 166.5, and lastly SHAP with 114 points. The average overall score per input text provided to the user is also shown. Remember that the average score ranges from +45 to -45, as each user could give from 1 point to -1 point depending on the response given.

	Total	**Average score per input text**
T-EBAnO	187	31.17
LIME	166.5	27.75
SHAP	114	19

Table 5.1: Scores for section 2 of the survey.

In this application, the user had no particular difficulty in recognising which of the two texts and explanations proposed was derived from a poorly performing model, and this can also be seen through the scores. The answers were mainly resolved for all explanations between the answers "explanation 1 is more appropriate than explanation 2", these being more affirmative (with the adverb "far") or not (with the adverb "rather").
Finally, it is interesting to note the distribution of answers given by the users per number of texts provided. We propose in detail the distributions of the different responses for each input text, evaluated according to the confidence given by the user. They can be seen respectively in the graphs proposed in Figure 5.13 for T-EBAnO, in Figure 5.14 for LIME and in Figure 5.15 for SHAP.

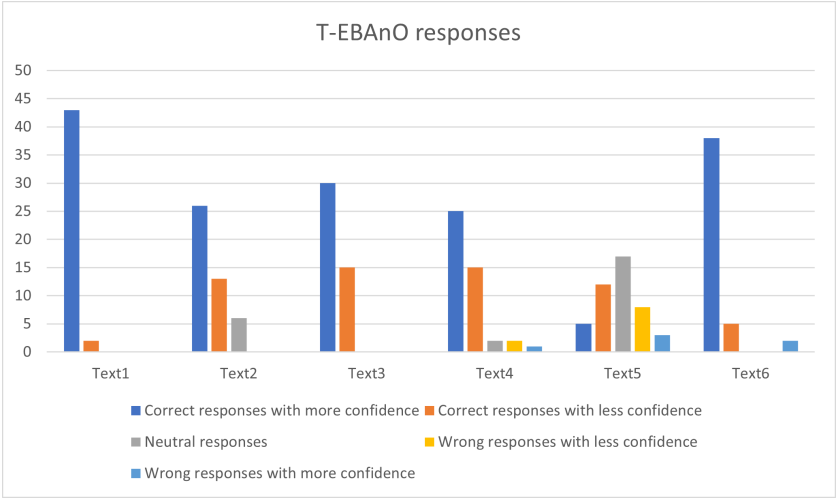


Figure 5.13: T-EBAnO distribution of responses per input text for section 2.

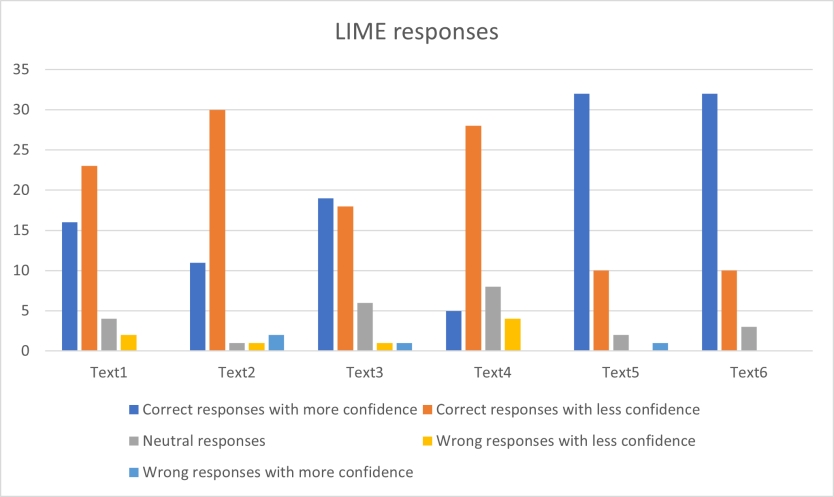


Figure 5.14: LIME distribution of responses per input text for section 2.

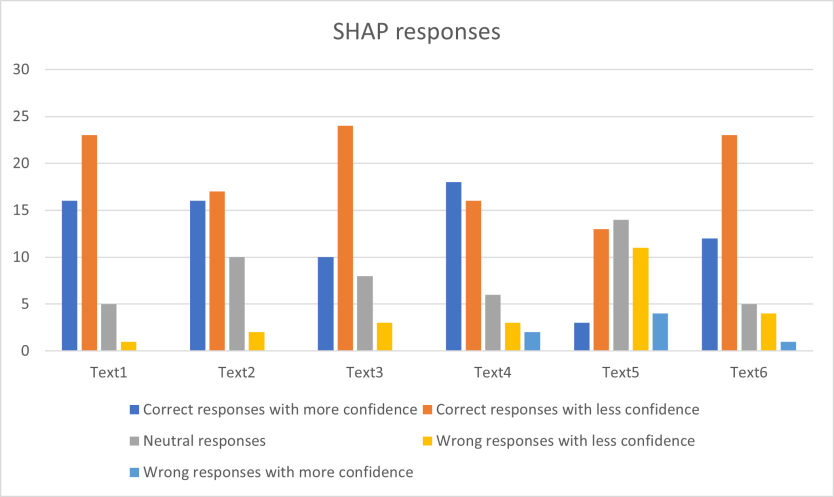


Figure 5.15: SHAP distribution of responses per input text for section 2.

Furthermore, in Table 5.2, for each input text and for each technique, the highest number of responses was counted. For example, for T-EBAnO the users, out of six input texts, gave a higher distribution of responses in the answers where there was the highest confidence. In contrast, for the SHAP technique, the distribution of responses was higher for responses that were correct but had less confidence. This is also visible in the graphs of the distributions of correct and incorrect responses above for each technique.

	Correct responses with more confidence	Correct responses with less confidence	Neutral responses	Wrong responses with less confidence	Wrong responses with more confidence
T-EBAnO	5		1		
LIME	3	3			
SHAP	1	4	1		

Table 5.2: Counting the highest number of responses per input text for section 2.

5.2.3 Third section: accuracy

In this section, as mentioned, an attempt has been made to assess the goodness of explanations in justifying the behaviour of the model.
Six questions were proposed for each explanation technique, making a total of 18 questions.

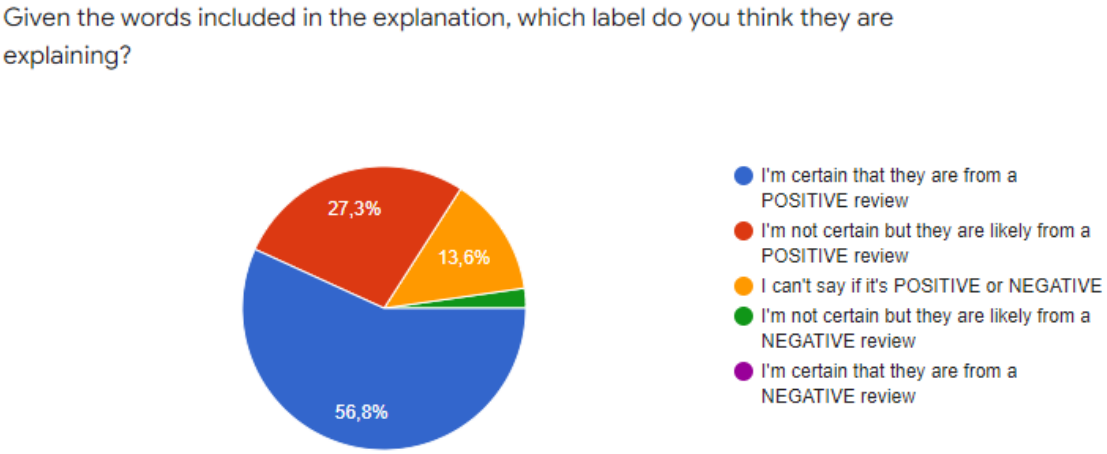


Figure 5.16: Example of response mapping for section 3.

Thanks to the scores setting, it was possible to map an overall result of the performance of the XAI techniques, shown in Table 5.3. It can be seen that T-EBAnO is again the best performing technique, with an overall score of 222.5, followed by LIME with a score of 169, and lastly SHAP with 145.5 points. The average overall score per input text provided to the user is also shown. Compared to the task in section 2, it can be seen that the average score of T-EBAnO increased from 31.17 to 37.08. LIME remained stationary with a previous average of 27.75 to 28.17, while SHAP also underwent a mediocre increase from 19 to 24.25.

	Total	**Average score per input text**
T-EBAnO	222.5	37.08
LIME	169	28.17
SHAP	145.5	24.25

Table 5.3: Scores for section 3 of the survey.

The results of this task showed that the user had no difficulty in recognising most of the explanations and which predicted label they referred to. This was mainly due to the high accuracy of the model and the validity of the explanation techniques. However, it was possible to recognise subtle differences between the three different techniques and their performance. We propose in detail the distributions of the different responses for each input text, evaluated according to the confidence given by the user. They can be seen respectively in the graphs proposed in Figure 5.17 for T-EBAnO, in Figure 5.18 for LIME and in Figure 5.19 for SHAP. As can be seen from these graphs, T-EBAnO appears to have a higher number of correct responses with high confidence, while the correct responses of both LIME and SHAP alternate between high and medium confidence, showing no preponderance on this side. On the other hand, it can be observed that LIME and, at a slightly higher level, SHAP present many more neutral (i.e. not decisive) responses than T-EBAnO, whose explanations turned out to be much clearer and more precise.

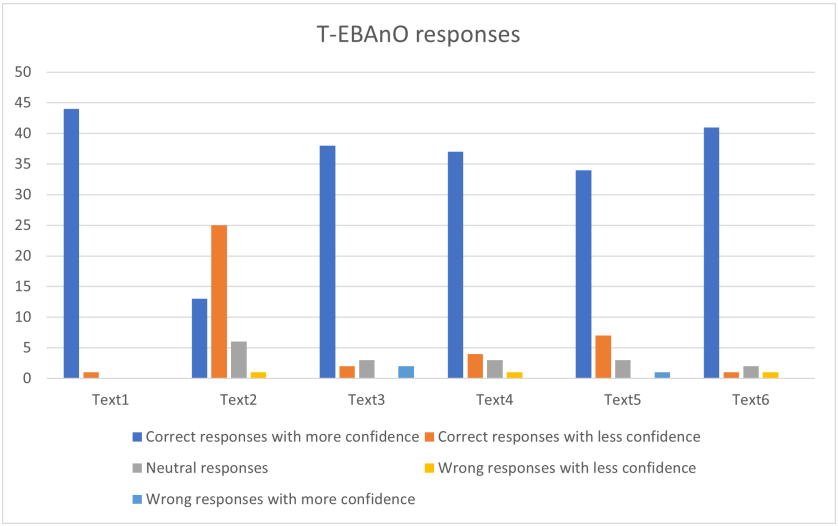


Figure 5.17: T-EBAnO distribution of responses per input text for section 3.

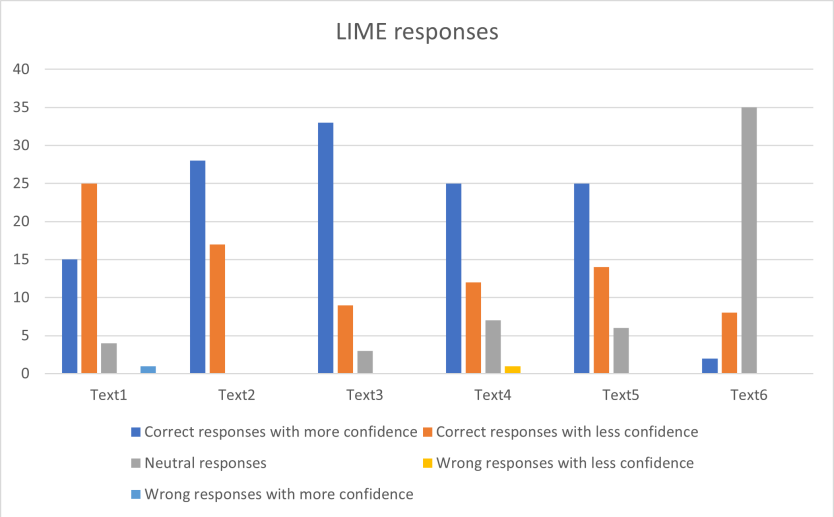


Figure 5.18: LIME distribution of responses per input text for section 3.

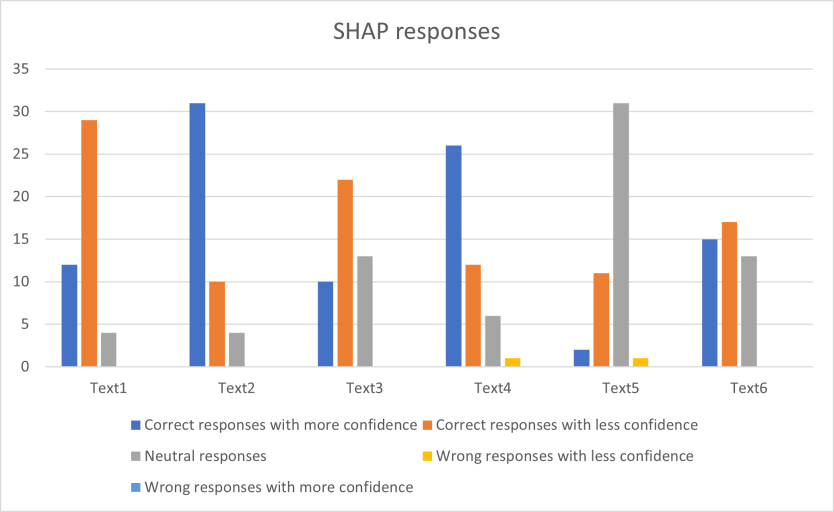


Figure 5.19: SHAP distribution of responses per input text for section 3.

In Table 5.4, for each input text and for each technique, the highest number of responses was counted. This table shows the phenomenon that presented this task, described above. T-EBAnO differs from the other two techniques by correct responses with high confidence and no prevalence of neutral or incorrect responses. It is followed by LIME with four prevalences of correct responses with high confidence, however with a prevalence of neutral responses, and finally SHAP with a prevalence of correct responses with medium confidence.

	Correct responses with more confidence	Correct responses with less confidence	Neutral responses	Wrong responses with less confidence	Wrong responses with more confidence
T-EBAnO	5		1		
LIME	4	1	1		
SHAP	2	3	1		

Table 5.4: Counting the highest number of responses per input text for section 3.

5.2.4 Fourth section: reliability

In this section, as mentioned, an attempt has been made to assess the goodness of explanations in helping humans investigate uncertain predictions of the model. Six questions were proposed for each explanation technique, making a total of 18 questions.

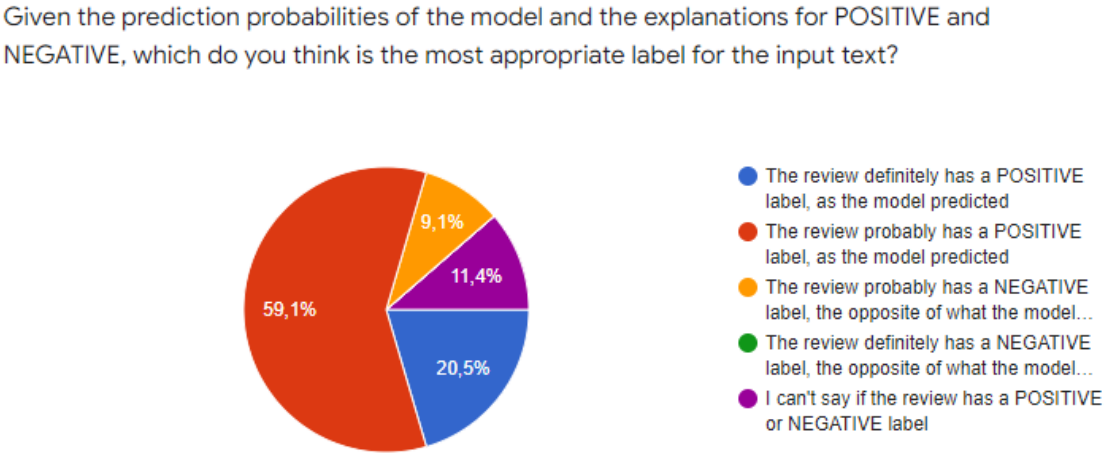


Figure 5.20: Example of response mapping for section 4.

Thanks to the scores setting, it was possible to map an overall result of the performance of the XAI techniques, shown in Table 5.5. We can immediately see a drastic decrease in the score compared to the previous two tasks, both for the total score and the average score per input text. This task, in fact, compared to the previous ones, made the user a little more uncertain about the responses to be selected. However, from the scores it is possible to observe the same trend of positioning of the techniques: T-EBAnO is positioned at the top with a score of 67.5, much lower than the scores of the previous tasks (187 and 225.5), however higher than the totals of the other two techniques. LIME follows with a score of 31.5 and SHAP is again in last position with an even negative score of -2.

It is not difficult to understand that this task was much more challenging for users to complete. This is mainly due to the construction of the task itself, i.e. starting from a model with uncertain starting predictions. By looking in detail at the distributions of responses for each text and technique, it is possible to discover the actual construction of the final scores. The distributions can be seen

	Total	Average score per input text
T-EBAnO	67.5	11.25
LIME	31.5	5.25
SHAP	-2	-0.33

Table 5.5: Scores for section 4 of the survey.

respectively in the graphs proposed in Figure 5.21 for T-EBAnO, in Figure 5.22 for LIME and in Figure 5.23 for SHAP. From the graph of T-EBAnO responses, it is possible to see a totally different trend in responses from the two previous tasks. In fact, while the previous responses were generally divided between correct responses with high and medium confidence, in this case the responses are prevalently correct but with medium confidence, or neutral, not to mention the «text 3», in which the prevalence of responses is wrong with high confidence. In contrast, for LIME and SHAP there is a general trend for both techniques in the prevalence of neutral responses. LIME also differs in correct responses with medium confidence with a prevalence in «text 1» of wrong responses with medium confidence. SHAP, on the other hand, has the opposite trend, that is, only for «text 2» does it have a prevalence of correct responses with medium confidence, while for almost all the other texts it has a prevalence of wrong responses with medium confidence.

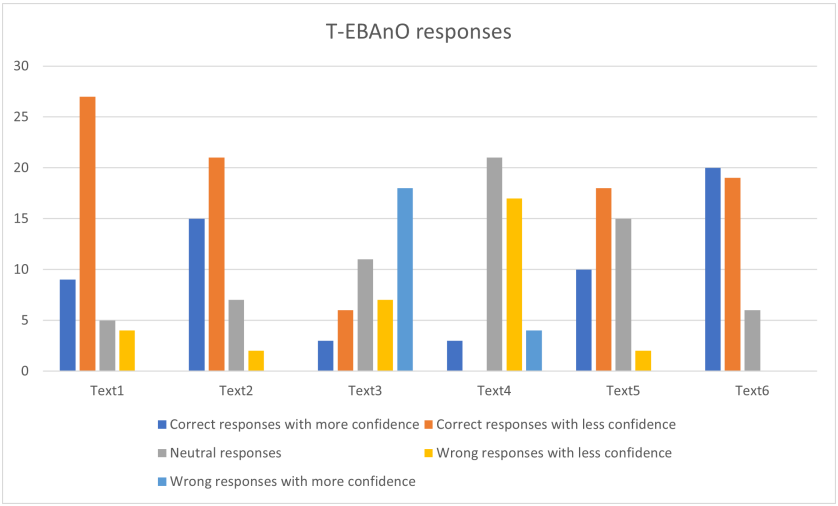


Figure 5.21: T-EBAnO distribution of responses per input text for section 4.

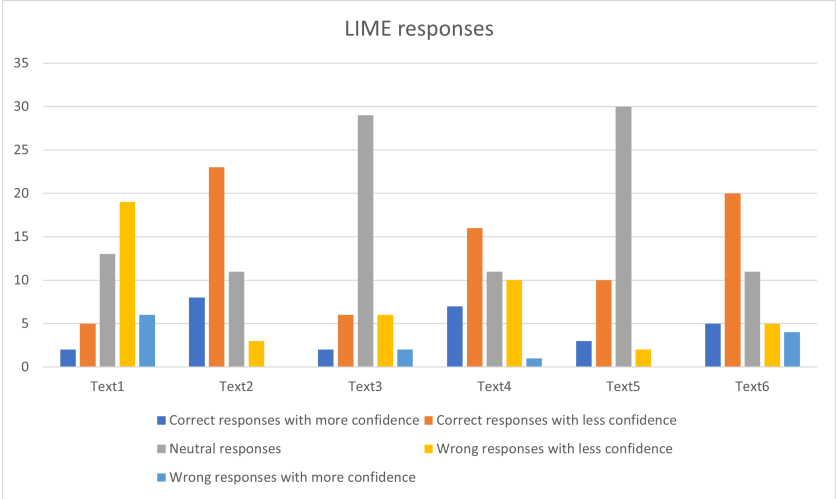


Figure 5.22: LIME distribution of responses per input text for section 4.

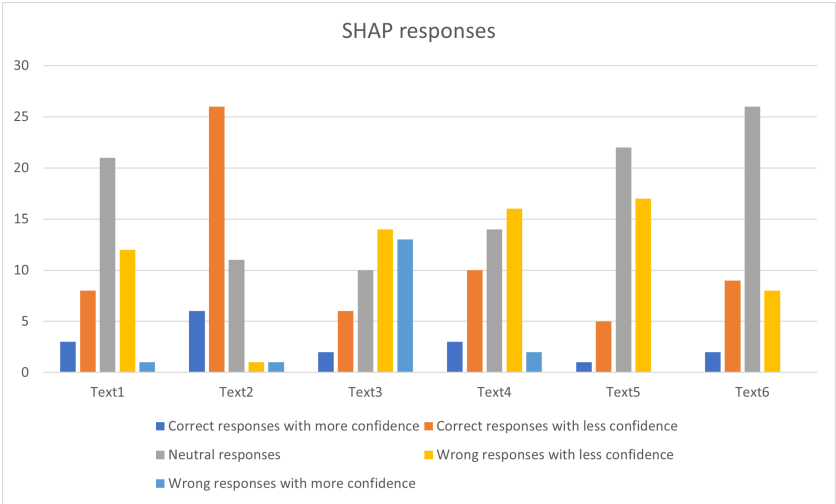


Figure 5.23: SHAP distribution of responses per input text for section 4.

In Table 5.6, for each input text and for each technique, the highest number of responses was counted. Again, it can be seen that the trend compared to the two previous tasks has changed. T-EBAnO is the only technique that remains with a prevalence of correct responses with high confidence, but this occurs only for an input text. Both T-EBAnO and LIME stand out for the majority of correct responses with medium confidence, while SHAP has a main prevalence of neutral responses, followed by wrong responses with medium confidence.

	Correct responses with more confidence	Correct responses with less confidence	Neutral responses	Wrong responses with less confidence	Wrong responses with more confidence
T-EBAnO	1	3	1		1
LIME		3	2	1	
SHAP		1	3	2	

Table 5.6: Counting the highest number of responses per input text for section 4.

5.2.5 Fifth section: comprehensibility, completeness and usefulness

In this section, as mentioned, an attempt has been made to study a more subjective evaluation, focusing on the first impact and the general understanding of the individual.
Three questions were proposed for each explanation technique, making a total of 9 questions.

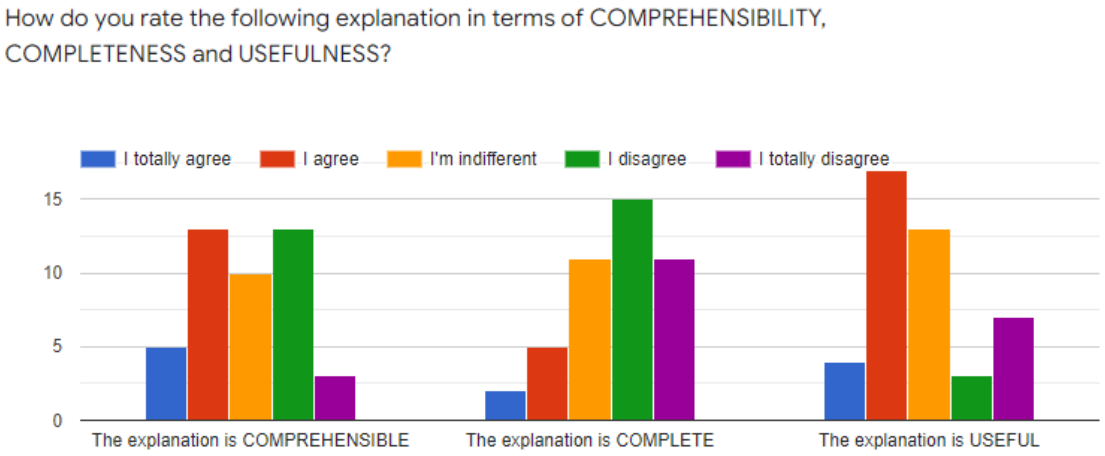


Figure 5.24: Example of response mapping for section 5.

Thanks to the scores setting, it was possible to map an overall result of the performance of the XAI techniques, respectively the total scores in Table 5.7 and the average scores per input text in Table 5.8. The way this task is formulated, it is not appropriate to compare it directly with the previous tasks, and this is also due to the different marks assigned to it. However, it is possible to make some general considerations on the three different subjective metrics proposed, namely comprehensibility, completeness and usefulness. As can be seen from the data proposed, the explanations of the T-EBAnO texts stand out for all three metrics considered, with a more marked detachment for usefulness, and a less marked one for completeness. SHAP is also in an intermediate position in all three metrics, with a large gap from T-EBAnO for comprehensibility and usefulness, but comes very close to T-EBAnO for completeness. Finally, LIME is the technique that is in last position for all three metrics, almost catching up with SHAP for comprehensibility

and usefulness, but pulling away completely for completeness, almost bordering on a zero overall score.

	Comprehensibility	**Completeness**	**Usefulness**	**Grand Total**
T-EBAnO	38.25	24	43.75	106
LIME	21	1.25	22.25	44.5
SHAP	25	18.5	29	72.5

Table 5.7: Total scores for section 5 of the survey.

	Comprehensibility (avg)	**Completeness (avg)**	**Usefulness (avg)**
T-EBAnO	12.75	8	14.58
LIME	7	0.42	7.42
SHAP	8.33	6.17	9.67

Table 5.8: Average scores per input text for section 5 of the survey.

The purpose of this task is very important to place it side by side with tasks that are much more objective. The results so far have shown that in all three of the above tasks, the ranking of techniques is drawn up with T-EBAnO ahead of LIME, followed by SHAP. However, attention must also be paid to the actual comprehensibility at first glance of the user and the usefulness of the explanation for the general understanding of the context. The different distributions can be seen respectively in the graphs proposed in Figure 5.25 for T-EBAnO, in Figure 5.26 for LIME and in Figure 5.27 for SHAP. T-EBAnO responses are predominantly «very high» or «high» for all three metrics, and interestingly there is almost no response of type «sufficient» or «scarce», except for input text 3. In the LIME responses graph, however, the distribution of responses is more differentiated, with different mean and negative scores. It is also interesting to note that for input text 3 the distribution on a «very high» evaluation of all three metrics is prevalent, which is significantly different from what was observed before on T-EBAnO, even though they represent two different explanations on the very same input text. Finally, the distribution of SHAP responses is much more focused on «high» and «average» responses, and has few «very high» responses compared to T-EBAnO. However, it is possible to understand the reason for the higher score compared to LIME, SHAP having a low preponderance of responses on «sufficient», except for input text 2.

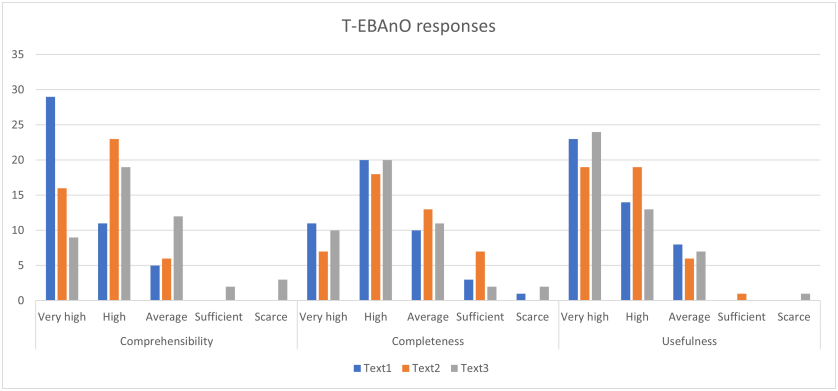


Figure 5.25: T-EBAnO distribution of responses per input text for section 5.

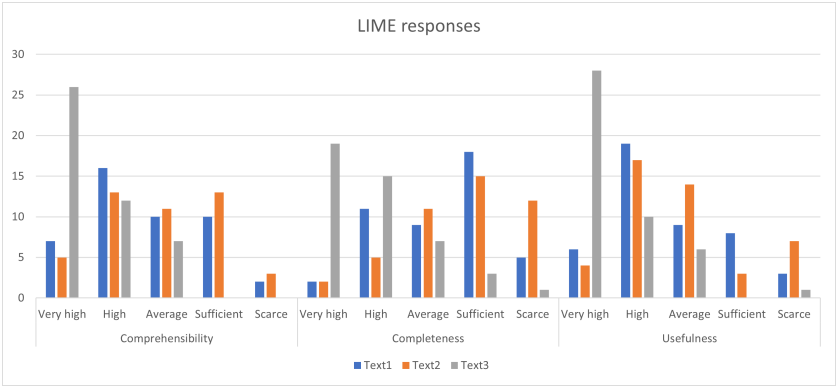


Figure 5.26: LIME distribution of responses per input text for section 5.

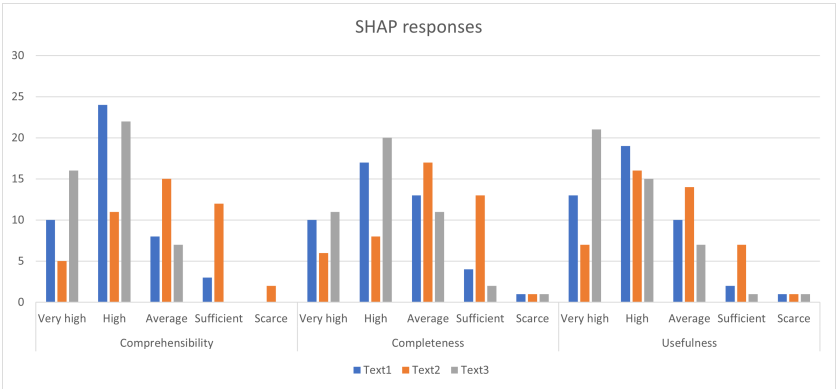


Figure 5.27: SHAP distribution of responses per input text for section 5.

In conclusion, in this task, differences in techniques are less evident in terms of metrics than in terms of text. This phenomenon may be due to the reduction of

the input text proposed to the user, so it would be advisable to carry out more in-depth analyses, proposing a higher number of input texts in a future survey.

5.2.6 Sixth section: visualisation

In this section, as mentioned, an attempt has been made to study the display of the different techniques in the first output they return, testing whether it is readable. One question was proposed for each explanation technique, making a total of 3 questions, since the visualisation is pretty much the same for all input texts.

The total scores for this last task are listed in Table 5.9. As can be seen, LIME stands out from the other techniques, as well as being the only technique with a positive score of 3 points. It is followed by T-EBAnO with -2.5 points and finally SHAP with a score of -8.

	Visualisation
T-EBAnO	-2.75
LIME	3
SHAP	-8

Table 5.9: Scores for section 6 of the survey.

In order to investigate the distribution of responses more accurately, the graph in Figure 5.28 is helpful. We can observe that only LIME receives an acceptable number of votes between «very good» (6 votes) and «good» (16 votes). The technique that stands out the most in the whole graph in terms of the number of «scarce» votes is SHAP, which in fact ranks last in the overall score. T-EBAnO, on the other hand, has a distribution of responses mainly on «good» and «average».

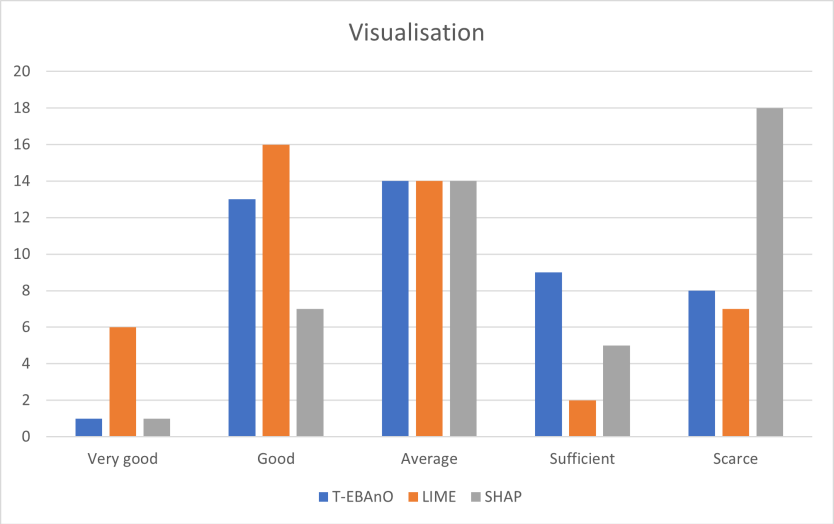


Figure 5.28: Distribution of responses per input text for section 6.

This result is very much dictated by the audience to whom the question is submitted, because it is purely based on personal interpretation for non-expert users. In order to better understand the results it is in fact necessary to place side by side the graphs of the first introductory section, so that we can understand the composition of the users who participated in this survey. As a matter of fact, we recall that 80% of the users were familiar with the concept of Artificial Intelligence, however only 17.8% were familiar with the techniques and the concept of Explainable AI. This mainly explains the results obtained in the latter section: LIME has to all intents and purposes a much more readable output than T-EBAnO and SHAP, as it is much more visually immediate to users who are not familiar with the functioning of XAI techniques. SHAP, in fact, placed in last position is a technique that offers more detail for expert users, as each word that composes the text is assigned a relevance score. This aspect, however, is confusing for non-expert users and may be too complex at first glance.

5.3 Study and comparison using objective metrics

In this section we are going to explore metrics that can be assessed objectively, i.e. through scores generated by a machine, without the help of humans. In particular, three very simple ones have been selected, but each emphasising a different desiderata of the techniques. In particular, this work is a continuation of the thesis work reported in [78].
To carry out the experiments, we decided to train a BERT (version base) [76]

model and use a binary data set (IMDb data set of 50k movie reviews [77]) and a multi-class data set (AG news data set of 120k news articles [79]) only for the last section. We trained the IMDb data set with 25k samples of the balanced training set (12.5k labelled as "Positive" and 12.5k labelled as "Negative"). The model obtained an accuracy on the other 25k samples of the test set of 93.7%. Instead, the model with AG news obtained an accuracy on the balanced test set of 7.6k of 94.7%. We trained both models for one epoch, i.e. using the whole training set once. The learning rate was 2e-5, the batch size was 8 and the optimiser was AdamW.

The difficulty already stated earlier in the various chapters is that there are still no formally totally objective and unique criteria that perfectly evaluate the different techniques, however in this thesis the following three metrics were explored:

- Execution time;
- Percentage of highlighted text;
- Prediction variance.

5.3.1 Execution time

Execution time is a more general metric, but one that nonetheless summarily quantifies the usability of an explainability technique. Execution time in fact directly mirrors the complexity of the algorithm. Execution time is measured in seconds and is directly returned by the algorithm when the run is executed.

5.3.2 Percentage of highlighted text

The percentage of highlighted text is a useful metric for quantifying the dimensionality of an explanation. This metric can be combined with the concept of comprehensibility of the explanation, as the more words highlighted in a textual explanation, the greater the degree of understanding the user has of how the model works.
The formula for this metric was calculated as follows:

$$\%\ highlighted\ text = \frac{n.highlighted\ words}{n.total\ words} * 100 \tag{5.1}$$

In order to better understand how this metric was calculated, we propose an example of a visualisation extract from one of the LIME explanation in the survey experiment.

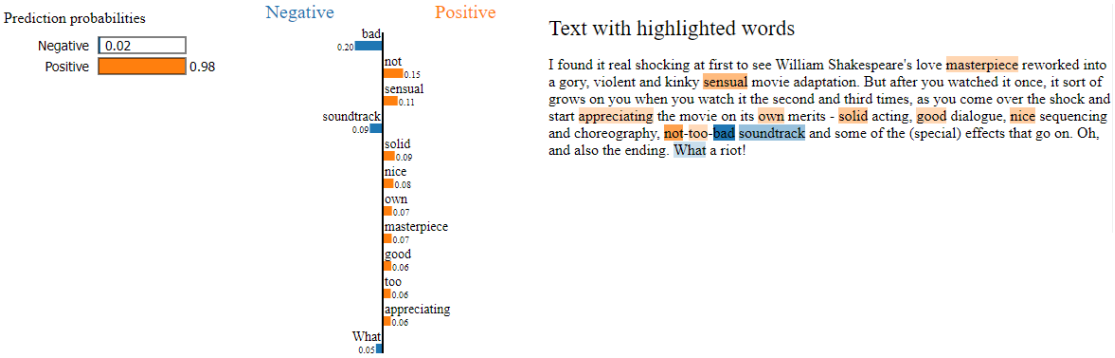


Figure 5.29: Example of the percentage of highlighted text.

In Figure 5.29, you can see the input text, the labels predicted by the model and the explanation provided by LIME. To calculate the percentage of highlighted text, we use 5.1. The number of highlighted words is 87, and the number of words in the input text is 12, so the ratio between these two numbers will be 0.138, i.e. a 13.79%.

5.3.3 Prediction variance

Prediction variance is a metric that aims to quantify the fidelity of the explanation. This metric, in fact, seeks to test the reliability of the explanation with respect to what the model predicted. The method with which this experiment was carried out is the same as that of the subsection SUBSECTION, since we wanted to attempt this experiment both from a human-based and an objective perspective. As we have already seen, the experiment consists of perturbing an input text according to which features were the most relevant in predicting a certain label, then removing these features and observing the variation of the prediction probability. If the model and explainability technique are working well, the probability of this prediction should drop significantly. For this metric we have considered both absolute and relative variation. They are calculated as follows.
The formula for the absolute variation was calculated as:

$$\%\ absolute\ variation = (final\ probability\ -\ initial\ probability) * 100 \quad (5.2)$$

The formula for the relative variation was calculated as:

$$\%\ relative\ variation = \frac{final\ probability\ -\ initial\ probability}{initial\ probability} * 100 \quad (5.3)$$

Let us now contextualise the metric by means of an example. Let's imagine we have a movie review, whose prediction for "Positive" is 90%.

Input text: "The movie is good".

The explanation for the label "Positive" is the highlighted text *"good"*. Let us now perturb this input text again, simply by removing the feature the explanation highlighted, i.e. "good". What will the prediction be?

Perturbed text: "The movie is ".

Now, let's imagine the prediction remained "Positive", but the probability has dropped to 10%. In this example, the initial probability will be 90% and the final probability will be 10%. We are now able to calculate the absolute variation from 5.2 and the relative variation from 5.3.

$$\%\ absolute\ variation = (0.10\ - 0.90) * 100 = 80\% \tag{5.4}$$

$$\%\ relative\ variation = \frac{0.10\ - 0.90}{0.90} * 100 = 88.9\% \tag{5.5}$$

5.4 Objective study results

In this section we propose and comment on all the results obtained in the different subsections of the objective study.

5.4.1 Execution time

This metric is not one of the most important, but it is very relevant because execution time is directly proportional to the feature selection time of an algorithm. If the execution times are very different from each other, it is an aspect that must be taken into account when performing an explainability technique. However, this metric depends on many other factors. The first is trivially the GPU of a computer, with which the algorithm is performed. Others are based on how the technique performs all the various steps of the algorithm. For example, T-EBAnO processing the input texts performs some steps in an aggregate way (e.g. embedding extraction) through inference. For this reason, an analysis of the complexity of the problem was not carried out, but we only wanted to analyse the average runtime as it takes for explanations on a "normal" computer. Three experiments were carried out and the results are shown in Table 5.10. The best performing technique, as can be seen, is T-EBAnO with an average time of 55.33 seconds. After that, LIME with an average time of 305 seconds and SHAP with an average time of 830.33 seconds.

	1 (s)	2 (s)	3 (s)
T-EBAnO	56	55	55
LIME	304	304	307
SHAP	717	650	1124

Table 5.10: Execution time (s) for three experiments.

5.4.2 Percentage of highlighted text

To conduct this experiment, a fixed percentage was taken from a chosen explanation technique (in this case T-EBAnO). After that, the same amount of percentage was taken for each input text as for the other techniques, LIME and SHAP. This methodology allowed greater accuracy than taking all input texts as a whole and reducing their percentage overall.
To perform the first experiment, a batch of 256 texts was taken from the IMDb data set [77] also used for human-based survey. A BERT model (version base) was then used with an accuracy of 91%. Table 5.11 shows the values of percentage of underlined text results in this experiment. For the same text, the technique that deviates slightly from the others is LIME.

	Highlighted text
T-EBAnO	17%
LIME	16%
SHAP	17%

Table 5.11: Percentage of highlighted text for IMDb data set experiment.

To perform the second experiment, a batch of 1024 texts was taken from the AG news data set [79]. A BERT model (version base) was then used with an accuracy of 93%. Table 5.12 shows the values of percentage of underlined text results in this experiment. For the same text, the technique that deviates slightly from the others is SHAP.

	Highlighted text
T-EBAnO	23%
LIME	23%
SHAP	22%

Table 5.12: Percentage of highlighted text for AG news data set experiment.

5.4.3 Prediction variance

IMDb sentiment analysis

To perform this experiment, a batch of 256 texts was taken from the IMDb data set [77] also used for human-based survey. A BERT model (version base) was then used with an accuracy of 91%. We also focused mainly on the absolute probability variation from Equation 5.2.
In order to make as fair a comparison as possible, the experiment was done by taking the same amount of text from all three explanation techniques under examination, and this value is represented by the average of what is represented for T-EBAnO, LIME and SHAP respectively. The texts were perturbed of the main features and their accuracy dropped significantly, as can be seen in Table 5.13. Specifically, we can see that the accuracy that dropped the most was that of T-EBAnO, up to 24%, followed by SHAP, up to 40% and then by LIME, up to 54%. It can be concluded that, on average, T-EBAnO predictions are those that have had the most loss of accuracy, therefore it was the technique that was most reliable with respect to the behaviour of the model.

	Accuracy of perturbed texts
T-EBAnO	0.24 (24%)
LIME	0.54 (54%)
SHAP	0.40 (40%)

Table 5.13: Comparison of the accuracy of perturbed texts

In the same way we can comment on the average decrease in the probability of the predicted class, with values visible in Table 5.14. Again, it can be seen that the loss of prediction probability affects the T-EBAnO technique more than the others.

Furthermore, we can see in detail in Table 5.15 the decrease in the mean probability of the predicted class divided by class, i.e. in this case "Positive" and "Negative". We can see that the values do not differ much between the various classes of the same technique, however they remain true to what has been said so far.

	Average decrease in the probability of the predicted class	
	μ	σ
T-EBAnO	0.77	0.32
LIME	0.48	0.44
SHAP	0.62	0.41

Table 5.14: Average decrease in the probability of the predicted class.

		Average decrease in the probability, divided by class	
		μ	σ
T-EBAnO	*Positive*	0.76	0.33
	Negative	0.77	0.31
LIME	*Positive*	0.46	0.45
	Negative	0.50	0.44
SHAP	*Positive*	0.62	0.42
	Negative	0.63	0.40

Table 5.15: Average decrease in the probability of the predicted class, divided by class

AG news topic detection

To perform this experiment, a batch of 1024 texts was taken from the AG news data set [79]. A BERT model (version base) was then used with an accuracy of 93%. We also focused mainly on the absolute probability variation from Equation 5.2.
The texts were perturbed of the main features and their accuracy dropped significantly, as can be seen in Table 5.16. Specifically, we can see that the accuracy that dropped the most was that of T-EBAnO, up to 43.8%, followed by SHAP, up to 55.4% and then by LIME, up to 68.7%. On this experiment, as well as the previous one, T-EBAnO predictions are those that have had the most loss of accuracy, therefore it was the technique that was most reliable with respect to the behaviour of the model.

	Accuracy of perturbed texts
T-EBAnO	0.43.8 (43.8%)
LIME	0.68.7 (68.7%)
SHAP	0.55.4 (55.4%)

Table 5.16: Comparison of the accuracy of perturbed texts.

We can comment as well on the average decrease in the probability of the predicted class, with values visible in Table 5.17. Again, it can be seen that the loss of prediction probability affects the T-EBAnO technique more and LIME continues to be the technique that least receives a drop in prediction probability. Compared to the previous task, the probabilities in this case are slightly lower. This may be due to the fact that more classes are present, and the original probability may have been lower. For further analysis, it may also be useful to study the relative probability detailed in Equation 5.3.

	Average decrease in the probability of the predicted class	
	μ	σ
T-EBAnO	0.58	0.43
LIME	0.30	0.42
SHAP	0.46	0.45

Table 5.17: Average decrease in the probability of the predicted class.

Moreover, we can see in detail in Table 5.18 again the decrease in the mean probability of the predicted class divided by class. In this case, there are four classes, divided according to the data set and topic detection: World, Sport, Business, Science/Tech. It can be seen that in contrast to the binary task of sentiment analysis, the values in the four classes differ slightly from each other. Examples of this are the class 4 of T-EBAnO with a mean of 0.79 and a standard deviation of 0.32, or the class 4 of LIME with a mean of 0.13 and a standard deviation of 0.28.

		Average decrease in the probability, divided by class	
		μ	σ
T-EBAnO	*Class 1*	0.45	0.45
	Class 2	0.45	0.46
	Class 3	0.62	0.41
	Class 4	0.79	0.32
LIME	*Class 1*	0.34	0.43
	Class 2	0.29	0.43
	Class 3	0.47	0.44
	Class 4	0.13	0.28
SHAP	*Class 1*	0.35	0.44
	Class 2	0.34	0.44
	Class 3	0.51	0.43
	Class 4	0.62	0.40

Table 5.18: Average decrease in the probability of the predicted class, divided by class

Chapter 6

Conclusion

6.1 Conclusions and future works

6.1.1 Conclusions

This thesis aimed to make a general state of the art on the subject, dealing with what Artificial Intelligence is and the importance of explanations, and their usefulness in today's world. A general classification of the main characteristics of the most common explanation techniques was made and a global overview of the surveys in the literature comparing explanation methods was proposed, along with a general and comprehensive methodology for comparing the different explanation techniques and their testing.
It should be reiterated that there is no absolute and completely objective comparison of these techniques, but increasingly refined methods can be constructed to indirectly assess their performance associated with the model. It should also be stressed again that a comparison of objective metrics associated with subjective metrics is the fairest way to evaluate an Explainable AI technique.

The general conclusions that can be drawn from the proposed comparative are clearly divided into the more human-based and the more objective comparative.
With regard to what emerges from the survey, it is possible to combine the overall scores of all sections. From this sum, the following results would come out: T-EBAnO scored 580.5 points, LIME scored 414.5 points and SHAP scored 322 points. The technique which, according to this survey, performs best, on average for all the different tasks proposed, is T-EBAnO. This result was also clearly visible in the individual results and graphs shown in most sections, where T-EBAnO proved to be the best performing technique in terms of clarity of explanations, human-readability and reliability in relation to the model.

As regards objective metrics, it is somewhat more difficult to draw up a final score for all the techniques, but here too it is possible to see from a general overview that T-EBAnO was the best performer in terms of execution time and the basis for the percentage of underlined text. In both prediction variance experiments, moreover, the data are very clear on the fact that T-EBAnO succeeds in selecting the most important features on which the model works, therefore by perturbing the text, it is the technique that received the greatest loss of accuracy.

6.1.2 Future works

This section proposes possible future work that can be carried out as a continuation of this thesis.
An initial work that could be taken forward is the addition of further Explainable AI techniques for evaluation by the proposed comparative. In fact, for the purposes of this thesis, the comparison of only three explainability techniques was sufficient; however, a total of 17 XAI techniques are proposed in the thesis, of which 11 support textual data. Another possible future work would be to expand this comparison to techniques that only support images.

Furthermore, in the subjective comparison, i.e. the survey, only the IMDb movie reviews data set was used, whereas for the objective comparison the AG news articles data set was also added. A future work that could be carried out is to add further data sets to make the comparison even more detailed and precise. This especially could be valid for the survey, for which only one data set was chosen due to the excessive length of compilation by the user. In addition, adding just the AG news data set could result in a more refined survey, as it is a multi-class data set, therefore with possible different performances.
Going further into the details of this possible future experiment, one could add many more input texts to the experiments and especially vary them in length. What was found in the visualisation section (Subsection 5.2.6) was that the users' rating was far better for one text than another. It would be good to see if this is something to do with the technique itself, or with the variety of texts, there being only three different ones for that section.

In addition, a further extension of the human-based comparison is surely to reach many more users, or to differentiate them. It would be possible to propose a comparison for expert users only and one for non-expert users only, and to analyse how the results differ from each other. For example, for the visualisation section, or for the interpretation of explanations, or for guessing the exact label when the model is uncertain, would anything change according to the user base?

Finally, as far as objective metrics are concerned, it is possible to greatly expand the experiment to other techniques and more data sets. It would be advisable to do a complexity analysis for the running time and expand the experiment by percentage of highlighted text. With regard to the prediction variance experiment, it would be possible to construct a similar experiment by adding a section to the survey and asking the user for their trustworthiness towards the model, using the explainability technique.

Bibliography

[1] Marco Ribeiro, Sameer Singh, and Carlos Guestrin. «"Why Should I Trust You?": Explaining the Predictions of Any Classifier». In: Feb. 2016, pp. 97–101. DOI: 10.18653/v1/N16-3020 (cit. on pp. 1, 3, 4, 14, 16–21, 24–27, 29, 30, 46, 51).

[2] Joshua A. Kroll, Joanna Huey, Solon Barocas, Edward W. Felten, Joel R. Reidenberg, David G. Robinson, and Harlan Yu. «Accountable algorithms». English (US). In: *University of Pennsylvania Law Review* 165.3 (Feb. 2017), pp. 633–705. ISSN: 0041-9907 (cit. on p. 2).

[3] David Danks and Alex John London. «Regulating Autonomous Systems: Beyond Standards». In: *IEEE Intelligent Systems* 32.1 (2017), pp. 88–91. DOI: 10.1109/MIS.2017.1 (cit. on p. 2).

[4] John Kingston. «Artificial Intelligence and Legal Liability». English. In: *Research and Development in Intelligent Systems XXXIII: Incorporating Applications and Innovations in Intelligent Systems XXIV*. Ed. by Max Brame and Miltiadis Petridis. Springer-Verlag, Dec. 2016, pp. 269–279. ISBN: 9783319471747. DOI: 10.1007/978-3-319-47175-4_20 (cit. on p. 2).

[5] Riccardo Guidotti, Anna Monreale, Salvatore Ruggieri, Franco Turini, Fosca Giannotti, and Dino Pedreschi. «A Survey of Methods for Explaining Black Box Models». In: *ACM Comput. Surv.* 51.5 (Aug. 2018). ISSN: 0360-0300. DOI: 10.1145/3236009. URL: https://doi.org/10.1145/3236009 (cit. on p. 2).

[6] Michael Hind. «Explaining Explainable AI». In: *XRDS* 25.3 (Apr. 2019), pp. 16–19. ISSN: 1528-4972. DOI: 10.1145/3313096. URL: https://doi.org/10.1145/3313096 (cit. on pp. 2, 9).

[7] https://towardsdatascience.com/human-interpretable-machine-learning-part-1-the-need-and-importance-of-model-interpretation-2ed758f5f476. In: () (cit. on p. 3).

[8] Ioannis Kakogeorgiou and Konstantinos Karantzalos. «Evaluating Explainable Artificial Intelligence Methods for Multi-label Deep Learning Classification Tasks in Remote Sensing». In: (Apr. 2021) (cit. on p. 3).

[9] T. Schnake, O. Eberle, J. Lederer, S. Nakajima, K. T. Schutt, K. Mueller, and G. Montavon. «Higher-Order Explanations of Graph Neural Networks via Relevant Walks». In: *IEEE Transactions on Pattern Analysis & Machine Intelligence* 01 (Sept. 5555), pp. 1–1. ISSN: 1939-3539. DOI: 10.1109/TPAMI.2021.3115452 (cit. on p. 3).

[10] Wojciech Samek, Gregoire Montavon, Sebastian Lapuschkin, Christopher Anders, and Klaus-Robert Muller. «Explaining Deep Neural Networks and Beyond: A Review of Methods and Applications». In: *Proceedings of the IEEE* 109 (Mar. 2021), pp. 247–278. DOI: 10.1109/JPROC.2021.3060483 (cit. on p. 3).

[11] Scott Lundberg and Su-In Lee. «A Unified Approach to Interpreting Model Predictions». In: Dec. 2017 (cit. on pp. 4, 14, 16, 18–21, 24, 28, 30, 68).

[12] Francesco Ventura, Salvatore Greco, Daniele Apiletti, and Tania Cerquitelli. «Explaining the Deep Natural Language Processing by Mining Textual Interpretable Features». In: (June 2021) (cit. on pp. 4, 14, 16, 18–21, 24, 39–41).

[13] Philip Howard and Bence Kollanyi. «Bots, StrongerIn, and Brexit: Computational Propaganda during the UK-EU Referendum». In: *SSRN Electronic Journal* (June 2016). DOI: 10.2139/ssrn.2798311 (cit. on p. 5).

[14] Samuel Woolley. «Automating power: Social bot interference in global politics». In: *First Monday* 21 (Mar. 2016). DOI: 10.5210/fm.v21i4.6161 (cit. on p. 5).

[15] Shikha Verma. «Weapons of Math Destruction: How Big Data Increases Inequality and Threatens Democracy». In: *Vikalpa: The Journal for Decision Makers* 44 (June 2019), pp. 97–98. DOI: 10.1177/0256090919853933 (cit. on pp. 5, 6).

[16] Bryce Goodman and Seth Flaxman. «EU regulations on algorithmic decision-making and a "right to explanation"». In: *AI Magazine* 38 (June 2016). DOI: 10.1609/aimag.v38i3.2741 (cit. on p. 6).

[17] Alexandra Chouldechova. «Fair Prediction with Disparate Impact: A Study of Bias in Recidivism Prediction Instruments». In: *Big Data* 5 (Oct. 2016). DOI: 10.1089/big.2016.0047 (cit. on p. 6).

[18] Amit Datta, Michael Tschantz, and Anupam Datta. «Automated Experiments on Ad Privacy Settings». In: *Proceedings on Privacy Enhancing Technologies* 1 (Apr. 2015). DOI: 10.1515/popets-2015-0007 (cit. on p. 6).

[19] Latanya Sweeney. «Discrimination in Online Ad Delivery». In: *Commun. ACM* 56.5 (May 2013), pp. 44–54. ISSN: 0001-0782. DOI: 10.1145/2447976.2447990. URL: https://doi.org/10.1145/2447976.2447990 (cit. on p. 6).

[20] Engin Bozdag and Jeroen van den hoven. «Breaking the filter bubble: democracy and design». In: *Ethics and Information Technology* 17 (Dec. 2015). DOI: 10.1007/s10676-015-9380-y (cit. on p. 6).

[21] Victoria Bellotti and Keith Edwards. «Intelligibility and Accountability: Human Considerations in Context-Aware Systems». In: *Hum.-Comput. Interact.* 16.2 (Dec. 2001), pp. 193–212. ISSN: 0737-0024. DOI: 10.1207/S15327051HCI16234_05. URL: https://doi.org/10.1207/S15327051HCI16234_05 (cit. on p. 6).

[22] Nicholas Diakopoulos. «Enabling Accountability of Algorithmic Media: Transparency as a Constructive and Critical Lens». In: May 2017, pp. 25–43. ISBN: 978-3-319-54023-8. DOI: 10.1007/978-3-319-54024-5_2 (cit. on p. 6).

[23] Brian Y. Lim, Anind K. Dey, and Daniel Avrahami. «Why and Why Not Explanations Improve the Intelligibility of Context-Aware Intelligent Systems». In: CHI '09. Boston, MA, USA: Association for Computing Machinery, 2009, pp. 2119–2128. ISBN: 9781605582467. DOI: 10.1145/1518701.1519023. URL: https://doi.org/10.1145/1518701.1519023 (cit. on p. 6).

[24] https://en.wikipedia.org/wiki/Machine_learning. In: () (cit. on p. 6).

[25] https://en.wikipedia.org/wiki/Explainable_artificial_intelligence. In: () (cit. on p. 7).

[26] Michael van Lent, William Fisher, and Michael Mancuso. «An Explainable Artificial Intelligence System for Small-unit Tactical Behavior». In: *AAAI*. 2004 (cit. on p. 7).

[27] Johanna D. Moore. «Explanation in Expert Systems : A Survey». In: 1988 (cit. on p. 7).

[28] David Gunning and David Aha. «DARPA's Explainable Artificial Intelligence (XAI) Program». In: *AI Magazine* 40 (June 2019), pp. 44–58. DOI: 10.1609/aimag.v40i2.2850 (cit. on p. 7).

[29] S. Barocas, S. Friedler, J. Kroll M. Hardt, S. Venka-Tasubramanian, and H. Wallach. «The FAT-ML Workshop Series on Fairness, Accountability, and Transparency in Machine Learning». In: () (cit. on p. 7).

[30] FICO. «Explainable Machine Learning Challenge». In: (). URL: https://community.fico.com/s/explainable-machine-learning-challenge (cit. on p. 7).

[31] Sina Mohseni, Niloofar Zarei, and Eric D. Ragan. «A Multidisciplinary Survey and Framework for Design and Evaluation of Explainable AI Systems». In: *ACM Trans. Interact. Intell. Syst.* 11.3–4 (Aug. 2021). ISSN: 2160-6455. DOI: 10.1145/3387166. URL: https://doi.org/10.1145/3387166 (cit. on pp. 7, 8, 10, 66, 68, 69).

[32] Emilee Rader, Kelley Cotter, and Janghee Cho. «Explanations as Mechanisms for Supporting Algorithmic Transparency». In: *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*. New York, NY, USA: Association for Computing Machinery, 2018, pp. 1–13. ISBN: 9781450356206. URL: https://doi.org/10.1145/3173574.3173677 (cit. on p. 8).

[33] Amina Adadi and Mohammed Berrada. «Peeking Inside the Black-Box: A Survey on Explainable Artificial Intelligence (XAI)». In: *IEEE Access* 6 (2018), pp. 52138–52160. DOI: 10.1109/ACCESS.2018.2870052 (cit. on pp. 8, 15, 17–20, 66).

[34] EU. «European Union General Data Protection Regulation (GDPR)». In: (). URL: http://www.eugdpr.org/ (cit. on p. 8).

[35] Serena Villata, Guido Boella, Dov M. Gabbay, and Leendert van der Torre. «A socio-cognitive model of trust using argumentation theory». In: *International Journal of Approximate Reasoning* 54.4 (2013). Eleventh European Conference on Symbolic and Quantitative Approaches to Reasoning with Uncertainty (ECSQARU 2011), pp. 541–559. ISSN: 0888-613X. DOI: https://doi.org/10.1016/j.ijar.2012.09.001. URL: https://www.sciencedirect.com/science/article/pii/S0888613X12001600 (cit. on p. 9).

[36] Robert Hoffman, Shane Mueller, Gary Klein, and Jordan Litman. «Metrics for Explainable AI: Challenges and Prospects». In: (Dec. 2018) (cit. on pp. 9, 10).

[37] Jianlong Zhou, Amir H. Gandomi, Fang Chen, and Andreas Holzinger. «Evaluating the Quality of Machine Learning Explanations: A Survey on Methods and Metrics». In: *Electronics* 10.5 (2021). ISSN: 2079-9292. DOI: 10.3390/electronics10050593. URL: https://www.mdpi.com/2079-9292/10/5/593 (cit. on pp. 11, 66).

[38] Aniek Markus, Jan Kors, and Peter Rijnbeek. «The role of explainability in creating trustworthy artificial intelligence for health care: A comprehensive survey of the terminology, design choices, and evaluation strategies». In: *Journal of Biomedical Informatics* 113 (Dec. 2020), p. 103655. DOI: 10.1016/j.jbi.2020.103655 (cit. on p. 11).

[39] Finale Doshi-Velez and Been Kim. «Towards A Rigorous Science of Interpretable Machine Learning». In: *arXiv: Machine Learning* (2017) (cit. on pp. 12, 13, 67–69).

[40] Riccardo Guidotti, Anna Monreale, Salvatore Ruggieri, Franco Turini, Fosca Giannotti, and Dino Pedreschi. «A Survey of Methods for Explaining Black Box Models». In: *ACM Comput. Surv.* 51.5 (Aug. 2018). ISSN: 0360-0300. DOI: 10.1145/3236009. URL: https://doi.org/10.1145/3236009 (cit. on pp. 12, 13, 21, 22, 66–68).

[41] Andrea Romei and Salvatore Ruggieri. «A multidisciplinary survey on discrimination analysis». In: *The Knowledge Engineering Review* 29 (2013), pp. 582–638 (cit. on pp. 13, 68).

[42] Yousra Abdul Alsahib S. Aldeen, Mazleena Salleh, and Mohammad Abdur Razzaque. «A comprehensive review on privacy preserving data mining». English. In: *SpringerPlus* 4.1 (Dec. 2015), pp. 1–36. ISSN: 2193-1801. DOI: 10.1186/s40064-015-1481-x (cit. on p. 13).

[43] Grégoire Montavon, Alexander Binder, Sebastian Lapuschkin, Wojciech Samek, and Klaus-Robert Müller. «Layer-Wise Relevance Propagation: An Overview». In: Sept. 2019, pp. 193–209. ISBN: 978-3-030-28953-9. DOI: 10.1007/978-3-030-28954-6_10 (cit. on pp. 14, 16, 18, 19, 21, 24, 31–33, 51).

[44] Avanti Shrikumar, Peyton Greenside, and Anshul Kundaje. «Learning Important Features through Propagating Activation Differences». In: *Proceedings of the 34th International Conference on Machine Learning - Volume 70*. ICML'17. Sydney, NSW, Australia: JMLR.org, 2017, pp. 3145–3153 (cit. on pp. 14, 16, 18, 19, 21, 24, 29, 30, 34–36).

[45] Ramprasaath R. Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. «Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization». In: *2017 IEEE International Conference on Computer Vision (ICCV)*. 2017, pp. 618–626. DOI: 10.1109/ICCV.2017.74 (cit. on pp. 14, 16, 18, 19, 21, 24, 37, 43, 53, 65).

[46] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. «Axiomatic Attribution for Deep Networks». In: *ArXiv* abs/1703.01365 (2017) (cit. on pp. 14, 16, 18, 19, 21, 36, 41–43, 51, 53).

[47] Vitali Petsiuk, Abir Das, and Kate Saenko. «RISE: Randomized Input Sampling for Explanation of Black-box Models». In: (June 2018) (cit. on pp. 14, 16, 18, 19, 21, 43–45, 53, 69).

[48] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. «Anchors: High-Precision Model-Agnostic Explanations». In: *AAAI*. 2018 (cit. on pp. 14, 16, 18, 19, 21, 46).

[49] Daniel Smilkov, Nikhil Thorat, Been Kim, Fernanda Viégas, and Martin Wattenberg. «SmoothGrad: removing noise by adding noise». In: (June 2017) (cit. on pp. 14, 16, 18, 19, 21, 47–49).

[50] David Alvarez-Melis and Tommi S. Jaakkola. «Towards Robust Interpretability with Self-Explaining Neural Networks». In: *Proceedings of the 32nd International Conference on Neural Information Processing Systems*. NIPS'18. Montréal, Canada: Curran Associates Inc., 2018, pp. 7786–7795 (cit. on pp. 14, 16, 18, 19, 21, 49–51).

[51] Yipei Wang and Xiaoqian Wang. «Self-Interpretable Model with Transformation Equivariant Interpretation». In: Nov. 2021 (cit. on pp. 14, 16, 18, 19, 21, 51–54).

[52] Christian F. Baumgartner, Lisa M. Koch, Kerem Can Tezcan, Jia Xi Ang, and Ender Konukoglu. *Visual Feature Attribution using Wasserstein GANs*. 2018. arXiv: 1711.08998 [cs.CV] (cit. on pp. 14, 16, 18, 19, 21, 54, 56, 58).

[53] C Bass, MD Silva, CH Sudre, P-D Tudosiu, SM Smith, and EC Robinson. «ICAM: Interpretable Classification via Disentangled Representations and Feature Attribution Mapping». In: Jan. 2020 (cit. on pp. 14, 16, 18, 19, 21, 56–58, 60).

[54] Michael Tsang, Sirisha Rambhatla, and Yan Liu. *How does this interaction affect me? Interpretable attribution for feature interactions*. 2020. arXiv: 2006.10965 [stat.ML] (cit. on pp. 14, 16, 18, 19, 21, 58, 60).

[55] Michael Tsang, Youbang Sun, Dongxu Ren, and Yan Liu. *Can I trust you more? Model-Agnostic Hierarchical Explanations*. 2018. arXiv: 1812.04801 [stat.ML] (cit. on pp. 14, 16, 18, 19, 21, 60–63).

[56] Andrei Kapishnikov, Tolga Bolukbasi, Fernanda Viégas, and Michael Terry. *XRAI: Better Attributions Through Regions*. 2019. arXiv: 1906.02825 [cs.CV] (cit. on pp. 14, 16, 18, 19, 21, 63–65).

[57] Leo Breiman. «Statistical Modeling: The Two Cultures (with comments and a rejoinder by the author)». In: *Statistical Science* 16.3 (2001), pp. 199–231. DOI: 10.1214/ss/1009213726. URL: https://doi.org/10.1214/ss/1009213726 (cit. on p. 15).

[58] Karen Simonyan and Andrew Zisserman. «Very Deep Convolutional Networks for Large-Scale Image Recognition». In: *arXiv 1409.1556* (Sept. 2014) (cit. on p. 32).

[59] Bolei Zhou, Aditya Khosla, Àgata Lapedriza, Aude Oliva, and Antonio Torralba. «Learning Deep Features for Discriminative Localization». In: Dec. 2016. DOI: 10.1109/CVPR.2016.319 (cit. on p. 36).

[60] https://towardsdatascience.com/understand-your-algorithm-with-grad-cam-d3b62fce353 In: () (cit. on pp. 38, 39).

[61] Martin Arjovsky, Soumith Chintala, and Léon Bottou. *Wasserstein GAN*. 2017. arXiv: 1701.07875 [stat.ML] (cit. on p. 54).

[62] Ambreen Hanif, Xuyun Zhang, and Steven Wood. «A Survey on Explainable Artificial Intelligence Techniques and Challenges». In: *2021 IEEE 25th International Enterprise Distributed Object Computing Workshop (EDOCW)*. 2021, pp. 81–89. DOI: 10.1109/EDOCW52865.2021.00036 (cit. on p. 66).

[63] Maksims Ivanovs, Roberts Kadikis, and Kaspars Ozols. «Perturbation-based methods for explaining deep neural networks: A survey». In: *Pattern Recognition Letters* 150 (Oct. 2021), pp. 228–234. DOI: 10.1016/j.patrec.2021.06.030 (cit. on p. 66).

[64] Piyawat Lertvittayakumjorn and Francesca Toni. *Human-grounded Evaluations of Explanation Methods for Text Classification*. 2019. arXiv: 1908.11355 [cs.CL] (cit. on pp. 66, 71–76).

[65] Udo Schlegel, Hiba Arnout, Mennatallah El-Assady, Daniela Oelke, and Daniel A. Keim. *Towards a Rigorous Evaluation of XAI Methods on Time Series*. 2019. arXiv: 1909.07082 [cs.LG] (cit. on p. 67).

[66] Xiao-Hui Li, Yuhan Shi, Haoyang Li, Wei Bai, Caleb Chen Cao, and Lei Chen. «An Experimental Study of Quantitative Evaluations on Saliency Methods». In: *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. New York, NY, USA: Association for Computing Machinery, 2021, pp. 3200–3208. ISBN: 9781450383325. URL: https://doi.org/10.1145/3447548.3467148 (cit. on p. 67).

[67] Yoseph Hailemariam, Abbas Yazdinejad, Reza M. Parizi, Gautam Srivastava, and Ali Dehghantanha. «An Empirical Evaluation of AI Deep Explainable Tools». In: *2020 IEEE Globecom Workshops (GC Wkshps)*. 2020, pp. 1–6. DOI: 10.1109/GCWkshps50303.2020.9367541 (cit. on p. 67).

[68] Yuyi Zhang, Feiran Xu, Jingying Zou, Ovanes L Petrosian, and Kirill V Krinkin. «XAI Evaluation: Evaluating Black-Box Model Explanations for Prediction». In: *2021 II International Conference on Neural Networks and Neurotechnologies (NeuroNT)*. IEEE. 2021, pp. 13–16 (cit. on p. 67).

[69] Nicholas Halliwell, Fabien Gandon, and Freddy Lecue. «User Scored Evaluation of Non-Unique Explanations for Relational Graph Convolutional Network Link Prediction on Knowledge Graphs». In: *Proceedings of the 11th on Knowledge Capture Conference*. K-CAP '21. Virtual Event, USA: Association for Computing Machinery, 2021, pp. 57–64. ISBN: 9781450384575. DOI: 10.1145/3460210.3493557. URL: https://doi.org/10.1145/3460210.3493557 (cit. on p. 67).

[70] Henrik Mucha, Sebastian Robert, Ruediger Breitschwerdt, and Michael Fellmann. «Interfaces for Explanations in Human-AI Interaction: Proposing a Design Evaluation Approach». In: *Extended Abstracts of the 2021 CHI Conference on Human Factors in Computing Systems*. New York, NY, USA: Association for Computing Machinery, 2021. ISBN: 9781450380959. URL: https://doi.org/10.1145/3411763.3451759 (cit. on p. 67).

[71] Yi-Shan Lin, Wen-Chuan Lee, and Z. Berkay Celik. *What Do You See? Evaluation of Explainable Artificial Intelligence (XAI) Interpretability through Neural Backdoors*. 2020. arXiv: 2009.10639 [cs.CV] (cit. on p. 67).

[72] Pang-Ning Tan, Michael Steinback, and Vipin Kumar. *Introduction to Data Mining*. Jan. 2006 (cit. on p. 68).

[73] Eliana Pastor and Elena Baralis. «Explaining Black Box Models by Means of Local Rules». In: *Proceedings of the 34th ACM/SIGAPP Symposium on Applied Computing*. SAC '19. Limassol, Cyprus: Association for Computing Machinery, 2019, pp. 510–517. ISBN: 9781450359337. DOI: 10.1145/3297280.3297328. URL: https://doi.org/10.1145/3297280.3297328 (cit. on p. 68).

[74] Xiao-Hui Li, Yuhan Shi, Haoyang Li, Wei Bai, Yuanwei Song, Caleb Chen Cao, and Lei Chen. *Quantitative Evaluations on Saliency Methods: An Experimental Study*. 2020. arXiv: 2012.15616 [cs.AI] (cit. on p. 69).

[75] Yoseph Hailemariam, Abbas Yazdinejad, Reza M. Parizi, Gautam Srivastava, and Ali Dehghantanha. «An Empirical Evaluation of AI Deep Explainable Tools». In: *2020 IEEE Globecom Workshops (GC Wkshps*. 2020, pp. 1–6. DOI: 10.1109/GCWkshps50303.2020.9367541 (cit. on p. 69).

[76] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. «BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding». In: *ArXiv* abs/1810.04805 (2019) (cit. on pp. 70, 96).

[77] https://www.kaggle.com/lakshmi25npathi/imdb-dataset-of-50k-movie-reviews. In: () (cit. on pp. 70, 97, 100, 101).

[78] https://webthesis.biblio.polito.it/20577/. In: () (cit. on p. 96).

[79] https://www.kaggle.com/amananandrai/ag-news-classification-dataset. In: () (cit. on pp. 97, 100, 102).