Norms, Rewards, and the Intentional Stance: Comparing Machine Learning Approaches to Ethical Training

Daniel Kasenberg, Thomas Arnold and **Matthias Scheutz**
Department of Computer Science
Tufts University
Medford, MA 02155

Abstract

The challenge of training AI systems to perform responsibly and beneficially has inspired different approaches for teaching a system what people want and how it is acceptable to attain that in the world. In this paper we compare work in reinforcement learning, in particular inverse reinforcement learning, with our norm inference approach. We test those two systems and present results. Using the idea of the "intentional stance", we explain how a norm inference approach can work even when another agent is acting strictly according to reward functions. In this way norm inference presents itself as a promising, more explicitly accountable approach with which to design AI systems from the start.

Introduction

The scope of AI ethics has spread into an interdisciplinary network of policy areas. Instead of viewing AI as a monolith that will either doom or save humanity, this broader view considers a richer landscape of challenges in how AI systems are designed, implemented, and evaluated (AI Now Institute 2017). Even so, the challenge of rendering systems that are "aligned" with human interests and values requires serious attention to computational architecture.

This challenge is particularly prevalent in social interaction, where robots and other artificial agents are being designed and sought for interactive, accountable roles. Because moral and social norms are important in such domains, it is often thought that artificial agents will need to be able to reason about and use these norms (Malle and Scheutz 2014; Scheutz and Malle 2014).

Moral and social norms are, however, viewed by some as too hard to represent stably. Because human values may change over time, and are far too complex to manually program into robotic architectures, what seems more statistically secure is *value alignment*– whatever we people decide we want, the system will be aligned with our preferences. This relies on *learning* human values as the centerpiece of computational approaches to ethical performance.

Inverse reinforcement learning (IRL) has been considered as an approach to the problem of learning in value alignment (Russell, Dewey, and Tegmark 2016). IRL allows agents to observe the behavior of (human or artificial) agents and deduce a reward function that may explain those behaviors. In this way, human values can be inferred by observing humans rather than pre-specified by the agent designer.

IRL relies on the *Markov assumption*: that the next state of the world depends only on the previous state of the world and the action that the agent takes from that state, rather than on the agent's entire history.

One alternative to reward-based approaches to value alignment is to explicitly specify norms in some logical language. This has the advantage of greater interpretability and generalizability, and such norms need not satisfy the Markov assumption. Recent work (Kasenberg and Scheutz 2017) has begun to address the problem of how such norms might be learned from behavior. In what follows, we refer to the process of learning explicitly represented norms by observing behavior as *norm inference*.

In this paper, we seek to directly compare norm inference with IRL for the task of value alignment. In particular, we argue that norm inference roughly corresponds to what (Dennett 1989) refers to as the "intentional stance:" a predictive strategy based on imputing some level of beliefs, desires, and concepts to an observed agent. As such norm inference shows interesting results as such a heuristic, even when the demonstrator agent is not explicitly governed by norms. In contrast, IRL by itself lacks sufficient representational power to represent the behavior of some norm-governed agents.

Preliminaries

The challenge of machine ethics has always been to balance adaptive, dynamic learning about values and behavior with enough stability to uphold ethical standards across contexts. Since a rule-based or law-based system seems to demand a level of exhaustive detail that threatens tractability, some have proposed variations on reinforcement learning. What has been proposed from this perspective are systems that learn the proper reward function for action in the world. In the case of inverse reinforcement learning (IRL), this means inferring that reward function (and hence preferences) from an agent's behavior.

Another kind of approach holds that explicitly representing rules, norms, or other ethical principles is essential for a system that will socially interact and act in concert with

other agents. Sometimes this can take the form of an ethical "governor" to provide guidance to human beings themselves (Shim and Arkin 2017). In previous work we have argued that competent social robots will need some representation of norms in their architecture (Arnold, Kasenberg, and Scheutz 2017). Here we extend this basic idea into the sphere of learning. Sometimes machine ethics can pit "bottom-up" learning (esp. with the growth of deep learning approaches) against "hand-coding" rules. It is assumed that the latter cannot provide enough flexibility and tractability for apprenticeship learning or learning from demonstration. We contend that a norm-based approach *can* learn adaptively in an open-ended environment. Here we propose a norm inference approach that seeks to grasp the action of another agent or agents. In looking for a norm one could say it is something of a normative "intentional" stance, where a cognizable norm is ascribed to the agent (Dennett 1989). While that may not be explicitly represented by the agent itself (say, through natural language), it is nonetheless gleaned and then used as a guide for the system's own behavior.

Further, we want to ask if a norm-based approach might even work acceptably well when the demonstrator is not themselves normatively guided; that is, if even when the demonstrator is in fact maximizing a reward function, the norm-seeking apprentice can well approximate the demonstrator's behavior.

Toward this end we attempt to base a test on what learning the appropriate behavior in a morally charged context necessitates. We explore a scenario of basic care, a plausible setting for showing how temporally complex, norm-based behavior is so crucial for autonomous systems to follow. This serves two main purposes. The first is to illustrate what apprenticeship learning will need to acquire in more care-oriented contexts, where the modeled behavior contains temporal and communicative complexity (e.g. consent). Second, it provides an initial test case for comparing how norm inference and IRL learn from behavior.

Markov Decision Processes

We focus on the problem of inferring moral and social norms by observing agent behavior in stochastic environments. In particular, we shall assume that these environments are modeled by Markov Decision Processes (MDPs).

Formally, we shall define a Markov Decision Process as a tuple $\langle S, A, P, \gamma, s_0, R\rangle$ where

- S is a finite set of *states*;
- A is a finite set of *actions*;
- $P : S \times A \times S \to [0, 1]$ is the *transition function* (where $P(s, a, s')$ is the probability of transitioning to state s' given that the agent is in state s and performs action a);
- $\gamma \in [0, 1)$ is a *discount factor*;
- $s_0 \in S$ is an *initial state*; and
- $R : S \times A \times S \to \mathbb{R}$ is a *reward function* specifying a reward for each transition (s, a, s').

At each time step t, the agent begins in some state s, performs some action a, and then transitions to a new state s' according to $P(s, a, \cdot)$, receiving reward $R(s, a, s')$ in the process. The goal of the agent is to pick a sequence of actions such that the discount sum of all rewards, $\sum_{t=0}^{\infty} \gamma^t R(s_t, a_t, s_{t+1})$, is maximized.

Note that the probability of transitioning to a new state s_{t+1} at time $t + 1$ depends only on the agent's current state s_t and action a_t. This is referred to as the *Markov property*. We also say that the reward function satisfies the Markov property because the reward received at time $t + 1$ depends only on the most recent transition (s_t, a_t, s_{t+1}).

Due to the Markov property, the "best action" to take at any time t depends only on the current state s_t. The problem of planning in MDPs thus is reduced to the problem of finding the optimal policy $\pi : S \times A \to [0, 1]$, where $\pi(s, a)$ is the probability of the agent performing action a at state s.

Inverse Reinforcement Learning

We define an MDP$\backslash R$ as an MDP that is missing its reward function (in other words, a tuple $\langle S, A, P, \gamma, s_0\rangle$).

We define a (finite) behavior trajectory τ as a sequence of state-action pairs, followed by a final state:

$$\tau := (s_0, a_0), (s_1, a_1), \cdots, (s_T, a_T), s_{T+1} \tag{1}$$

Inverse reinforcement learning (IRL) is thus the problem of determining a reward function R given an MDP$\backslash R$ and a set of trajectories $\tau^{(1)}, \cdots, \tau^{(m)}$. The reward function should "explain" the observed behavior in some way (typically, the observed behavior should be close to optimal for the inferred reward function R).

There are a wide variety of IRL algorithms, based on a wide variety of principles. A detailed description of these algorithms is beyond the scope of this paper. While we will focus on the original algorithm by Ng and Russell (Ng and Russell 2000), our arguments are general (we will discuss possible exceptions in the discussion section).

Norm inference

The general task of learning moral and social norms is sometimes referred to as *norm learning*. Because this term may refer to learning norms in a variety of ways (e.g., through natural language (Dzifcak et al. 2009)), we will instead use the term *norm inference* to describe the task of learning moral and social norms from behavior.

In particular, we define norm inference as the problem of determining a logical statement ϕ from some (given) logical language, given an MDP($\backslash R$) and set of trajectories $\tau^{(1)}, \cdots, \tau^{(m)}$. As in the case of IRL, the logical statement should "explain" the observed trajectories in that these trajectories should approximate the behavior of an agent attempting to satisfy ϕ.

Under this definition, norm inference is fundamentally tied to the *norm planning* problem: given an MDP($\backslash R$) and some logical statement ϕ, attempt to satisfy ϕ "as well as possible". This may involve some notion of "better" and "worse" violations encoded in a *violation cost* function, in which case the norm planning agent attempts to minimize

the expected violation cost. The same notion of violation cost can then be leveraged by norm inference algorithms.

The approach that we will consider for norm inference is that of (Kasenberg and Scheutz 2017). This approach represents norms in linear temporal logic (LTL) (Pnueli 1977), a propositional logic representing time linearly:

$$\phi ::= \top \mid \bot \mid p \mid \neg\phi_1 \mid \phi_1 \vee \phi_2 \mid \phi_1 \wedge \phi_2 \mid \phi_1 \rightarrow \phi_2 \\ \mid \mathsf{X}\phi_1 \mid \mathsf{G}\phi_1 \mid \mathsf{F}\phi_1 \mid \phi_1 \,\mathsf{U}\, \phi_2$$

where p belongs to some set Π of atomic propositions. Here $\mathsf{X}\phi_1$ means "at the next time step, ϕ_1"; $\mathsf{G}\phi_1$ means "now and at all future time steps, ϕ_1"; $\mathsf{F}\phi_1$ means "now or at some future time step, ϕ_1", and $\phi_1 \,\mathsf{U}\, \phi_2$ means "eventually ϕ_2 will hold, and ϕ_1 will hold until then".

The truth of each statement ϕ in LTL is evaluated over an infinite sequence of valuations $\sigma_0, \sigma_1, \cdots$, where each $\sigma_i \subseteq \Pi$ represents the set of atomic propositions which are true at time step t. By augmenting an MDP($\backslash R$) with a "labeling function" $\mathcal{L} : S \rightarrow 2^{\Pi}$ where $\mathcal{L}(s)$ is the set of propositions in Π that are true in state s, we can thus evaluate the truth or falsehood of ϕ over infinite sequences of states $s_0, s_1, \cdots$.

This process is typically done by means of a deterministic Rabin automaton (DRA), a finite state machine over infinite strings. In this case, the input alphabet is the set 2^{Π} of valuations. The DRA can be combined with the MDP to produce a *product MDP* with an augmented state space. This product space keeps track of as much of the agent's history as is needed to know whether the agent has satisfied the norm. The concept of the product MDP is crucial to the efficacy of norm inference, and we will discuss it throughout this paper.

We can also construct a notion of violation cost, which roughly defines the "severity" of a violation. This particular violation cost assumes that violating ϕ "for a long time" is worse than violating ϕ "for a short time". In particular, we assume that the agent can "remove" certain timesteps from its infinite sequence $s_0, s_1, \cdots$ of states. That is, removing time steps 1 and 3 would result in the sequence $s_0, s_2, s_4, s_5, \cdots$. The violation cost of an infinite sequence is thus the minimum number of such "skipped" states required for the resulting infinite sequence to entail ϕ (discounted by γ so as to remain finite):

$$\mathrm{Viol}(s_0, s_1, \cdots) = \min_{\substack{N \subseteq \mathbb{N} \\ s_0, s_1, \cdots \backslash N \vDash \phi}} \sum_{t=0}^{\infty} \gamma^t \mathbb{1}_{t \in N} \qquad (2)$$

This notion of violation costs can also be extended to policies (in the product space) . The policy that optimizes the violation cost is in general nonstationary on the original MDP $\mathcal{M}$ (that is, it depends on the agent's entire history), but it is a stationary policy *on the product MDP*. That is, the product MDP contains all the extra information about the agent's history that is necessary to minimize the violation cost.

Norm inference then is formulated as a multi-objective optimization problem over the space of all LTL formulas. The objective functions in question are (1) formula complexity, measured simply by the length $\ell(\phi)$ of the formula ϕ in symbols, and (2) an objective function based on the violation cost. This objective function is computed by

Table 1: An argument in favor of norm inference

Apprentice	Reward-driven demonstrator	Norm-governed demonstrator
IRL	✓	Cannot represent temporally complex norms
Norm inference	Can uncover behavior properties	✓

- constructing the product MDP for the candidate norm ϕ;
- constructing a product space policy $\pi^{\otimes}$ out of the observed behavior trajectories $\tau^{(1)}, \cdots, \tau^{(m)}$;
- evaluating

$$\mathrm{Obj}^{\mathrm{Viol}}(\phi) = \mathrm{Viol}_{\phi}(\pi^{\otimes}) - \mathrm{Viol}_{\phi}(\pi^{rand}) \qquad (3)$$

  where $\mathrm{Viol}_{\phi}(\pi)$ is the violation cost of the (product-space) policy π and π^{rand} is the random policy.

The intuition is that a good norm ϕ is such that the observed trajectories will conform much more closely to ϕ than will random behavior, and thus the difference between their objective costs should be very negative. This encourages norms that both are (more or less) satisfied by the observed trajectories, and which are difficult to satisfy without attempting to (as measured by the violation cost of the random policy). The actual optimization over LTL is done in (Kasenberg and Scheutz 2017) by genetic programming.

"Normative wager"

Having defined IRL and norm inference, we now turn to the task of comparing them. We aim to demonstrate by this comparison that norm inference is a promising approach to learning moral and social norms. While there is much work still to be done towards the norm inference problem, we nevertheless aim to show some concrete advantages over IRL.

IRL with reward-driven demonstrators

We will accept without argument that IRL is effective at learning behavior from demonstration where the demonstrator is driven by rewards. Note that this relies on the "true" reward function satisfying the Markov property. If not, IRL is not guaranteed to approximate the correct behavior.

IRL with norm-governed demonstrators

Previous work (Arnold, Kasenberg, and Scheutz 2017) showed a toy MDP in which a logical specification described a desired pattern of agent behavior on an MDP that could not be captured by a (Markovian) reward function, and thus could not be reproduced by IRL. We now describe a problem with similar, though not identical, properties. Although this is still something of a toy problem and makes a few unrealistic assumptions, it is grounded in the notion of consent and framed in the context of a care-providing robot. We demonstrate IRL's inadequacy for this problem by running an IRL algorithm on the given MDP and showing its inability to capture the relevant behavior.

Consider a caregiving robot responsible for assisting an ailing human. The robot is responsible for giving care if and only if the person desires it. The person will not constantly express their desire or lack of desire for care, however, and will only ask for the robot to care or not to care at discrete points in time. Thus, the person will periodically give signals that they desire or do not desire care, and the robot must remember their current preference.

In order to convert this into a simple Markov Decision Progress, we make a number of simplifying assumptions. We assume that the current state consists of the values of three propositions: whether the robot provided care for the human at the previous time step ($careGiven$), whether the human gave the robot consent to care ($consentGiven$), and whether the consent was withdrawn ($consentWithdrawn$). We further assume that the robot has two actions, $care$ and $nocare$. At each time step, $careGiven$ is set according to the robot's last action. The signals $consentWithdrawn$ and $consentGiven$ are each given with probability ϵ, and are mutually exclusive. We assume that the human's preference will not change without signaling, so that the last signal encodes the human's current preference. Notably, the human's current preference *is not included in the state space* (we will discuss the obvious objections to this shortly).

The agent's goal is simply to perform the action $care$ if and only if the human's last signal was $consentGiven$. This can be represented by a norm in temporal logic as follows:

$$\begin{aligned}\phi := \mathsf{G}((&consentGiven \rightarrow \\ &((\mathsf{X}careGiven)\ \mathsf{U}\ consentWithdrawn)) \\ &\wedge (consentWithdrawn \rightarrow \\ &((\neg\mathsf{X}careGiven)\ \mathsf{U}\ consentGiven))) \quad (4)\end{aligned}$$

The desired behavior is non-Markovian: the optimal course of action depends on which signal ($consentGiven$ or $consentWithdrawn$) the agent has last seen, information not contained in the current state (except during those time steps in which the signal occurs). Because for any reward function that can be inferred by IRL (at least in its standard formulation) the reward-maximizing policy is Markovian, IRL will be inadequate for learning to obey the norm.

Indeed, this is borne out in simulation. We implemented this simulation in BURLAP (MacGlashan 2014), with $\epsilon = 0.1$. We used BURLAP's standard implementation of IRL.

In each trial, we used our norm planner with the norm (4) to demonstrate the desired behavior in 30 trajectories of 30 time steps each. We used IRL to infer a reward function and a corresponding policy from the observed trajectories, and then demonstrated the learned behaviors in 30 trajectories of 30 time steps each. For IRL, we considered each state as its own feature; this resulted in 6 features total.

Figure 1 shows the results of IRL in this domain over 500 trials. Note that the IRL agent performed the wrong action (providing care without consent, or neglecting care when care was required) 30% to 70% of time steps in every trial.

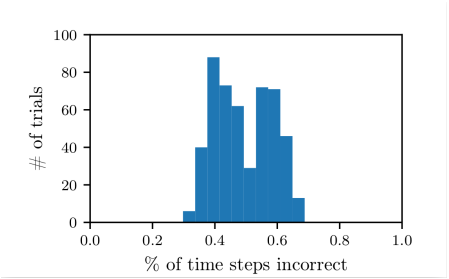

Figure 1: Histogram of the performance of IRL-induced policy over 500 trials. IRL performs the incorrect action ($care$ when $nocare$ is required, and vice versa) between 30% and 70% of the time.

Objection 1: "Add it to the state space" IRL advocates will likely be unconvinced by the preceding argument. The obvious counter-argument is that the state space in the above example can easily be augmented with the information about the agent's current consent/lack of consent. This is true, but in an important respect it misses the point: an agent that is attempting to learn moral and social norms will not know a priori what extra information needs to be added to the state space in order to maintain the Markov property, and neither will the system's designers.

In the IRL framework, for every moral/social norm that an agent wishes to learn, the agent's current state must contain all information about the agent's history necessary in order to satisfy the norm. Perhaps in narrow domains, agent designers can include all of this information in the state space.

Those who believe IRL sufficient for learning norms will likely make at least one of the following assumptions:

(a) *Agent designers will have sufficient foresight to include all morally relevant facts about the agent's history within the state space*. In general domains requiring social interaction, this assumption is almost certainly false. Knowing a priori what aspects of the agent's history will be relevant to an as-yet-unknown norm is virtually equivalent to knowing the norm, which would render IRL unnecessary.

(b) *All information about the agent's history that could possibly be relevant to the norms that will eventually be learned will be included in the agent's state space*. The only upper bound on possibly-relevant information about the agent's history that is known before runtime is the agent's entire trajectory. Operating in "trajectory space", or treating entire agent trajectories as states, is obviously intractible (the agent will never be in the same state in trajectory space twice, and the state space will be infinite (or, in continuous state spaces, the dimensionality of the state space will increase every time step).

(c) *Along with IRL, agents will have a mechanism to dynamically alter the state space so as to "fix" the Markov assumption*. When learning a new norm/reward function, this mechanism would alter the state space so that it stores precisely the history required for the new norm/reward function to be Markovian. Note that this is *precisely what norm inference does*. In the case of LTL, the DRA stores exactly the information needed about the agent's prior history in order to make the problem Markovian. Norm inference has the added advantage of interpretability (which

is not the primary focus of this paper, but is emphasized in (Arnold, Kasenberg, and Scheutz 2017)), but in principle other mechanisms that co-learn a state space augmentation and a reward function may overcome the temporal complexity problem. (In practice, this augmentation could perhaps be accomplished using the hidden states of recurrent neural networks to store the relevant information, although the lack of interpretability of this approach may make it unsuitable for ethics.)

Objection 2: "What about propositions?" In response to our preceding argument, critics may argue that while norm inference (or at least the implementation of it described in this paper) may help to fix problems with agent memory, it does so at the cost of relying on some set of propositions in the state space that encode morally relevant signals. This point is well-taken: we recognize that this assumption may not be well-founded in reality. However, IRL algorithms also tend to rely on the existence of a set of state (or state-action) features from which the reward function can be computed.

In practice, agents in the real world will need to map sensory information onto these features/percepts. Recent work in deep inverse reinforcement learning (Wulfmeier, Ondruska, and Posner 2015, for example) allows the system to map raw sensory data onto features from which a reward function can be computed. It is possible that similar approaches may help to identify a set of relevant propositions for norm inference from sensory data (though this may undermine the interpretability of the system), especially when combined with natural language instruction.

Norm inference with norm-governed demonstrators

When the demonstrator is norm-governed (where we use "norm-governed" to mean "consistent with some normative principle that may be complex, but can be articulated explicitly"), it is valuable to attempt to deduce that principle explicitly in a way that can be reasoned about, understood, and easily corrected. Our work in (Kasenberg and Scheutz 2017), in which we introduced our norm inference algorithm, showed encouraging early results in this capacity.

While IRL is unable to produce the correct behavior in the aforementioned consent domain, the behavior can be represented in temporal logic (indeed, the demonstrator is a norm planner using the norm (4))), and so in principle norm inference will be able to find it. (In practice, the complexity of the norm means that norm inference will take a long time to find it - see the discussion section for more details.)

Norm inference with reward-driven demonstrators

Finally, we evaluate the results of using norm inference to attempt to infer norms where none exist: where the demonstrator is driven by a (random) reward function.

We employed these experiments in a 4×4 GridWorld domain. Each state corresponds to the agent being in one of the grid's 16 cells. Available to the agent are the four actions $north$, $south$, $east$, and $west$. The dynamics are stochastic: with probability $1 - \epsilon$, the agent moves one cell in the intended direction; with probability ϵ the agent moves randomly in one of the three other directions. In this case we set $\epsilon = 0.2$. If an action would cause the agent to collide with the wall, the agent instead remains in the same cell.

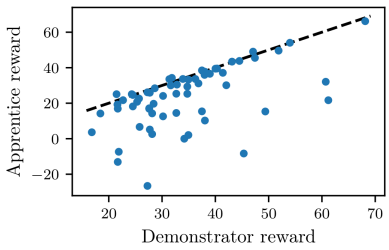

Figure 2: Average reward per episode (30 time steps) for an apprentice using norm inference, vs the average reward per episode of the reward-driven demonstrator. Each data point represents the average over 30 episodes. The dashed line represents the identity function.

In each trial, a reward function was generated by selecting the reward for each individual cell independently from $\mathcal{N}(0, 1)$, the normal distribution with zero mean and unit variance. After using value iteration to determine the optimal policy, the agent demonstrated that policy using 30 trajectories of 30 time steps each.

Norm inference was then applied to the generated trajectories (where a proposition $C_{i,j}$ corresponding to each cell was assumed) to determine a the Pareto-efficient candidate norms. The agent then performed norm planning for each of the Pareto-efficient norms to determine an optimal (product-space) policy, and executed this policy for 30 trajectories of 30 time steps each. We performed a total of 50 trials.

Figure 2 shows the average reward per 30 time steps of the policy found by norm inference vs that of the reward-driven demonstrator. Whenever multiple norms were Pareto-efficient in norm inference, data points corresponding to each recovered norm are plotted. The line corresponds to the identity function: if a data point is on the line, the policy constructed by norm inference maximizes reward as well as the demonstrator's behavior (the apprentice may achieve more reward than the demonstrator, but this is because of stochasticity rather than a superior policy). In many cases, norm inference well approximated the reward-optimal policy, as is evident from the number of points close to this line.

Norms and the intentional stance

Dennett (Dennett 1989) describes three "stances" that serve as strategies for an observer attempting to describe and predict an object's behavior. The first stance, the *physical stance*, considers the object as a set of atoms and molecules. The second stance, the *design stance*, considers the object as if it were designed for a purpose, and considers form and function in accordance with that supposed purpose. The third stance, the *intentional stance*, considers the object as an agent with beliefs, desires, and intentions of its own. Im-

portantly, Dennett notes that the intentional stance can be a useful abstraction for predicting an object's behavior, *even when the object in question may not in fact have beliefs, desires, or intentions*.

We argue that learning explicit rules associated with an agent's behavior (e.g., using norm inference) can be thought of as adopting an intentional stance toward that agent. To assume that there is some norm that a demonstrator is attempting to satisfy is to ascribe to that demonstrator an objective that can be reasoned and communicated about. Further, the temporal complexity of behaviors described by LTL formulas in particular imply that the agent has at least a *rudimentary* notion of memory beyond that supplied by the MDP state space. While agents maximizing Markovian reward functions can be argued to "desire" reward, such agents lack (1) an *explicit* objective and (2) memory beyond that already encoded in their environment. Like the intentional stance, the assumption that agents possess explicit, temporally complex objectives and norms forms a predictive strategy that can be *useful*, even if it is not always *true* (as in the case of the agents in GridWorlds with random reward).

The norm inference form of an"intentional stance" thus shows some promise as a means of mapping even non-normative behavior. One practical issue to pursue further is what tradeoffs come with deciding (at least within a certain context) to approach behavior with norm inference (and not infer an agent's actual reward function) as opposed to using IRL when there is actually a norm guiding the observed behavior.

Discussion and future work

Our proposal to learn an agent's behavior by inferring norms, not rewards, has shown suggestive and promising results. It can capture temporally complex rules that can guide behavior, thereby showing features of context and memory that usual descriptions of state space do not consider. For ethically charged contexts like that of personal care, with an interactive dimension like consent, these subtleties may be crucial to represent and explain a system's behavior.

That said, a great deal of work is left to be done. While our norm inference algorithm is capable in principle of capturing temporally complex norms, in practice optimizing over statements in a formal language is slow. Recent work on inferring formal specifications from demonstration (Vazquez-Chanlatte et al. 2017) searches over the space of specifications more efficiently, but this relies on specifications whose truth can be evaluated in some bounded number of time steps. The general problem of efficiently inferring logical norms from behavior remains open.

It is also important to note that IRL and norm inference may be able to work together to tackle certain problems– what remains is to spell where the comparative advantage lies. Aside from its strengths relative to IRL, norm inference work will also need to incorporate efforts to map which norms are most critical (especially for human and artificial agents (Malle, Scheutz, and Austerweil 2017)), in order to address how conflicting norms can best be managed.

The norm inference approach should also feed back into larger spheres of AI discussions of learning, and what counts as machine learning. Though deep learning reigns as the presumed growing edge of promising AI, commentators have begun to point out how fairly simple statistical mechanisms (e.g. perturbation methods) can cause drastic errors in deep learning approaches (Perez 2017). Marcus has also pointed out that some top-down concepts are needed to guide truly adaptive learning in an environment (Marcus 2017). For machine ethics, norms are a strong candidate, both in how an agent can learn and interpret action and in how accessible, accountable direction can be given to that agent by initial design and subsequent instruction.

On a general level, it is critical to keep computational architecture within the spotlight of algorithmic ethics. Current critiques of AI systems have understandably begun to center on the data on which the algorithms train, and the applications wherein the algorithms lead to consequential decisions (e.g. job applications, loan approval, sentencing). The question remains, however: what should algorithms themselves look like? Even with fair, socially justifiable data, what kinds of algorithms would best support human flourishing and, in Dignum's terms, uphold accountability, transparency, and our own responsibility (Dignum 2017)? The comparative evaluation of IRL (and related RL models), norm inference, and other approaches of machine ethics should not overshadow their important joint commitment: to make algorithms themselves as ethically beneficial and accountable as possible.

Conclusion

In this paper we argued in favor of the norm inference over reward-driven approaches to learning morality. While norm inference can uncover useful properties of agent behavior even when the demonstrator is explicitly not norm-governed, IRL can have great difficulty adapting when the reward function is non-Markovian.

Continuing to test and find new tests for the ethical performance of machine learning will be essential for machine ethics to serve the larger landscape of ethics and policy discussions for AI. Diplomatically generated slogans about AI can mask serious disagreement (Boddington 2017). Surfacing that disagreement respectfully and honestly is a duty for those genuinely committed to holding artificial intelligence to account. By putting algorithms and architectures to a better test, we can more ably chart what AI ethics should look like in (and beside) the flesh.

Acknowledgements

This project was supported in part by ONR MURI grant N00014-16-1-2278 and by NSF IIS grant 1723963.

References

AI Now Institute. 2017. AI Now 2017 Report.

Arnold, T.; Kasenberg, D.; and Scheutz, M. 2017. Value alignment or misalignment – what will keep systems accountable? In *Proceedings of the 3rd International Workshop on AI, Ethics, and Society*.

Boddington, P. 2017. Some suggestions for how to proceed. In *Towards a Code of Ethics for Artificial Intelligence*. Springer. 99–111.

Dennett, D. C. 1989. *The intentional stance*. MIT press.

Dignum, V. 2017. Responsible autonomy. *arXiv preprint arXiv:1706.02513*.

Dzifcak, J.; Scheutz, M.; Baral, C.; and Schermerhorn, P. 2009. What to do and how to do it: Translating natural language directives into temporal and dynamic logic representation for goal management and action execution. In *Proceedings - IEEE International Conference on Robotics and Automation*, 4163–4168.

Kasenberg, D., and Scheutz, M. 2017. Interpretable apprenticeship learning with temporal logic specifications. In *Proceedings of the 56th IEEE Conference on decision and control (CDC 2017)*.

MacGlashan, J. 2014. Brown-umbc reinforcement learning and planning (burlap). `http://burlap.cs.brown.edu/`. Accessed: 2017-11-12.

Malle, B. F., and Scheutz, M. 2014. Moral competence in social robots. In *Ethics in Science, Technology and Engineering, 2014 IEEE International Symposium on*, 1–6. IEEE.

Malle, B. F.; Scheutz, M.; and Austerweil, J. L. 2017. Networks of social and moral norms in human and robot agents. In *A World with Robots*. Springer. 3–17.

Marcus, G. 2017. Artificial intelligence is stuck: Here's how to get it moving. *New York Times*.

Ng, A., and Russell, S. 2000. Algorithms for inverse reinforcement learning. In *Proceedings of the Seventeenth International Conference on Machine Learning*, volume 0, 663–670.

Perez, C. E. 2017. Why probability theory should be thrown under the bus.

Pnueli, A. 1977. The temporal logic of programs. In *18th Annual Symposium on Foundations of Computer Science (sfcs 1977)*, 46–57.

Russell, S.; Dewey, D.; and Tegmark, M. 2016. Research priorities for robust and beneficial artificial intelligence. *arXiv preprint arXiv:1602.03506*.

Scheutz, M., and Malle, B. F. 2014. ?think and do the right thing??a plea for morally competent autonomous robots. In *Ethics in Science, Technology and Engineering, 2014 IEEE International Symposium on*, 1–4. IEEE.

Shim, J., and Arkin, R. C. 2017. An intervening ethical governor for a robot mediator in patient-caregiver relationships. In *A World with Robots*. Springer. 77–91.

Vazquez-Chanlatte, M.; Jha, S.; Tiwari, A.; and Seshia, S. A. 2017. Specification inference from demonstrations. *arXiv preprint arXiv:1710.03875*.

Wulfmeier, M.; Ondruska, P.; and Posner, I. 2015. Deep inverse reinforcement learning. *CoRR, abs/1507.04888*.