Proceedings of the 23rd International Conference on Principles of Knowledge Representation and Reasoning
KR meets Machine Learning and Explanation

Normative Narrator: Guiding and Explaining Reinforcement Learning Agents

Emery Neufeld , Kees van Berkel
Institute of Logic and Computation, TU Wien, Austria
emeric.neufeld@tuwien.ac.at, kees.van.berkel@tuwien.ac.at

Abstract

A *normative supervisor* is an external module that uses a formal reasoning engine to impose normative constraints on reinforcement learning agents, by either dynamic action masking or feeding the agent additional punishments when violations of norms occur (or both). In this paper, we use a normative supervisor implemented with a solver for deontic answer set programming (ASP) — `deolingo` — as a basis for the construction of a *normative narrator*, which uses `deolingo`'s ability to interface with the explainable solver `xclingo` to construct a module capable of both regulating behaviour through action masking and additional punishments, and providing contrastive explanations of why a given action was allowed while others were not, relative to the normative system being enforced. The explanations are modelled after the two tiers — internal and external — of explanation. We demonstrate this approach's ability to provide understandable and descriptive explanations in a scenario where a taxi driver agent must act in accordance to a normative system governing its normal duties and provisions that must be made in case of an emergency.

1 Introduction

Over the past decade, a significant effort has been invested in devising ways to enforce ethical (or, in general, normative) behaviour on reinforcement learning (RL) agents. These are agents that learn optimal behaviour by exploring an environment and collecting rewards based on the actions they take. While often very effective, since these agents stringently prioritize high rewards, they tend toward opportunistic behaviour which may violate safety standards and human norms; hence the effort into constraining them for safety, legal, and ethical purposes. However, these approaches often suffer from a lack of transparency and the absence of facilities enabling comprehensive explanations of behaviour, which is crucial for both troubleshooting and auditing if, e.g., the agent does not behave as expected.

Explanations answer why-questions, and are often framed contrastively (Miller 2021; Stepin et al. 2021). In this paper, we are interested in norm conflicts and violations that agents face; in the context of normative agents we may ask causal questions, such as "Why did the agent do A, rather than B?", or normative questions like "Why *should* the agent do A, rather rather B?", which become relevant when, e.g., a violation of a norm takes place. Where normative agents make decisions with the aim to maximize outcome while simultaneously optimizing compliance, contrastive explanations are particularly promising (van Berkel and Straßer 2024).

Contribution. In this paper, we propose a method that generates contrastive explanations of agentive behaviour in reference to the violation of norms caused by an action, its relevant alternatives, and the values promoted by these norms. To this end, we used as a base a normative supervisor (Neufeld et al. 2021) — a module which can guide training and operation by either limiting an agent's pool of possible actions, or assigning a punishment when an unrestricted agent strays from this limited pool — implemented with a solver for deontic answer set programming (ASP), `deolingo`. We build on this foundation to construct a *normative narrator*, which uses `deolingo`'s ability to interface with the explainable solver `xclingo` to construct a module that both regulates behaviour, and provides explanations for actions taken. Furthermore, our module allows for a values-based assessment of actions when more than one action is deemed normatively optimal.

Related Work. Over the past decade, following significant advancements in the field of RL (e.g., (Mnih et al. 2015; Silver et al. 2016)), various approaches have arisen for teaching RL agents normative behaviour (though, most approaches, while amenable to some legal or social norms, focus on ethics). Top-down approaches, which begin with given norms/values and then, e.g., design reward functions reflecting them, include (Rodriguez-Soto et al. 2022; Rodriguez-Soto et al. 2023; Ecoffet and Lehman 2021; Neufeld 2024; Neufeld, Ciabattoni, and Tulcan 2024). Other approaches propose using human demonstrations or feedback to create an ethical utility function, for example (Riedl and Harrison 2016; Wu and Lin 2018; Balakrishnan et al. 2019; Noothigattu et al. 2019). For a more comprehensive discussion of these approach, we recommend the reader examine (Vishwanath, Dennis, and Slavkovik 2024).

Our approach is based on the (less expressive and less explainable) approach in (Neufeld et al. 2021; Neufeld et al. 2022), and like (Adam and Eiter 2025), we use answer set programming to describe normative behaviour. However, unlike these approaches, our focus is on explanations — a topic which has been largely neglected by the abovementioned approaches.

2 Preliminaries

2.1 Reinforcement Learning

RL is a learning paradigm in which an agent learns optimal behaviour by exploring a given environment and adjusting its behaviour based on the rewards and punishments it receives from the environment. This environment is formally defined as a *Markov decision process* (MDP):

Definition 1 (Labelled MDP)**.** *A labelled MDP can be represented with the tuple: $\langle S, A, Pr, L, R\rangle$, where S is a set of states, $A : S \to 2^{Acts}$ is a function to a subset of the set of all actions $Acts$ (giving the actions which are possible in a given state), and $Pr : S \times Acts \times S \to [0, 1]$ is a transition probability function, where $Pr(s, a, s')$ is the probability of transitioning to state s' from state s with action a. $L : S \times Acts \to 2^{AP}$ is a mapping from state-action pairs (s, a) to a set of true atomic propositions, and $R : S \times A \times S \to \mathbb{R}$ is a reward function that assigns some real-valued reward to a transition (s, a, s').*

The goal of RL is to learn an optimal policy π, often by learning an action-value function Q^π. We will focus on model-free value-based methods, which assume that we don't know the transition function Pr. In this context, the optimal policy is a map from states to optimal actions ($\pi : S \to Acts$) defined as:

$$\pi(s) \in \arg\max_{a \in A(s)} Q^\pi(s, a) \tag{1}$$

Q^π gives the sum of future expected rewards if the agent takes action a in state s and then follows π.

RL agents learn an optimal policy through experience; given the current state of the environment, the agent will take an action in $\pi(s)$, and when the environment transitions to the next state, it will collect a reward and update Q^π (and therefore π) based on this reward. During training, with a certain probability, the agent takes a random action, so that it is sure to explore the entire state space. This process continues until Q^π converges, yielding an optimal policy π^*.

2.2 Normative Reasoning

Norms are rules that describe what the world *should* look like, expressing ideality, not mere actuality. We deal with two types of norms: (1) constitutive norms, which take the form "X counts as Y in context C" (Searle 1969), and can be used to, e.g., define terms within a normative system; and (2) regulative norms, which regulate behaviour — e.g., obligations, prohibitions, and permissions (Pigozzi and van Der Torre 2018). Reasoning about norms is a complex topic, explored formally in the field of deontic logic. Challenging dynamics and paradoxes abound in the literature, including dilemmas (when two norms cannot be complied with simultaneously), and contrary-to-duty obligations (obligations that come into force when another obligation is violated) (Gabbay et al. 2021). Furthermore, norm codes are often conflict-sensitive by design. The aim is not to get rid of conflicting norms, but to identify resolution mechanisms that enable agents to apply norms that hold all things considered (Nute 1997; Parent and van der Torre 2018).

DELX. In order to express norms formally, we use a non-monotonic deontic logic: deontic equilibrium logic with explicit negation (DELX) (Cabalar, Ciabattoni, and van der Torre 2023). Equilibrium Logic (EL) (Pearce 1996) is the logical characterization of answer set programming (ASP) (Marek and Truszczyński 1999; Niemelä 1999), an established, expressive non-monotonic knowledge representation paradigm. DELX extends EL with the deontic operators **O** (for obligation) and **F** (for prohibition), as well as explicit negation (Aguado et al. 2019), an additional type of negation which resembles classical negation (as opposed to the default negation or negation as failure typically used in ASP). Formally, the language of DELX is:

$$\varphi ::= p \mid \bot \mid \varphi \wedge \varphi \mid \varphi \vee \varphi \mid \sim \varphi \mid \varphi \to \varphi \mid \mathbf{O}\varphi \mid \mathbf{F}\varphi$$

where $p \in AP$ is an atomic proposition. We can further define default negation $\neg\varphi := (\varphi \to \bot)$, and an explicit permission operator $\mathbf{P}\varphi := \sim \mathbf{F}\varphi$. Note that $\mathbf{O}(\sim \varphi)$ is equivalent to $\mathbf{F}\varphi$. We also note the defined operators $\mathbf{O^d}(\varphi) := \neg\mathbf{P}(\sim \varphi) \to \mathbf{O}(\varphi)$ (default obligations), and $\mathbf{O^{nf}}(\varphi) = \mathbf{O}(\varphi) \wedge \neg\varphi$ (non-fulfilled obligation).

Deolingo. As DELX characterizes an extension of EL, it corresponds to a language for deontic answer set programming. Fortunately, a tool exists to compute deontic answer sets: `deolingo` (Cabalar and Manteiga 2025). `deolingo` interfaces with `xclingo` (Cabalar, Fandinno, and Muñiz 2020), allowing for the generation of both deontic answer sets, and explanations tracing the derivation of these answer sets. This is accomplished via *annotations* which accompany the rules in an answer set program. In this paper, we make use of this tool to build a *normative narrator*, which guides and explains the decisions of RL agents.

2.3 Explanation in AI

Due to the presence of norm conflicts, and the possibility of violation, agentive behaviour needs to be explained in the context of norm tensions. For this reason, contrastive explanations are most promising for normative contexts (van Berkel and Straßer 2024). A norm conflict is a tension between multiple norms and possible actions, hence contrastive explanations support understanding of how these norms and possible actions lead to a final decision. Due to the focus on conflict avoidance (here, violation), we draw from existing work on explanation in argumentative dispute. First, contrastive explanations answer *why* questions in reference to a claim and a foil: e.g., "why did the agent perform x (claim), despite the possibility to perform y (foil)?" We adopt the two tier model by Johnson (Johnson 2000) to answer such questions: An explanation in light of conflict contains the internal, illative, tier which provides reasons in favour of a given action (both the claim and the foil), by listing relevant facts, including norms, factual context, and, in our case, the number of violations that may occur. Then, to explain why the claim precedes over the foil, such an explanation contains an external, dialective, tier, which makes explicit how counter-reasons (from the foil), do not suffice for different behaviour (cf. "despite").

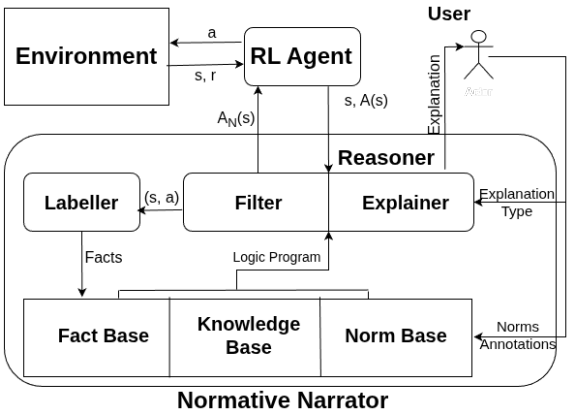

Figure 1: Interaction between the Normative Narrator, the RL agent, the environment, and the user. Boxes with rounded corners are computational modules (the agent, the reasoner, the labeller), and the rectangles signify some mutable information source (the environment and the fact/knowledge/norm base).

The Role of Values. Values play an essential role in dispute. For instance, in legal disputes a judge may refer to the spirit of the law, which represents the values that the law aims to promote (Perelman 2012; Bench-Capon 2002), this is referred to as judiciary discretion (Dik and Markovich 2024). There is a strong connection between normative judgements and value judgements (Hansson 2001). For this reason, using values to resolve (and explain) conflict has been extensively investigated in the context of symbolic AI, formal argumentation in particular (Atkinson and Bench-Capon 2021). A feature of these approaches is that values enter the stage when the conflict cannot be resolved by an appeal to norms alone. In those cases, resolve is sought in a preferential balancing of the values they promote.

3 Normative Narrator

The normative narrator's structure resembles that of the normative supervisor (Neufeld et al. 2021), with the notable addition of an **Explainer**; see Fig. 1 for an overview of the architecture. The normative narrator works together with an RL agent at training or operation time; for each new state the RL agent transitions into, it sends the narrator the state s and those actions available in that state $A(s)$, and for each action, a fact base FB is generated (via an implementation of the labelling function from Definition 1, noted as the **Labeller** in Fig. 1). The result is a list of propositions that are true in that state if that action is taken, such as $raining :=$ "it is raining". A knowledge base KB provides domain knowledge about the environment that can help the narrator determine which actions will cause a violation; e.g., there might be a rule that says that moving north from coordinate $(0, 0)$ will leave the agent in $(0, 1)$.

The norm base NB is provided by the user, containing the constitutive and regulative norms that form the normative system the agent will be subject to, in terms of the the vocabulary established in $FB \cup KB$ (note that constitutive norms may introduce new terms). Each norm can be accompanied by annotations which will inform the explanation given by the **Explainer** if requested. For example, the norm $emergency \rightarrow \mathbf{O}(atSafety)$ might be accompanied by an annotation: `"Agent must be at a safe place because of emergency."`

$FB \cup KB \cup NB$ will be transformed into a logic program LP syntactically amenable to `deolingo`. E.g., the above norm and annotation would generate the following code:

```
%!trace_rule {"Agent must be at a safe place
                because of emergency"}
&obligatory{atSafety} :- emergency.
```

The generated logic program is then sent to the **Reasoner.** In the filter, for each action $a \in A(s)$ `deolingo` will generate an answer set $Ans(a)$ containing all of the facts and obligations that hold in state s if action a is taken. The number of violations (computed from each $Ans(a)$, as those obligations $\mathbf{O}(p) \in Ans(a)$ such that $p \notin Ans(a)$ and those $\mathbf{F}(p) \in Ans(a)$ while $p \in Ans(a)$) is counted. The set $A_{\mathcal{N}}(s)$ is then composed of those actions with the fewest number of violations. Formally:

$$A_{\mathcal{N}}(s) := \arg\min_{a \in A(s)} Viol(a)$$

where

$$Viol(a) = |\{p \mid \mathbf{O}(p) \in Ans(a) \wedge p \notin Ans(a)\}| + |\{p \mid \mathbf{F}(p), p \in Ans(a)\}|$$

This way *only* those actions deemed normatively optimal are passed to the agent; it has no other option but to choose one of these.

The narrator, like the normative supervisor, has a second functionality, in which the narrator remains active during training; it still computes $A_{\mathcal{N}}(s)$ for every state, but instead of passing this to the agent, limiting its choices, it allows the agent to choose whatever action it wants, and if this action is not in $A_{\mathcal{N}}(s)$ the module assigns a punishment of magnitude r. This punishment is then added to the $R(s, a, s')$ the agent receives from the MDP. This allows the narrator to use *reward shaping* to change the behaviour learned by the agent (similar to the safe RL technique presented in (De Giacomo et al. 2019)). This approach, especially when combined with the use of the **Filter** at runtime, can result in a significant improvement in performance (Neufeld 2022).

Value-Based Assessment. In the normative supervisor, $A_{\mathcal{N}}(s)$ would be sent directly to the agent; however, we propose an extra step where we optionally complete a *value based assessment* of the normatively optimal actions if $|A_{\mathcal{N}}(s)| > 1$ and the actions therein result in at least one violation. In this case, we make an appeal to the (ordered) values promoted by the violated norms when multiple norms per action are violated, asserting that the agent should strive not to violate the norm promoting the most preferred value.

For example, in the taxi scenario we will work with in Section 4, the agent is obligated to head to safety in the case of an emergency, which promotes the value of $Safety$; it is also obligated to take the passenger to their destination, promoting the value of $Duty$. Given data from the original filtering process (about which actions violate which obligations, represented as, e.g., `viol(north, atSafety)`, we get a logic program LP' containing, for example:

```
%!trace_rule {"Action % opposes the value
               Safety.", A}
opposeSafety(A) :- viol(A, atSafety),
                   action(A).
```

This logic program will further contain rules reasoning about what should be done given which actions oppose certain values, such as:

```
perfect(A) :- action(A), not opposeSafety(A),
              not opposeDuty(A).
&obligatory{A} :- not opposeSafety(A),
                  opposeDuty(A),
                  not optimal(A).
```

We then select all the actions which are obligatory in the answer set Ans_{val} output when we feed the above logic program to `deolingo`, and pass these instead to the agent.

Explainer. When queried with a state and an action, the **Explainer** generates an explanation. In our framework, we propose a two-tiered contrastive explanation (Section 2.3):

Definition 2 (Two-Tier Explanation)**.** *Let $v(s, a)$ be the set of norms violated by action a in state s. Our two-tiered explanation will consist of:*

1. *An illative explanation. This contains the data informing why the agent might prefer action a over another action a' by providing reasons for these actions. It consists of data already generated by the **Filter**, broken up into brute facts (those facts generated from (s, a) by the **Labeller**), institutional facts (those facts generated by applying relevant constitutive norms), and deontic facts (regulative norms).*
2. *A dialective explanation.*
   (a) *A trace (generated by `xclingo`) describing which norms have been triggered in the reasoning process.*
   (b) *In case there is more than one normatively optimal action (all of which violate at least one norm), a value-based assessment is adopted that describes which violations promote or oppose individual values, and shows that no alternative action promotes a value strictly preferred over any value opposed by $v(s, a)$.*

To generate this explanation, the **Explainer** makes use of the answer sets generated by the **Filter** and the corresponding traces enabled by the above annotations. We trace:

```
%!show_trace &non_fulfilled_obligation{A}.
%!show_trace &fulfilled_obligation{A}.
```

Essentially, we are asking `deolingo` to trace the fulfilled and unfulfilled violations. The user can query the **Explainer** at any time with a state and an action, and receive these traces which explain which obligations were violated by that action in contrast to other actions, and which rules were triggered resulting in these violations being derived. We will see an example of such an explanation in the next section.

4 Case Study

The `gymnasium` (Towers et al. 2024) taxi environment is a maze in which a taxi agent picks up a passenger at one point, and drops it off at another. The agent is rewarded for dropping off the passenger correctly. We have used a modified version of this environment where it is possible for rainy weather to develop into a hurricane. In addition, variable points on the map have been appointed as the locations of a "home" (which may or may not be subject to a flood risk) area and a "shelter" area.

Narrator	Psger. Sat.	atSafety	twrd(atSafety)	atDest	twrd(atDest)
w/o	98 %	10.01	6.97	2.66	0.39
w/	55 %	7.98	0.0	6.24	0.0

Table 1: For an agent trained for 2 million steps: statistics are given for 100 episodes and without the normative narrator, giving percentage of times the passenger was delivered, and the average violations of each regulative norm.

There are a number of constitutive norms that are relevant in this domain. The first is that a hurricane counts as an emergency: $hurricane \rightarrow emergency$. The second is that being at the shelter counts as being at a safe location: $atShelter \rightarrow atSafety$. However, being at home also counts as being safe, if there is no risk of a flood: $\neg floodrisk \wedge atHome \rightarrow atSafety$.

For regulative norms, the first is that if a passenger is in the taxi, it is obligatory that the taxi is at the destination: $hasPassenger \rightarrow \mathbf{O^d}(atDestination)$. We make this a default obligation, because we will establish a higher priority norm soon. Of course, immediately upon picking up the passenger, being at the destination is impossible, so there is a *contrary-to-duty* obligation that comes into force when this obligation is violated: $\mathbf{O^{nf}}(atDestination) \rightarrow \mathbf{O}(toward(atDestination))$. These two norms promote the value we call $Duty$ (as in doing one's job). A similar pair of norms exists regarding getting to safety if there is an emergency: $emergency \rightarrow \mathbf{O}(atSafety)$ and $\mathbf{O^{nf}}(atSafety) \rightarrow \mathbf{O}(toward(atSafety))$, which promote the value $Safety$. We establish the strong permission that defeats the duty to be at the destination (in favour of arriving at a safe place in the case of an emergency): $emergency \rightarrow \mathbf{P}(\sim atDestination)$[1]. The performance of the normative narrator enforcing this normative system is given in Table 1. Note that compliance with norms sometimes comes at the cost of success; sometimes the taxi does not get the chance to pick up/drop off the passenger before the environment times out, as it spent too much time waiting at a safe point.

Our focus here, however, is the **Explainer** feature of the narrator. To explore this functionality, consider the following scenario: the passenger asks the taxi to head east, toward a hotel, but the taxi heads west instead, and the passenger demands an explanation, because the taxi has violated its duty to get the passenger to their destination. We provide snippets of the explanation below[2].

For the illative tier of the explanation, the report contains (see Defn. 2) the contents of each answer set (each associated with an individual action) generated in the **Filter**, broken into brute facts, institutional facts, deontic facts, and the number of violations. For example, for the ac-

[1]This is discarded for the explanation given below, as doing so yields a conflict, where $Safety$ takes precedence over $Duty$.

[2]The full explanation is in the supplementary material, on the repository https://github.com/lexeree/normative-narrator.

tion $west$, brute facts include $toward(atSafety)$, institutional facts include $emergency$, and deontic facts include $obligatory\{atSafety\}$, and there are 3 violations.

For the first phase of the dialective tier of the explanation the **Explainer** uses the traces generated by `xclingo`:

```
|__Obligation to be at toward(atSafety) was fulfilled
|  |__Taxi is going toward the shelter, which counts as
      going toward safety.
|  |__Taxi is not at safety; it must head toward it.
|  |  |__Obligation to be at atSafety was not fulfilled
|  |  |  |__Taxi must be at a safe place because of
            emergency.
|  |  |  |  |__There is a hurricane; it counts as an
               emergency.

|__Obligation to be at toward(atDestination) was not
   fulfilled
|  |__Taxi has a passenger and is not at destation; it
      must head toward it.
|  |  |__Obligation to be at atDestination was not
         fulfilled
|  |  |  |__Taxi must be at destination because it has
            a passenger.
```

The second phase of the dialective tier does the value-based assessments of all actions with the lowest number of violations.

```
|__east promotes Duty, but opposes Safety, and
   Safety > Duty; avoid east.
|  |__Action east opposes the value Safety.

|__west opposes Duty, but promotes Safety, and
   Safety > Duty: do west.
|  |__Action west opposes the value Duty.

|__north promotes Duty, but opposes Safety, and
   Safety > Duty; avoid north.
|  |__Action north opposes the value Safety.
```

Based on this analysis, we can see clearly that $west$ was, in fact, the only viable option.

5 Future Work

For future work, we aim to investigate explanations that involve temporal components. Sometimes, an agent learns to violate a norm in order to avoid an increase in norm violations in the future (Neufeld 2024). This anticipatory behaviour must be explained, constrastively, in light of temporal intervals that demonstrate the learned expected number of violations over possible courses of actions.

Other promising XAI methods for normative agents which we hope to explore include are argument-based explanations (Čyras et al. 2021), counterfactual explanations (Guidotti and others 2022), and interactive explanations (Miller 2023), all of which are promising with respect to dialogues (Mindlin et al. 2025; van Berkel and Straßer 2024). Moreover, we aim to experiment with our framework domains beyond the limited taxi domain we used for illustrative purposes above.

Acknowledgements

This research was funded in whole or in part by the Vienna Science and Technology Fund (WWTF) project 10.47379/ICT22023, the Austrian Science Fund (FWF) projects 6372-N and 10.55776/COE12.

AI Declaration

No AI was used at any stage in writing this paper.

References

Adam, S., and Eiter, T. 2025. Asp-driven emergency planning for norm violations in reinforcement learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, 14772–14780.

Aguado, F.; Cabalar, P.; Fandinno, J.; Pearce, D.; Pérez, G.; and Vidal, C. 2019. Revisiting explicit negation in answer set programming. *Theory and Practice of Logic Programming* 19(5-6):908–924.

Atkinson, K., and Bench-Capon, T. 2021. Value-based argumentation. *Journal of Applied Logics* 8(6):1543–1588.

Balakrishnan, A.; Bouneffouf, D.; Mattei, N.; and Rossi, F. 2019. Incorporating behavioral constraints in online ai systems. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, 3–11.

Bench-Capon, T. J. 2002. Value-based argumentation frameworks. In Benferhat, S., and Giunchiglia, E., eds., *In Proceedings of Non Monotonic Reasoning*. 444–453.

van Berkel, K., and Straßer, C. 2024. Towards deontic explanations through dialogue. In Kampik, T.; Kristijonas, C.; Rago, A.; and Cocarascu, O., eds., *Proceedings of Argumentation for eXplainable AI 2024 (ArgXAI)*, volume 3768 of *CEUR Workshop Proceedings*, 29–40. CEUR (ISSN 1613-0073).

Cabalar, P., and Manteiga, O. 2025. deolingo: Extending answer set programming with deontic reasoning. *17th DEON* 61.

Cabalar, P.; Ciabattoni, A.; and van der Torre, L. 2023. Deontic equilibrium logic with explicit negation. In *European Conference on Logics in Artificial Intelligence*, 498–514. Springer.

Cabalar, P.; Fandinno, J.; and Muñiz, B. 2020. A system for explainable answer set programming. *arXiv preprint arXiv:2009.10242*.

Čyras, K.; Rago, A.; Albini, E.; Baroni, P.; and Toni, F. 2021. Argumentative XAI: A survey.

De Giacomo, G.; Iocchi, L.; Favorito, M.; and Patrizi, F. 2019. Foundations for restraining bolts: Reinforcement learning with ltlf/ldlf restraining specifications. In *Proceedings of the international conference on automated planning and scheduling*, volume 29, 128–136.

Dik, J., and Markovich, R. 2024. Modeling judicial discretion with nuanced permissions. In *Legal Knowledge and Information Systems*. IOS Press. 48–59.

Ecoffet, A., and Lehman, J. 2021. Reinforcement learning under moral uncertainty. In *International Conference on Machine Learning*, 2926–2936. PMLR.

Gabbay, D.; Horty, J.; Parent, X.; Van der Meyden, R.; van der Torre, L.; et al. 2021. *Handbook of deontic logic and normative systems*. College Publications, 2021.

Guidotti, R., et al. 2022. Counterfactual explanations and how to find them: literature review and benchmarking. *Data mining and knowledge discovery*.

Hansson, S. O. 2001. *The structure of values and norms*. Cambridge university press.

Johnson, R. H. 2000. *Manifest rationality: A pragmatic theory of argument*. Routledge.

Marek, V. W., and Truszczyński, M. 1999. Stable models and an alternative logic programming paradigm. In *The logic programming paradigm: A 25-year perspective*. Springer. 375–398.

Miller, T. 2021. Contrastive explanation: A structural-model approach. *The Knowledge Engineering Review* 36.

Miller, T. 2023. Explainable ai is dead, long live explainable ai! hypothesis-driven decision support using evaluative ai. In *Proceedings of the 2023 ACM conference on fairness, accountability, and transparency*, 333–342.

Mindlin, D.; Beer, F.; Sieger, L. N.; Heindorf, S.; Esposito, E.; Ngonga Ngomo, A.-C.; and Cimiano, P. 2025. Beyond one-shot explanations: a systematic literature review of dialogue-based xai approaches. *Artificial Intelligence Review* 58(3):81.

Mnih, V.; Kavukcuoglu, K.; Silver, D.; Rusu, A. A.; Veness, J.; Bellemare, M. G.; Graves, A.; Riedmiller, M.; Fidjeland, A. K.; Ostrovski, G.; et al. 2015. Human-level control through deep reinforcement learning. *nature* 518(7540):529–533.

Neufeld, E.; Bartocci, E.; Ciabattoni, A.; and Governatori, G. 2021. A normative supervisor for reinforcement learning agents. In *Proceedings of CADE 28*, 565–576. Springer.

Neufeld, E. A.; Bartocci, E.; Ciabattoni, A.; and Governatori, G. 2022. Enforcing ethical goals over reinforcement-learning policies. *Journal of Ethics and Information Technology*.

Neufeld, E. A.; Ciabattoni, A.; and Tulcan, R. F. 2024. Norm compliance in reinforcement learning agents via restraining bolts. In *Legal Knowledge and Information Systems*. IOS Press. 119–130.

Neufeld, E. A. 2022. Reinforcement learning guided by provable normative compliance. In *Proceedings of ICAART 2022*, 444–453.

Neufeld, E. A. 2024. Learning normative behaviour through automated theorem proving. *KI-Künstliche Intelligenz* 1–19.

Niemelä, I. 1999. Logic programs with stable model semantics as a constraint programming paradigm. *Annals of mathematics and Artificial Intelligence* 25(3):241–273.

Noothigattu, R.; Bouneffouf, D.; Mattei, N.; Chandra, R.; Madan, P.; Varshney, K. R.; Campbell, M.; Singh, M.; and Rossi, F. 2019. Teaching ai agents ethical values using reinforcement learning and policy orchestration. In *Proc. of IJCAI: 28th International Joint Conference on Artificial Intelligence*. ijcai.org.

Nute, D. 1997. *Defeasible deontic logic*, volume 263. Springer Science & Business Media.

Parent, X., and van der Torre, L. 2018. I/O logics with a consistency check. In Broersen, J. M.; Condoravdi, C.; Shyam, N.; and Pigozzi, G., eds., *Deontic Logic and Normative Systems, 14th International Conference, DEON 2018*, 285–299. College Publications.

Pearce, D. 1996. A new logical characterisation of stable models and answer sets. In *International Workshop on Non-monotonic Extensions of Logic Programming*, 57–70. Springer.

Perelman, C. 2012. *Justice, law, and argument: Essays on moral and legal reasoning*. Springer Science & Business Media.

Pigozzi, G., and van Der Torre, L. 2018. Arguing about constitutive and regulative norms. *Journal of Applied Non-Classical Logics* 28(2-3):189–217.

Riedl, M. O., and Harrison, B. 2016. Using stories to teach human values to artificial agents. In *AI, Ethics, and Society, Papers from the 2016 AAAI Workshop*, volume WS-16-02 of *AAAI Technical Report*. AAAI Press.

Rodriguez-Soto, M.; Serramia, M.; Lopez-Sanchez, M.; and Rodriguez-Aguilar, J. A. 2022. Instilling moral value alignment by means of multi-objective reinforcement learning. *Ethics and Information Technology* 24(1):1–17.

Rodriguez-Soto, M.; Rădulescu, R.; Rodriguez-Aguilar, J. A.; Lopez-Sanchez, M.; and Nowé, A. 2023. Multi-objective reinforcement learning for guaranteeing alignment with multiple values. In *2023 Adaptive and Learning Agents Workshop at AAMAS*.

Searle, J. R. 1969. *Speech Acts: An Essay in the Philosophy of Language*. Cambridge, England: Cambridge University Press.

Silver, D.; Huang, A.; Maddison, C. J.; Guez, A.; Sifre, L.; Van Den Driessche, G.; Schrittwieser, J.; Antonoglou, I.; Panneershelvam, V.; Lanctot, M.; et al. 2016. Mastering the game of go with deep neural networks and tree search. *nature* 529(7587):484–489.

Stepin, I.; Alonso, J. M.; Catala, A.; and Pereira-Farina, M. 2021. A survey of contrastive and counterfactual explanation generation methods for explainable artificial intelligence. *IEEE Access* 9:11974–12001.

Towers, M.; Kwiatkowski, A.; Terry, J.; Balis, J. U.; De Cola, G.; Deleu, T.; Goulão, M.; Kallinteris, A.; Krimmel, M.; KG, A.; et al. 2024. Gymnasium: A standard interface for reinforcement learning environments. *arXiv preprint arXiv:2407.17032*.

Vishwanath, A.; Dennis, L. A.; and Slavkovik, M. 2024. Reinforcement learning and machine ethics: a systematic review. *arXiv preprint arXiv:2407.02425*.

Wu, Y.-H., and Lin, S.-D. 2018. A low-cost ethics shaping approach for designing reinforcement learning agents. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32.