Formal Interpretability with Merlin-Arthur Classifiers

A Preprint

Stephan Wäldchen
Zuse Institute Berlin
waeldchen@zib.de

Kartikey Sharma
Zuse Institute Berlin
kartikey.sharma@zib.de

Max Zimmer
Zuse Institute Berlin
zimmer@zib.de

Berkant Turan
Zuse Institute Berlin
turan@zib.de

Sebastian Pokutta
Zuse Institute Berlin
pokutta@zib.de

Abstract

We propose a new type of multi-agent interactive classifier that provides provable interpretability guarantees even for complex agents such as neural networks. These guarantees consist of bounds on the mutual information of the features selected by this classifier. Our results are inspired by the Merlin-Arthur protocol from Interactive Proof Systems and express these bounds in terms of measurable metrics such as soundness and completeness. Compared to existing interactive setups we do not rely on optimal agents or on the assumption that features are distributed independently. Instead, we use the relative strength of the agents as well as the new concept of Asymmetric Feature Correlation which captures the precise kind of correlations that make interpretability guarantees difficult. We test our results through numerical experiments on two small-scale datasets where high mutual information can be verified explicitly.

1 Introduction

Safe deployment of Neural Network (NN) based AI systems in high-stakes applications requires that their reasoning be subject to human scrutiny. The field of Explainable AI (XAI) has thus put forth a number of interpretability approaches, among them saliency maps (Mohseni et al., 2021), mechanistic interpretability (Olah et al., 2018) and self-explaining networks (Alvarez Melis & Jaakkola, 2018). These have had some successes, such as detecting biases in established datasets (Lapuschkin et al., 2019). However, these approaches are motivated primarily by heuristics and come without any theoretical guarantees. Thus, their success cannot be verified. It has also been demonstrated for numerous XAI-methods that they can be manipulated by a clever design of the NNs (Slack et al., 2021, 2020; Anders et al., 2020; Dimanov et al., 2020). On the other hand, formal approaches to interpretability run into complexity barriers when applied to NNs and require an exponential amount of time to guarantee useful properties (Macdonald et al., 2020; Ignatiev et al., 2019). This makes any "right to explanation," as codified in the EU's *GDPR* (Goodman & Flaxman, 2017), unenforceable.

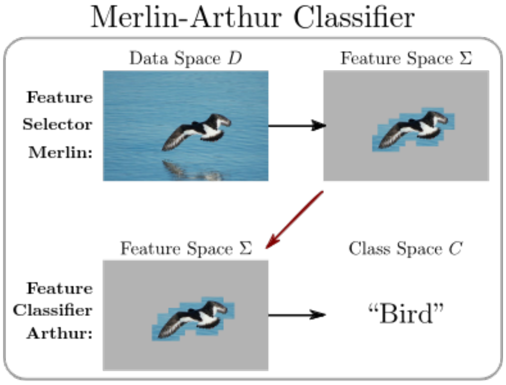

Figure 1: The Merlin-Arthur classifier consists of two interactive agents that communicate over an exchanged feature. This feature serves as an interpretation of the classification.

In this work, we design a classifier that guarantees feature-based interpretability under reasonable assumptions. For this, we connect classification to the Merlin-Arthur protocol (Arora & Barak, 2009) from Interactive Proof Systems (IPS), see Figure 1. Our setup consists of a classifer called Arthur (verifier) and 2 feature selectors referred to Merlin and Morgana (provers). Merlin and Morgana choose features from the input image and send them to Arthur. Merlin aims to send features that cause Arthur to correctly classify the underlying data point. Morgana instead selects features to convince Arthur of the wrong class. Arthur does not know who sent the feature and is allowed to say "Don't know!" if he cannot discern the class. In this context, we can then translate the concepts of *completeness* and *soundness* from IPS to our setting. Completeness describes the probability that Arthur classifies correctly based on features from Merlin. Soundness is the probability that Arthur does not get fooled by Morgana, thus either giving the correct class or answering "Don't know!". These two quantities can be measured on a test dataset and are used to lower bound the information contained in features selected by Merlin.

1.1 Related Work

Formal approaches to interpretability, such as mutual information (Chen et al., 2018) or Shapley values Frye et al. (2020), generally make use of partial inputs to the classifier. These partial inputs are realised by considering distributions over inputs conditioned on the given information. However, modelling these distributions is difficult for non-synthetic data. This has been pursued practically by training a generative model as in Chattopadhyay et al. (2022). But as of yet there is no approach that provides a bound on the quality of these models. We discuss these approaches and their challenges in greater detail in Appendix A.2.

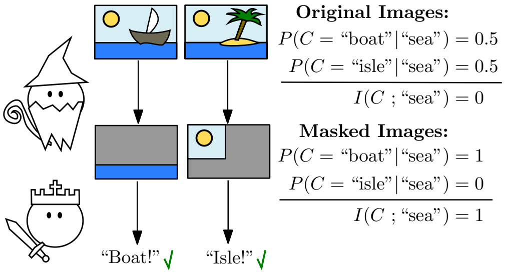
Original Images:
$P(C = \text{“boat”} | \text{“sea”}) = 0.5$
$P(C = \text{“isle”} | \text{“sea”}) = 0.5$
$I(C\ ; \text{“sea”}) = 0$
Masked Images:
$P(C = \text{“boat”} | \text{“sea”}) = 1$
$P(C = \text{“isle”} | \text{“sea”}) = 0$
$I(C\ ; \text{“sea”}) = 1$
“Boat!” ✓
“Isle!” ✓

Figure 2: Illustration of "cheating" behaviour. In the original dataset, the features "sea" and "sky" appear equally in both classes "boat" and "island". In the new set of images that Merlin creates by masking features of the original image, the "sea" feature is visible only in the images labelled "boat" and the "sky" feature is visible only in the images labelled "island". Thus, these features now strongly indicate the class of the image. This allows Merlin to communicate the correct class with uninformative features — in contrast to our concept of an interpretable classifier.

Interactive classification in form of a prover-verifier setting has emerged as a way to design inherently interpretable classifiers (Lei et al., 2016; Bastings et al., 2019). In this setup, the feature selector chooses a feature from a data point and presents it to the classifier who decides the class, see Figure 2. The classification accuracy is meant to guarantee the informativeness of the exchanged features. However, it was noted by Yu et al. that the selector and the classifier can cooperate to achieve high accuracy while communicating over uninformative features, see Figure 2 for an illustration of this "cheating". Thus, one cannot hope to bound the information content of features via accuracy alone. Chang et al. include an adversarial selector to prevent the cheating. The reasoning is that any "cheating" strategy can be exploited by the adversary to fool the classifier into stating the wrong class, see Figure 3 for an illustration. Anil et al. investigate scenarios in which the three-player setup converges to an equilibrium of perfect completeness and soundness. However, this work assumes that a perfect strategy exists and can be reached through training. For many classification problems, such as the ones we explore in our experimental section, no strategies are perfectly sound and complete when the size of the certificate is limited.

Alternative adversarial setups have been proposed in Yu et al. (2019) and Irving et al. (2018), but no information bounds were formulated for them. We discuss these ideas and their challenges in detail in Appendix A.4.

An additional theoretical focus has been the learnability of interpretations Goldwasser et al. (2021); Yadav et al. (2022). In this work, we do not focus on the question of learnability. We instead propose to evaluate soundness and completeness directly on the test dataset, as state-of-the-art models are to complex to guarantee generalisation from a realistic number of training samples.

The closest work is the framework proposed by Chang et al.. The setup is basically the same, except that for choices that matter only for numerical implementation, see Appendix A.3 for an in-depth discussion. The authors show that the best strategy for the provers is to select features with high mutual information wrt to the class. However, these results have three restriction that we resolve in this work: **(i)** The features are assumed to be independently distributed. This is an unrealistic assumption for most datasets where features are generally correlated. In this regime simply modelling the data distribution directly is possible. **(ii)** The provers can only select one feature at a time without context. This strategy is unlikely to yield useful rationalisations for most types of data where the importance of a feature strongly depends on the features surrounding it, like images and text. The authors do not impose this restriction for their numerical investigation. **(iii)** The result is non-quantitative. Since we cannot expect the agents to play optimally on

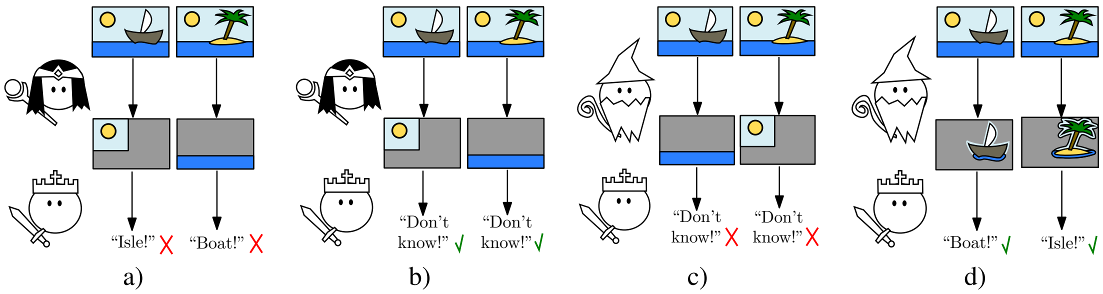

Figure 3: Strategy evolution with Morgana. a) Due to the “cheating” strategy from Figure 2, Arthur expects the “sea” feature for boats and the “sky” for islands. Morgana can exploit this and send the “sky” feature to trick Arthur into classifying a “boat” image as an “island” (and vice versa with “sea”). b) To not be fooled into the wrong class when represented with an ambiguous feature, Arthur refrains from giving a concrete classification. c) Since Arthur does not know who sends the features, he now cannot leverage the uninformative features sent by Merlin. d) Merlin adapts his strategy to only send unambiguous features that cannot be used by Morgana to fool Arthur.

complicated data we need measures on how well they play and how that relates to the mutual information of the features.

1.2 Contribution

We provide, what we believe, the first quantitative lower bound on the information content of the features in an interpretive setup without the need to trust a model of the data distribution. Additionally, we improve existing analyses in the following ways:

1. We do not assume our agents to be optimal. In Theorem 2.8 Merlin is allowed to have an arbitrary strategy and in Theorem 2.12 all three players can play suboptimally. We rather rely on the relative strength of Merlin and Morgana for our bound. We also allow our provers to select the features with the context of the full datapoint.
2. We do not make the assumption that features are independently distributed. Instead, we introduce the notion of Asymmetric Feature Correlation (AFC) that captures which correlations make an information bound difficult. In Theorem 2.8 we circumvent the issue by reducing the dataset, and in Theorem 2.12 we incorporate the AFC explicitly. In Section 4 we discuss why the AFC also matters for other interactive settings.

2 Theoretical Framework

In this section we develop the theoretical framework for the Merlin-Arthur classifier. A key aspect is the notion of a feature. What reasonably constitutes a feature strongly depends on the context and prior work often considered subsets of the input as features. W.l.o.g we will stay with this convention for ease of notation. But nothing in our framework relies on these specifics and our theoretical results can be extended to more abstract queries Chen et al. (2018); Ribeiro et al. (2018).

Given a vector $\mathbf{x}$ of dimension d, we use $\mathbf{x}_S$ to represent a vector made of the components of $\mathbf{x}$ indexed by the set $S \subseteq \{1, \dots, d\}$.

Definition 2.1. *Given a dataset $D \subset [0,1]^d$, we define the corresponding* partial *dataset D_p as*

$$D_p = \bigcup_{\mathbf{x} \in D} \bigcup_{S \subset [d]} \mathbf{x}_S.$$

The set D is possibly infinite, e.g. the set of all images of hand-written digits. $\mathcal{D}$ is a distribution on this set. The finite training and test sets, e.g. MNIST, for our algorithms are assumed to be faithful samples from this distribution.

Every vector $\mathbf{x} \in D \subset [0,1]^d$ can be uniquely represented as a set $\{(1, x_1), (2, x_2), \dots, (d, x_d)\}$. A partial vector $\mathbf{z} \in D_p$ can then be a subset of $\mathbf{x}$. Thus, $\mathbf{z} \subseteq \mathbf{x}$ indicates that $\mathbf{x}$ contains the feature $\mathbf{z}$. The set D_p might be further restricted to include only connected sets (for image or text data) or only sets of a certain size as in our numerical investigation.

We now define a data space. In our theoretical investigation we restrict ourselves to two classes and assume the existence of unique class for every data point. These are restrictions that we hope to relax in further research.

Definition 2.2 (Two-class Data Space). *We consider the tuple* $\mathfrak{D} = (D, \mathcal{D}, c)$ *a* two-class data space *consisting of the dataset* $D \subseteq [0,1]^d$, *a probability distribution* $\mathcal{D}$ *along with the ground truth class map* $c : D \to \{-1, 1\}$. *The* class imbalance B *of a two-class data space is* $\max_{l \in \{-1,1\}} \mathbb{P}_{\mathbf{x} \sim \mathcal{D}}[c(\mathbf{x}) = l] / \mathbb{P}_{\mathbf{x} \sim \mathcal{D}}[c(\mathbf{x}) = -l]$.

Note that our assumption on the ground truth class map requires there to be a unique class for any datapoint. This may not necessarily be true for many data sets that are described by a common distribution of data and label.

Remark 2.3. *We will oftentimes make use of restrictions of the set* D *and measure* $\mathcal{D}$ *to a certain class, e.g.,* $D_l = \{\mathbf{x} \in D \,|\, c(\mathbf{x}) = l\}$ *and* $\mathcal{D}_l = \mathcal{D}|_{D_l}$.

We now introduce the notion of a feature selector (as prover) and feature classifier (as verifier).

Definition 2.4 (Feature Selector). *For a given dataset* D*, we define a* feature selector *as a map* $M : D \to D_p$ *such that for all* $\mathbf{x} \in D$ *we have* $M(\mathbf{x}) \subseteq \mathbf{x}$. *This means that for every data point* $\mathbf{x} \in D$ *the feature selector* M *chooses a feature that is present in* $\mathbf{x}$. *We call* $\mathcal{M}(D)$ *the space of all feature selectors for a dataset* D.

Definition 2.5 (Feature Classifier). *We define a* feature classifier *for a dataset* D *as a function* $A : D_p \to \{-1, 0, 1\}$. *Here,* 0 *corresponds to the situation where the classifier is unable to identify a correct class. We call the space of all feature classifiers* $\mathcal{A}$.

2.1 Mutual Information, Entropy and Precision

We consider a feature to carry class information if it has high mutual information with the class. For a given feature $\mathbf{z} \in D_p$ and a data point $\mathbf{y} \sim \mathcal{D}$ the mutual information is

$$I_{\mathbf{y} \sim \mathcal{D}}(c(\mathbf{y}); \mathbf{z} \subseteq \mathbf{y}) := H_{\mathbf{y} \sim \mathcal{D}}(c(\mathbf{y})) - H_{\mathbf{y} \sim \mathcal{D}}(c(\mathbf{y}) \,|\, \mathbf{z} \subseteq \mathbf{y}).$$

When the conditional entropy $H_{\mathbf{y} \sim \mathcal{D}}(c(\mathbf{y}) \,|\, \mathbf{z} \subseteq \mathbf{y})$ goes to zero, the mutual information becomes maximal and reaches the pure class entropy $H_{\mathbf{y} \sim \mathcal{D}}(c(\mathbf{y}))$ which measures how uncertain we are about the class a priori. A closely related concept is *precision*. Given a data point $\mathbf{x}$ with feature $\mathbf{z}$, precision is defined as $\Pr(\mathbf{z}) := \mathbb{P}_{\mathbf{y} \sim \mathcal{D}}[c(\mathbf{y}) = c(\mathbf{x}) \,|\, \mathbf{z} \subseteq \mathbf{y}]$ and was introduced in the context of interpretability by Ribeiro et al. (2018) and Narodytska et al. (2019). We extend this definition to a feature selector.

Definition 2.6 (Average Precision). *For a given two-class data space* $\mathfrak{D}$ *and a feature selector* $M \in \mathcal{M}(D)$, *we define the* average precision *of* M *with respect to* $\mathcal{D}$ *as*

$$Q_{\mathcal{D}}(M) := \mathbb{E}_{\mathbf{x} \sim \mathcal{D}}[\mathbb{P}_{\mathbf{y} \sim \mathcal{D}}[c(\mathbf{y}) = c(\mathbf{x}) \,|\, M(\mathbf{x}) \subseteq \mathbf{y}]].$$

The average precision $Q_{\mathcal{D}}(M)$ can be used to bound the *average* conditional entropy of Merlin's features, defined as

$$H_{\mathbf{x},\mathbf{y} \sim \mathcal{D}}(c(\mathbf{y}) \,|\, M(\mathbf{x}) \subseteq \mathbf{y}) := \mathbb{E}_{\mathbf{x} \sim \mathcal{D}}[H_{\mathbf{y} \sim \mathcal{D}}(c(\mathbf{y}) \,|\, M(\mathbf{x}) \subseteq \mathbf{y})], \tag{1}$$

and accordingly the average mutual information. For greater detail, see Appendix B. We can lower-bound the mutual information as follows,

(2)

When the precision goes to 1, the binary entropy $H_b(p)$ goes to 0 and the mutual information becomes maximal. Our results are easier to state in terms of $Q_{\mathcal{D}}(M)$, because of the infinite slope of the binary entropy.

We can connect $Q_{\mathcal{D}}(M)$ back to the precision of any feature selected by M in the following way.

Lemma 2.7. *Given* $\mathfrak{D} = (D, \mathcal{D}, c)$ *, a feature selector* $M \in \mathcal{M}(D)$ *and* $\delta \in [0,1]$. *Let* $\mathbf{x} \sim \mathcal{D}$, *then with probability* $1 - \delta^{-1}(1 - Q_{\mathcal{D}}(M))$, $M(\mathbf{x})$ *is a feature s.t.*

$$\mathbb{P}_{\mathbf{y} \sim \mathcal{D}}[c(\mathbf{y}) = c(\mathbf{x}) \,|\, M(\mathbf{x}) \subseteq \mathbf{y}] \geq 1 - \delta.$$

The proof follows directly from Markov's inequality, see Appendix B. We will now introduce a new framework that will allow us to prove bounds on $Q_{\mathcal{D}}(M)$ and thus assure feature quality. For I and H, we will leave the dependence on the distribution implicit when it is clear from context.

2.2 Merlin-Arthur Classification

For a feature classifier A (Arthur) and two feature selectors M (Merlin) and $\widehat{M}$ (Morgana) we define

$$E_{M,\widehat{M},A} := \left\{ x \in D \,\middle|\, A(M(\mathbf{x})) \neq c(\mathbf{x}) \;\vee A\left(\widehat{M}(\mathbf{x})\right) = -c(\mathbf{x}) \right\} \tag{3}$$

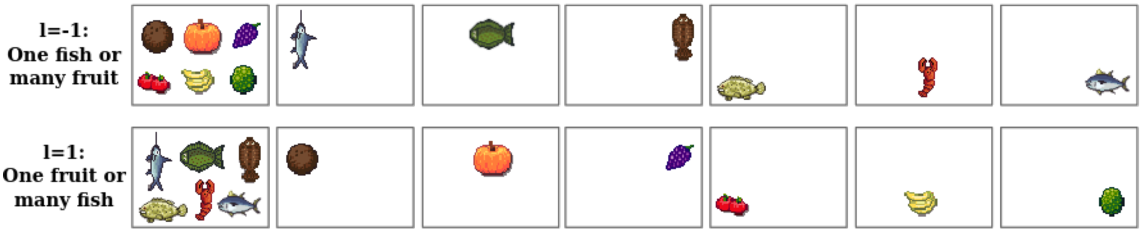

Figure 4: Example of a dataset an AFC $\kappa = 6$. The "fruit" features are concentrated in one image for class $l = -1$ but spread out over six images for $l = 1$ (vice versa for the "fish" features). Each individual feature is not indicative of the class as it appears exactly once in each class. Nevertheless, Arthur and Merlin can exchange "fruits" to indicate "$l = 1$" and "fish" for "$l = -1$". The images where this strategy fails or can be exploited by Morgana are the two images on the left. Applying Theorem 2.8, we get $\epsilon_M = \frac{1}{7}$ and the set D' corresponds to all images with a single feature. Restricted to D', the features determine the class completely.

as the set of data points for which Merlin fails to convince Arthur of the correct class or Morgana is able to trick him into returning the wrong class, in short, the set of points where Arthur fails. We can now state the following theorem connecting the competitive game between Arthur, Merlin and Morgana to the class conditional entropy.

Theorem 2.8. *[Min-Max] Let $M \in \mathcal{M}(D)$ be a feature selector and let*

$$\epsilon_M = \min_{A\in\mathcal{A}} \max_{\widehat{M}\in\mathcal{M}} \mathbb{P}_{\mathbf{x}\sim\mathcal{D}}\left[\mathbf{x}\in E_{M,\widehat{M},A}\right].$$

Then a set $D' \subset D$ with $\mathbb{P}_{\mathbf{x}\sim\mathcal{D}}[\mathbf{x}\in D'] \geq 1-\epsilon_M$ exists such that for $\mathcal{D}' = \mathcal{D}|_{D'}$ we have

$$Q_{\mathcal{D}'}(M) = 1, \quad \textit{thus} \quad H_{\mathbf{x},\mathbf{y}\sim\mathcal{D}'}(c(\mathbf{y}) \mid \mathbf{y}\in M(\mathbf{x})) = 0.$$

The proof is in Appendix B. This theorem states that if Merlin's strategy allows Arthur to classify almost perfectly, i.e., small ϵ_M, then there exists a set that covers almost the entire original dataset and on which the class entropy conditioned on the selected features is zero. Note that these guarantees are for the set D' and not the original set D. A bound for the set D, such as $Q_{\mathcal{D}}(M) \geq 1-\epsilon_M$, is complicated by a factor we call *asymmetric feature correlation (AFC).*

2.3 Asymmetric Feature Correlation:

AFC describes a possible quirk of datasets, where a set of features is strongly concentrated in a few data points in one class and spread out over almost all data points in another. We give an illustrative example in Figure 4. If a data space $\mathfrak{D}$ has a large AFC κ, Merlin can use features that individually appear equally in both classes (low precision) to indicate the class where they are spread over almost all points. Morgana can only fool Arthur in the other class where these features are highly concentrated, thus only in a few data points. This ensures a small ϵ_M even with uninformative features.

For a set of features $F \subset D_p$ we define

$$F^* := \{\mathbf{x}\in D \mid \exists\, \mathbf{z}\in F : \mathbf{z}\subseteq\mathbf{x}\},$$

the set of all datapoints that contain a feature from F.

Definition 2.9 (Asymmetric feature correlation)**.** *Let $(D, \mathcal{D}, c)$ be a two-class data space, then the asymmetric feature correlation κ is defined as*

$$\kappa = \max_{l\in\{-1,1\}} \max_{F\subset D_p} \mathbb{E}_{\mathbf{y}\sim\mathcal{D}_l|_{F^*}}\left[\max_{\substack{\mathbf{z}\in F\\ s.t.\ \mathbf{y}\in\mathbf{z}}} \kappa_l(\mathbf{z},F)\right]$$

with

$$\kappa_l(\mathbf{z},F) = \frac{\mathbb{P}_{\mathbf{x}\sim\mathcal{D}_{-l}}[\mathbf{z}\subseteq\mathbf{x}\,|\,\mathbf{x}\in F^*]}{\mathbb{P}_{\mathbf{x}\sim\mathcal{D}_{l}}[\mathbf{z}\subseteq\mathbf{x}\,|\,\mathbf{x}\in F^*]}.$$

We derive this expression in more detail in Appendix B.3, but give an intuition here. The probability $\mathbb{P}_{\mathbf{x}\sim\mathcal{D}_l}[\mathbf{z}\subseteq\mathbf{x}\,|\,\mathbf{x}\in F^*]$ for $\mathbf{z}\in F$ is a measure of how correlated the features are. If all features appear in the same datapoints this quantity takes a maximal value of 1 for each $\mathbf{z}$. If no features share the same datapoint the value is minimally $\frac{1}{|F|}$ for the average $\mathbf{z}$. The $\kappa_l(\mathbf{z},F)$ thus measures the difference in correlation between the two classes. In the example in Figure 4 the worst-case F for $l = -1$ correspond to