Evaluating the Interpretability of Threshold Operators

Guendalina Righetti[1(✉)], Daniele Porello[2(✉)], and Roberto Confalonieri[1,3(✉)]

[1] Faculty of Computer Science, Free University of Bozen-Bolzano, 39100 Bolzano, Italy
guendalina.righetti@stud-inf.unibz.it
[2] Dipartimento di Antichità, Filosofia e Storia, Università di Genova, 16126 Genova, Italy
daniele.porello@unige.it
[3] Dipartimento di Matematica, Universitá degli Studi Padova, Via Trieste, 63, 35121 Padova, Italy
roberto.confalonieri@unibz.it

Abstract. Weighted Threshold Operators are n-ary operators that compute a weighted sum of their arguments and verify whether it reaches a certain threshold. They have been extensively studied in the area of circuit complexity theory, as well as in the neural network community under the name of *perceptrons*. In Knowledge Representation, they have been introduced in the context of standard Description Logics (DL) languages by adding a new concept constructor, the Tooth operator ($\mathbb{W}$). Tooth expressions can provide a powerful yet natural tool to represent local explanations of black box classifiers in the context of Explainable AI. In this paper, we present the result of a user study in which we evaluated the interpretability of tooth expressions, and we compared them with Disjunctive Normal Forms (DNF). We evaluated interpretability through accuracy, response time, confidence, and perceived understandability by human users. We expected tooth expressions to be generally more interpretable than DNFs. In line with our hypothesis, the study revealed that tooth expressions are generally faster to use, and that they are perceived as more understandable by users who are less familiar with logic. Our study also showed that the type of task, the type of DNF, and the background of the respondents affect the interpretability of the formalism used to represent explanations.

Keywords: Threshold operators · Explainable AI · Interpretability · User study

1 Introduction

Predictive models based on machine and deep learning techniques have become ubiquitous in many decision making scenarios. Whilst these models are typically

This research is partially supported by Italian National Research Project PRIN2020 2020SSKZ7R and by unibz RTD2020 project HULA.

O. Corcho et al. (Eds.): EKAW 2022, LNAI 13514, pp. 136–151, 2022.
https://doi.org/10.1007/978-3-031-17105-5_10

very performative, they behave like black boxes, lacking transparency and leading to unfair and discriminative outcomes [23]. To this end, a lot of attention has been given to approaches that can explain black box models to increase trust by all users in why and how decisions are made [2,5,21].

Explainable AI (XAI) has been identified as a key factor for developing trustworthy AI systems [2,6]. The reasons for equipping AI systems with explanation capabilities are not only limited to enable diagnostics to prevent bias, unfairness, and discrimination [8], but also to user rights and acceptance (e.g., see Article 22 of the GDPR law [24]).

XAI focuses on developing approaches for explaining black box models by achieving good explainability without sacrificing system performance [18]. One typical approach is the extraction of local or global post-hoc explanations that approximate the behaviour of a black box model by means of an interpretable proxy. For instance, LIME is a local post-hoc explanation approach that explains model instances by means of linear expressions [26]. Other approaches advocate a tighter integration between symbolic and non-symbolic knowledge, e.g., by combining symbolic and statistical methods of reasoning [9,17].

Symbolic knowledge plays a key role for the creation of intelligible explanations. In [9], it has been shown that the integration of DL ontologies in the creation of explanations can enhance the perceived *interpretability*[1] of post-hoc explanations by human users. Furthermore, linking explanations to formal background knowledge brings multiple advantages. It does not only enrich explanations (or the elements therein) with semantic information—thus facilitating common-sense reasoning—, but it also creates a potential for supporting the customisation of the levels of specificity and generality of explanations to specific user profiles [19].

Motivated by the conventional wisdom that disjunctive normal form (DNF) is considered as a benchmark in terms of both expressivity and interpretability of logic-based knowledge representations [12], we assume to have local explanations of black box models modeled as a DNF formula. An example explanation from a loan agent could be: 'I grant a loan when the subject has no children and is married or when he has high income range' (i.e., $(\neg Parent \sqcap Married) \sqcup Rich$). Prior works raised the questions of whether DNF is always the most interpretable representation, and whether alternate representation forms enable better interpretability [4,12]. In particular, [4] evaluated several forms of DNFs in terms of their interpretability when presented to human users as logical explanations for different domains of application. In this work we aim at comparing the intepretability of DNFs and threshold operators.

Weighted Threshold Operators are n-ary operators which compute a weighted sum of their arguments and verify whether it reaches a certain threshold. These operators have been extensively studied in the area of circuit complexity theory (see e.g., [30]), and they are also known in the neural network community by perceptrons (see e.g., [3]). Threshold operators have been studied in the context of Knowledge Representation and integrated within DLs in [25], by adding a novel

[1] Interpretability describes the possibility to comprehend a black box model and to present the underlying basis for decision-making in a way that is understandable to humans [13].

concept constructor, the "Tooth" operator ($\mathbb{W}$). From now on, we shall use, more specifically "tooth operators" and "tooth expressions". Tooth operators allow for introducing weights into standard DL languages to assess the importance of the features in the definition of the concepts. For instance, as we shall see, the concept $\mathbb{W}^1((Parent, -1), (Rich, 2), (Married, 1))$ classifies those instances for which the sum of the satisfied weighted concepts reaches the threshold 1.

In the context of XAI, tooth expressions provide a powerful yet natural tool to represent local explanations of black box classifiers. In [14,16] a link between tooth-expressions and linear classifiers has been established, where it is shown that tooth-operators behave like *perceptrons*. More precisely, a (non-nested) tooth expression is a linear classification model, which enables to *learn* weights and thresholds from real data (in particular, from sets of assertions about individuals), exploiting standard linear classification algorithms. Thus, they could be used to represent post-hoc local explanations. Furthermore, adding tooth operators to any language including the booleans does not increase the expressivity and complexity of the language. Tooth expressions are indeed equivalent to standard DNFs,[2] i.e., canonical normal form of logical formulas consisting of a disjunction of conjunctions of literals [16]: they are 'syntactic sugar' for languages that include the booleans. They allow, however, for crisper formulas, being thus less error-prone and, putatively, more understandable by users.

In this paper, we present the results of a user study we conducted to measure the interpretability of tooth expressions versus their translation into standard DNFs. In the user study, respondents were asked to carry out different classification tasks using concepts represented both as a tooth-expressions and as DNFs. In line with previous works evaluating the interpretability of explanation formats (e.g., [1,4,9,10,20]), we used the metrics of accuracy, time of response, and confidence in the answers as a proxy for evaluating the interpretability of the two representations. We expected that tooth expressions could be perceived as more interpretable. In line with our hypothesis, our study revealed that the type of task, the background of the respondents, and the size of the DNF formula affect the interpretability of the formalism used.

2 Background

2.1 Tooth Operator - Preliminary Definitions

In this section, we delineate the formal framework necessary to introduce $\mathbb{W}$ (Tooth) expressions. Following the work done in [25], we extend standard DL languages with a class of m-ary operators denoted by the symbol $\mathbb{W}$ (spoken 'tooth'). Each operator works as follows: (*i*) it takes a list of concepts, (*ii*) it associates a weight (i.e., a number) to each of them, and (*iii*) it returns a complex concept that applies to those instances that satisfy a certain combination of concepts, i.e., those instances for which, by summing up the weights of the satisfied concepts, a certain threshold is met. More precisely, we assume a vector

[2] More precisely, non-nested tooth-expressions are not able to represent the XOR. Nested tooth can however overcome this difficulty.

of m weights $\vec{w} \in \mathbb{R}^m$ and a threshold value $t \in \mathbb{R}$. If $C_1, \ldots, C_m$ are concepts of $\mathcal{ALC}$, then $\mathbb{W}^t_{\vec{w}}(C_1, \ldots, C_m)$ is a concept of $\mathcal{ALC}_{\mathbb{W}}$. For C'_i concept of $\mathcal{ALC}$, the set of $\mathcal{ALC}_{\mathbb{W}}$ concepts is described by the grammar:

$$C ::= A \mid \neg C \mid C \sqcap C \mid C \sqcup C \mid \forall R.C \mid \exists R.C \mid \mathbb{W}^t_{\vec{w}}(C'_1, \ldots, C'_m)$$

To better visualise the weights an operator associates to the concepts, we often use the notation $\mathbb{W}^t((C_1, w_1), \ldots, (C_m, w_m))$ instead of $\mathbb{W}^t_{\vec{w}}(C_1, \ldots, C_m)$.

The semantics of $\mathcal{ALC}_{\mathbb{W}}$ just extends the usual semantics of $\mathcal{ALC}$ to account for the interpretation of the Tooth operator, as follows.

Let $I = (\Delta^I, \cdot^I)$ be an interpretation of $\mathcal{ALC}$. The interpretation of a $\mathbb{W}$-concept $C = \mathbb{W}^t((C_1, w_1), \ldots, (C_m, w_m))$ is:

$$C^I = \{d \in \Delta^I \mid v^I_C(d) \geq t\} \tag{1}$$

where $v^I_C(d)$ is the *value* of $d \in \Delta^I$ under the concept C, defined as:

$$v^I_C(d) = \sum_{i \in \{1,\ldots,m\}} \{w_i \mid d \in C^I_i\} \tag{2}$$

We refer the interested reader to [14,15,25] for a more precise account of the properties of the operator.

In the context of Knowledge Representation, tooth expressions provide a powerful tool to represent concepts. Tooth operators have indeed been applied in DL with a variety of goals. As already mentioned, in [14,16] a link between tooth-expressions and linear classifier has been established. In [16], in particular, it was shown that even simple tooth-expressions are expressive enough to represent complex concepts derived from real use cases in the context of the Gene Ontology.

Tooth operators are also useful in the representation of different cognitively relevant phenomena related to human concept combination and categorisation [27,28]. More precisely, the representation of tooth expressions is inspired by the design of Prototype Theory [29]. Tooth operators, and generally weighted logics [22], are thus more cognitively grounded than standard logic languages, allowing for a representation of concepts that is, arguably, more in line with the way humans think of them.

In particular, Tooth expressions are equivalent to standard DNFs, i.e., canonical normal form of logical formulas consisting of a disjunction of conjunctions of literals [16].

Let us imagine, for instance, to model the explanation for approving a loan from a loan agent, as described in the Introduction, by means of the tooth-operator. This could be captured through an axiom using a tooth expression as follows: $\exists isGranted.Loan \sqsubseteq \mathbb{W}^1((Parent, -1), (Rich, 2), (Married, 1))$.

The practical advantages for knowledge acquisition and cognitive science are thus gained without any increase in computational complexity: adding Tooth operators to $\mathcal{ALC}$ does not increase the expressivity of the language. The reason is that $\mathcal{ALC}$ is closed under Boolean operators, so any Tooth concept can be translated into a DNF of concepts of $\mathcal{ALC}$, see ([25], Sec. 3.1). Moreover, any

ontology in $\mathcal{ALC}$ plus Tooth concepts can be translated into an ontology in the language of $\mathcal{ALC}$, see ([14] Sec. 2).

By representing our running example by $D = \mathbb{W}^1((A,-1),(B,2),(C,1))$, we show that it is extensionally equivalent to the DNF $(\neg A \sqcap C) \sqcup B$. In one direction, if $d \in ((\neg A \sqcap C) \sqcup B)^I$, then $d \in (\Delta^I \setminus A^I) \cap C^I$ or $d \in B^I$. In the first case, d scores 1 because $d \in C^I$; in the second case, d scores 2 because $d \in B^I$. Therefore, in both cases, $v_D^I(d) \geq 1$, so $d \in D^I = \mathbb{W}^1((A,-1),(B,2),(C,1))^I$.

In the other direction, suppose by contraposition that $d \notin ((\neg A \sqcap C) \sqcup B)^I$. So $d \notin (\Delta^I \setminus A^I) \cap C^I$ and $d \notin B^I$. We have two cases, if $d \notin (\Delta^I \setminus A^I)$, then $d \in A^I$, so d scores -1. Since $d \notin B^I$, d does not score 2, so $v_D^I(d) < 1$. If $d \notin C^I$, then d does not score 1, and since $d \notin B^I$, again $v_D^I(d) < 1$. Thus, in both cases, $v_D^I(d) < 1$, so $d \notin D^I = \mathbb{W}^1((A,-1),(B,2),(C,1))^I$.

2.2 Disjunctive Normal Forms - Preliminary Definitions

A disjunctive normal form (DNF) is a logical formula consisting of a disjunction of one or more conjunctions, of one or more literals. It can also be described as an OR of ANDs, as the only propositional operators in DNF are the and ($\wedge$), the or ($\vee$), and the negation ($\neg$). In our study, we used DL symbols ($\sqcap$, $\sqcup$) to interpret conjunctions and disjunctions of concepts.

Henceforth, we will follow the definitions proposed by Darwiche and Marquis [12]. Accordingly, DNF is a strict subset of the Negation Normal Form language. An NNF formula can be characterised as a *rooted*, *directed*, *acyclic* graph, where each leaf node is labeled with a propositional variable or its negation, and each internal node is labeled with a conjunction or a disjunction. A DNF is a *flat* NNF, i.e., an NNF whose maximum number of edges from the root to some leaf is 2. Moreover, DNFs satisfies the property of *simple conjunction*, i.e., each propositional variable occurs at most once in each conjunction. An example is provided in Fig. 1.

One can consider different NNF subsets by imposing one or more of the following conditions on the formulas: (i) *Decomposability*: an NNF is decomposable (DNNF) iff for each conjunction in the NNF, the conjuncts do not share variables. Each DNF is decomposable by definition. (ii) *Determinism*: an NNF is deterministic (d-NNF) iff for each disjunction in the NNF, every two disjuncts are logically contradictory. (iii) *Smoothness*: NNFs satisfy smoothness (sd-NNF) iff for each disjunction formula, each disjunct mentions the same variables. When looking at DNF, the class of formulas satisfying determinism and smoothness is called MODS.

In what follows, we will consider three sets of DNF, obtained by adding different conditions on the formulae (and leading to formulas of different sizes).

- **DNF1:** Simple (decomposable) DNFs ($DNF1 \subsetneq DNNF$), corresponding to the shorter formulas. The only requirement for the formulas is to satisfy the property of simple conjunction. See (i) in Fig. 1 for an example.
- **DNF2:** Deterministic DNFs ($DNF2 \subsetneq d-NNF$), for which each couple of disjuncts is required to be logically contradictory. See (ii) in Fig. 1 for an example.

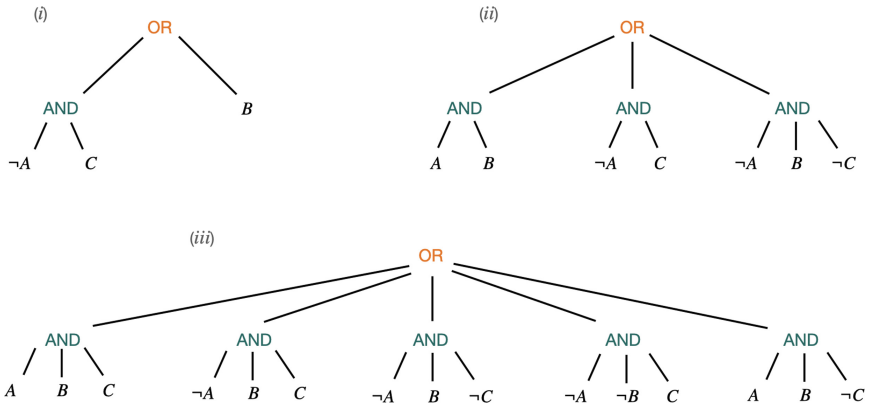

Fig. 1. Three different variants of the same DNF modeling the running example.

– **DNF3:** Deterministic, smooth DNFs ($DNF3 \subsetneq MODS$), corresponding to the longest possible DNFs. DNF3 collect all the formula models. See (*iii*) in Fig. 1 for an example.

3 Evaluating the Interpretability of Explanations

The notion of interpretability of symbolic representations has gained popularity in recent years (e.g., [1,4,20]), also due to an increasing interest in Explainable AI. How to precisely characterise interpretability is however far from being obvious, and there is, in general, no consensus on a precise definition.

From now on, we adhere to the taxonomy of interpretability evaluation proposed by Doshi-Velez and Kim [13] which supports using 'human-grounded metrics' with real users to evaluate the perceived quality of an explanation. According to this view, the evaluation focuses on the perceived interpretability of explanations rather than in their mechanistic creation. Thus, it is not important how the explanations are computed, but whether these explanations are perceived as interpretable by humans.

To operationalise this idea, different strategies have been adopted in the literature (see e.g., [1,9,20]). In order to measure the interpretability of an explanation, subjects are usually asked to perform the same task (often, a classification task) using different explanation formats. Across the different studies, the evaluation metrics can then vary, but they normally range between four metrics, namely *accuracy* (how many times did the subjects reply correctly), *time of response* (how fast they were in carrying out the task), *confidence* (how confident did they feel in their reply), and *perceived understandability* (to what extent an explanation is perceived as understandable by the user). In [20], for instance, the interpretability of decision tables, binary decision trees, and propositional rules is measured by combining the metrics of accuracy, time of response, and confidence. Allahyari and Lavesson [1] focus on the interpretability of decision tree

models and rule-based models, using the perceived understandability as the only metric for the evaluation. The metrics of accuracy, time of response, confidence, and perceived understandability are taken into account in [9] to measure the interpretability of decision trees. More precisely, the paper extends Trepan [11], an algorithm that explains ANNs by means of decision trees, to include ontologies that model domain knowledge when generating explanations. The paper shows that decision trees generated taking into account domain knowledge are perceived as more understandable by users.

Booth et al. [4] compare different propositional theories and evaluates their interpretability in different domains of application. To the best of our knowledge, [4] constitutes the most thorough attempt to evaluate the interpretability of different logical languages in terms of human-grounded metrics. In their study, the authors presented subjects with natural language explanations translating different propositional languages (varying from DNFs, CNFs and other variations of NNFs), across different domains. They thus evaluated subjects' comprehension of these explanations in terms of accuracy, confidence, and time of response. They observed that while decomposability resulted in a statistically significant increase in confidence, simple conjunction did not always show an effect in their dataset. Interestingly, they also observed that the domain, in which the explanations were presented, affected the perceived understandability of the formulas.

In the following, we present a user study that compares the interpretability of tooth expressions and DNF formulas. In the study, respondents were asked to carry out two tasks using Tooth expressions and DNF formula. We evaluated the interpretability of formulas by means of accuracy in the responses, time of response, confidence in the reply, and perceived understandability of the formula used. To avoid any bias due to prior knowledge about a certain domain, we kept the presentation of the input at an abstract level, that is, respondents were provided with logical formulas not bounded to any domain in particular.

We use variables (e.g., A, B, C) for concepts occurring in the DNFs as well as for concepts occurring in the Tooth expressions. Formally, those variables range over concepts of $\mathcal{ALC}$. However, in practice, we do not present participants concepts defined by means of restricted quantifications (i.e., $\forall R.C$ and $\exists R.C$) and we focus on the Boolean operators of $\mathcal{ALC}$. Moreover, concepts in the scope of the tooth expression are simple, i.e., we do not allow for Boolean combinations. These two simplifications allow for a direct comparison between Tooth and DNFs. More complex cases shall be analysed in a longer dedicated study.

4 Experimental Evaluation

The main research hypothesis in which we were interested was whether Tooth operators are more effective and perceived as more interpretable than DNFs by human users. More precisely, we were interested in determining under what metrics this was the case (see Sect. 3), and for which types of DNF formulas (see Sect. 2.2). To verify or refute this hypothesis we designed and ran a user study.

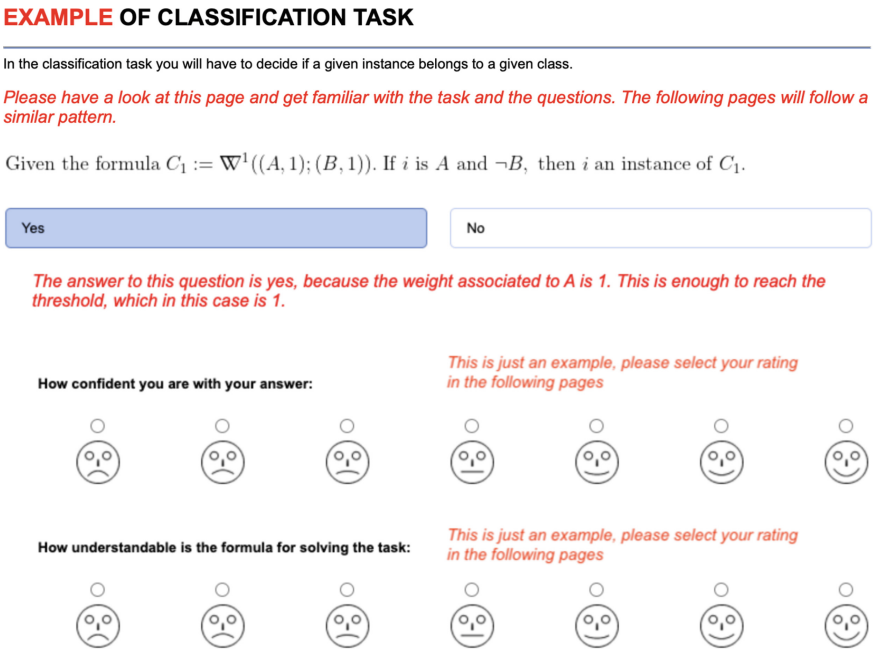

Fig. 2. The introductory page of the classification task for the Tooth-operator questionnaire.

4.1 Method

Materials. We used examples of concepts defined by means of DNF formulas (the three variants) and by means of the Tooth operator. We had 6 concept definitions of different complexities, varying in the number of symbols used and length. For each concept, we constructed four formulas, one for each of the formats (i.e., DNF1, DNF2, DNF3 and Tooth expression). In this manner, we obtained 24 distinct concept definitions. We had two questionnaires, one for the DNFs and one for Tooth expressions. In the user study, each participant was shown a total of twelve formulas corresponding to concept definitions. That is, participants were asked to carry out both questionnaires, in separate sessions, in random order. Concept definitions were randomly shuffled for each of the participants in the user study.

Procedure. The experiment used an online questionnaires on the usage of logical formulas to carry out certain tasks. The questionnaire was run in a controlled environment (i.e., in a classroom). The questionnaire contained an introductory and an experimental phase. In the introductory phase, subjects were shown a short description of either DNFs or Tooth operator, and how its semantics is determined. Each introduction had the same duration, and consisted of the same number of slides (and examples) for DNFs and Tooth expressions.

The experimental phase was divided into two tasks: classification, and inspection. Each task starts with an instruction page describing the task to be performed (an example for the classification task is shown in Fig. 2). In these tasks the participants were presented with six formulas corresponding to one of the two representations (one of the variants of the DNFs and the Tooth operator). In the classification task, subjects were asked to decide if a certain combination of literals is an instance of a given formula (e.g., *Given the formula* $C_1 := (\neg A \sqcap C) \sqcup B$. *If i is* $\neg A$, B, *and* $\neg C$, *then i is an instance of* C_1). In the inspection task, participants had to decide on the truth value of a particular statement, referring to if some given conditions of an instance are necessary for the instance to belong to a given class (e.g., *Given the formula* $C_1 := (\neg A \sqcap C) \sqcup B$. *Having* B *is necessary for being classified as* C_1). The main difference between the two types of questions used in the two tasks is that the former provides all details necessary for performing the decision, whereas the latter only specifies whether a subset of the features influence the decision. In these two tasks, for each formula, we recorded:

- Correctness of the response.
- Confidence in the response, as provided on a Likert scale from 1 to 7.
- Response time measured from the moment the formula was presented.
- Perceived formula understandability, as provided on a Likert scale from 1 to 7.

Participants. 58 participants volunteered to take part in the experiment. The participants were recruited among students with different backgrounds. In particular we had two groups of students, 33 students with a background in computer science and 25 students with a background in philosophy. Each group repeated the questionnaire twice, once using DNFs and once using Tooth expressions. In the analysis, we will denote these groups as GroupI and GroupII respectively.

4.2 Results

As it can be appreciated in Table 1, when looking at the two groups together, respondents carried out both tasks correctly, performing better in the classification task than in the inspection task. This is in line with our assumption that the classification task was simpler than the inspection task, due to the fact that more information was provided for making the decision. Remarkably, the influence of the type of formula on the percentage of correct answers is not significant in our dataset. More specifically, the answers to tasks containing DNFs are slightly more accurate than those containing Tooth expressions, but this difference is not statistically significant. Nonetheless, we observed a significant influence ($p < .0001$) of Tooth expressions on the time of response within both tasks, showing that when using Tooth operators respondents carried out the tasks in a quicker way. This suggests that Tooth expressions are more cognitively friendly than standard DNFs. Interestingly, Tooth operators were perceived as more understandable in carrying out the inspection task. Similarly, users were more confident with their

Table 1. Mean values of correct answers, time of response, user confidence, and user understandability for formulas represented using DNFs and Tooth operator (standard deviations are reported in parenthesis).

Task	Measure	DNFs	Tooth
Classification	%Correct Responses	0.91 (0.28)	0.90 (0.29)
	Time (sec)	46.78 (58.90)	29.87 (20.72)
	Confidence	5.74 (1.32)	5.65 (1.51)
	User Understandability	5.80 (1.24)	5.55 (1.44)
Inspection	%Correct Responses	0.87 (0.34)	0.83 (0.37)
	Time (sec)	28.67 (28.78)	19.78 (19.78)
	Confidence	5.70 (1.32)	5.82 (1.49)
	User Understandability	5.79 (1.24)	5.81 (1.43)

Table 2. Mean values of correct answers, time of response, user confidence, and user understandability for formulas represented using DNFs and Tooth operator for **GroupI** and **GroupII** (standard deviations are reported in parenthesis).

Group	Measure	DNFs	Tooth
Computer Science	%Correct Responses	0.90 (0.32)	0.88 (0.31)
	Time (sec)	37.29 (55.29)	25.23 (17.96)
	Confidence	5.98 (1.29)	5.73 (1.71)
	User Understandability	6.11 (1.17)	5.61 (1.65)
Philosophy	%Correct Responses	0.86 (0.30)	0.90 (0.34)
	Time (sec)	36.39 (28.06)	24.80 (16.77)
	Confidence	5.44 (1.28)	5.88 (1.15)
	User Understandability	5.43 (1.20)	5.84 (1.10)

answers when using Tooth operators in the inspection task. This is in line with our assumption that Tooth operators could be perceived as simpler representations when the task can benefit from a more compact representation of the concepts. On the contrary, DNFs were perceived better than Tooth operators in the classification task, and respondents were more confident with their answers.

When looking at the two groups separately (Table 2), the percentages of correct answers are slightly different when using DNFs and Tooth operators, but this difference is again not significant. Thus, generally, we can conclude that the type of formula used does not have any significant effects or interactions on the accuracy of responses. Tooth operators yielded faster responses in both groups. This seems to suggest that having more compact information, like in the case of Tooth operators, could speed up the human decision-making process.

Table 3. Mean values of correct answers, time of response, user confidence, and user understandability for formulas represented using DNF1, DNF2, DNF3 and Tooth operator for both groups (standard deviations are reported in parenthesis).

Measure	DNF1	DNF2	DNF3	Tooth
%Correct Responses	0.91 (0.28)	0.90 (0.30)	0.78 (0.42)	0.83 (0.38)
Time (sec)	21.03 (10.84)	25.01 (19.99)	39.97 (42.28)	19.78 (12.81)
Confidence	6.21 (1.19)	5.72 (1.20)	5.18 (1.37)	5.82 (1.49)
User Understandability	6.14 (1.00)	5.99 (1.19)	5.24 (1.34)	5.81 (1.43)

Interestingly, faster decision making can yield more correct responses, but surprisingly faster decision-making is not always associated with highest perceived understandability and highest confidence. Respondents with computer science background were more confident with DNFs and perceived them as more understandable than Tooth operators. On the contrary, respondents with a background in philosophy found Tooth operators more understandable and were more confident with their answers when using Tooth operators. This behaviour can be motivated by the fact that computer scientists were introduced to logic and DNF formulas in their curricula, but not to Tooth operators. Thus, being more proficient in DNFs, they did not face the 'learning curve' in understanding a new representation formalism such Tooth operators. Respondents with a background in philosophy, on the other hand, studied neither DNFs nor Tooth operators. From this study, we can conclude that Tooth operators are better representation for users who are not familiar with logic, and with DNFs in particular.

When looking at results of different DNFs vs Tooth operator (Table 3), we can observe that simpler DNF formats, namely DNF1 and DNF2, yielded more accurate responses. Tooth operators perform better compared to DNF3. This is expected since formulas in DFN3 format tend to be very long (see examples in Sect. 2.2). DNF1 and DNF2 performs similarly in our study. This is expected, since they are quite similar in lengths and they do not impose a cognitive burden on the users w.r.t. DNF3 (as also shown in the previous study comparing them directly [4]). As far as time is concerned, we still observe that Tooth operators are faster than any of the DNF formats. Remarkably, the response time obtained using DNF1 is similar to the one obtained when using the Tooth operator. This can be motivated by observing that DNF1 format can be considered still a concise representation. Thus, the 'interformat' analysis seems to suggest that DNF1 and Tooth operator have quite similar understandability from the performance point of view and also from the subjective point of view. On the other hand, DFN2 and DNF3 require longer time of response and were perceived as less understandable than Tooth operators.

5 Conclusion and Future Works

In this paper, we studied the intepretability of threshold operators, by comparing them with a standard logical formalism, i.e. the DNFs. To model threshold operators in a logical setting and to facilitate the comparison with DNFs, we presented the threshold operators as concept constructors on top of $\mathcal{ALC}$, i.e. the Tooth expressions. Then, we proposed a user study aiming at comparing the interpretability of Tooth expressions and DNFs.

On the one hand, DNFs are conventionally considered a benchmark in terms of both expressivity and interpretability of logical languages [12]. On the other hand, Tooth expressions [25] provide a more concise representation of formulas. Furthermore, they are cognitively grounded, since their design is inspired by Prototype Theory [29]. Thus, they should allow for a representation of concepts that is, arguably, more in line with the way humans think of them. We hypothesised tooth expressions to be generally more interpretable than DNFs.

In the user study, we compared Tooth expressions with equivalent DNFs of different complexity and length, by imposing different conditions on the DNFs used (see Appendix A). We asked users to carry out two distinct tasks, namely classification and inspection (see Sect. 4), using Tooth expressions and DNFs. The interpretability of Tooth expressions and DNFs was measured through human-grounded metrics, namely accuracy in the responses, time of response, confidence in the responses, and perceived understandability.

In line with our hypothesis, the study revealed that Tooth expressions are generally faster to use, leading to a lower time of response. This was observed across all different DNFs formats considered in the study. Moreover, Tooth expressions were perceived as more understandable than DNFs in the inspection task (suggesting that they are better suited to tasks that benefit from a more compact representation of knowledge). The same was not generally observed in the classification task. Whilst the time of response was much lower for Tooth expressions than DNFs and the percentage of correct responses was almost the same for Tooth expressions and DNFs, the confidence in the reply and the perceived understandability were higher in the case of DNF formulas. By distinguishing different DNF formats, we observed that longer DNFs (e.g., DNF3) were perceived as less understandable than Tooth expressions. This result was also affected by the background of the respondents. Tooth operators, in particular, resulted in better performances and in a higher level of perceived understandability for users who were not familiar with logic.

The results obtained open several directions for future work. Firstly, we plan a second user study, where both Tooth expressions and DNFs are translated into natural language. This would allow to further test whether the algorithm of classification which stands behind the Tooth operator is more interpretable and easy to use. Secondly, we plan to compare decision trees and Tooth expressions [7]. Decision trees and Tooth expressions seem to have complementary pros and cons when considered in the context of XAI. Analysing the different performances of

users in either the representations might provide useful insights on which representation format would be more suitable in relation to different contexts, tasks, and applications.

Acknowledgment. The authors thank Oliver Kutz, Nicolas Troquard, Pietro Galliani, and Antonella De Angeli for taking the pre-test and providing precious feedback about the user study.

A Examples used in the questionnaires

1. – DNF1: $A \sqcup B$
   – DNF2: $A \sqcup (\neg A \sqcap B)$
   – DNF3: $(A \sqcap B) \sqcup (\neg A \sqcap B) \sqcup (A \sqcap \neg B)$
   – Tooth: $\mathbb{W}^1((A,1),(B,1))$
2. – DNF1: $(\neg A \sqcap C) \sqcup B$
   – DNF2: $(A \sqcap B) \sqcup (\neg A \sqcap C) \sqcup (\neg A \sqcap B \sqcap \neg C)$
   – DNF3: $(A \sqcap B \sqcap C) \sqcup (\neg A \sqcap B \sqcap C) \sqcup (\neg A \sqcap B \sqcap \neg C) \sqcup (\neg A \sqcap \neg B \sqcap C) \sqcup (A \sqcap B \sqcap \neg C)$
   – Tooth: $\mathbb{W}^2((\neg A,1),(B,2),(C,1)) \equiv \mathbb{W}^1((A,-1),(B,2),(C,1))$
3. – DNF1: $(\neg A \sqcap B) \sqcup C$
   – DNF2: $(\neg A \sqcap B) \sqcup (A \sqcap \neg B \sqcap C) \sqcup (A \sqcap B \sqcap C) \sqcup (\neg A \sqcap \neg B \sqcap C)$
   – DNF3: $(\neg A \sqcap B \sqcap C) \sqcup (\neg A \sqcap B \sqcap \neg C) \sqcup (A \sqcap \neg B \sqcap C) \sqcup (A \sqcap B \sqcap C) \sqcup (\neg A \sqcap \neg B \sqcap C)$
   – Tooth: $\mathbb{W}^2((A,-1),(B,2),(C,3))$
4. – DNF1: $(A \sqcap B) \sqcup (B \sqcap C) \sqcup (A \sqcap C)$
   – DNF2: $(A \sqcap B) \sqcup (A \sqcap \neg B \sqcap C) \sqcup (\neg A \sqcap B \sqcap C)$
   – DNF3: $(A \sqcap B \sqcap C) \sqcup (\neg A \sqcap B \sqcap C) \sqcup (A \sqcap \neg B \sqcap C) \sqcup (A \sqcap B \sqcap \neg C)$
   – Tooth: $\mathbb{W}^2((A,1),(B,1),(C,1))$
5. – DNF1: $(A \sqcap D) \sqcup (A \sqcap B \sqcap C) \sqcup (D \sqcap B) \sqcup (D \sqcap C)$
   – DNF2: $(A \sqcap D) \sqcup (A \sqcap B \sqcap C \sqcap \neg D) \sqcup (\neg A \sqcap B \sqcap D) \sqcup (\neg A \sqcap \neg B \sqcap C \sqcap D)$
   – DNF3: $(\neg A \sqcap \neg B \sqcap C \sqcap D) \sqcup (\neg A \sqcap B \sqcap \neg C \sqcap D) \sqcup (\neg A \sqcap B \sqcap C \sqcap D) \sqcup (A \sqcap \neg B \sqcap \neg C \sqcap D) \sqcup (A \sqcap B \sqcap C \sqcap \neg D) \sqcup (A \sqcap B \sqcap C \sqcap D)$
   – Tooth: $\mathbb{W}^5((A,3),(B,1),(C,1),(D,4))$
6. – DNF1: $(A \sqcap B) \sqcup (A \sqcap C) \sqcup (A \sqcap D) \sqcup (B \sqcap D)$
   – DNF2: $(A \sqcap B \sqcap \neg D) \sqcup (\neg A \sqcap B \sqcap C \sqcap D) \sqcup (A \sqcap \neg B \sqcap C \sqcap \neg D) \sqcup (\neg A \sqcap B \sqcap \neg C \sqcap D) \sqcup (A \sqcap D)$
   – DNF3: $(\neg A \sqcap B \sqcap \neg C \sqcap D) \sqcup (\neg A \sqcap B \sqcap C \sqcap D) \sqcup (A \sqcap \neg B \sqcap \neg C \sqcap D) \sqcup (A \sqcap \neg B \sqcap C \sqcap \neg D) \sqcup (A \sqcap \neg B \sqcap C \sqcap D) \sqcup (A \sqcap B \sqcap \neg C \sqcap \neg D) \sqcup (A \sqcap B \sqcap \neg C \sqcap D) \sqcup (A \sqcap B \sqcap C \sqcap \neg D) \sqcup (A \sqcap B \sqcap C \sqcap D)$
   – Tooth: $\mathbb{W}^3((A,2),(B,1.5),(C,1),(D,1.5))$
7. – DNF 1: $(A \sqcap B) \sqcup (A \sqcap C \sqcap D) \sqcup (B \sqcap C \sqcap D)$
   – DNF 2: $(A \sqcap B) \sqcup (\neg A \sqcap B \sqcap C \sqcap D) \sqcup (A \sqcap \neg B \sqcap C \sqcap D)$
   – DNF 3: $(A \sqcap B \sqcap C \sqcap D) \sqcup (A \sqcap B \sqcap \neg C \sqcap \neg D) \sqcup (A \sqcap B \sqcap \neg C \sqcap D) \sqcup (A \sqcap B \sqcap C \sqcap \neg D) \sqcup (\neg A \sqcap B \sqcap C \sqcap D) \sqcup (A \sqcap \neg B \sqcap C \sqcap D)$
   – Tooth: $\mathbb{W}^4((A,2),(B,2),(C,1),(D,1))$

References

1. Allahyari, H., Lavesson, N.: User-oriented assessment of classification model understandability. In: SCAI 2011 Proceedings, vol. 227, pp. 11–19. IOS Press (2011)
2. Barredo Arrieta, A., et al.: Explainable Explainable Artificial Intelligence (XAI): concepts, taxonomies, opportunities and challenges toward responsible AI. Inf. Fusion **58**(October 2019), 82–115 (2020). https://doi.org/10.1016/j.inffus.2019.12.012
3. Bishop, C.M.: Pattern Recognition and Machine Learning, 5th Edition. Information science and statistics, Springer, New York (2007). ISBN 9780387310732. https://www.worldcat.org/oclc/71008143
4. Booth, S., Muise, C., Shah, J.: Evaluating the interpretability of the knowledge compilation map. In: Kraus, S. (ed.) Proceedings of IJCAI, pp. 5801–5807 (2019)
5. Coba, L., Confalonieri, R., Zanker, M.: RecoXplainer: a library for development and offline evaluation of explainable recommender systems. IEEE Comput. Intell. Mag. **17**(1), 46–58 (2022). https://doi.org/10.1109/MCI.2021.3129958
6. Confalonieri, R., Coba, L., Wagner, B., Besold, T.R.: A historical perspective of explainable artificial intelligence. WIREs Data Min. Knowl. Disc. **11**(1), e1391 (2021). https://doi.org/10.1002/widm.1391
7. Confalonieri, R., Galliani, P., Kutz, O., Porello, D., Righetti, G., Troquard, N.: Towards knowledge-driven distillation and explanation of black-box models. In: Confalonieri, R., Kutz, O., Calvanese, D. (eds.) Proceedings of the Workshop on Data meets Applied Ontologies in Explainable AI (DAO-XAI 2021) part of Bratislava Knowledge September (BAKS 2021), CEUR Workshop Proceedings, Bratislava, Slovakia, 18–19 September 2021, vol. 2998. CEUR-WS.org (2021)
8. Confalonieri, R., Lucchesi, F., Maffei, G., Solarz, S.C.: A unified framework for managing sex and gender bias in AI models for healthcare. In: Sex and Gender Bias in Technology and Artificial Intelligence. Elsevier, pp. 179–204 (2022)
9. Confalonieri, R., Weyde, T., Besold, T.R., Moscoso del Prado Martín, F.: Using ontologies to enhance human understandability of global post-hoc explanations of black-box models. Artif. Intell. **296** (2021). https://doi.org/10.1016/j.artint.2021.103471
10. Confalonieri, R., Weyde, T., Besold, T.R., del Prado Martín, F.M.: Trepan reloaded: a knowledge-driven approach to explaining black-box models. In: Proceedings of the 24th European Conference on Artificial Intelligence, pp. 2457–2464 (2020). https://doi.org/10.3233/FAIA200378
11. Craven, M.W., Shavlik, J.W.: Extracting tree-structured representations of trained networks. In: NIPS 1995, pp. 24–30. MIT Press (1995)
12. Darwiche, A., Marquis, P.: A knowledge compilation map. J. Artif. Intell. Res. **17**, 229–264 (2002). https://doi.org/10.1613/jair.989
13. Doshi-Velez, F., Kim, B.: Towards a rigorous science of interpretable machine learning (2017)
14. Galliani, P., Kutz, O., Porello, D., Righetti, G., Troquard, N.: On knowledge dependence in weighted description logic. In: Calvanese, D., Iocchi, L. (eds.) GCAI 2019. Proceedings of the 5th Global Conference on Artificial Intelligence, EPiC Series in Computing, Bozen/Bolzano, Italy, 17–19 September 2019, vol. 65, pp. 68–80. EasyChair (2019)

15. Galliani, P., Kutz, O., Troquard, N.: Perceptron operators that count. In: Homola, M., Ryzhikov, V., Schmidt, R.A. (eds.) Proceedings of the 34th International Workshop on Description Logics (DL 2021) part of Bratislava Knowledge September (BAKS 2021), CEUR Workshop Proceedings, Bratislava, Slovakia, 19–22 September 2021, vol. 2954. CEUR-WS.org (2021)
16. Galliani, P., Righetti, G., Kutz, O., Porello, D., Troquard, N.: Perceptron connectives in knowledge representation. In: Keet, C.M., Dumontier, M. (eds.) EKAW 2020. LNCS (LNAI), vol. 12387, pp. 183–193. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-61244-3_13
17. Garcez, A.D., Gori, M., Lamb, L.C., Serafini, L., Spranger, M., Tran, S.N.: Neural-symbolic computing: an effective methodology for principled integration of machine learning and reasoning. IfCoLoG J. Log. Appl. **6**(4), 611–631 (2019)
18. Guidotti, R., Monreale, A., Ruggieri, S., Turini, F., Giannotti, F., Pedreschi, D.: A survey of methods for explaining black box models. ACM Comp. Surv. **51**(5), 1–42 (2018)
19. Hind, M.: Explaining explainable AI. XRDS **25**(3), 16–19 (2019). https://doi.org/10.1145/3313096
20. Huysmans, J., Dejaeger, K., Mues, C., Vanthienen, J., Baesens, B.: An empirical evaluation of the comprehensibility of decision table, tree and rule based predictive models. Decis. Support Syst. **51**(1), 141–154 (2011)
21. Mariotti, E., Alonso, J.M., Confalonieri, R.: A framework for analyzing fairness, accountability, transparency and ethics: a use-case in banking services. In: 30th IEEE International Conference on Fuzzy Systems, FUZZ-IEEE 2021, Luxembourg, 11–14 July 2021, pp. 1–6. IEEE (2021). https://doi.org/10.1109/FUZZ45933.2021.9494481
22. Masolo, C., Porello, D.: Representing concepts by weighted formulas. In: Borgo, S., Hitzler, P., Kutz, O. (eds.) Formal Ontology in Information Systems - Proceedings of the 10th International Conference, FOIS 2018, Frontiers in Artificial Intelligence and Applications, Cape Town, South Africa, 19–21 September 2018, vol. 306, pp. 55–68. IOS Press (2018). https://doi.org/10.3233/978-1-61499-910-2-55
23. Mittelstadt, B., Russell, C., Wachter, S.: Explaining explanations in AI. In: Proceedings of the Conference on Fairness, Accountability, and Transparency - FAT* 2019, pp. 279–288. ACM Press, New York (2019). https://doi.org/10.1145/3287560.3287574
24. Parliament and Council of the European Union: General Data Protection Regulation (2016)
25. Porello, D., Kutz, O., Righetti, G., Troquard, N., Galliani, P., Masolo, C.: A toothful of concepts: towards a theory of weighted concept combination. In: Simkus, M., Weddell, G.E. (eds.) Proceedings of the 32nd International Workshop on Description Logics, CEUR Workshop Proceedings, Oslo, Norway, 18–21 June 2019, vol. 2373. CEUR-WS.org (2019)
26. Ribeiro, M.T., Singh, S., Guestrin, C.: "Why should I trust you?": explaining the predictions of any classifier. In: Proceedings of the 22nd International Conference on Knowledge Discovery and Data Mining, KDD 2016, pp. 1135–1144. ACM (2016)
27. Righetti, G., Masolo, C., Troquard, N., Kutz, O., Porello, D.: Concept combination in weighted logic. In: Sanfilippo, E.M., et al. (eds.) Proceedings of the Joint Ontology Workshops 2021 Episode VII, CEUR Workshop Proceedings, vol. 2969. CEUR-WS.org (2021)
28. Righetti, G., Porello, D., Kutz, O., Troquard, N., Masolo, C.: Pink panthers and toothless tigers: three problems in classification. In: Cangelosi, A., Lieto, A. (eds.)

Proceedings of the 7th International Workshop on Artificial Intelligence and Cognition, CEUR Workshop Proceedings, vol. 2483, pp. 39–53. CEUR-WS.org (2019)
29. Rosch, E., Lloyd, B.B.: Cognition and categorization (1978)
30. Vollmer, H.: Introduction to Circuit Complexity: A Uniform Approach. Springer, Heidelberg (1999). https://doi.org/10.1007/978-3-662-03927-4