Explainability and Robustness of AI with an outlook on military applications

TPT-DATAAI951 - AI Ethics - 07.02.2023

Thomas Wimmer
TU Munich, IP Paris
thomas.wimmer@ip-paris.fr

Jules Soria
IMT Atlantique, IP Paris
jules.soria@ip-paris.fr

Maximilien Chau
Télécom SudParis, IP Paris
maximilien.chau@ip-paris.fr

Mathilde Bonin
IP Paris
mathilde.bonin@ip-paris.fr

Abstract

Explainable AI (XAI) is a rapidly growing field that aims to make the decision-making process of artificial intelligence (AI) models transparent and interpretable to humans. The ability to understand and trust the output of an AI model is becoming increasingly important as the use of AI expands into critical applications such as healthcare, finance, and autonomous systems, even being used in autonomous warfare. This paper aims to (1) provide an overview of the different methods used to achieve XAI, covering evaluation methods and challenges in XAI research, (2) provide an overview on how and to what extent AI can be proven to be robust and finally (3) an outlook on military applications in the form of lethal autonomous weapon systems (LAWS).

1. Introduction

As the use of artificial intelligence (AI) continues to expand into critical applications such as healthcare, finance, and autonomous systems, the need for Explainable AI (XAI) is becoming increasingly important. XAI aims to make the decision-making process of AI models transparent and interpretable to humans, allowing for greater understanding and trust in the output of these models. However, the use of AI in autonomous warfare as lethal autonomous weapon systems (LAWS) has raised even more critical concerns about the ability to understand and trust AI-driven decisions.

XAI plays a key role in addressing ethical concerns related to the use of AI, such as accountability and bias. Without XAI, it can be difficult to understand how an AI model arrived at a certain decision, making it difficult to identify and address any potential ethical issues.

This paper aims to give a comprehensive overview of the various techniques used to achieve Explainable AI (XAI) in Section 2, with a focus on the current challenges and evaluation methods in the field. Additionally, we explore the methods to ensure robustness of AI models in Section 3. Lastly, the paper delves into the military applications of AI, specifically in the form of Lethal Autonomous Weapon Systems (LAWS), in Section 4 providing an outlook on the future developments and ethical concerns.

2. Explainable AI

Explainable AI (XAI) was first needed as early as the 1970s to simplify the decisions of expert systems [18], a form of symbolic AI. However, with the uprise of machine learning and especially deep neural networks in the 2010s, XAI research has gained new momentum. As AI-based systems are being used in various real-world applications, sometimes even safety-critical, new requirements have emerged from both a user and a legal perspective. As a result, research in this area has seen significant growth in recent years, due in part to official research programmes such as the DARPA [7].

Making AI explainable or interpretable means making the internals of an AI system plain or comprehensible [5]. There is a difference between global and local explainability. Global explainability means making the whole functioning of an algorithm understandable, while local explainability is a slightly weaker notion, which is nonetheless important as it means that one can find explanations for a particular algorithmic decision made by the AI (e.g. for a specific user). Often, we are in fact more interested in being able to find these local explanations, hence a lot of research has been carried out in this area in recent years.

In the context of ethical considerations, explainability is a key tool to achieve accountability, which is the ability to demonstrate and accept responsibility for the proper functioning of an AI system. In other words, it is a key aspect of designing responsible AI solutions [5].

The following sections will provide an overview on the different categories of XAI concepts and the most popular methods. Finally, we will also discuss how to evaluate the explainability of a model and challenges in the field of XAI.

2.1. Methods

The following section will introduce different approaches to creating XAI. Most methods in XAI today focus on post-hoc explanations, meaning that they try to explain the behavior of a trained model that is in some cases even treated as a black-box model. The first two subsections will concentrate on these methods while first introducing methods for global (Sec. 2.1.1) and in the further course local explainability (Sec. 2.1.2). While this differentiation between global and local explanations might be intuitive, another way of structuring this section could be to differ between methods using surrogate models, statistical methods, feature-effect-based approaches and methods working with examples as mean for explanations.

A fundamentally different idea is including explainability already when designing learning algorithms, so called explainability by design. Methods that fall into this area are presented in Section 2.1.3.

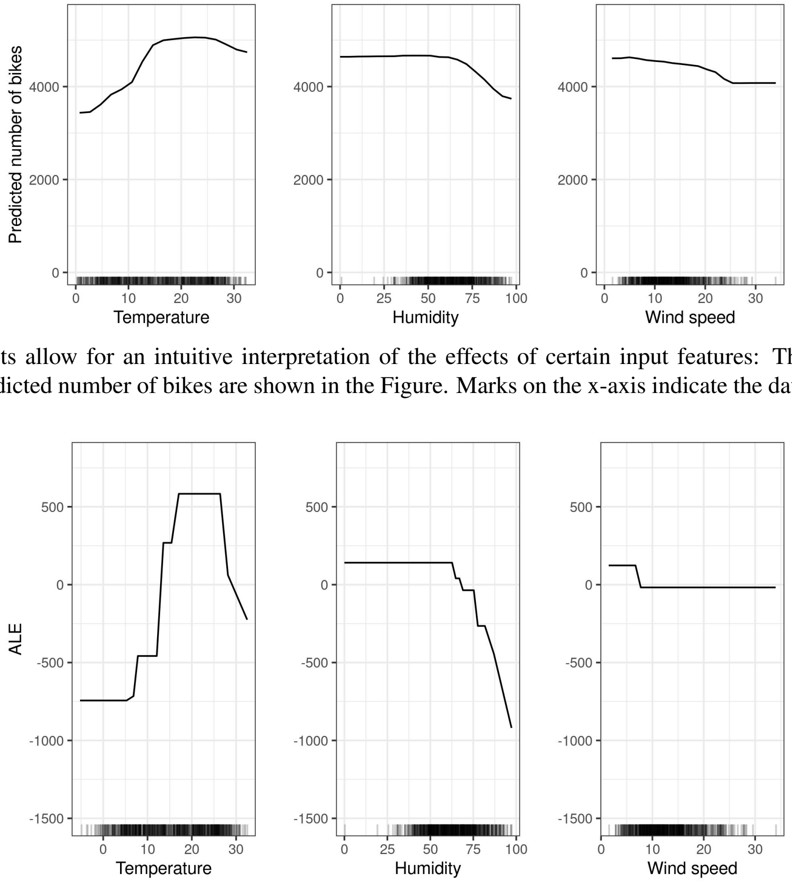
2.1.1 Global Explainability

Finding global explanations accounts to the problem of giving an understandable and interpretable way of the general behavior of an AI method. Global XAI methods thus often describe the average behavior of a machine learning model and work with the notion of expected values based on the distribution of the data [13].

This kind of explanation giving a global interpretation of the model's behavior is of immense relevance for ethical considerations around AI as these general findings can provide valuable details about the fairness of a model, through interpreting the way it treats input groups with different values for certain sensitive attributes, or even provide ways to simplify and approximate complex decision functions, as deep neural networks, with simpler methods that are easier to interpret.

Feature Effect Methods Various methods exist that try to explain a model's behavior through the effects of different input values on the prediction.

Often, we are interested in the effects of one, two or only a few of the usually many input variables to a ML model. In this case, we can analyze and visualize the behavior of the observed model through Partial Dependence Plots [6]. This method marginalizes out the other input features, through taking a simple mean over the predictions of all data samples with a fixed value for the feature that is in focus. We can repeat this process for all values of the focused attribute that we are interested in and then plot it in a plot similar to the one shown in Figure 1a. This method allows for an intuitive interpretation of the general effects of certain input features and it is easy to



(a) Partial Dependence Plots allow for an intuitive interpretation of the effects of certain input features: The effects of three different weather features on the predicted number of bikes are shown in the Figure. Marks on the x-axis indicate the data distribution. Figure taken from [13].

(b) ALE plots on the same setting. The plots are centered around zero, allowing for an intuitve interpretation of the effects of changing the specific input features on the predicted outcome. Figure taken from [13].

Figure 1. Feature Effect Methods allow for easy-to-understand global explanations.

implement. However, one assumes independence between the different input features which is a very strong and often unrealistic assumption.

Accumulated Local Effects (ALE) plots aim at avoiding this strong assumption of PDPs and work as well with correlated input features. They are centered at zero with positive values showing a positive effect on the prediction result compared to the average of the data (and vice versa for negative values) [4]. They offer an easy interpretability and are usually the preferred of the two methods but are in general more complex to implement and still sensitive to strong correlations of input features. A corresponding example is shown in Figure 1b

Global Surrogate Models Global surrogate models follow the idea of approximating the usually complex machine learning model with a simpler model that can be easily interpreted [13]. To be more precise, these methods try to fit a glassbox model (refer to Sec. 2.1.3 to input-output pairs of the complex predictor function. They thus are able to explain methods of arbitrary structure, as the ML models are treated as black-box predictors where only the outputs for a given input are needed.

Glassbox models that are considered interpretable are trained only on the outputs of the model that should be explained, not on real labels. It is, however, still necessary to have a dataset with inputs that are sampled from the data distribution on which we want to find explanations of the model's behavior to train the surrogate model.

The use of global surrogate models is an intuitive way to deal with the problem of explainability. It is flexible as it handles the models as a black-box and can thus be applied to classic ML methods, as well as more sophisticated Deep Learning methods.

With the usage of surrogate models, we inherit all advantages but also all disadvantages of the specific models used as a surrogate. This method is thus highly dependent on the specific glassbox model chosen as a surrogate. More than that, since we are training the surrogate the same way as any other ML model, we face the same challenges of balancing bias and variance, as overfitting the explaining model to the outputs of the interpreted model could have a negative impact on the explainability. On the other hand, it might be challenging to even match the predictions of a complex ML model with a usually simpler surrogate model, especially with a large input diversity. Finally, one must be careful to not step into the trap of interpreting the surrogate's behavior as an insight over the data as we are just drawing conclusions about the interpreted model and not about the data.

2.1.2 Local Explainability

Local explainability refers to the ability of the AI to explain a particular decision, meaning that it should be able to explain why an output was given with respect to its input. Local explainability is partly responsible to justify the use of an AI to solve a problem. Indeed, good results provided by an algorithm are not always the only requirement of using this algorithm. Being able to give precise explanations to decisions is especially important in high stakes applications. Moreover, metrics used to measure the performance of an AI model are not always easy to find or do not always measure what we aim to measure. Practical explanations of representative cases and edges are complementary way of assessing the performance of one mode.

Local XAI enables the algorithm maintainers to tweak and improve the model if some explanation is not satisfying. Adding local explainability to a model can also act as a way of revealing if the AI model reasons similarly to humans. In some way, we could argue that XAI acts as additional expert in the domain of use. Through collaboration with true experts, it can ease their decisions since they should be looking at the same features. Even if they are not looking at the same features, it brings room for improvement on both sides. This category of explainability can at first sight be particularly useful for industrial applications. When a client must choose between different systems, the opacity of AI often comes as a hindrance. Local explainability could provide trust in this option. It also enables checking whether an algorithm respects human rights and the legal system it is part of.

Local Surrogate Models Local surrogate models are tools that provide insights into the inner workings of black box models. These models simplify the complexity of advanced AI algorithms, making it easier to understand and interpretable. By breaking down the complexities, local surrogate models allow for a clearer understanding of how AI models arrive at their individual predictions.

Various techniques exist for implementing local surrogate models. One notable approach is ”Local Interpretable Model-Agnostic Explanations” (**LIME**) [16]. LIME examines models through repeated inputs of synthetic data that are created to mimic the properties of the original query data. It is important to note that the input perturbations can be carefully selected by experts to evaluate the reasoning and explanations of the model. Then, LIME deploys a simple, interpretable model such as a Decision Tree to fit the generated data. The inputs are weighted based on their proximity to the original query, allowing the white-box model to provide locally-explainable results. LIME provides insights into the underlying reasoning of complex AI models, making it an important tool for improving transparency and accountability in machine learning.

SHAP (Shapley Additive exPlanations) is another local surrogate model [11]. It is based on the Shapley values and the coalition game theory. Thus, it provides accuracy and consistency values to the explanation of the model. In this model, the features are the players forming the coalition. After computing the prediction for multiple coalitions, the model can average the impact of each feature of the coalition into the prediction. This method seems better than LIME but it requires a longer computation time. SHAP can be implemented with a faster computation

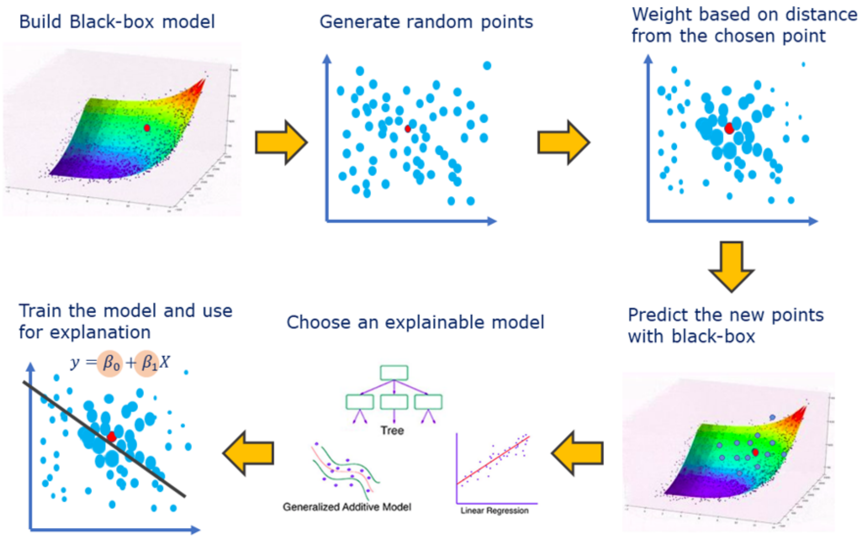

Build Black-box model
Generate random points
Weight based on distance from the chosen point
Predict the new points with black-box
Choose an explainable model
Tree
Generalized Additive Model
Linear Regression
Train the model and use for explanation
$y = \beta_0 + \beta_1 X$

Figure 2. Steps of the LIME algorithm [25]

time, however it would mean weakening the accuracy of the explanation. This issue must be balanced with the ethical outcomes that could derive from the use of this kind of less optimised model.

Both methods reveal great potential in providing accurate explanations and justifying the use of AI models for various applications. Nonetheless, those local surrogate models can be fooled by malicious users using adversary entities. Those entities can mask discriminatory explanations of strongly biased models [20]. The explanations would then only explain that the decision was taken by non-discriminatory features, not reflecting the way the original model took decisions. This leads us to understand that local surrogate models are not yet very robust, and they should be carefully implemented to avoid misleading interpretations.

Saliency Maps Saliency maps are a tool to interpret the classification decision AI models. They are local methods as they highlight the specific regions of one input that led to the classification. These maps act as a feedback mechanism for maintainers, allowing them to better understand the complex decision-making process of neural networks. The intuitive visual representation of the map makes it useful to understand the underlying mechanisms of the model.

Among the different saliency map methods, we can find the Layer-Wise Relevance Propagation (LRP). Despite not understanding how a neural network (black box model) fully reasons, LRP uses the weights composing the network and the classification for a specific input to propagate the relevance of the output class to the input. In the case of an image input, LRP will propage the relevance of the output class to the pixels or super pixels composing the image. An additional interesting aspect of LRP is that it is also effective for text and audio data. Several other methods were built on top of LRP like DeepLiFT, which can for instance provide more robust explanations to non-linear activation functions in neural networks.

Gradient-based approaches like Grad-CAM can also provide meaningful interpretations for predictions. One interesting aspect of Grad-CAM is that it performs well on localisation of objects. This method relies on the computation of the gradients on the output prediction and then produces a heatmap in a faster way than the previous methods.

Example-based Methods Example-based explanation methods are a type of model-agnostic interpretation technique. They provide explanations for a model's outputs by comparing the input data to similar examples in the training data. The idea is to identify the closest examples in the training set and use instances of the dataset to explain the prediction made by the model. Therefore, the training set size and diversity is important for those techniques to be effective. The data used for those models should be easily interpreted by humans as example-based explanations rely on inputs and not features of the input.

The anchors method uses rules deduced from examples to anchor a group of instances in a non linear area. When an anchor is created, changes in other features of instances from the area should not change the classification. Those rules are characterised by their precision and coverage. Reinforcement learning techniques lead the method to find the scoped rules and that interpretation technique requires a wide exploration of the dataset.

Counterfactual explanations are one of the most popular example-based explanation methods. When an explanation technique uses rules to deduce the prediction, it creates a causal relationship between the hypothesis of the rule and the deduction. Counterfactual explanations change one cause and check changes in the output. The modified instance can be generated and might not belong to the input dataset. The goal of counterfactual explanations is to find the smallest change modifying the output.

2.1.3 Explainability by Design

Explainability by design refers to the principle of embedding transparency and interpretability into the development of machine learning models. By incorporating explainability from the design phase, AI systems can increase trust and accountability, and minimize the risks of unintended consequences.

Glassbox Models Some ML methods are considered to be intrinsically interpretable or explainable.

The interpretability of models varies depending on their complexity. Linear models such as linear and logistic regression are considered interpretable, but their simple structure limits their ability to handle more complex data. Other ML methods such as Naive Bayes, RuleFit [6], decision trees (with limited depth), and Nearest Neighbor are also considered explainable, but usually come with drawbacks in performance as well.

Generalized Linear Models (GLMs) [14] and Generalized Additive Models (GAMs) [8] are extensions of linear regression introducing abilities to handle (limited) non-linear behavior and non-Gaussian output distributions. They are also considered explainable, but introduce more barriers to direct understanding of the model's workings.

Explainable Boosting Machines (EBMs) [10] are a recent method building upon GAMs that uses gradient boosting and ensemble learning in addition and are transparent in their explanation of the effects of input variables on the output. EBMs perform better than other glassbox models but require more training time, especially on larger datasets.

Modifying model and learning objectives Other methods that try to include model explainability directly into the model design and training procedure are presented in [5]: Modifying the loss function [26] or adversarial setups with a built-in explanator [9] aim at integrating explainability through modifying the learning objective.

Modifying the model architecture, through learning sparse representations of the data or weight vectors [1], instead of the learning objective is another method used to inject high-level explainability into models. Causal methods aiming at detecting actual causal relations [3] fall into the same category of methods.

Hybrid AI Hybrid AI aims at connecting symbolic AI and machine learning methods. Pre-processing the inputs or post-processing the outputs of ML models through symbolic rules are often used examples in this field.

Building neural networks around logic rules, through rewarding the network for following interpretable rules and representations, and thus putting constraints on the network intrinsics through symbolic reasoning are another way to create hybrid AI [5].

2.2. Evaluation

Explainable Artificial Intelligence methods are designed to provide transparency and accountability in decisions made by AI systems. Evaluation metrics are needed to ensure the explainable models are delivering on this promise. A good explainable model should be able to provide objectively accurate and consistent explanations, and convince users of their reliability. The question that arises is, how we actually measure the quality of an explanation.

[24] describes several challenges for evaluating XAI:

- XAI methods are still not enough structured and explainability itself needs a clear definition that the scientific community agrees upon.
- The definition of explainability for AI needs to be adapted to a wider range of applications, reaching from medicine to social sciences.
- The evaluation of XAI methods is a complex field of study that needs to directly include humans as they are the final consumers of explanations. Possible solutions could be adapted from the research field of human-computer interaction.
- Objective evaluation of provided explanations is not always possible. While methods treating images are easier to handle with mathematical metrics and enough human labeling, this might not be the case for more complex scenarios. Here, we need a human-in-the-loop evaluation, as known from social sciences.

Scholars have proposed several metrics to formally and objectively evaluate the methods of explainability. The interpretability of these models vary depending on the learning algorithm, the learning architecture and its hyper-parameters. For example, in the case of image semantic segmentation, the metric, location instability, could be used to check if a CNN locates relevant parts of an object at a constant relative distance. Other proposed metrics such as BLEU, METEOR, and CIDEr focus on automatic evaluation of textual explanations. A number of researchers proposed formal analysis among explainable models by evaluating their robustness to input perturbation. Intuitively, if the input being explained undergoes a slight modification that doesn't affect the prediction made by the model, then the explanation provided by the explainable model should not change much. On the other hand, if we focus on the resulting explanations, metrics can be used to assess the completeness of the explanations by analyzing partial derivatives of the output or network weights [22].

Using explainability techniques to provide insight into a model's decision-making process can be beneficial, but it's important to exercise caution. As the distance between the model's implementation and its end user increases, the user's understanding of how the model operates decreases. The trust in the model can lead to negative outcomes due to the potential risks it presents. The magnitude of these risks is proportional to the trust placed in the model, as the user may fall victim to automation bias. This bias minimises the effectiveness of collaboration between human users and XAI systems that have reasoning capabilities. If only one of these actors is reasoning, the automation bias will cause the human user to follow the reasoning of the XAI system, which may be biassed. This danger is even greater when explainable models are not robust, as they can be manipulated by malicious entities.

Human mental models, as explained in the literature [17] are small scaled models of how systems work. In the case of a cooperation with XAI, this mental model can be modified to lead to greater automation bias. Experiences have shown that the automation bias was strengthened through the cooperation of XAI and human actors. For this reason, AI end users should both have knowledge in the problem they are dealing with and in the AI boundaries

3. Robustness in AI

Over the past decade, there has been a surge of both technical and theoretical, as well as both hardware and software, advancements, oriented towards the exponential use of Deep neural network models. As they are able to complete more difficult tasks than their previous alternatives, they are being used in diverse fields and applications.

Indeed, today neural networks are participating in safety-critical, security-critical, and socially critical tasks. But neural networks are delicate and it is of utmost importance that we prove that they are well-behaved when applied in critical settings.

Robustness is the most widely researched correctness property in neural networks due to its generality and the fact that deep learning models are known for their vulnerability. This means that small perturbations to inputs should not cause significant changes in the neural network's output. For instance, altering just a few pixels in a photo shouldn't cause the network to mistake the person for a cupboard, or adding unobtrusive noise to a recorded lecture shouldn't cause the network to believe it's about the Ming dynasty in the 15th century. Furthermore, an autonomous vehicle equipped with cameras that rely on traffic signs is susceptible to missing a stop sign if it is slightly vandalized. This could result in a collision. Another example is a neural network designed to identify malware. It is crucial that even a slight modification to the malware's binary code does not cause the detector to mistakenly classify it as safe to install. [2]

3.1. Adversarial Examples

It is widely known that Deep Neural Networks (DNNs) are susceptible to adversarial perturbations. In this example this means that slight changes in points that are correctly classified can be misclassified. Below is a demonstration of an adversarial attack on a trained a classifier on the MNIST dataset, the most basic dataset in Machine Learning, composed of handwritten digits.

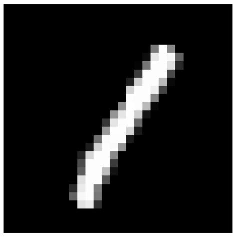


Figure 3. Handwritten 1 from the MNIST dataset

The robustness test is quite simple mathematically, it consists of creating a set $\mathcal{X}$ of images close (with regards to some distance ϵ) to the one in Figure 3 and checking if the following statement holds :

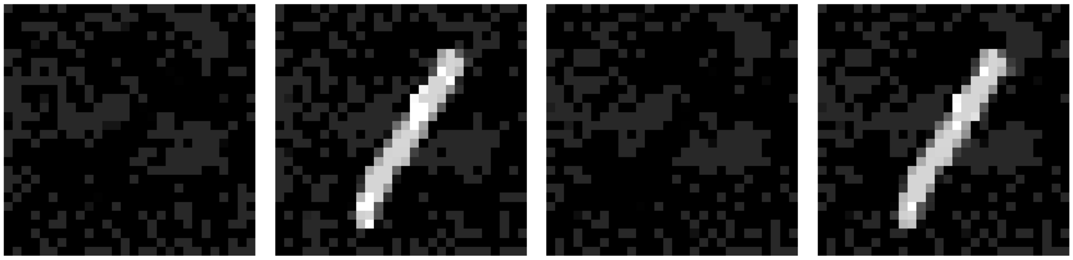
$$x \in \mathcal{X} \Rightarrow f(x) = y \in \mathcal{Y}$$

where $\mathcal{Y}$ represents the output set, and in our case corresponds to the image being classified as a 1. Most programming languages have operational logical verification libraries which allow to check whether such statement holds or not, and can return a counter-example if it is not the case. Here, in Figure 4, the following images are misclassified as an 8 and a 4, respectively, instead of a 1. As we can see, the perturbed images are quite close to the original one, and yet the pretrained classifier confuses them with another digit.

3.2. Adversarial Paradigms

As the use of AI progresses, the attacks on machine learning algorithms, or adversarial attacks, will also gain in importance.

These attacks aim at destabilizing the AI in order to produce false results, either by feeding it faulty data during training, slightly altering the input to make it unusable, or predicting the behaviour of the model in order to counter it. All of these are hardly detectable by humans, leading to inconspicuous mistakes in the case of a black-box AI with no defenses. Rosbustness, as defined previously, prevents the AI from accumulating mistakes and thus ending up useless. Coupled with explainability, it allows the user to check the results for logical faults that may come from



(a) perturbation ($1 \rightarrow 8$) (b) perturbed image ($1 \rightarrow 8$) (c) perturbation ($1 \rightarrow 4$) (d) perturbed image ($1 \rightarrow 4$)

Figure 4. Examples of misclassified adversarial images

an adversarial attack. Adversarial attacks can be separated in four categories: data poisoning attacks (tampering with training data), evasion attacks (modifying the input to make it impossible to use during inference), and model extraction or oracle attacks [21].

Evasion Adversarial attacks can be classified into two types: white-box and black-box attacks. White-box attacks are the simpler to execute as they have complete knowledge of the model parameters, which allows the attacker to use gradient information to generate adversarial examples. By back-propagating the gradient computation to the input, the attacker can create a similar image that is misclassified.

Black-box attacks, on the other hand, are more challenging as the attacker has no information about the model parameters nor access to the training stage. As a result, gradient information cannot be used to determine adversarial examples. The model may only provide class confidence scores or predicted labels, making it harder for the attacker.

We also differentiate between targeted and untargeted attacks. Untargeted attacks aim to alter the pixel intensities in a way that reduces the confidence of the original class until it is no longer the highest in the prediction vector. They do not consider which class the model should predict instead, but simply aim to fool the model.

Targeted attacks are more sophisticated and aim to modify the input towards a specified target class y' which causes the model to misinterpret the input as the desired class of the attacker.

We now desire to avoid having neural networks compromised by such attacks, and thus there has been development in methods to prove that a model is robust to perturbations and adversarial attacks, to some arbitrary degree.

3.3. Certifiable Correctness

In the context of reliable artificial intelligence in cyber-physical systems (CPS), the objective is to have collaborating devices that interact with their environment, with sensing, computation, and communication and control capabilities. As discussed earlier, we want autonomous driving cars and healthcare related AI to be consistently reliable and trustworthy. We have just seen that neural networks can sometimes be manipulated into giving the wrong output, so there is a need for a safe and robust design for such critical models. The list is qualifications desired in autonomous CPS is as follows:

- object detection and classification should be robust to change in lighting, physical attacks, and adversarial noise
- robots need to operate in unknown, uncertain, and dynamic environments

The question that arises is how do we specify and analyze a neural network?

3.3.1 Reachability Analysis

Reachability analysis allows for safety and robustness verification in neural networks. The principle is to over-approximate the sets of reachable states by flowpipes: successively proving the bounds on some maximum distance to a reference trajectory, or obstacle avoidance paradigm. To put it in simpler terms, we wish to solve the Reach-Avoid problem: to reach the target region while avoiding unsafe regions. We do this by adding noisy initial conditions and adding external disturbances over time to see if the robustness test is conclusive. An illustration of such analysis is found below, for an autonomous system.

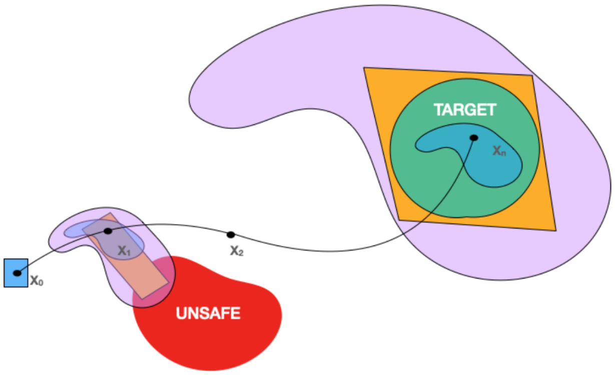


Figure 5. Illustrative example of the Reach-Avoid problem. In this example, the under-approximation of the noisy path passes by an unsafe region, so the robustness test is inconclusive. It is not determined that the model is safe within the test standards, neither is it determined that it is definitely unsafe

It is important to note that, at the time of writing, the current state-of-the-art methods that are used and developed consist of mathematically proving that the model is robust to some form of noise, for a specific sample and we cannot assert general robustness of a model to noise. We can, however, argue that an AI is very likely to be robust against noise if we can prove the robustness of the model for a large number of samples. The predictions for new, unseen samples that come from the same data distribution will then likely be robust against this specific noise as well. Proving AI systems to be robust in general and against any form of noise, whether artificial or natural, is still an open challenge.

3.3.2 Abstraction Based Verification

Abstraction based verification is a way to check the accuracy of algorithms that are used in critical settings. It helps find a balance between being thorough and manageable. This method plays a crucial role in guaranteeing responsible and trustworthy usage of algorithms in decision-making processes.

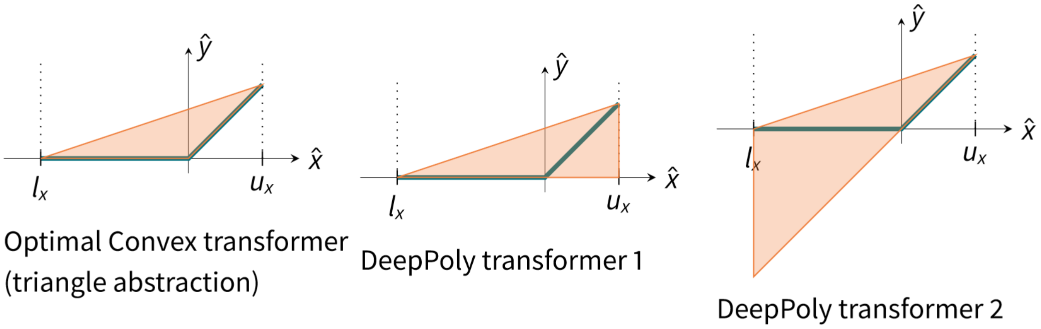
One type of abstraction used in this method is numerical abstract domains, such as boxes, zonotopes, and polyhedra. This method can also be applied to verifying neural networks, which are made up of multiple layers of interconnected nodes. The technique involves considering the impact of changes in one node while taking into account the changes in all the nodes connected to it in the previous layer. This is a vital step towards ethical AI, as it helps prevent unintended biases or unethical outcomes in the algorithm's decision-making.

Another important tool in verifying neural networks is zonotopes, which provide a compact representation of uncertainty in the input data. This information can be used to prove the robustness of the network and even simplify it by reducing the number of nodes and layers. This contributes greatly to upholding AI ethics, as it helps ensure that the decisions made by the algorithm are trustworthy and transparent, which is an essential component in building confidence in AI systems.

3.3.3 DeepPoly Convex Relaxation

In 2018, a new approach called *DeepPoly* was proposed for certifying neural networks. It combines mathematical models and custom tools to analyze the behavior of neural networks, including functions like ReLU, sigmoid, and tanh. The goal is to balance the accuracy and efficiency of the analysis.

Below, in Figure 6, is an example of abstract transformers for the ReLU activation function, the most commonly-used function in practice.



Optimal Convex transformer (triangle abstraction)

DeepPoly transformer 1

DeepPoly transformer 2

Figure 6. Abstract Transformers for the ReLU activation. It can be observed that the two DeepPoly transformers cover more area than the optimal convex transformer, but less than a whole rectangle for the first transformer, and less than the whole parallelepiped for the second. Furthermore, depending on the input interval, one is better than the other at minimizing the covered area.

Despite its benefits, the deep poly convex relaxation is generally more accurate but more costly than Zonotopes. The increased accuracy comes at the cost of increased computational resources, making the deep poly convex relaxation more suitable for small to medium-sized neural networks. However, for larger networks, the increased computational cost may make the deep poly convex relaxation less practical [19].

In conclusion, proof of robustness is a very active research field in AI, and once an AI algorithm is guaranteed to produce trustworthy results within some conditions, society will be able to rely on and use AI in critical, and previously unavailable, roles.

4. Outlook on Military Applications

The use of Artificial Intelligence in military applications has been broadened in the past decades. From surveillance to autonomous defensive systems, AI has been slowly invading every step of the decision process, from training troops or commanders, to complementing military intelligence. This gain in importance raises numerous questions concerning the safety of AI as well as the accountability of systems, users, and manufacturers.

Governments cannot afford to loose the opportunities offered by AI, as it is superior to humans in multiple critical tasks: identifying a specific battle tank in a satellite image, identifying high-value targets in a crowd using facial recognition, translating text for open-source intelligence, and text generation for use in information operations [12]. Unfortunately, it is not yet as reliable as its human counterparts, and thus requires safeguards, especially regarding the decision making processes.

4.1. Explainability for accountability

Accountability in data-driven algorithms applications has been an important problematic since the rise of such algorithms in the 1990's, and it is even more so important when human lives are at stakes, including military applications.

Nissenbaum [15] defines four barriers to accountability:

- many hands, to refer to the problem of attributing moral responsibility for outcomes caused by multiple moral actors;
- "bugs," the way software developers might shrug off responsibility by suggesting software errors are unavoidable;
- computer as scapegoat, the shifting of blame to computers as if they were moral actors;
- and ownership without liability, the free pass to the software industry to deny responsibility over their creations.

Military, through its very hierarchical organization, is already familiar with the many hands problem, but the remaining three, especially bugs, are important issues when it comes to the use of AI for military purposes.

An example of the importance of explainability in AI use for the military can be the training of troops, and more importantly of officers. For training, most Western countries (US, France, and Germany for example) use some form of simulation. Full Spectrum Command [23] is one of the simulations used by the US Army to train their commanding officers. The simulation relies heavily on an AI that simulates the behaviour of troops. It is evident that in such case, the explainability of the AI decisions is paramount to the training: after each actions and choice, the trainee must be able to understand the consequences in order to to correct their decision-making process, and the AI must thus be able to hand over the logical steps that lead to the chosen simulated answer. A black-box AI would be counter-productive and may, ironically, be the equivalent of feeding heavily biased data to an algorithm for the trainee.

4.2. XAI for the use of LAWS

More than that, AI Explainability helps to ensure that the decision-making processes of autonomous weapons systems are transparent and can be understood by human operators. This is important for verifying that the weapons are functioning as intended and making decisions that are in line with ethical and legal guidelines. In the context of autonomous warfare, human-in-the-loop verification is particularly important as decisions made by lethal autonomous weapons have the potential to result in significant harm to human lives. Without explainability, it may be difficult or impossible for human operators to understand how the weapons arrived at a particular decision and determine whether that decision was appropriate. Explainability also helps to build trust in autonomous weapons systems and increase accountability for their actions. By providing transparency into the decision-making processes, stakeholders can better understand the limitations and capabilities of the weapons, which can help to prevent unintended consequences and mitigate the risk of harm.

However, explainability also poses a potential risk of tricking military users of lethal autonomous weapons systems (LAWS) into a false trust in the AI. This is known as automation bias, which refers to the tendency for people to place too much trust in automated decision-making systems and to overlook their limitations and potential errors, as mentioned in Section 2.2.

Therefore, while explainability is an important factor in increasing transparency and accountability in LAWS, it is also crucial to ensure that military users are adequately trained and equipped to understand and interpret the information provided by these systems in order to mitigate the risk of automation bias.

4.3. Certifiability in LAWS

Certifiability refers to the process of ensuring that an AI system meets certain standards, such as reliability, safety, and accountability. When it comes to LAWS, certifiability is important for a number of reasons. First and foremost, humans are being pushed back both physically and cognitively as machines make life-or-death decisions, and certifiability helps to ensure that LAWS are not used to cause harm to innocent people. With certifiable AI systems, there is a clear understanding of how the system will make decisions and what factors it takes into account. This can help to minimize the risk of unintended harm, such as collateral damage or the use of force in non-combat situations.

Additionally, certifiability provides a level of accountability, allowing individuals and organizations to take responsibility for the actions of LAWS. This is particularly important in the context of LAWS, as there is often a lack of human oversight and decision-making. By having a certifiable AI system in place, organizations can demonstrate that they are taking their responsibilities seriously and are committed to using AI ethically and responsibly.

Certifiable robustness of AI, particularly in the context of neural networks, could be useful for the case of lethal autonomous weapons systems (LAWS) as it provides a means to assess the robustness and reliability of the AI's decisions. This involves mathematically proving that the AI's decision-making processes are resistant to certain types of adversarial attacks and that the AI's decisions are consistent with specified safety and security constraints.

The advantage of certifiable robustness is that it provides a more systematic and rigorous way to evaluate the reliability of the AI's decisions compared to traditional testing methods. It can also help to increase transparency and accountability in LAWS, as it provides a clear and verifiable means to assess the AI's decision-making processes.

However, certifiable robustness also has limitations. For example, it may not be possible to mathematically prove the robustness of the AI's decisions in all possible scenarios, as explained in Section 3. Additionally, certifiable robustness only evaluates the AI's decisions under specific conditions, and it may not be possible to fully anticipate all potential real-world scenarios.

Therefore, while certifiable robustness can be a useful tool in ensuring the robustness and reliability of the AI's decisions in LAWS, it is important to consider its limitations and to use it in combination with other methods, such as human-in-the-loop verification, to ensure that the AI's decisions are ethical and lawful.

5. Conclusion

In conclusion, AI explainability is a critical component for ensuring ethical considerations in the development and deployment of AI technology. With AI increasingly permeating various aspects of society, it is imperative to ensure that its actions and decisions are transparent, interpretable, and accountable. This can be achieved through explainable AI, which aims at providing a clear understanding of how AI systems make decisions, and why they behave the way they do.

By enabling stakeholders to understand and scrutinize AI's decision-making processes, explainability can play a crucial role in promoting accountability, transparency, and trust in AI systems. Moreover, it can help prevent unintended consequences and ensure that AI aligns with ethical and moral values. In light of these benefits, it is important that the field of AI research and development continues to prioritize explainability, so that AI can be harnessed for the greater good of society.

In addition to ensuring that AI algorithms are explainable, we have surveyed some of the methods that have been developed to assess their robustness against noise and adversarial attacks. This property is key for AI models in critical applications as it ensures their reliable performance even in challenging conditions. By demonstrating robustness, AI systems can gain the trust of society and be granted more control and decision-making responsibilities.

References

[1] Maruan Al-Shedivat, Avinava Dubey, and Eric Xing. Contextual explanation networks. *The Journal of Machine Learning Research*, 21(1):7950–7993, 2020. 6

[2] Aws Albarghouthi et al. Introduction to neural network verification. *Foundations and Trends® in Programming Languages*, 7(1–2):1–157, 2021. 8

[3] David Alvarez-Melis and Tommi S Jaakkola. A causal framework for explaining the predictions of black-box sequence-to-sequence models. *arXiv preprint arXiv:1707.01943*, 2017. 6

[4] Daniel W Apley and Jingyu Zhu. Visualizing the effects of predictor variables in black box supervised learning models. *arXiv preprint arXiv:1612.08468*, 2016. 3

[5] Valérie Beaudouin, Isabelle Bloch, David Bounie, Stéphan Clémençon, Florence d'Alché Buc, James Eagan, Winston Maxwell, Pavlo Mozharovskyi, and Jayneel Parekh. Flexible and context-specific ai explainability: a multidisciplinary approach. *arXiv preprint arXiv:2003.07703*, 2020. 2, 6

[6] Jerome H Friedman. Greedy function approximation: a gradient boosting machine. *Annals of statistics*, pages 1189–1232, 2001. 2, 6

[7] David Gunning and David Aha. Darpa's explainable artificial intelligence (xai) program. *AI magazine*, 40(2):44–58, 2019. 2

[8] Trevor J Hastie. Generalized additive models. In *Statistical models in S*, pages 249–307. Routledge, 2017. 6

[9] Guang-He Lee, David Alvarez-Melis, and Tommi S Jaakkola. Towards robust, locally linear deep networks. *arXiv preprint arXiv:1907.03207*, 2019. 6

[10] Yin Lou, Rich Caruana, Johannes Gehrke, and Giles Hooker. Accurate intelligible models with pairwise interactions. In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 623–631, 2013. 6

[11] Scott M Lundberg and Su-In Lee. A unified approach to interpreting model predictions. *Advances in neural information processing systems*, 30, 2017. 4

[12] Paul Maxwell. Artificial intelligence is the future of warfare (just not in the way you think), 2020. 11

[13] Christoph Molnar. *Interpretable Machine Learning*. 2 edition, 2022. 2, 3

[14] John Ashworth Nelder and Robert WM Wedderburn. Generalized linear models. *Journal of the Royal Statistical Society: Series A (General)*, 135(3):370–384, 1972. 6

[15] Helen Nissenbaum. Accountability in a computerized society. *Science and engineering ethics*, 2:25–42, 1996. 11

[16] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. "why should i trust you?" explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 1135–1144, 2016. 4

[17] Max Schemmer, Niklas Kühl, Carina Benz, and Gerhard Satzger. On the influence of explainable ai on automation bias. *arXiv preprint arXiv:2204.08859*, 2022. 7

[18] Edward H Shortliffe and Bruce G Buchanan. A model of inexact reasoning in medicine. *Mathematical biosciences*, 23(3-4):351–379, 1975. 1

[19] Gagandeep Singh, Timon Gehr, Markus Püschel, and Martin Vechev. An abstract domain for certifying neural networks. *Proceedings of the ACM on Programming Languages*, 3(POPL):1–30, 2019. 11

[20] Jia Slack, Hilgard and Singh Lakkaraju. Fooling lime and shap: Adversarial attacks on post hoc explanation methods. *AIES '20: Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*, pages 180–186, 2020. 5

[21] Elham Tabassi, Kevin J. Burns, Michael Hadjimichael, Andres D. Molina-Markham, and Julian T. Sexton. A taxonomy and terminology of adversarial machine learning. 9

[22] Jasper van der Waa, Elisabeth Nieuwburg, Anita Cremers, and Mark Neerincx. Evaluating xai: A comparison of rule-based and example-based explanations. *Artificial Intelligence*, 291:103404, 2021. 7

[23] Michael van Lent, William Fischer, and Michael Mancuso. An explainable artificial intelligence system for small-unit tactical behavior. In *American Association for Artificial Intelligence Emerging Applications*, pages 900–907, 2004. 12

[24] Giulia Vilone and Luca Longo. Notions of explainability and evaluation approaches for explainable artificial intelligence. *Information Fusion*, 76:89–106, 2021. 7

[25] Giorgio Visani, Enrico Bagli, and Federico Chesani. Optilime: Optimized lime explanations for diagnostic computer algorithms. *arXiv preprint arXiv:2006.05714*, 2020. 5

[26] Quanshi Zhang, Ying Nian Wu, and Song-Chun Zhu. Interpretable convolutional neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8827–8836, 2018. 6