Dialogue Explanations for Rule-Based AI Systems

DOI:
10.1007/978-3-031-40878-6_4

Document Version
Accepted author manuscript

Published in:
5th International Workshop on EXplainable and TRAnsparent AI and Multi-Agent Systems (EXTRAAMAS 2023)

Dialogue Explanations for Rule-based AI Systems

Yifan Xu[0000−0003−2303−1531], Joe Collenette[0000−0001−6179−2038], Louise Dennis[0000−0003−1426−1896], and Clare Dixon[0000−0002−4610−9533]

Department of Computer Science, The University of Manchester
{yifan.xu, joe.collenette, louise.dennis, clare.dixon}@manchester.ac.uk

Abstract. The need for AI systems to explain themselves is increasingly recognised as a priority, particularly in domains where incorrect decisions can result in harm and, in the worst cases, death. Explainable Artificial Intelligence (XAI) tries to produce human-understandable explanations for AI decisions. However, most XAI systems prioritize factors such as technical complexities and research-oriented goals over end-user needs, risking information overload. This research attempts to bridge a gap in current understanding and provide insights for assisting users in comprehending the rule-based system's reasoning through dialogue. The hypothesis is that employing *dialogue* as a mechanism can be effective in constructing explanations. A dialogue framework for rule-based AI systems is presented, allowing the system to explain its decisions by engaging in "Why?" and "Why not?" questions and answers. We establish formal properties of this framework and present a small user study with encouraging results that compares dialogue-based explanations with proof trees produced by the AI System.

1 Introduction

Reasoning, the process of synthesising facts and beliefs to make new decisions, is a fundamental component of humans' explanatory mechanisms [11]. Giving the current generation of AI systems human-like capabilities for explaining themselves is challenging because their data-driven nature makes it hard to identify reasoning-like processes. In contrast, in the early days of AI, explainability was regarded as an easy task since most systems were logic-based [26]. Such *Rule-based systems* (RBS) may be learned, and, in particular, there have been recent results in the extraction of decision trees (and rules) from neural networks for the purposes of improved explainability [21,31]. Even in this work, however, the assumption is that once converted to the RBS, the resulting system is inherently explainable. Because when the rule-chaining process of such a system becomes very complex, their explanations are difficult to follow [14].

As a starting point, we focus on explaining hand-crafted RBS, with the aim of extending our learned rules to RBS extracted from machine learning models in the future. The utility of an explanation depends upon the user's context – why they are seeking an explanation. Are they surprised by a recommendation and want to know more? Do they want to challenge a recommendation? In particular,

we have focused on situations where the user's information is different from that possessed by the system and we've used the user's ability to discover this mismatch following the explanatory process as one of our metrics for assessing the utility of the explanation.

We propose a formal framework for dialogues involving two participants (presumed to be a RBS and a user) that specifies allowable utterances (in the form of questions or "one step" explanations) and how each participant's mental model of the other is updated given these utterances. We have implemented this framework together with a simple RBS based on rules around Covid-19 restrictions. To assess our explanation, consider Miller's [16]findings that a good explanation must be short, be selected, and be social, we compared the dialogue system with providing the RBS' deduction tree with encouraging results.

2 Related Work

Early rule-based expert system explanations [22] focused in particular on the explanation framework [5,29,25,19], and the human-computer interface (HCI) through which the explanation was supplied [13,24]. The most sophisticated approaches involved an "intelligent" conversation with the system user that was done in simple terms and using interactive methods [9]. Naturalness was recognised as a condition for a good explanation [12,17] so the social aspect of explanation was known. The user's inquiry is restricted to asking why this information is being requested by the system [5]. However, little progress was made in terms of enabling users to really guide an explanation to a desired outcome, also it becomes challenging to construct a coherent explanation when there are numerous chained rules involved [14].

To solve the issue mentioned above, several dialogue models for explanation have been proposed [27,23]. Walton's shift model for dialogue proposes an explanation and examination dialogue with three stages and two rules governed by the explainee to determine the success of an explanation [28,3]. These models, however, don't appear to have iterative aspects like cyclic dialogues and lack a data-based foundation or validation. Madumal introduced an interaction protocol for interactive explanations by analyzing transcripts from real explanation dialogue datasets [15].

Argumentation, as an important reasoning strategy, has also been incorporated into dialogue models to enhance the explainability of AI systems [26,20,7,2,18]. Walton and Bex [3] utilize argumentation models and dialogue and enable the explainee to question and dispute the provided explanations which are modeled as arguments. This enables the explainee to query and interrogate the provided explanations in order to achieve better comprehension. Although the proposed framework offers a high-level structure for explanation-based conversations, it does not place a strong emphasis on explaining rule-based deductions or using arguments to fully comprehend the beliefs of the other person. Furthermore, there are very few actual human experiments that have been done to evaluate the efficacy of such arguments.

A dialogue framework has been developed to explain the behavior of a system programmed using the BDI (Beliefs-Desires-Intentions) paradigm which has many similarities to RBS [8]. It defines a turn-based system and allows users to ask questions about the reasons behind selecting plans of action within the system, but does not provide a way to explain deductive reasoning (which is our focus). Building upon the foundational works of Dennis and Oren [8], we aim to ensure that the user gains a genuine comprehension of the explanation without overwhelming them with excessive information.

Miller highlights the importance of concise, carefully chosen, and socially relevant explanations [16]. He emphasizes that explanations serve as answers to "why" questions. Similarly, Winikoff also emphasizes the significance of addressing "why" questions when providing explanations [30]. Our dialogue explanations also prioritize addressing both "why" and "why not" questions to generate collaborative explanations.

3 Framework

Our starting point is two "players" (assumed to be some RBS and a user). Each possesses a set of facts (F) and a set of rules (R) and uses these to deduce whether some conclusion (C) is true or false. Deductions are represented as trees. When the players disagree they engage in a dialogue. Each player can ask *why* a particular node in a tree is believed in which case they are informed that it was either an initial fact, or it was deduced from its parent nodes using a rule. A player can also ask *why not* questions. In this case, the other player turns this around and asks the other player why they believe that something does hold. Note we assume that both players reason correctly.

3.1 Proof Trees

We assume:

- A language of terms, $\mathcal{L}$, defined in the standard way (See [10], p. 99).
- A set of labels L which include two special labels: *initial* and *unprovable*.
- A set of initial facts, F (positive literals in $\mathcal{L}$).
- A set of rules, R. A rule is a clause consisting of a non-empty set of literals in $\mathcal{L}$ (the antecedants, A), a consequent, a positive literal $C \in \mathcal{L}$, and a label $l \in L \backslash \{initial, unprovable\}$, written as $l : A \rightarrow C$. We assume that labels in R are unique and that rules that are identical up to the renaming of variables have the same label[1].

We use the notation $pos(A)$ for the set of terms that appear positively in some set of literals, A, and $neg(A)$ for the set of terms that appear negatively in some set of literals A (i.e. if $t \in neg(A)$ then $\neg t \in A$)

[1] We don't need to label rules for our system to work, but labels are a useful convenience when referring to rules.

Definition 1. Proof Tree

A proof tree *is a directed rooted tree written* $\langle N, E\rangle$*, where* N *is a set of nodes of the form* (t, l) *where* $t \in \mathcal{L}$ *is a ground positive literal and* $l \in L$ *is a label.* $E \subseteq N \times N$ *is the set of edges. An edge between two nodes* n_1 *and* n_2 *is written as* $n_1 \mapsto n_2$*.*

We use standard terminology so the **root** of a proof tree is the single node, n such that there is no edge $n' \mapsto n$. The **parent nodes** of a node n are the set of nodes n' such that there exists an edge $n \mapsto n'$. The **parent trees** of a node n are the set of sub-trees with a parent of n as their root.

If (t, l) is the root node of a tree, then we refer to t as the **root term** of the tree.

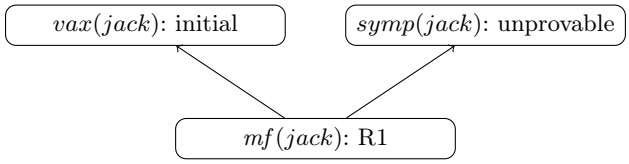

Fig. 1: A Proof Tree showing why Jack can meet his friends using $R1 : \{vax(X), \neg symp(X)\} \rightarrow mf(X)$. R1: You can meet friends if you have been vaccinated and display no symptoms, and the initial fact set $\{vax(jack)\}$ means Jack is vaccinated.

Definition 2. Provable, Unprovable and Undecided in T

If $\langle N, E\rangle = T$ *is a proof tree and* t *is a ground positive literal in* $\mathcal{L}$*. We say:* t *is* **provable** *in* T *iff there exists a node* $(t, l) \in N$ *such that* $l \neq unprovable$*;* t *is* **unprovable** *in* T *iff* $(t, unprovable) \in N$*;* t *is* **undecided** *in* T *iff there is no node* $(t, l) \in N$*.*

Therefore, in Figure 1, if our proof tree is T, then $vax(jack)$ and $mf(jack)$ are both provable in T, $symp(jack)$ is unprovable in T and any other term (e.g., $fever(jack)$) is undecided in T.

Definition 3. Proof Tree for F **and** R

A Proof Tree, T, for a set of facts, F, and rules, R is defined recursively as follows:

- $\langle\{(t, initial)\}, \emptyset\rangle$ is a proof tree for F and R iff $t \in F$
- $\langle\{(t, unprovable)\}, \emptyset\rangle$, is a proof tree for F and R iff no proof tree, T', for F and R exists such that t is provable in T'
- If $E \neq \emptyset$ then a proof tree $T = \langle N, E\rangle$ with root node (t, l) is a proof tree for F and R iff:
  - The parent trees of (t, l) are all proof trees for F and R

- There exists a rule, $l : A \rightarrow C \in R$ and a substitution, θ for the free variables in A and C such that $C\theta = t$ and $t \notin F$, and
  - if (t', l') is a parent of (t, l) in T then either
    - $\exists t_i \in pos(A).\, t_i\theta = t'$ and $l' \neq unprovable$ or,
    - $\exists t_i \in neg(A).\, t_i\theta = t'$ and $l' = unprovable$; and
  - $\forall t' \in A\theta$ there exists a unique label, l' such that (t', l') is a parent node of (t, l) in T.

A proof tree with some statement t at its root (either as a provable or unprovable statement) can be constructed from F and R by standard backward reasoning with negation as a failure as used in logic programming languages such as Prolog [6]. From this point, we will stop referring to substitutions, θ, etc. for reasons of readability and present our theory only for the case where rules contain no free variables. Our proofs can be adapted straightforwardly to the more general case.

Note that our proof trees are essentially SLDNF-trees (Selective Linear Definite Clause with Negation as Failure) from logic programming [1] extended with rule labels. We assume that our facts and rules are such that SLDNF-resolution is complete – for instance that they represent an acyclic program [4].

4 Dialogues

We formalise the idea of a disagreement between two RBSs as a difference in their initial facts or rules. The purpose of a dialogue will be to identify at least one such difference from a starting point where one RBS has deduced some fact to be the case and the other has deduced that it is not the case.

Definition 4. Deduction

We formalise a deduction as a tuple $\mathcal{D}(F, R, \mathcal{T})$ where F is a set of initial facts, R a set of rules and $\mathcal{T}$ is a set of proof trees for F and R. We will refer to $\mathcal{T}$ as the deduction trees.

Our problem is: given two deductions $\mathcal{D}(F_1, R_1, \mathcal{T}_1) \neq \mathcal{D}(F_2, R_2, \mathcal{T}_2)$ which disagree about some deduced fact can we identify the disagreement in terms of their initial facts or rules? More formally if there exists a $T_1 \in \mathcal{T}_1$ (resp. $T_2 \in \mathcal{T}_2$) which has some provable root term t that is unprovable in at least one $T_2 \in \mathcal{T}_2$ (resp. $T_1 \in \mathcal{T}_1$), can we identify at least one fact, t' such that $t' \in F_1$ and $t' \notin F_2$ (or vice versa) or at least one rule r such that $r \in R_1$ and $r \notin R_2$ (or vice versa).

We can trivially identify the differences if we have full access to F_1, F_2, R_1, R_2, etc., so we assume that this is not the case but take the viewpoint of one of the parties making the deduction – so either we have access to F_1 and R_1 but not F_2 and R_2 or vice versa. We do assume that rules with the same label in R_1 and R_2 are identical up to the renaming of variables – i.e., if $l : A_1 \rightarrow C_1 \in R_1$ and $l : A_2 \rightarrow C_2 \in R_2$ then $A_1 = A_2$ and $C_1 = C_2$. This means we can use rule labels without loss of generality as proxies for the rules themselves rather than having to match antecedents and consequents.

Definition 5. Provable/Unprovable for Deductions
Given a deduction $D = \mathcal{D}(F, R, \mathcal{T})$ *we say a term* t *is* provable in D *if* t *is provable in some* $T \in \mathcal{T}$ *and that* t *is* unprovable in D *if* t *is unprovable in some* $T \in \mathcal{T}$.

To simplify our proofs we introduce a completeness property for deductions. This specifies that if some term is unprovable in the deduction then the deduction contains the evidence for why it is unprovable – in particular it contains proof trees for all the antecedents of any rule with the term as its consequent. These can then be inspected to understand why that rule did not apply.

Definition 6. Complete Deduction *We say that a deduction* $D = \mathcal{D}(F, R, \mathcal{T})$ *is* complete *if, for any* t *that is unprovable in* D*, if there is a rule* $l : A \rightarrow t \in R$ *then all terms* $t' \in pos(A) \cup neg(A)$ *are either provable or unprovable in* D.

In practice, we can generate necessary additional proof trees on the fly during a dialogue and add them to deductions in order to make them complete. But this process complicates the presentation here so we assume our dialogue starts out with all the proof information it needs to justify an agent's conclusions.

A dialogue is a sequence of moves taken by two players. P_1 knows all the information in $D_1 = \mathcal{D}(F_1, R_1, \mathcal{T}_1)$ while P_2 knows all the information in $D_2 = \mathcal{D}(F_2, R_2, \mathcal{T}_2)$.

We will extend our simple example from Figure 1 into a scenario involving two players, P_1 and P_2, that will be used to illustrate our dialogue definition. There's already one rule ($R1$), and we introduce another rule:

$$R2 : \{\neg tns(X)\} \rightarrow symp(X)$$

(if X has lost their sense of taste and smell (tns) then they have symptoms).

Scenario:

- $F_1 = \{vax(jack), tns(jack)\}$ while $F_2 = \{vax(jack)\}$. So the difference between our two players is that one is aware that Jack retains his sense of taste and smell while the other is not.
- Both players have rules $R1$ and $R2$ in their rule set.
  - $R1 : \{vax(X), \neg symp(X)\} \rightarrow mf(X)$
  - $R2 : \{\neg tns(X)\} \rightarrow symp(X)$
- P_1 has deduced that Jack can meet his friends and P_2 has deduced he can not. We start the dialogue with complete deductions.
  - $D_1 = \mathcal{D}(F_1, R_1, \mathcal{T}_1)$ where $\mathcal{T}_1$ contains the proof tree shown in Figure 1 and a proof tree consisting of a single node $(tns(jack), initial)$ (This is because, for the deduction to be complete, we need the antecedents of $R2$ to be either provable or unprovable in D_1).
  - $D_2 = \mathcal{D}(F_2, R_2, \mathcal{T}_2)$ and $\mathcal{T}_2$ contains a proof tree consisting of the single node $(mf(jack), unprovable)$. For deduction to be complete the antecedents for $R1$ must be provable or unprovable in D_2. Therefore $\mathcal{T}_2$ also contains the proof tree shown in Fig. 2 and a proof tree consisting of the single node $(vax(jack), initial)$.

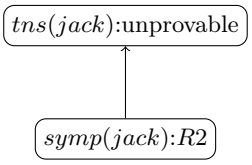

Fig. 2: P_2's proof tree for why Jack has symptoms.

Note that $mf(jack)$ is provable in D_1 and unprovable in D_2; $symp(jack)$ is provable in D_2 and unprovable in D_1; $tns(jack)$ is unprovable in D_2 and provable in D_1; and $vax(jack)$ is provable in both D_1 and D_2.

The two players gradually build up a *mental model* of how the other player has reasoned. This model consists of four sets OB_{ij}, OF_{ij}, OD_{ij} and OR_{ij}:

- OB_{ij} consists of terms t that P_i has established that P_j believes. We refer to OB_{ij} as the *opponent belief set*.
- OF_{ij} consists of terms t that P_i has established that P_j had as an initial fact. Note that $OF_{ij} \subseteq OB_{ij}$. We refer to OF_{ij} as the *opponent fact set*.
- OD_{ij} consists of terms t that P_i has established that P_j does not believe. We refer to OD_{ij} as the *opponent disbelief set*.
- OR_{ij} consists of labels l that P_i has established label one of P_j's rules. We refer to OR_{ij} as the *opponent rule set*.

There are seven possible statements that can be made in the course of a dialogue:

1. $df(t, i, j)$ (the two players have different initial facts) – $t \in F_i$ and $t \notin F_j$.
2. $dr(l : A \to C, i, j)$ (the two players have different rules) – $l : A \to C \in R_i$ and $l : A \to C \notin R_j$.
3. $initial(t)$ – t is an initial fact for the player.
4. $l : A \to t$ – the player deduced t from the terms in A using the rule labelled l
5. $why(t)$ - why do you believe t?
6. $whynot(t)$ – why don't you believe t?
7. $pass$ – the dialogue participant has no question to ask and skips its turn.

The first two statements terminate the dialogue.

Definition 7. Player State *The* state *of P_i at statement k in a dialogue with P_j is $S_k^i = \langle D_i, OB_{ij}, OF_{ij}, OD_{ij}, OR_{ij} \rangle$ where D_i is a deduction, and $OB_{ij}, OF_{ij}, OD_{ij}, OR_{ij}$ are P_i's opponent belief set, fact, set, disbelief set and rule set respectively.*

The initial state of the two players is one where the only thing they know is that they disagree on some term t. So their opponent's belief sets etc., are empty.

Definition 8. Initial Player State *The initial state of P_i is either $\langle \mathcal{D}(F_i, R_i, \mathcal{T}_i), \{t\}, \emptyset, \emptyset, \emptyset \rangle$ where t is unprovable in $\mathcal{T}_i$ or $\langle \mathcal{D}(F_i, R_i, \mathcal{T}_i), \emptyset, \emptyset, \{t\}, \emptyset \rangle$ where t is the root term of some $T_i \in \mathcal{T}_i$.*

Definition 9. Dialogue State *S_k is the state of the dialogue after the utterance of the kth statement. It consists of the two-player states, the last dialogue statement, stmt, and whose turn it is, P_i. $S_k = \langle S_k^1, S_k^2, stmt, P_i \rangle$*

A dialogue is a sequence of dialogue states $S_0, \ldots, S_n$. The starting point for the dialogue is the disagreement over the term t in Definition 8. Without loss of generality, we assume this is provable in $\mathcal{T}_1$ and unprovable in $\mathcal{T}_2$. Therefore, $S_0 = \langle S_0^1, S_0^2, stmt_0, P_i \rangle$ where $S_0^1 = \langle D_1, \emptyset, \emptyset, \{t\}, \emptyset \rangle$, $S_0^2 = \langle D_2, \{t\}, \emptyset, \emptyset, \emptyset \rangle$, and either $P_i = P_1$ and $stmt_0 = why(t)$ (P_2 started the dialogue by asking P_1 why they believe t and it is now P_1's turn) or $P_i = P_2$ and $stmt_0 = whynot(t)$ (P_1 started the dialogue by asking P_2 why they don't believe t).

Suppose $S_k = \langle S_k^1, S_k^2, stmt_k, P_i \rangle$ is the state of a dialogue at utterance k and $S_{k+1} = \langle S_{k+1}^1, S_{k+1}^2, stmt_{k+1}, P_j \rangle$ is the next state. We define what it means for S_{k+1} to be a legal next state. S_{k+1}^i defines how each player has updated their mental model of the other in response to $stmt_k$ and $stmt_{k+1}$ is the next utterance.

First, we consider how the two players update their state. P_j ($j \neq i$) does not alter their state – they uttered the last statement and have not learned any new information. So $S_{k+1}^j = S_k^j$.

P_i, on the other hand has gained information from P_j's utterance and so their state changes. Before the utterance their state was $S_k^i = \langle D_i, OB_{ij}, OF_{ij}, OD_{ij}, OR_{ij} \rangle$. We provide four rules below that govern the state and can be updated.

Upd.1 If $stmt_k = initial(t)$ then $S_{k+1}^i = \langle D_i, OB_{ij} \cup \{t\}, OF_{ij} \cup \{t\}, OD_{ij}, OR_{ij} \rangle$ (P_i adds t to the things P_j believes and P_j's initial facts).

Upd.2 If $stmt_k = l$, $l : A \rightarrow C \in R_i$ then $S_{k+1}^i = \langle \mathcal{D}(F_i, R_i, \mathcal{T}), OB_{ij} \cup pos(A), OF_{ij}, OD_{ij} \cup neg(A), OR_{ij} \cup \{l\} \rangle$ (P_i adds all the positive literals in A to OB_{ij} (these are things the other player believes) and all the negative literals in A to OD_{ij} (these are all the things the other player does not believe), and adds l to OR_{ij}).

Upd.3 If $stmt_k = why(t)$, $D_i = \mathcal{D}(F_i, R_i, \mathcal{T})$, and t is provable in $\mathcal{T}$ then $S_{k+1}^i = \langle \mathcal{D}(F_i, R_i, \mathcal{T}), OB_{ij}, OF_{ij}, OD_{ij} \cup \{t\}, OR_{ij} \rangle$ (P_i adds t to OD_{ij} (the other player doesn't believe t)).

Upd.4 If $stmt_k = whynot(t)$, and t is unprovable in D_i then $S_{k+1}^i = \langle D_i, OB_{ij} \cup \{t\}, OF_{ij}, OD_{ij}, OR_{ij} \rangle$ (P_i adds t to OB_{ij} (the other player believes t)).

Note: The states where P_i is asked either why it believes something it does not, or why it does not believe something that it does should not occur in a legal dialogue and so these have been omitted. For the purposes of our theoretical results, we assume that if this does occur the dialogue terminates with an error and no next state is generated. We will prove that error states can not arise as corollaries to lemmas 4 and 5.

We now consider the utterances P_i can make – possible values for $stmt_{k+1}$. In some dialogue states, there may be several possible utterances.

Utt.1 $stmt_{k+1} = initial(t)$ is legal iff $stmt_k = why(t)$ and $t \in F_i$

Utt.2 $stmt_{k+1} = l$ is legal iff: $l : A \to C \in R_i$; $stmt_k = why(t)$; $t \notin F_i$; and there exists a proof tree $\langle N, E\rangle \in \mathcal{T}_{i_k}$ such that $(t, l) \in N$.

Utt.3 $stmt_{k+1} = whynot(t)$ is legal iff: $\forall t'.stmt_k \neq why(t') \wedge stmt_k \neq whynot(t')$ (you can not answer a question by asking why not); $\forall l.l \leq k \to stmt_l \neq whynot(t)$ (this question has not been asked before); t is provable for D_i; and $t \in OD_{ij}$. P_i identifies a term t that it believes and it has established the other doesn't and asks why not.

Utt.4 $stmt_{k+1} = why(t)$ is legal iff either $stmt_k = whynot(t)$; or $\forall t'.\ stmt_k \neq why(t') \wedge stmt_k \neq whynot(t')$ (you can not answer a question by asking $why(t)$ unless that question was $whynot(t)$); $\forall l.l \leq k \to stmt_l \neq why(t)$ (this question has not been asked before); t is unprovable for D_i; and $t \in OB_{ij}$ P_i identifies a term t that it does not believe and it has established the other does and asks why.

Utt.5 $stmt_{k+1} = df(t, j, i)$ is legal iff $t \in OF_{ij}$ and $t \notin F_i$

Utt.6 $stmt_{k+1} = df(t, i, j)$ is legal iff $t \in OD_{ij}$ and $t \in F_i$

Utt.7 $stmt_{k+1} = dr(l, j, i)$ is legal iff $l \in OR_{ij}$ and there is no rule $l : A \to C \in R_i$

Utt.8 $stmt_{k+1} = pass$ is legal iff no other utterance is legal and $stmt_k \neq pass$.

Finally, the player whose turn it is is switched.

Figure 3 shows an example dialogue for our scenario. We show the opponent's belief, fact, disbelief, and rule sets for each player as they are built up, as well as the statement uttered and whose turn it is next. We also comment on the changes with reference to the updates and utterances defined by the dialogue framework.

5 Theoretical Results

We demonstrate that error states in dialogues cannot arise, that opposing belief sets etc., are correct representations of the other player's deductions, and that the debate process ends when a discrepancy is discovered.

We establish via a set of lemmas that the assumptions made by the update process are correct (for instance in Lemma 1 that if one player has uttered $initial(t)$ then t is indeed an initial fact for that player).

Lemma 1 (Statements about initial facts are truthful). *If the current dialogue state is $\langle S_k^1, S_k^2, initial(t), P_i\rangle$, $i \neq j$ and $D_j = \mathcal{D}(F_j, R_j, \mathcal{T}_j)$ then $t \in F_j$ and is provable for D_j.*

Lemma 2 (Statements about the use of rules are truthful). *If the current dialogue state is $\langle S_k^1, S_k^2, l : A \to t, P_i\rangle$, $i \neq j$ and $D_j = \mathcal{D}(F_j, R_j, \mathcal{T}_j)$ is P_j's deduction then there exists a proof tree, $T_j \in \mathcal{T}_j$ such that (t, l) is a node in T_j; $l : A \to t \in R_j$, for all $t \in pos(A)$, t is provable in D_j; and for all $t \in neg(A)$, t is unprovable D_j.*

k	P_i State				$stmt_k$	j
	OB_{ij}	OF_{ij}	OD_{ij}	OR_{ij}		
0	$\emptyset$	$\emptyset$	$\{mf(jack)\}$	$\emptyset$	$whynot(mf(jack))$	2

P_1 has asked why P_2 thinks Jack can't meet friends

1	$\{mf(jack)\}$	$\emptyset$	$\emptyset$	$\emptyset$	$why(mf(jack))$	1

Upd.4 applies but P_2 already knows P_1 thinks Jack can meet friends as our initial condition; P_2 asks why P_1 thinks Jack can meet friends (**Utt.3**)

2	$\emptyset$	$\emptyset$	$\{mf(jack)\}$	$\emptyset$	$R1$	2

Upd.3 applies but makes no change; P_1 responds with the Rule it used (**Utt.2**)

3	$\{mf(jack),$ $vax(jack)\}$	$\emptyset$	$\{symp(jack)\}$	$\{R1\}$	$whynot(symp(jack))$	1

Upd.2 applies and changes P_2's state. P_2 asks why P_1 does not believe Jack has symptoms – note they can't ask why P_1 believes Jack has been vaccinated because they could only ask this if they disagreed with this belief.

4	$\{symp(jack)\}$	$\emptyset$	$\{mf(jack)\}$	$\emptyset$	$why(symp(jack)$	2

Upd.4 applies and changes P_1's state; P_1 asks why P_2 believes Jack has symptoms (**Utt.3**)

5	$\{mf(jack),$ $vax(jack)\}$	$\emptyset$	$\{symp(jack)\}$	$\{R1\}$	$R2$	1

Upd.3 applies but does not change P_2's state. P_2 responds with the rule it used.

6	$\{symp(jack)\}$	$\emptyset$	$\{mf(jack),$ $tns(jack)\}$	$\{R2\}$	$whynot(tns(jack))$	2

Upd.2 applies and changes P_1's state. P_1 asks why P_2 does not believe Jack has a sense of taste and smell.

7	$\{mf(jack),$ $vax(jack),$ $tns(jack)\}$	$\emptyset$	$\{symp(jack)\}$	$\{R1\}$	$why(tns(jack))$	1

Upd.4 applies but makes no change to P_2's state; P_2 asks why P_1 believes Jack has a sense of taste and smell.

8	$\{symp(jack)\}$	$\emptyset$	$\{mf(jack),$ $tns(jack)\}$	$\{R2\}$	$initial(tns(jack))$	2

Upd.3 applies but does not change P_1's state; P_1 replies that this is an initial fact.

9	$\{mf(jack),$ $vax(jack),$ $tns(jack)\}$	$\{tns(jack)\}$	$\{symp(jack)\}$	$\{R1\}$	$df(tns(jack), 1, 2)$	1

Upd.1 applies and changes P_2's state; P_2 replies announcing it has found a different fact and terminating the dialogue.

Fig. 3: Sample Dialogue for our scenario showing the current player's opponent belief, fact, disbelief and rule sets and the statement the player has uttered.

Lemma 3 (A player only asks the other "why not" about statements it believes to be true). *If the current dialogue state is* $\langle S^1_{stmt}, S^2_{stmt}, whynot(t), P_i\rangle$, $i \neq j$ *and* D_j *is* P_j*'s deduction then* t *is provable in* D_j.

Lemmas 1, 2 and 3 follow trivially from the rules for legal utterances in dialogue. The equivalent to Lemma 3 for $why(t)$ is Lemma 6 but we need a few other results before we can prove this, in particular, we need to know that the dialogue participants' mental models of each other are correct.

Dialogue Mental Models are correct We establish that $t \in OF_{ij}$ iff $t \in F_j$ (i.e., P_i only decides P_j has t as an initial fact if P_j does indeed have t as an initial fact). The same for OB_{ij}, OD_{ij} etc. As a result, we can also show that the error states (where a participant is asked $why(t)$ for some term t they do not believe or $whynot(t)$ for some t they do believe) never occur.

Theorem 1 (The opponent fact set is correct). *Given two players P_i and P_j in a legal dialogue, if $\mathcal{D}(F_j, R_j, \mathcal{T}_j)$ is P_j's deduction and OF_{ij} is P_i's opponent fact set, $OF_{ij} \subseteq F_j$.*

Proof Sketch The proof follows by induction on the size of OF_{ij} using Lemma 1

Theorem 2 (The opponent belief set is correct). *Given two players P_i and P_j in a legal dialogue where D_j is P_j's deduction and OB_{ij} is P_i's opponent belief set, then all terms $t \in OB_{ij}$ are provable in D_j.*

Proof Sketch The proof follows by induction on the size of OB_{ij} using Lemmas 1, 2 and 3.

Lemma 4 (A player is only asked why about things it believes to be true). *If the dialogue state is $\langle S_k^1, S_k^2, why(t), P_i\rangle$ and D_i is P_i's deduction then t is provable in D_i.*

Proof. This holds in the initial state. Otherwise, $why(t)$ has been uttered because $t \in OB_{ji}$ (**Utt.4**) and this follows from Theorem 2 or $why(t)$ has been uttered in response to $whynot(t)$ (**Utt.5**) and this follows from Lemma 3.

Corollary *If the dialogue state is $\langle S_k^1, S_k^2, why(t), P_i\rangle$ then the error state does not arise.*

Theorem 3 (The opponent disbelief set is correct). *Given two players P_i and P_j in a legal dialogue where D_j is P_j's deduction and OD_{ij} is P_i's opponent disbelief set, then all terms $t \in OD_{ij}$ are unprovable in D_j.*

Proof Sketch The proof follows by induction on the size of OD_{ij} using Lemmas 2 and 4.

Lemma 5 (A player is only asked why not about things it does not believe to be true). *If the dialogue state is $\langle S_k^1, S_k^2, whynot(t), P_i\rangle$ and D_i is P_i's deduction then t is unprovable for P_i.*

Proof. P_j can only ask $whynot(t)$ if $t \in OD_{ji}$ (**Utt.3**) so t is unprovable in D_i by Theorem 3.

Corollary *If the dialogue state is* $\langle S_k^1, S_k^2, whynot(t), P_i\rangle$ *then the error state doesn't arise.*

Lemma 6 (A player only asks the other player why about things it believes are not the case). *If the dialogue state is* $\langle S_k^1, S_k^2, why(t), P_i\rangle$ $(i \neq j)$ *then* t *is unprovable for* P_j*.*

Proof. P_j can only ask $why(t)$ if either a) P_i asked $whynot(t)$ in which case t is unprovable for D_j by Lemma 5; or b) t is unprovable for D_j (**Utt.4**) .

Theorem 4 (The opponent rule set is correct). *Given* P_i *and* P_j *in a legal dialogue where* $\mathcal{D}(F_j, R_j, \mathcal{T}_j)$ *is* P_j*'s deduction and* OR_{ij} *is* P_i*'s opponent rule set, then* $\forall l.l \in OR_{ij}.\exists A, C.\, l : A \rightarrow C \in R_j$

Proof Sketch The proof follows by induction on the size of OR_{ij} using Lemma 2 and the definition of proof trees.

5.1 Termination

Theorem 5. *Let* $D_1 = \mathcal{D}(F_1, R_1, \mathcal{T}_1)$ *and* $D_2 = \mathcal{D}(F_2, R_2, \mathcal{T}_2)$ *be two complete deductions. If* $\mathcal{T}_1$ *and* $\mathcal{T}_2$ *contain a finite number of finite proof trees then any dialogue starting from* D_1 *and* D_2 *terminates.*

Proof Sketch By assumption, there are only a finite number of terms in $\mathcal{T}_1$ and $\mathcal{T}_2$. Therefore $why(t)$ can only be asked a finite number of times. The number of times all other utterances can be made depends upon how many times $why(t)$ is asked. Therefore the dialogue terminates.

Note this means that dialogues only terminate if complete deductions can be created from the attempt to prove or disprove some term t and this depends on the facts, rules, and t. However many sets of facts and rules have this property for given terms.

In order to show that when dialogues terminate a disagreement between the facts or rules of the two players has been found, we need to show that it is always possible for a player to ask questions about terms in OB_{ij} or OD_{ij} which requires these terms to be provable or unprovable in that player's deduction (because of the conditions on **Utt.3** and **Utt.4**). We establish this in two lemmas whose proofs rely on our completeness property for deductions.

Lemma 7. *Given two dialogue participants* P_i *and* P_j *where* D_i *is* P_i*'s deduction and* OB_{ij} *is* i*'s opponent belief set, then all terms* $t \in OB_{ij}$ *are either provable or unprovable in* D_i*.*

Lemma 8. *Given two dialogue participants* P_i *and* P_j *where* D_i *is* P_i*'s deduction and* OD_{ij} *is* i*'s opponent disbelief set, then all terms* $t \in OD_{ij}$ *are either provable or unprovable in* D_i*.*

Proof Sketch The proofs for both these lemmas proceed by induction on the size of OB_{ij} (resp. OD_{ij}), noting that the property holds at the start of the dialogue and exploiting Theorems 2 and 3 and the completeness of deductions together with Lemma 6 in the step case.

Having established this we then introduce the concept of a *disagreement tree* in order to prove that all dialogues terminate with a statement that either facts or rules are different.

Definition 10. *A* disagreement tree *is a tree that reveals the inference processes behind the disagreements between two dialogue participants. Every node in the tree is a tuple* $\langle t, i, lbl \rangle$ *where* t *is a term that is provable for one dialogue participant and unprovable for the other;* i *is the participant for which the term is provable, and* lbl *is either initial (meaning* $t \in F_i$*),* l^- *(meaning* t *was deduced by* i *using rule* l *and rule* l *is not in the rule set for the other participant) or* l^+ *(meaning* t *was deduced by* i *using rule* l *and rule* l *is in the rule set for the other participant). Nodes labeled initial or* l^- *are leaf nodes. Nodes labeled* l^+ *have child nodes consisting of all terms in* $pos(a)$ *which are provable for* i *and not for* j *and all terms in* $neg(a)$ *which are provable for* j *and not for* i*.*

Note that all nodes l^+ must have at least one child node. Figure 4 shows the disagreement tree for our scenario. The two players disagree on the truth of $mf(jack)$ which P_1 has deduced using $R1$ but which P_2 could not deduce because P_1 and P_2 disagree on the truth of $symp(jack)$ and so on.

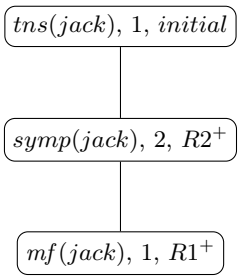

Fig. 4: The Disagreement Tree for Scenario 1.

Lemma 9. *Consider two players in a legal dialogue and a disagreement tree,* DT*, which has the initial disagreement term as its root. Let* NT *be the set of node terms closest to the root of* DT *(there may be several such terms since this is a tree) about which* $why(t)$ *has not been asked.* $\forall t \in NT$ *the dialogue will continue deterministically until* t *is in the belief or disbelief set for at least one player.*

Proof Sketch The proof observes that $why(t)$ will have been asked for the parents of each of these nodes and so a player either has or will, respond with **Utt.1** or **Utt.2** which has or will trigger an appropriate update in a player's state.

Theorem 6. *If the kth state in a legal sequence of dialogue states is* $\langle S_k^1, S_k^2, s, P_i \rangle$ *and* $s \neq df(t,i,j)$, $s \neq df(t,j,i)$ *and* $s \neq dr(l,i,j)$ *then there is a legal next dialogue state.*

Proof Sketch We use Lemmas 7, 8 and 9 to show that one player exists who can ask $why(t)$ if it is their turn and they are not required by the framework to make some other utterance. If the current player is not capable of asking $why(t)$, then they can utter *pass*, and the other player will be able to respond.

Corollary *If a dialogue terminates the last statement is:* $df(t,i,j)$, $df(t,j,i)$, $dr(l,i,j)$ *or* $dr(l,j,i)$.

6 Implementation

We applied our framework to an RBS that functions as a Covid Advice System (CAS) implemented in Prolog. This consists of a simple backward-chaining rule-based system with sets of example rules and facts based on Covid-19 restrictions paired with an implementation of the dialogue framework. The dialogue framework implementation tracks both participants' dialogue states and allows the human user to choose between legal next utterances. As a result, a theoretically legal dialogue can be generated, even where the human is not sure of the legal moves, or may not be reasoning correctly with their facts and rules.

Our implementation differs slightly from the theory in that dialogues did not start with complete deductions, instead, one participant starts with a deduction that contains a single proof tree consisting only of an unprovable node. Additional proof trees were generated on-the-fly during the dialogue as needed.

We present an example of a dialogue in our system. In this example both players have a rule that says two people can meet if a) they are both vaccinated, b) neither of them has been "pinged" by a contact tracing app and c) neither has symptoms. Harry and Sara wish to meet but the CAS is unaware that Harry has been vaccinated and so states that they may not. The user thinks Harry and Sara should be able to meet and so a dialogue starts with a why not question from the user. The dialogue system responds on behalf of the CAS with a why question using **Utt.4** and displays the possible legal user responses (Figure 5).

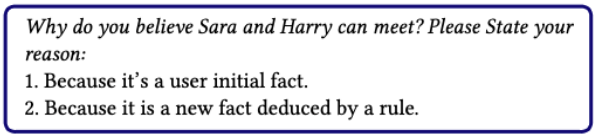

Fig. 5: The computer asks why

The user selects 2 because they've used a rule. The dialogue system has stored the rules provided to the test participants so it offers a choice of these rules (Figure 6). The user selects rule 1. The dialogue system updates the CAS

PLEASE SELECT A RULE NUMBER FROM YOUR RULES:
1. If both X and Y are vaccinated, and none of them have been pinged (close contact with someone who has Covid-19), and none of them have symptoms, then X and Y can meet.
2. If X does not have taste or smell, then X has symptoms.
3. If X has a fever, then X has symptoms.
4. If X has a cough, then X has symptoms.

Fig. 6: The user is offered a choice of rules

Why do you believe Harry is vaccinated? Please state your reason:
1. Because it's a user initial fact.
2. Because it is a new fact deduced by a rule.

Fig. 7: Why does the user believe Harry is vaccinated?

	Ex1		**Ex2**		**Ex3**		**Ex4**		**Ex5**		**Ex6**	
Type	Tree	Dlog.	Tree	Dlog.	Tree	Dlog.	Tree	Dlog.	Tree	Dlog.	Tree	Dlog.
Ease	0.5	2.5	2.25	2.5	1.5	3	0.25	2.75	2.5	2.75	3	3
Helpful	0.25	2.5	2	2	1.75	2.5	0.25	2.25	2.5	2	2.75	2.25
Correct	0	100%	100%	100%	75%	75%	0	75%	100%	75%	100%	100%

Table 1: Our results are broken up by scenario. Each participant marked their explanation on a scale of 0-4 for how easy it was to understand and how helpful they found it - we show the average mark for each explanation style. Additionally, we show what percentage of users correctly identified the difference between their facts and rules and that of the CAS.

mental model of the user and consults the system's proof trees for a mismatch. In this case, it identifies that $vaccinated(harry)$ is unprovable for the CAS. The dialogue system asks why the user believes this (Figure 7). The user selects 1. The dialogue system then terminates announcing that a difference has been found.

7 User Evaluation

The purpose of our user evaluation was to test our hypothesis that dialogue is a useful mechanism for building explanations. The proof trees generated by the deductive process were used as an alternative explanation for comparison. We created six scenarios in which the CAS and the User were given different sets of facts and rules (differing either by one fact or by one rule) and the CAS presented a conclusion which the user should not be able to derive (if the user reasoned correctly). The user was then either shown the proof tree generated

by the CAS or allowed to participate in a dialogue. Our expectation was that dialogue explanations would have an advantage firstly in situations where the CAS deduced something was unprovable (and so produced a proof tree consisting of a single unprovable node) and secondly, as proof trees grew beyond a certain size.

Our study comprised 24 volunteers from the Department of Computer Science. Each participant was presented with two scenarios (one where they viewed a proof tree and one where they could use the dialogue system). Each scenario was completed by the same number of participants, and followed by a short questionnaire. We summarise the features of the six scenarios in Table 2 – as can be seen, two of the examples feature trees consisting of a single unprovable node, while the others have trees of varying, though modest, sizes.

Ex.	**Nodes**	**Unprovable Nodes**	**Cause**
1	1	1	CAS missing Fact
2	14	4	User missing Fact
3	18	4	User missing Rule
4	1	1	CAS missing Rule
5	7	1	CAS missing Fact
6	6	3	CAS and User have different Rules

Table 2: Our examples, showing how many nodes the initial proof tree contains, how many of those nodes are unprovable and the cause of the disagreement between user and CAS

Out of 24 responses, 20 preferred the dialogue explanation, and 18 found the dialogue explanation easy. Table 1 shows a breakdown of our results by example. As can be seen, the dialogue explanations have a clear advantage where no meaningful tree was provided (Scenarios 1 and 4) while there is not much to tell between the two explanation styles in most other cases. Scenario 3, with the largest number of nodes, suggests that the dialogue explanation was beginning to outperform the tree in terms of ease of use and perceived helpfulness, but the sample size is too small (4 people) to draw strong conclusions. Classifying whether a user had correctly identified the difference proved more challenging than we expected. We allowed freeform answers to the question "What do you think the difference was between your information and the computer information?" and in some cases these answers were very minimal (e.g., "There was a different rule") and in some cases, it is difficult to decide whether or not they should count as correct (e.g., in scenario 2 one respondent correctly identified that they did not possess a rule, but stated the rule's antecedents incorrectly). We allowed minimal but correct answers to count as correct but did not allow

other mistakes to count as correct. For Scenario 6 we counted as correct both answers which noted that they had slightly different rules for deducing whether someone was required to get a Covid test and answers which noted that the user did not have the rule the computer was using. Future study will define the rules in a way that is easier for general users to understand because in this experiment the rules provided to the user require a high cognitive load and consider the user with different background.

8 Discussion

We have proposed a dialogue approach to explain the reasoning in systems where derivations are represented as trees, typical of rule-based AI systems. A dialogue system assumes that an explanation is a collaborative process in which the system determines what information it is that the user wants. We have established some theoretical properties of the dialogue framework and performed a small user study.

The study shows a clear advantage for the dialogue process where no meaningful proof tree can be presented. There is some evidence that, as the amount of information in the proof tree increases, the dialogue explanation becomes more useful, and in further work, we intend to extend our study with larger scenarios. We also intend to examine how our explanations could be adapted to explore "what-if" scenarios which would allow a dialogue to progress beyond identifying a source of disagreement to exploring whether eliminating that disagreement would change the system's conclusion, and to evaluate whether dialogue is a useful explanatory process when applied to RBS extracted from statistical models as in [21]. In the future, we'll take into account the agreed-upon situation in which the user can request further information without causing a disagreement.

Acknowledgements: This work is supported by EPSRC, through EP/W01081X (*Computational Agent Responsibility*).

Data Access Statement The code and data supporting the findings reported in this paper are available for open access at https://github.com/xyfLily/Rule-based-system (Code) and https://doi.org/10.6084/m9.figshare.22220494.v3 (User Evaluation).

Ethical Approval We performed a light-touch ethical review for the user evaluation, using a tool provided by our university. This tool advised that since the only personal data gathered was names on consent forms and these were stored in a locked cabinet separate from the rest of the gathered data, further ethical approval was not required.

References

1. Apt, K.R., Van Emden, M.H.: Contributions to the theory of logic programming. Journal of the ACM (JACM) **29**(3), 841–862 (1982)
2. Arioua, A., Tamani, N., Croitoru, M.: Query answering explanation in inconsistent datalog+/- knowledge bases. In: Database and Expert Systems Applications. pp. 203–219. Springer (2015)
3. Bex, F., Walton, D.: Combining explanation and argumentation in dialogue. Argument & Computation **7**(1), 55–68 (2016)
4. Cavedon, L., Lloyd, J.: A completeness theorem for SLDNF resolution. The Journal of Logic Programming **7**(3), 177–191 (1989), https://www.sciencedirect.com/science/article/pii/0743106689900204
5. Clancey, W.J.: The epistemology of a rule-based expert system—a framework for explanation. Artificial intelligence **20**(3), 215–251 (1983)
6. Clocksin, W.F., Mellish, C.S.: Programming in Prolog. Springer, 5 edn. (2003). https://doi.org/10.1007/978-3-642-55481-0
7. Cocarascu, O., Stylianou, A., Čyras, K., Toni, F.: Data-empowered argumentation for dialectically explainable predictions. In: ECAI 2020, pp. 2449–2456. IOS Press (2020)
8. Dennis, L.A., Oren, N.: Explaining BDI agent behaviour through dialogue. In: Proc. of the 20th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2021). International Foundation for Autonomous Agents and Multiagent Systems (IFAAMAS) (2021)
9. Fiedler, A.: Dialog-driven adaptation of explanations of proofs. In: International Joint Conference on Artificial Intelligence. vol. 17, pp. 1295–1300. Citeseer (2001)
10. Huth, M., Ryan, M.: Logic in Computer Science: Modellling and Reasoning about Systems. Cambridge University Press (2004)
11. Johnson-Laird, P.N.: Mental models in cognitive science. Cognitive science **4**(1), 71–115 (1980)
12. Kass, R., Finin, T., et al.: The need for user models in generating expert system explanations. International Journal of Expert Systems **1**(4) (1988)
13. Lacave, C., Díez, F.J.: A review of explanation methods for bayesian networks. The Knowledge Engineering Review **17**(2), 107–127 (2002)
14. Lacave, C., Diez, F.J.: A review of explanation methods for heuristic expert systems. The Knowledge Engineering Review **19**(2), 133–146 (2004)
15. Madumal, P., Miller, T., Sonenberg, L., Vetere, F.: A grounded interaction protocol for explainable artificial intelligence. In: Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent Systems. pp. 1033–1041 (2019)
16. Miller, T., Howe, P., Sonenberg, L.: Explainable ai: Beware of inmates running the asylum or: How i learnt to stop worrying and love the social and behavioural sciences. arXiv preprint arXiv:1712.00547 (2017)
17. Moore, J.D., Paris, C.L.: Requirements for an expert system explanation facility. Computational Intelligence **7**(4), 367–370 (1991)
18. Oren, N., Deemter, K.v., Vasconcelos, W.W.: Argument-based plan explanation. In: Knowledge Engineering Tools and Techniques for AI Planning, pp. 173–188. Springer (2020)
19. Reggia, J.A., Perricone, B.T.: Answer justification in medical decision support systems based on bayesian classification. Computers in Biology and Medicine **15**(4), 161–167 (1985)

20. Sendi, N., Abchiche-Mimouni, N., Zehraoui, F.: A new transparent ensemble method based on deep learning. Procedia Computer Science **159**, 271–280 (2019)
21. Shams, Z., Dimanov, B., Kola, S., Simidjievski, N., Terre, H., Scherer, P., Matjašec, U., Abraham, J., Jamnik, M., Liò, P.: REM: An integrative rule extraction methodology for explainable data analysis in healthcare (2021)
22. Shortliffe, E.H., Axline, S.G., Buchanan, B.G., Merigan, T.C., Cohen, S.N.: An artificial intelligence program to advise physicians regarding antimicrobial therapy. Computers and Biomedical Research **6**(6), 544–560 (1973)
23. Singh, R., Miller, T., Newn, J., Sonenberg, L., Velloso, E., Vetere, F.: Combining planning with gaze for online human intention recognition. In: Proceedings of the 17th international conference on autonomous agents and multiagent systems. pp. 488–496 (2018)
24. Studer, R., Benjamins, V.R., Fensel, D.: Knowledge engineering: principles and methods. Data & knowledge engineering **25**(1-2), 161–197 (1998)
25. Swartout, W.R.: Xplain: A system for creating and explaining expert consulting programs. Artificial intelligence **21**(3), 285–325 (1983)
26. Vassiliades, A., Bassiliades, N., Patkos, T.: Argumentation and explainable artificial intelligence: a survey. The Knowledge Engineering Review **36** (2021)
27. Walton, D.: A dialogue system specification for explanation. Synthese **182**, 349–374 (2011)
28. Walton, D.: A Dialogue System for Evaluating Explanations, pp. 69–116. Springer International Publishing, Cham (2016). https://doi.org/10.1007/978-3-319-19626-8_3, https://doi.org/10.1007/978-3-319-19626-8_3
29. Wick, M.R., Thompson, W.B.: Reconstructive expert system explanation. Artificial Intelligence **54**(1-2), 33–70 (1992)
30. Winikoff, M., Sidorenko, G., Dignum, V., Dignum, F.: Why bad coffee? Explaining BDI agent behaviour with valuings. Artificial Intelligence **300**, 103554 (2021)
31. Zarlenga, M.E., Shams, Z., Jamnik, M.: Efficient decompositional rule extraction for deep neural networks. arXiv preprint arXiv:2111.12628 (2021)