(Extended Abstract) How to model classifier and its explanation in modal logic

Xinghan Liu[1] and Emiliano Lorini[2]

[1] ANITI, Toulouse University, France Xinghan.Liu@univ-toulouse.fr
[2] IRIT-CNRS Toulouse University, France Emiliano.Lorini@irit.fr

1 Motivation

A classifier is a function Nowadays, systems resulting from machine learning (ML) achieve unprecedented successes in use. But their decisions/predictions are notoriously hard to explain. Recently, there has been a renewed interest for the notion of explanation in the context of classifier systems [2, 13, 5]. Artificial feed-forward neural networks are special kinds of classifier systems aimed at learning or, at least approximating, the function mapping instances of the input data to their corresponding outputs. Classifiers can be seen as "black boxes" computing given (Boolean) functions in the context of a decision or prediction task.

Explaining why the system has classified a given instance in a certain way and identifying the set of features that is necessary (minimally) sufficient for the classification is crucial for making the system intelligible and for finding biases in the classification process. A variety of notions of this "local explanation" have been discussed in the area of explainable AI (XAI) including abductive, contrastive and counterfactual explanation [3, 1, 10, 5, 11, 9].

A binary classifier is a partition on S5 model Traditionally a Boolean function is expressed by propositional logic, particularly in DNF. Why bother mentioning modal logic? We see a modal logic for classifier both natural and fruitful. Natural, because a classifier can be seen as a partition on a complete S5 model, where each possible world stands for a possible input. Moreover, as Quine [12] found, a Boolean function can be uniquely expressed by its prime implicants. And it has a meaning of modality: the classification is necessary given the prime implicant. Fruitful, because we can investigate classifiers with resources in modal logic, e.g. dynamic and epistemic extensions we will give. These considerations motivate us to come up with a modal logic for binary input classifier. This paper is the extended abstract of the full paper [8].

The language $\mathcal{L}(Atm)$ To our end we introduce a modal language $\mathcal{L}(Atm)$:

$$\varphi ::= p \mid \neg\varphi \mid \varphi_1 \wedge \varphi_2 \mid [X]\varphi$$

where p ranges over Atm the set of atomic formulas and X ranges over $\wp(Atm)$. $[X]$ and its dual $\langle X\rangle =_{def} \neg[X]\neg\varphi$ have a *ceteris paribus* nature to represent that a formula is necessarily or possibly true when valuations in X being equal, and are first introduced in [4]. Atm consists of (finite) variables that occur in a given classifier. To encode decision values, Atm has a set of finite atoms formed like $\mathtt{t}(x), \mathtt{t}(y), \ldots$. We call them *decision atoms* and note $\mathit{Dec} = \{\mathtt{t}(x) : x \in \mathit{Val}\}$, where Val stands for the set of decision values of the decision function f defined below. Atoms occurring in inputs thus belong to $(\mathit{Atm} \setminus \mathit{Dec})$. Finally, let $\mathit{Atm}(\varphi)$ denote atoms that occur in φ.

2 Model

Classifier Model A classifier model (CM) is simply a pair $C = (S, f)$ where $S = 2^{\mathit{Atm} \setminus \mathit{Dec}}$ and $f : S \to \mathit{Val}$ is a function, where Val denotes a set of decision values. For any $s \in S$, we have (C, s) as a pointed model with the following satisfaction relation:

- $(C, s) \models p$ for $p \in (\mathit{Atm} \setminus \mathit{Dec})$, if $p \in s$
- $(C, s) \models \mathtt{t}(x)$ for $\mathtt{t}(x) \in \mathit{Dec}$, if $f(s) = x$
- $(C, s) \models \neg\varphi$, if it is not the case that $(C, s) \models \varphi$
- $(C, s) \models \varphi \wedge \psi$, if $(C, s) \models \varphi$ and $(C, s) \models \psi$
- $(C, s) \models [X]\varphi$, if $\forall s' \in S$, if $(s \cap X) = (s' \cap X)$ then $(C, s') \models \varphi$

We may abbreviate the notion as $s \models \varphi$ when the context is clear. The class of all such models is noted **CM**. Satisfiability and validity are defined as usual.

The very virtue of a classifier model is its faithfulness to the classifier: only the information of f is needed for its construction. For any Boolean formula, its truth value in s is determined by the valuation of all atoms with respect to either s or $f(s)$. The only exception is $[X]\varphi$. Its ceteris paribus nature can be read as changing valuations of atoms outside of X and checking whether φ holds in all resulting states. But surely that depends on again some state s' and $f(s')$.

Besides the formulas above, in our language we can define an operator of counterfactual conditional $\Rightarrow$ and its satisfaction relation. So $\varphi \Rightarrow \psi$ has to be read as "if that φ were the case, other things being equal, then it would that ψ", which is nothing but an abbreviation of the following formula:

$$\Big(\bigwedge_{Y \subseteq (\mathit{Atm} \setminus \mathit{Dec}) : |Y| = k} \langle Y \rangle \varphi \wedge \bigwedge_{Y \subseteq (\mathit{Atm} \setminus \mathit{Dec}) : k < |Y|} [Y] \neg\varphi \Big) \to \bigwedge_{Y \subseteq (\mathit{Atm} \setminus \mathit{Dec}) : |Y| = k} [Y](\varphi \to \psi).$$

Messy as it may seem, it represents some Lewisian counterfactual where similarity is determined by the number of common valuations of atoms in the scope $(\mathit{Atm} \setminus \mathit{Dec})$. One can show it satisfies all semantic conditions of Lewis' VC logic [7]. We will see $\Rightarrow$ can be used to express contrastive explanation (CXp).

Notation Obviously we intend to interpret the function f in (S, f) as a classifier. So some terminologies of classifier are needed for that purpose. For any $s \in S$, call $\widehat{s} =_{def} \bigwedge_{p \in s} \wedge \bigwedge_{p \notin s} \neg p$ an *instance*. Instances are special *terms*, a.k.a. *properties*, where a term $\lambda =_{def} \bigwedge_{p \in X} p \wedge \bigwedge_{p \in Y} \neg p$ for some $X, Y \subseteq (\mathit{Atm} \setminus \mathit{Dec})$ such that

$X \cap Y \neq \emptyset$; and by $\overline{\lambda}$ we mean $\bigwedge_{p \in X} \neg p \wedge \bigwedge_{p \in Y} p$. By convention $\top$ is a term of zero conjuncts. We say $\lambda \subseteq \lambda'$ when λ' (propositional logically) entails λ, and $\lambda \subset \lambda'$ when λ' entails λ and λ does not entail λ'. Additionally, to define bias we may distinguish the set of protected features PF, like gender and race, and the set of non-protected features NF, s.t. $\mathsf{PF} \cup \mathsf{NF} = (\mathit{Atm} \setminus \mathit{Dec})$ and $\mathsf{PF} \cap \mathsf{NF} = \emptyset$.

Model classifier explanation We show how to express some existing notions of classifier explanation in e.g. [2, 5, 6, 13] in our classifier model and language. Let us exemplify them by the following toy model.

Example 1. Given $\mathcal{L}(\mathit{Atm})$ with $\mathsf{PF} = \{m, c\}, \mathsf{NF} = \{e, o\}, \mathit{Dec} = \{\mathsf{t}(x), \mathsf{t}(y)\}$, where each stands for *m*ale, in *c*ity center, *e*mployed, *o*wning property, to accept and to reject resp., let f be a classifier of loan s.t. $f = (\mathsf{t}(x) \leftrightarrow ((e \wedge o) \vee (m \wedge c))$. Suppose $s = \{c, e\}$ standing for the state of Alice, and Alice is rejected by the classifier. Now Alice wants to know 1) the "minimal" reasons of the rejection (abductive explanation), 2) the "minimal" changes to make a different outcome (constrastive explanation), and 3) whether this rejection is a bias.

Definition 1 (AXp, CXp and Bias). *Given a classifier model $C = (S, f) \in \mathbf{CM}$ and $s \in S$, s.t. $f(s) = x$. We call a property λ* abductively explains *decision x at s, noted $\mathsf{AXp}(\lambda, x)$, if*

$$(C, s) \models \lambda \wedge [\mathit{Atm}(\lambda)]\mathsf{t}(x) \wedge \bigvee_{p \in \mathit{Atm}(\lambda)} \langle \mathit{Atm}(\lambda) \setminus \{p\} \rangle \neg \mathsf{t}(x).$$

We call a property λ contrastively explains *decision x at s, noted $\mathsf{CXp}(\lambda, x)$, if*

$$(C, s) \models \mathsf{t}(x) \wedge \lambda \wedge \langle \mathit{Atm} \setminus \mathit{Atm}(\lambda) \rangle \neg \mathsf{t}(x) \wedge \bigwedge_{p \in \mathit{Atm}(\lambda)} [(\mathit{Atm} \setminus \mathit{Atm}(\lambda)) \cup \{p\}]\mathsf{t}(x).$$

We call the decision x is biased*, if*

$$(C, s) \models \mathsf{t}(x) \wedge \bigvee_{X \subseteq \mathsf{PF}} \langle \mathit{Atm} \setminus X \rangle \neg \mathsf{t}(x).$$

The three conjuncts of AXp means that 1) λ is a "part" of instance $\widehat{s}$; 2) atoms in $\mathit{Atm}(\lambda)$ staying the same valuation as in s, $\mathsf{t}(x)$ necessarily takes place regardless of other atoms; 3) λ is the "minimal" such property, in the sense that any its proper part $\lambda' \subset \lambda$ fails condition 2).

We can also explain the decision from a contrastive viewpoint, to find what is minimally needed if Alice could change herself for a different decision outcome. The four conjuncts of CXp means that 1) the decision value for the current state s is x; 2) λ is a part of instance $\widehat{s}$; 3) all atoms in $\mathit{Atm}(\lambda)$ staying the same valuation as in s, $\neg\mathsf{t}(x)$ is possible; 4) λ is a minimal such property, since any its proper part $\lambda' \subset \lambda$ fails condition 3).

As for Bias it says that 1) the decision for s is x; 2) for any state whose atom in X not staying the same valuation as in s, $\neg\mathsf{t}(x)$ is possible.

Proposition 1. *The following validities show the relation between CXp, counterfactual conditional and bias.*

$$\models_{\mathbf{CM}} \mathsf{CXp}(\lambda, x) \to (\overline{\lambda} \Rightarrow \mathsf{t}(x))$$

$$\models_{\mathbf{CM}} \mathsf{Bias}(x) \leftrightarrow \bigvee_{Atm(\lambda) \subseteq \mathsf{PF}} \mathsf{CXp}(\lambda, x)$$

Example 2. Following Example 1., we have $(C, s) \models \mathsf{AXp}(\neg m \wedge \neg o, y) \wedge \mathsf{CXp}(\neg m, y) \wedge \mathsf{CXp}(\neg o, y) \wedge (o \Rightarrow \neg \mathsf{t}(y)) \wedge (m \Rightarrow \neg \mathsf{t}(y)) \wedge \mathsf{Bias}(y)$. Being female and not owing property together abductively explains, and individually contrastively explains the rejection. Alice would be accepted if she were a male or owning a property, and so the decision is biased.

3 BCL Logic and Extensions

We provide a sound and complete axiomatics for the language $\mathcal{L}(Atm)$ and the semantics for **CM**, which we call binary classifier logic (BCL) as follows. Checking satisfiability of φ relative to **CM** is NP-complete. For notational convenience, for every finite $X, Y \subseteq Atm$, let $\mathsf{cn}_{Y,X} =_{def} \bigwedge_{p \in Y} p \wedge \bigwedge_{p \in X \setminus Y} \neg p$.

$$\big([\emptyset]\varphi \wedge [\emptyset](\varphi \to \psi)\big) \to [\emptyset]\psi \qquad (\mathbf{K}_{[\emptyset]})$$

$$[\emptyset]\varphi \to \varphi \qquad (\mathbf{T}_{[\emptyset]})$$

$$[\emptyset]\varphi \to [\emptyset][\emptyset]\varphi \qquad (\mathbf{4}_{[\emptyset]})$$

$$\varphi \to [\emptyset]\langle\emptyset\rangle\varphi \qquad (\mathbf{B}_{[\emptyset]})$$

$$[X]\varphi \leftrightarrow \bigwedge_{Y \subseteq X} \big(\mathsf{cn}_{Y,X} \to [\emptyset](\mathsf{cn}_{Y,X} \to \varphi)\big) \qquad (\mathbf{Red}_{[\emptyset]})$$

$$\bigvee_{x \in Val} \mathsf{t}(x) \qquad (\mathbf{AtLeast}_{\mathsf{t}(x)})$$

$$\mathsf{t}(x) \to \neg \mathsf{t}(y) \text{ if } x \neq y \qquad (\mathbf{AtMost}_{\mathsf{t}(x)})$$

$$\bigwedge_{Y \subseteq \left(Atm \setminus Dec\right)} \Big(\big(\mathsf{cn}_{Y,\left(Atm \setminus Dec\right)} \wedge \mathsf{t}(x)\big) \to [\emptyset]\big(\mathsf{cn}_{Y,\left(Atm \setminus Dec\right)} \to \mathsf{t}(x)\big)\Big) \qquad (\mathbf{Def})$$

$$\bigwedge_{X \subseteq \left(Atm \setminus Dec\right)} \langle\emptyset\rangle \mathsf{cn}_{X,\left(Atm \setminus Dec\right)} \qquad (\mathbf{Comp})$$

$$\frac{\varphi \to \psi \quad \varphi}{\psi} \qquad (\mathbf{MP})$$

$$\frac{\varphi}{[\emptyset]\varphi} \qquad (\mathbf{Nec}_{[\emptyset]})$$

It can be seen that $[\emptyset]$ is an S5 style modal operator, $\mathbf{Red}_{[\emptyset]}$ reduces any $[X]$ to $[\emptyset]$. $\mathbf{AtLeast}_{\mathsf{t}(x)}$, $\mathbf{AtMost}_{\mathsf{t}(x)}$, **Def** represent the decision function syntactically, that every $\widehat{s}$ maps to some unique $\mathsf{t}(x)$.[3] **Comp** ensures the function is total.

[3] Notice that $\mathsf{cn}_{Y,Atm \setminus Dec}$ is just another expression of $\widehat{s}$ where $s = Y$.

As mentioned, a modal language for classifier is fruitful, for we can extend the logic with more modal operators. Here we mention two and their uses.

Dynamic Extension We introduce a dynamic operator $[x := \varphi]$ with $x \in \mathit{Dec}$ into $\mathcal{L}(\mathit{Atm})$, and denote the resulting language $\mathcal{L}^{dyn}(\mathit{Atm})$. We extend our language by dynamic operators of the form $[x := \varphi]$ with $x \in \mathit{Dec}$. The formula $[x := \varphi]\psi$ has to be read as "ψ holds after every decision is set to x in context φ." The resulting language, noted $\mathcal{L}^{dyn}(\mathit{Atm})$, is defined by the following grammar:

$$\varphi ::= p \mid \neg\varphi \mid \varphi_1 \wedge \varphi_2 \mid [X]\varphi \mid [x{:=}\varphi]\psi$$

The interpretation of formula $[x := \varphi]\psi$ is relative to a pointed classifier model (C, s) with $C = (S, f)$ as follows:

$$(C, s) \models [x{:=}\varphi]\psi \Longleftrightarrow (C^{x:=\varphi}, s) \models \psi,$$

where $C^{x:=\varphi} = (S, f^{x:=\varphi})$ is the updated classifier model where, for every $s' \in S$:

$$f^{x:=\varphi}(s') = \begin{cases} x \text{ if } (C, s') \models \varphi, \\ f(s') \text{ otherwise.} \end{cases}$$

Call the class of all such models **DCM**. It can be used to model classifier's revision, e.g. training classifier in ML. A simple example is that we have an initial $C = (S, f)$ where $\forall s \in S, f(s) = 0$. Then we update C with a DNF φ to obtain $C^{1:=\varphi}$. The resulting model is $\mathit{Atm}(\varphi)$-essential and for any property λ, $\lambda \subseteq \varphi$ if and only if λ is a prime implicant at 1 of $f^{1:=\varphi}$.

Epistemic Extension A classifier can be conceived as an agent and we can represent its epistemic state and uncertainty. To this end we need to enrich our set of atoms. Note Atm_0 the original set of atoms excluding decision atoms. Then, define $\mathit{Atm} = \mathit{Atm}_0 \cup \mathit{Dec} \cup \mathit{ObsAtm}$, where $\mathit{ObsAtm} = \{\mathsf{o}(p) : p \in \mathit{Atm}_0\}$, and $\mathsf{o}(p)$, as an *observability* atom, has to be read "the agent can see the truth value of p". Thus we extend our language as $\mathcal{L}^{epi}(\mathit{Atm})$ by adding the operator $\mathsf{K}\varphi$ to represent what an agent knows in light of what it sees, defined by the following grammar:

$$\varphi ::= p \mid \neg\varphi \mid \varphi_1 \wedge \varphi_2 \mid [X]\varphi \mid \mathsf{K}\varphi.$$

Now the epistemic classifier model is a triple $C^{epi} = (S, \sim, f)$, where $\forall s, s' \in S, s \sim s' \iff \mathit{Obs}(s) = \mathit{Obs}(s')$ and $\forall p \in \mathit{Obs}(s), p \in s$ iff $p \in s'$, namely (i) what f can see are the same at s and s'; (ii) the truth values of the visible variables are the same at s and s'. In this case, the agent f cannot tell s from s'. Thus we have the satisfaction relation: $(C, s) \models \mathsf{K}\varphi \Longleftrightarrow \forall s' \in S :$ if $s \sim s'$ then $(C, s') \models \varphi$. Call the class of all such models **ECM**.

Suppose the model is X-definite, then the agent (namely f) has no uncertainty about its classification, if $(\bigwedge_{p \in X} \mathsf{o}(p) \wedge \mathsf{t}(x)) \to \mathsf{Kt}(x)$ holds. Otherwise the agent's classification is fallible. We can surely extend $\mathcal{L}^{epi}(\mathit{Atm})$ to multi-agent case by considering a set of operators K_i to obtain $\mathcal{L}^{epi}(\mathit{Atm}, \mathit{Agt})$.

Both checking satisfiability of φ relative to **DCM** and **ECM** are NP-complete.

References

1. Or Biran and Courtenay Cotton. Explanation and justification in machine learning: A survey. In *IJCAI-17 workshop on explainable AI (XAI)*, volume 8(1), pages 8–13, 2017.
2. Adnan Darwiche and Auguste Hirth. On the reasons behind decisions. In *ECAI 2020 - 24th European Conference on Artificial Intelligence*, volume 325 of *Frontiers in Artificial Intelligence and Applications*, pages 712–720. IOS Press, 2020.
3. Amit Dhurandhar, Pin-Yu Chen, Ronny Luss, Chun-Chen Tu, Paishun Ting, Karthikeyan Shanmugam, and Payel Das. Explanations based on the missing: Towards contrastive explanations with pertinent negatives. In *Advances in neural information processing systems*, pages 592–603, 2018.
4. Davide Grossi, Emiliano Lorini, and François Schwarzentruber. The ceteris paribus structure of logics of game forms. *Journal of Artificial Intelligence Research*, 53:91–126, 2015.
5. Alexey Ignatiev, Nina Narodytska, Nicholas Asher, and Joao Marques-Silva. From contrastive to abductive explanations and back again. In *International Conference of the Italian Association for Artificial Intelligence*, pages 335–355. Springer, 2020.
6. Alexey Ignatiev, Nina Narodytska, and Joao Marques-Silva. Abduction-based explanations for machine learning models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 1511–1519, 2019.
7. David Lewis. Counterfactual dependence and time's arrow. *Noûs*, pages 455–476, 1979.
8. Xinghan Liu and Emiliano Lorini. A logic for binary classifiers and their classification. In *Fourth International Conference, CLAR 2021*. Springer, forthcoming.
9. David Martens and Foster Provost. Explaining data-driven document classifications. *Mis Quarterly*, 38(1):73–100, 2014.
10. Brent Mittelstadt, Chris Russell, and Sandra Wachter. Explaining explanations in ai. In *Proceedings of the conference on fairness, accountability, and transparency*, pages 279–288, 2019.
11. Ramaravind K. Mothilal, Amit Sharma, and Chenhao Tan. Explaining machine learning classifiers through diverse counterfactual explanations. In *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*, pages 607–617, 2020.
12. Willard V. Quine. A way to simplify truth functions. *The American mathematical monthly*, 62(9):627–631, 1955.
13. Weijia Shi, Andy Shih, Adnan Darwiche, and Arthur Choi. On tractable representations of binary neural networks. *arXiv preprint arXiv:2004.02082*, 2020.