Autonomous Agents and Multi-Agent Systems (2019) 33:628–644
https://doi.org/10.1007/s10458-019-09418-w

Summarizing agent strategies

Ofra Amir[1] · Finale Doshi-Velez[2] · David Sarne[3]

Published online: 26 July 2019

Abstract

Intelligent agents and AI-based systems are becoming increasingly prevalent. They support people in different ways, such as providing users with advice, working with them to achieve goals or acting on users' behalf. One key capability missing in such systems is the ability to present their users with an effective summary of their strategy and expected behaviors under different conditions and scenarios. This capability, which we see as complementary to those currently under development in the context of "interpretable machine learning" and "explainable AI", is critical in various settings. In particular, it is likely to play a key role when a user needs to collaborate with an agent, when having to choose between different available agents to act on her behalf, or when requested to determine the level of autonomy to be granted to an agent or approve its strategy. In this paper, we pose the challenge of developing capabilities for strategy summarization, which is not addressed by current theories and methods in the field. We propose a conceptual framework for strategy summarization, which we envision as a collaborative process that involves both agents and people. Last, we suggest possible testbeds that could be used to evaluate progress in research on strategy summarization.

Keywords Strategy summarization · Human–agent interaction · Explainable AI

✉ Ofra Amir
oamir@technion.ac.il

Finale Doshi-Velez
finale@seas.harvard.edu

David Sarne
sarned@cs.biu.ac.il

[1] Faculty of Industrial Engineering and Management, Technion - Israel Institute of Technology, Haifa, Israel

[2] John Paulson School of Engineering and Applied Sciences, Harvard University, Cambridge, MA, USA

[3] Department of Computer Science, Bar-Ilan University, Ramat Gan, Israel

1 Introduction

Intelligent systems play a growing role in our daily lives, from voice-controlled assistants to tools for recognizing cancerous tumors and computer-assisted driving [59]. Many of these systems go even further, autonomously carrying out tasks on behalf of their user rather than simply providing advice. These can either take the form of virtual agents (e.g., bots bidding for ad placement, automatic news feed generators) or physical ones (e.g., autonomous cars, vacuum robots). The behavior of these systems is often opaque to human users. For example, a robotic vacuum may be equipped with several coverage algorithms and the choice of which algorithm to use at a specific time may depend on environment conditions, which might be outside the user's knowledge or understanding.

In a clinical setting, the way in which an agent combines a patient's current measures and history may be more involved than a clinician could immediately process. In both settings, ideally the user could gain a general understanding of how and when to deploy the agent in advance rather than have to learn this situation-by-situation.

Users' familiarity with the strategies and expected behaviors of agents[1] under different conditions and scenarios is essential for many purposes. First, an understanding of agents' behaviors can facilitate choosing between interchangeable systems (e.g., Siri, Cortana, Alexa). Second, knowing the agent's strengths and weaknesses can improve the ability of people to collaborate with agents (e.g., being familiar with a surgical robot's strategy can support a surgeon in the operating room). Last, users may need to determine how much autonomy to grant to an agent, and knowing its expected behavior can help them make more informed decisions, and trust that the agent could perform its designated role.

However, explaining agent behavior to users is challenging because the strategies of these agents are often determined using sophisticated computational techniques (e.g., machine-learning models, deep learning, Markov decision processes and in many cases an ensemble of methods). People are inherently bounded-rational and find it difficult to map from a design and logic to actual behavior of the system. It has been shown that users' mental models of system behaviors are incomplete, parsimonious and unstable; that people are limited in their ability to "run" these models to predict expected behavior; and that people often confuse different mental models [49]. Additionally, humans do not reason in the same way that autonomous systems do: they are likely to use different state representations and different reasoning mechanisms, making it potentially harder to understand agents' behavior. Moreover, attempting to specify the system's actual behavior in each world state is typically infeasible because the space of possible world states the system may run into is often far more immense than what a human can manage. For example, the state space in which autonomous vehicles make decisions is based on speed, weather, road conditions, distance and much other data gathered by a variety of sensors, including cameras, LiDARs (Light Detection and Ranging), and radars. While the user may be a very experienced driver, it is very likely that she has never experienced or ever considered many of the possible states in this space (e.g., spotting a child running out to retrieve a bouncing ball or spotting an exploding tire of a close-by car).

The idea behind strategy summarization is to provide users with some form of a summary (either textual or visual, through an interactive interface) that demonstrates system behavior in carefully selected world states. With this new paradigm, users gain a better understanding of the system in a range of diverse key world states, in a relatively short time. Recent

[1] The use of the term "agent" in this paper refers to any system for which strategy can be formally captured or simulated. Autonomous agents are a specific case in that sense.

efforts developed methods for explaining one-shot decisions made by autonomous agents (e.g., [32]) or machine learning algorithms (e.g., [51]) in retrospect, or *ex-post*. In contrast, strategy summarization offers a complementary important capability, which is the cohesive description of the behavior an agent is likely to exhibit, *ex-ante*. A few recent works developed user interface designs enabling users to query an agent's policy (e.g., [27]), yet these require much sophistication, knowledge and effort from users in order to properly understand the system, and do not generate automated summaries. In contrast, the strategy summarization approach aims to communicate the actual agent behavior in a scenario-based manner, rather than conveying its underlying decision-making model (e.g., a decision tree, the coefficients of a logistics regression model). It reduces user effort in that it presents agent behavior in a range of scenarios rather than requiring users to specify many queries on their own.

The study of strategy summarization will contribute to theories and methods in the areas of decision making under uncertainty, human–agent interaction, explainable AI and multi-agent literature. It requires expertise in diverse fields, in particular human–agent interaction, planning and learning algorithms and representations, interpretability, and machine learning. Still, we argue it is both feasible and worth pursuing, as methods that help people better understand agents are expected to have an impact in many areas, including domains of societal importance such as healthcare, education and transportation.

The goal of this position piece is to place strategy summarization in the context of other work in explainable AI (Sect. 2), provide a framework for strategy summarization algorithms (Sect. 3), and propose considerations for evaluation (Sect. 4). In our framework section, we identify three key components of strategy summarization systems, which we hope will help organize research in this nascent field. In our evaluation section, we not only suggest potential benchmark applications, but also call out considerations for evaluating a global property such as a summary.

2 Related work

Strategy summarization falls within the broad area of "Explainable AI" and "Human-Aware AI". In particular, it relates to works on explaining robot and agent behavior, interpretable machine learning and user understanding of system behavior. In this section we argue that the majority of work in these areas, to date, focused on ex-post explanation of local agent decisions. Strategy summarization, on the other hand, aims to effectively convey to the user a strategy as a whole, ex-ante. While both approaches aim to support users in understanding the behavior of AI-based systems, they require diffeent methods and designs. We next review related works, emphasizing these differences in approaches.

Explaining robot and agent behavior Most closely related to strategy summarization, Huang et al. [30] recently presented a method for extracting trajectories of agent behavior that would support users in inferring the agent's objectives. Their approach extracts trajectories that enable recovering the agent's strategy using inverse reinforcement learning. This work provides one approach to tackle one aspect of the strategy summarization problem, namely automatic extraction of states to present to users. In this paper, we discuss additional potential approaches for extracting summaries of agent behavior, and further present a conceptual framework for the problem of agent strategy summarization that includes additional aspects of the problem (e.g., choosing state representations, interacting with users) and discuss evaluation approaches for assessing the effectiveness of strategy summarization methods.

Other prior work has developed methods that enable users to “debug” a robot or interact with it in simulation to gain better understanding of its behavior. For example, Nikolaidis and Shah [48] proposed a cross-training approach to help parties develop a better understanding of their teammate. Lomas et al. [42] developed a system that enables a user to ask robots questions. Brooks et al. [7] developed a system that visualizes all the past actions of a robot and includes explanations for the actions. Hayes and Shah [27] proposed several methods for explaining robot policies to people using past execution traces, enabling users to query the agent’s behavior in different states and request explanations. Recent work [23,72] developed methods for explaining the decision-making of agents playing Atari games by visualizing the (pixel) regions the agent relies on for choosing its actions. While these approaches do provide users with some ways of exploring what actions a robot will take in different scenarios, they do not attempt to carefully select which behaviors to demonstrate. Thus, they require substantial manual effort from users to explore the behavior of the robot and do not ensure that users understand the overall strategy of the robot.

The majority of prior work in the area of explaining robot and agent behavior focused on explanations of specific decisions, without attempting to demonstrate their global strategies. Sreedharan et al. [58] developed methods for generating plan explanations to users that have different level of expertise in the domain. Another line of work used argumentation approaches to explain agent behavior [2,8,57]. For example, Caminada et al. [8] developed a dialogue-based system which allows a user to query the system about justification for actions, which the system answers based on actions’ preconditions and effects. Several prior works suggested methods for explaining recommendations given by MDP-based intelligent assistants [13,15,16,31,32] or explaining plans [55]. Wang et al. generated explanations of robot reasoning based on Partially Observable Markov Decision Problems (POMDPs) [68] and similar approaches have been developed for explaining decisions in the context of Hierarchical Task Networks (HTNs) planning, explaining an agent’s actions based on its task model [45,46]. The problem we address differs from the problem of generating explanations for specific decisions, as rather than providing justifications to a specific action, we aim to describe *which* actions would be taken in information-critical states, with the overarching goal of providing a *global* understanding of the agent’s behavior. To illustrate, a *local* explanation of a medical treatment policy might describe the reasons that an agent chose to prescribe a particular medication to a patient; in contrast, a *global* summary might show different states or trajectories of patients’ medical state and would describe which treatments the agent would assign in each of these cases.

Finally, several works considered summarizing hierarchical plans or generating plans that are more understandable to people in the first place. For example, Zhang et al. [73] proposed measures for plan explicability and predictability and developed methods for synthesizing plans that are more explicable by considering a human model of the domain. This problem is distinct from the strategy summarization problem in that it aims to generate, rather than *describe* plans or policies. Myers [47] proposed a method for summarizing plans represented as HTNs to help people in reviewing and comparing them, emphasizing features such as the allocation of roles to agents, tasks included in the plan and tasks absent. However, this approach is limited in the sense that it is only applicable to fairly restricted short-term plans toward achieving a specific goal and relies on very specific features of HTNs. Therefore it cannot be used to summarize MDP policies that determine agents’ behavior in a large state-space.

In sum, while there exists a substantial body of work addressing questions related to explaining agents’ plans to people, the strategy summarization approach is distinct from these works as it attempts to convey the global behavior of an agent to users. Therefore, it requires the development of new algorithms for extracting summaries as well as new interaction designs for presenting summaries to users. Importantly, we note that strategy summarization is complementary to the existing approaches described in this section and can be integrated with them, for instance by adding explanations for local decisions as part of a summary of a strategy.

Interpretable machine learning Recently, many approaches have been proposed for developing interpretable machine learning models, that is, models that are understandable to people [14,41,52,66]. These approaches typically seek to explain a decision made by a machine learning model. For example, LIME [51] explains the features that determined the classification of a particular instance by presenting users with a simpler locally correct model that applies only to that particular region of the feature space. Similar to the methods for explaining MDP decisions, these approaches explain a single decision in retrospect rather than provide a description of a strategy or behavior of an agent in different situations. As such, they address a different problem than that of strategy summarization.

Some recent works developed methods for choosing a set of instances to present to users along with their classification in order to provide them with a global understanding of the model [33,34,51]. For example, the SP-LIME algorithm [51] selects a set of instances to show to users along with their respective classification and an explanation of the classification of these instances. The instances are chosen to cover different regions of the state space. While such approaches provide users with a better global understanding of a machine learning model's classification decisions, they are not applicable to *sequential decision-making* settings, where an agent follows a long-term policy. In particular, they do not account for the effect of decisions on outcomes as there is no reward function or transition functions, which are important aspects for agents that act in the world.

Users' understanding of system behavior The strategy summarization idea relates to the literature on users' mental models of systems (software, robots). In a study of users' trust in personal assistants, Glass et al. [21] found that system transparency is important to users and suggested that explanations of system behavior can facilitate trust. More accurate mental models of users about robots' behavior can result in improved performance when collaborating with robots [12]. However, research in HCI has shown that people face difficulties in forming accurate mental models of systems and in practice their models of system behaviors are incomplete, parsimonious and unstable; and they are limited in their ability to predict a system's behavior [49]. Letting users interact with a robot's behavior has been shown to help them establish appropriate mental model [60]. We hypothesize that presenting users with summaries of agent strategies will also help them establish mental models of these agents, facilitating trust and collaboration.

3 A conceptual framework for strategy summarization

While strategy summarization is a complex task, we argue that it can be broken down into manageable subcomponents. We suggest a conceptual framework for the process of strategy summarization, illustrated in Fig. 1, which we envision as a collaborative effort that involves both agents and people. We next describe the main components required for generating and presenting summaries. The objective of this section is to define each of these components, which we believe will provide a substrate for future work in strategy summarization. For

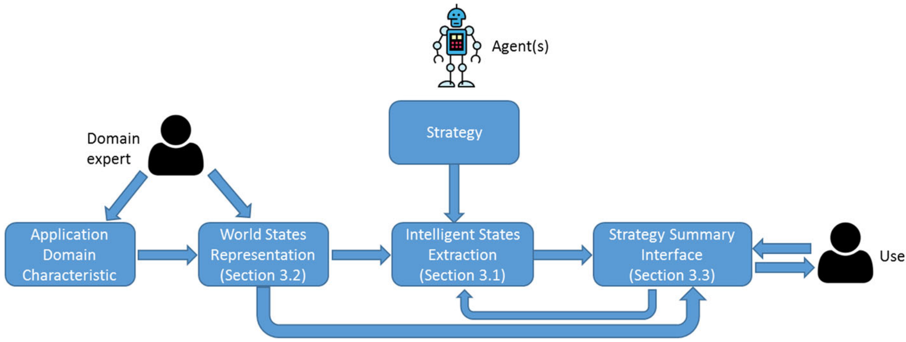
Agent(s)
Strategy
Domain expert
Application Domain Characteristic
World States Representation (Section 3.2)
Intelligent States Extraction (Section 3.1)
Strategy Summary Interface (Section 3.3)
User

Fig. 1 A conceptual model of the summarization process

example, future works could focus on a particular component or the pipeline as a whole. We summarize the components below, and then describe conjectures and considerations on how each component might be built.

Framework overview A key capability required for strategy summarization is identifying states that are likely to be of interest to people interacting with the agent, such that the behavior of the agent in these states can be conveyed to users. To this end, the first key component is **intelligent state extraction** that takes into account the sequential nature of the decisions made, the dynamic transitions between states and the outcomes of actions (rewards). The state extraction takes as input the strategy of the agent (or agents) that needs to be summarized and identifies and prioritizes a subset of states to include in the summary, together with the agent's behavior in those states. The state extraction component requires a specification of the desired properties of a set of world states to present in a summary. We discuss these in Sect. 3.1.

Extracting states presupposes that we have a way to **represent world states** and present them to users. We expect that internally, the agent may have a highly complex representation of the world that it uses for decision-making. For example, a decision to invest in a stock may depend not only on current stock prices but also on long-term and short-term trends as well as a variety of different types of external financial and political events. This representation might not align with that of a human user, as people might take into account different factors or integrate less information than that of the agent. A good world-state representation will thus substantially reduce the potentially immense and inscrutable space of agent internal representations to those that are meaningful to the user. We discuss potential approaches to determining the appropriate state representation in Sect. 3.2.

The final component is the **strategy summary interface**. Users will be the ultimate consumers of the strategy summary, and thus an appropriate interface is essential in the process of determining what is relevant to them. The interface must facilitate mixed-initiative interaction, where the exploration of the summary is guided by both the user and the system. We discuss key design and computational problems toward supporting this collaborative exploration in Sect. 3.3.

When describing the proposed methods, we will consider the task of summarizing the strategies of robots used for search and rescue (SAR) missions [71], one of the domains in which we propose to evaluate strategy summarization methods. SAR robots are deployed to help with rescue missions (e.g., following an earthquake) and are used to gather information such as the location of victims and collapses. The SAR missions are challenging in the sense that they usually involve diverse and dynamic nature scenarios which require a high-level of autonomy and versatile decision-making capabilities [53]. The robots are typically managed by a human operator, hence having a better understanding of the behavior of different robots and their exploration strategies can facilitate and improve the collaborative effort. Recent advances in this field suggest fully-autonomous robotic solutions for executing complex SAR missions in unstructured environments [53]. These rely on a flexible system architecture which integrates several learning-based capabilities (e.g., for target recognition and interaction). While such system architectures allow the execution of high-level missions in a fully unsupervised manner (i.e., without human intervention), the complexity of the designs calls for strategy-summarization like capabilities to support users in deciding between several available implementations for a specific task,[2] which becomes highly crucial in the event of crisis.

3.1 Intelligent state extraction

Implementing a state extraction module presents two main problems. First, it requires defining the desired properties of a set of world-states to present in a summary. We hypothesize that the desired summary characteristics will depend on the context of use. Second, once the properties of the summary are defined, the problem of automatically extracting summaries with these properties needs to be solved. While the specific requirements for effective strategy summaries will likely vary across problem domains, we hypothesize that there is a basic set of requirements that are common across different settings: the summary should provide a high coverage of states that are likely to be of interest to users and should be of reasonable length such that it does not require too much time/cognitive effort to review but still provides the user with sufficient information about the agent's strategy. In addition, the summary should be relevant to the user's goal (e.g., whether it is to choose between alternative agents or work together with an agent), and should provide information that would help the user to contextualize the information presented (e.g., the likelihood of encountering different states).

In the following we assume that the agent uses a Markov Decision Process (MDP) to represent the world. Formally, an MDP is described by the following: A is the set of actions available to the agent; S is the set of states; R: $S \times A \rightarrow \mathbb{R}$ is the reward function, which maps each state and action to a reward. Tr is the transition probability function, i.e., $Tr(s', a, s)$ defines the probability of reaching state s', when taking action a in state s.

We formalize the problem of extracting a summary of an agent's behavior as follows: given access to a simulator of the agent, choose a set of state-action pairs $T = \langle\langle s_1, a_1\rangle, \ldots, \langle s_k, a_k\rangle\rangle$ to include in the summary. The problem definition can either specify a limited budget k for the length of the summary, such that $|T| = k$, or be formulated as an optimization problem of extracting the smallest summary which provides some guarantees (e.g., coverage of the state or feature space).

Below, we suggest three possible directions for the development of methods that determine which state-action pairs to include in the summary. Each has different computational tradeoffs, and will emphasize different states. An important research direction will be to determine what kinds of state extraction methods work best for different contexts.

[2] For example, when considering the autonomous operation of UAVs in SAR missions, various recent solutions can be considered [1,6,54,61,62].

Direction: Generating state-action summaries based on states of interest Different heuristics can be used for choosing states that expose an agent's decision-making process. First, some states might be more *important* than others, in the sense that the decisions made in such states have a higher impact on the agent's utility. For example, knowing which action a search and rescue robot will make in a state in which some action could result in a collapse might be of interest to a user operating the robot. Such states can be identified using the agent's own decision making model. For instance, the distribution of Q-values of different actions in a given state could be used to determine the importance of the choice of action in that state, as the Q-values reflect the expected utility of the agent from taking those actions and then following its policy in future states. Similar ideas have been used in the past in the context of student-teacher reinforcement learning to determine effective teaching opportunities for agents [4,11,63]. In preliminary work on strategy summarization, we used this approach to generate summaries [3]. Specifically, we developed the HIGHLIGHTS algorithm which extracts the most important states from simulations of agent behavior, using the following criteria to determine the importance I_s of a state s:

$$I(s) = \max_a Q^{\pi}_{(s,a)} - \min_a Q^{\pi}_{(s,a)} \tag{1}$$

Intuitively, a state is considered important if taking a wrong action in that state can lead to a significant decrease in future rewards, as determined by the agent's Q-values. In contrast, if there is a small difference between the best and worst action at a particular state, it would suggest that the state is not important. Human-subject studies showed that participants were more successful at evaluating the capabilities of agents when presented with summaries generated by HIGHLIGHTS compared to baseline summaries, and rated them as more helpful [3].

Second, it might be important to consider *coverage* of the state space. While the importance of a state provides a heuristic for choosing situations that a person would find useful to review, it is not a sufficient consideration. If state selection is only based on importance, it is possible that the extracted summary will include states that are very similar to each other. For example, in a search and exploration scenario, an importance-based summary might include many situations in which the robot avoids being trapped in a collapse. In contrast, a summary with high coverage would describe a robot's behavior in a wider range of states, such as navigating when there are visibility problems due to collapses, encountering a fire, and handling scenarios where victims are spread out in different locations. The coverage heuristic can be formalized using state similarity measures, which can be tuned by the user or by domain experts. In our preliminary work on state extraction, adding the coverage consideration substantially improved the ability of participants to assess agents' capabilities compared to summaries that only considered state importance [3].

In addition to state importance and the coverage of the state space, the *likelihood of encountering states* can also be taken into consideration when extracting states for a summary. This would ensure that the summary assists users in understanding the agent's behavior in states that are more likely to be encountered.

Finally, if the goal of a user observing summaries is to decide which agent to use, the summary can highlight *policy disagreement*. That is, the summary can emphasize regions of the state-space in which different agents differ in their decisions, thus enabling a user to assess the differences between the agents. For example, different search and rescue agents may vary in how they prioritize different tasks, which could be demonstrated to the user by showing the distinct decisions these agents make in similar key states.

Direction: Generating state-action summaries based on policy reconstruction accuracy An alternate way to identify important states is to optimize them for the ability of a human to reconstruct the agent's policy. This approach is closely related to the work on machine teaching [74], where an agent is provided with demonstrations of expert behavior, and attempts to learn a policy based on those demonstrations. There are two main approaches for learning a policy based on demonstrations: imitation learning (IL) and Inverse Reinforcement Learning (IRL). In IL, a classifier is trained to predict an action based on state features. In this case, state-action pairs can be chosen for a summary to optimize the accuracy of the classifier using active learning [36]. IRL methods try to infer the agent's reward function based on the demonstrations. Then, the inferred reward function can be used to learn a policy. In this framework, the aim is to generate summaries that result in high action prediction accuracy when they are given as demonstrations to IL or IRL methods. Our hypothesis is that summaries that are useful for reconstructing the agent's policy would also be helpful to people, as they capture meaningful information which allows to generalize the agent's behavior to unseen states. We have begun using this approach in preliminary work [37] to extract summaries. This approach has also been used recently by Huang et al. [30].

A key challenge in this approach to extracting summaries is overcoming the fact that we do not know how people would generalize based on demonstrations. For example, when a user sees a summary of a search and rescue robot, would she try to infer the robot's reward function (like IRL), or would she attempt to learn a direct mapping from states to actions (like IL)? In recent work [36,37], we showed that it is important that the model used to extract a summary matches the model used to reconstruct a policy, as in many cases a mismatch in models results in low policy reconstruction accuracy. One possible approach to address this problem is to generate an array of different summaries that vary in the reconstruction methods and choose summaries that are most robust to different models. An alternative approach is to develop human-in-the-loop methods that will elicit people's computational models when extrapolating based on summaries and adjust summaries accordingly.

Direction: Generating summaries based on peer-designed agents The third direction utilizes people's judgment to identify states that will likely be of importance to users. An example for possible methods that take this approach is the use of Peer-Designed Agents (PDAs) [9,17,18,44]. These agents have been used in recent years for generating realistic behaviors in various application domains (e.g., automated negotiation [40], security [39] various social dilemmas [75], online markets [43] and the design of parking lots [10]) for the purpose of predicting human behavior and their reaction to changes in their environment. The idea is to provide people with a skeleton agent equipped with all the required functionality except for its behavioral layer and have people (either directly or through the mediation of a programmer) design and program into it the strategy they would have used in similar decision situations.[3] The PDAs' logic can then be used as a reflection of what their developers considered to be important (or worth distinguishing) when reasoning about the strategy to be used. This could enable synthesizing the set of world states that are most relevant for demonstrating the system behavior, either through a manual code review, seeking for points of code divergence, or by applying standard clustering algorithms for identifying those states that are highly distinguishable in terms of the code used in large by the population (when using several PDAs).

[3] The method is inspired by the "strategy method" paradigm from behavioral economics [56] in the sense of eliciting people's strategy. Nevertheless, while in the strategy method people state their action for every possible situation that may arise in their interaction (i.e., a state-machine-like description) with PDAs people are actually required to program their (not-necessarily-state-based) strategy into an agent.

3.2 World-state representation

Deciding how to encode the state representation such that states could be effectively conveyed to people and such that the space of states to consider for the summary will be reduced is a hard problem. We expect that this would require either analysis of large sets of strategies (e.g., a set of strategies programmed into PDAs for that application domain) or the design of effective processes for eliciting state representations from domain experts. The idea in both approaches is to reason about what types of raw states can be logically aggregated to a single state, as far as the user is concerned, and to what extent prior events as well as various measurable factors should be considered for distinguishing between states of interest. For example, the underlying state representation used by a search and rescue robot might include low-level data such as inputs from its various sensors, as well as higher-level factors such as a map of the space. In collaboration with domain experts, this representation could be reduced to include only higher level features such as the current map and likelihood of collapses in different areas. This simpler representation could then be used when extracting a summary, which would reduce the complexity of the process due to the pruned state space.

State representation encoding using experts is inherently a manual process, and designing a process for querying experts in an efficient way (e.g., using active learning approaches) will be key to making this process feasible. Extraction based on strategies could be performed either based on manual code analysis or using unsupervised clustering over the raw world states. The latter approach would identify states for which a similar action is used by a large subset of strategies in peer-designed agents.

3.3 Strategy summary interface

Strategy summarization poses several challenges with respect to the design of collaborative interfaces through which users can review and explore agents' strategies. In this section we discuss three aspects in the design of such interfaces.

Consideration: Summary presentation Naturally, different forms of presenting summaries would be appropriate in different settings. For example, for physical agents such as self-driving cars or home robots, it might be more helpful to visually show their actual behavior than showing a textual summary of their expected behavior.This can be done by showing a summary video of their execution constructed as a sequence of short clips, each demonstrating the agent's behavior in a given situation. However, for virtual agents such as a finance investment advising agent, some form of a textual summary or a static visual summary might be more appropriate. For example, states might be described by showing plots of financial trends and other key events, with the actions taken listed for the demonstrated states. A key question is thus how to present summaries to people. There are additional important questions such as how to provide people with sufficient context about the states shown in the summary without overwhelming them with non-important low-level details.

Consideration: User-guided exploration of agent strategies Automatically extracted summaries of agent strategies can provide an effective starting point for a user to develop an understanding of an agent's strategy. However, these summaries may be insufficient in addressing all of the user's needs. For instance, they might not describe the behavior of the agent in a region of the state space which is of particular interest to the user. Therefore, we propose the design of collaborative interfaces which will allow users to: (1) guide the generation of summaries by stating preferences, and (2) directly query the agent's strategy and explore its behavior in different situations. For instance, in the search and rescue domain, the user (human operator) might ask the system to generate summaries that best distinguish the strategies of different robots, or summaries that focus on the most risky states that can be encountered. She might also query the system directly to ask what a robot would do in a particular state by describing a specific rescue scenario, or request to further extend a given segment of the summary, showing the behavior in a varied set of world states that follow the demonstrated scenario. A key design challenge in developing these interaction methods is to ensure that users can express their needs in their own language, rather than being asked to specify low-level system parameters. Moreover, exploring the behavior of an agent can be tedious. To make the process more efficient, the design of mixed-initiative interactions [29] where the system tries to help the user in exploring the agent's strategy will likely be required.

The use of collaborative interfaces for exploring agents' strategies would also facilitate continuous improvement of the state extraction and state representation methods. By observing and analyzing the information users required beyond that provided in the automatically generated summaries, it would be possible to learn more about users' needs and the criteria for effective summaries, thus informing future design of methods for extracting summaries.

Consideration: Understanding users' extrapolation from summaries The intelligent summary extraction methods make various assumptions about how the summaries will be interpreted and generalized from when shown to users. As demonstrated in our preliminary work [36,37], it is important that these assumptions match with users' reasoning. To address such potential mismatches, interactive processes for eliciting users' inference and extrapolation processes can be designed. For example, the system could show users simple summaries (perhaps of a small region in the state space, or a simple policy), and assess their ability to appropriately extrapolate from the summary to the behavior of the agent in other states. Such a process can involve the assessment of alternative feature representations, and different summaries generated using different optimization criteria. Based on this evaluation, the system could both tailor the summary shown to the way in which users generalize from it, and also provide better explanations to people about the way in which they should interpret behaviors shown in the summary.

4 Evaluation methodologies

To assess progress in the area of strategy summarization and ensure that generated summaries are helpful for users, it is essential to thoughtfully consider means of evaluation. In the following, we consider three aspects of evaluating strategy summarization approaches: evaluation domains, user and task characteristics and evaluation metrics.

Evaluation domains The domain characteristics could have an impact on the applicability and usefulness of different summary generation and visualization methods. For example, for domains where the agent is physical (e.g., a robot), showing a video demonstrating its behavior could be feasible, while for virtual agents (e.g., an agent recommending medical treatment) a different visualization might be needed. In addition, the size of the state-space and its feature representation as well as the structure of the reward function can also effect the usefulness of different summarization approaches. Therefore, it is important to evaluate strategy summarization methods in a variety of domains and assess the generalizability of the developed methods.

To this end, there are several testbeds of common use in the AI-community which can be used for evaluation, including the following:

- Search and rescue robots [35,65,67,69,71].
- Automated trading and negotiation (e.g., TAC [22,64,70] or ANAC [5] simulators).
- Route planning [24,25]

There are several advantages to using these existing testbeds. First, they provide infrastructure where different levels of complexity can be implemented. That is, the domain can be modified to vary the size of the state-space as well as the difficulty of the tasks. Second, there is already a large set of implemented agents which vary in their strategies and capabilities. That is, it is possible to control the level of complexity both of the environment and of the agents, thus allowing for testing summarization methods in increasingly more complex settings.

Alongside experimentation with the above testbeds, it is important to have some real test-cases to validate the developed approaches. In such domains, researchers developing strategy summarization methods can work closely with domain experts to ensure that summaries serve their needs. Example applications range from clinical decision-support (e.g., for treatment management [19,20]) to autonomous vehicles. The simpler domains which have been previously studied in the AAMAS community will facilitate relatively fast cycles of testing and improvement of the developed methods, enabling gradual progress towards evaluation in the more complex domains, where substantial implementation and careful experimental design will be required.

User and task characteristics We hypothesize that different methods might be more fitting depending on the user's goals and expertise. For example, if a summary is used to identify blind-spots of search and rescue robots, emphasizing important states might best support a human operator, while if the goal is to choose between alternative agents, summaries emphasizing their differences might be more appropriate. Therefore, we suggest evaluating summaries in a wide range of settings. In addition to differences in users' tasks and goals, other characteristics such as the user's expertise in the domain at hand and their technical knowledge (end-user vs. agent developer) might also affect the choice of summaries. Thus, experiments should also consider different user types.

Evaluation metrics Strategy summarization methods should be evaluated using both computational measures and human-centered measures, as typically done in the context of human–robot interaction and human–agent interaction. The main measures we suggest include:

- *Computational complexity of generating a summary (computational)* The complexity of generating a summary can be important in some cases, as we might want to generate new summaries online based on user interactions.
- *Policy reconstruction accuracy (computational and user-based)* assuming some computational model of policy reconstruction, it is possible to compute the accuracy of the reconstructed policy based on the summary. We have used this measure in our recent preliminary work [36,37], where this metric revealed that matching the computational models used for summary extraction and summary reconstruction can be important in order to reconstruct the policy. Similarly, users' ability to extrapolate from a summary and predict agent behavior can be assessed by asking users what they expect the agent to do in different (unseen) scenarios. This metric has been used recently by Huang et al. [30] to evaluate people's ability to predict the behavior of an agent in a driving scenario.

In both computational and human-based assessments, the accuracy of action predictions for subsets of states, e.g. important states or common states, can be considered.

- *Understanding of agents' reward functions (computational and user-based)* Given a summary, IRL methods can be used to infer the agent's reward and this inferred reward function can be compared to the agent's true reward function. Users' understanding of the agent's reward function can be assessed by directly asking the users about the agent's goals or its valuation of different world-states.
- *Ability to collaborate with the agent (user-based)* in settings where a user will need to collaborate with the agent (e.g., a human operator of a search and rescue robot), the objective performance of the human–agent team after the user reviews summaries can be measured in a simulated collaborative activity. Several measures such as task effectiveness, attention demand and interaction effort have been proposed in the HRI literature [50] and can be applied to evaluate the contribution of summaries to collaboration. In addition, team fluency can be assessed using additional measures from the HRI literature such as concurrent activity and time to complete task [28,48].
- *Ability to predict agent performance (user-based)* Assess the human users' ability to determine which of several agents would perform better on a task and ability to spot blind-spots in the agent's strategy. We used this metric to assess our importance-based summarization approach [3], and were able to find differences in people's ability to predict the performance of agents based on different summaries.
- *Users' perceived understanding of the agent's behavior (user-based)* Eliciting people's (subjective) confidence when making predictions about agents' behavior to see whether they are confident in their understanding of the agent's behavior, and importantly, whether their confidence correlates with their ability to predict the agent's behavior.
- *Users' cognitive load when reviewing summaries (user-based)* Summaries may differ in the cognitive required to extrapolate from them. To assess this, users' cognitive load can be measured using standard questionnaires such as NASA-TLX [26].

Considering a variety of evaluation measures can help identify the strengths and weaknesses of summarization methods and possibly reveal tradeoffs between them. For example, some summaries might better support a user in predicting the agent's behavior while posing higher cognitive demands.

5 Discussion

With the increasing prevalence of intelligent agents, it has become paramount to ensure that the behavior of such agents is understandable to their users. Thus, there is a growing interest in the development of "explainable AI" [38] and "human-aware AI". In this paper, we pose the challenge of summarizing agent behavior to people, which we view as a complementary approach to existing methods in the area of explainable AI. We introduce a conceptual framework for the strategy summarization problem which consists of three main components: identifying appropriate world-state representations, extracting informative trajectories of agent behavior and presenting this information to users through collaborative interfaces. This framework aims to outline initial potential research directions and approaches toward addressing the strategy summarization problem. We believe this area is ripe with interesting and important research problems beyond those discussed in the paper, such as summarizing strategies of agent teams (as opposed to individual agents), developing summaries for users with different levels of expertise and generating a set of diverse summaries that optimize for different summary criteria.

The study of strategy summarization will contribute to various research areas within AI, as discussed throughout the paper. More importantly, its deliverables will enable both novice and expert users to better understand the systems they use, in particular complex (AI-based) systems. With the growing use of AI-based agents and shift toward the design of intelligent agents that can collaborate effectively with people [59], we expect that improved user understanding of agents' capabilities and limitations will lead to improved outcomes in many areas.

One key area in which we expect the developed methods to make a substantial impact is the emerging use of autonomous and semi-autonomous vehicles. There is no doubt that we are on the verge of a shift in the way vehicles, humans and the transportation infrastructures interact. The successful operation of autonomous transportation systems (e.g., the autonomous car or Amazon's UAVs) requires providing the highest level of assurance to legislators, authorities (e.g., highway authorities) and users (e.g., car buyers). With strategy summarization, both legislators and users will be able to better understand the way in which these systems work, potentially leading to shorter approval cycles and more effective use. For example, a better understanding of the expected behavior of an autonomous car will help drivers anticipate situations in which the car needs to transfer control to them.

Strategy summarization could also be beneficial in areas of vast societal importance such as education and healthcare. In education, the methods to be developed have potential to enable parents and educators to make better choices when deciding on the educational systems students will become engaged with, hence improving education development. In medical domains, strategy summarization could help patients better understand treatment protocols, potentially leading to a better state of mind while being treated; professionals would be able to reason about the fit of such plans to patients, potentially improving patients' outcomes.

Acknowledgements The research was partially supported by a J.P. Morgan faculty research award and by the Israel Science Foundation (Grant No. 1162/17).

References

1. Abrahamsen, H. B. (2015). A remotely piloted aircraft system in major incident management: Concept and pilot, feasibility study. *BMC Emergency Medicine*, *15*(1), 12. https://doi.org/10.1186/s12873-015-0036-3.
2. Amgoud, L., & Prade, H. (2009). Using arguments for making and explaining decisions. *Artificial Intelligence*, *173*(3–4), 413–436.
3. Amir, D., & Amir, O. (2018). Highlights: Summarizing agent behavior to people. In *Proceedings of the 17th international conference on autonomous agents and multi-agent systems (AAMAS)*.
4. Amir, O., Kamar, E., Kolobov, A., & Grosz, B. J. (2016). Interactive teaching strategies for agent training. In *International joint conferences on artificial intelligence*.
5. Baarslag, T., Hindriks, K., Jonker, C. M., Kraus, S., & Lin, R. (2012). The first automated negotiating agents competition (anac 2010). In T. Ito, M. Zhang, V. Robu, S. Fatima, & T. Matsuo (Eds.), *New trends in agent-based complex automated negotiations* (Vol. 383, pp. 113–135). Berlin, Heidelberg: Springer.
6. Bejiga, M. B., Zeggada, A., Nouffidj, A., & Melgani, F. (2017). A convolutional neural network approach for assisting avalanche search and rescue operations with uav imagery. *Remote Sensing, 9*(2). https://doi.org/10.3390/rs9020100. http://www.mdpi.com/2072-4292/9/2/100.
7. Brooks, D. J., Shultz, A., Desai, M., Kovac, P., & Yanco, H. A. (2010). Towards state summarization for autonomous robots. In *AAAI fall symposium: Dialog with robots* (Vol. 61, p. 62).
8. Caminada, M. W., Kutlak, R., Oren. N., & Vasconcelos, W. W. (2014). Scrutable plan enactment via argumentation and natural language generation. In *Proceedings of the 2014 international conference*

on Autonomous agents and multi-agent systems, international foundation for autonomous agents and multiagent systems (pp. 1625–1626).

9. Chalamish, M., Sarne, D., & Lin, R. (2012). The effectiveness of peer-designed agents in agent-based simulations. *Multiagent and Grid Systems*, *8*(4), 349–372.
10. Chalamish, M., Sarne, D., & Lin, R. (2013). Enhancing parking simulations using peer-designed agents. *IEEE Transactions on Intelligent Transportation Systems*, *14*(1), 492–498.
11. Clouse, J. A. (1996). On integrating apprentice learning and reinforcement learning. PhD thesis, University of Massachusetts
12. Devin, S., & Alami, R. (2016). An implemented theory of mind to improve human–robot shared plans execution. In *2016 11th ACM/IEEE international conference on human–robot interaction (HRI)* (pp. 319–326). IEEE.
13. Dodson, T., Mattei, N., & Goldsmith, J. (2011). A natural language argumentation interface for explanation generation in Markov decision processes. In *Algorithmic decision theory* (pp. 42–55).
14. Doshi-Velez, F., & Kim, B. (2017). Towards a rigorous science of interpretable machine learning. arXiv preprint arXiv:1702.08608.
15. Elizalde, F. (2008). Policy explanation in factored Markov decision processes. In *Proceedings of the 4th European workshop on probabilistic graphical models (PGM 2008)* (pp. 97–104).
16. Elizalde, F., Sucar, L. E., Reyes, A., & de Buen, P. (2007). An MDP approach for explanation generation. In *ExaCt* (pp. 28–33).
17. Elmalech, A., & Sarne, D. (2014). Evaluating the applicability of peer-designed agents for mechanism evaluation. *Web Intelligence and Agent Systems*, *12*(2), 171–191.
18. Elmalech, A., Sarne, D., & Agmon, N. (2016). Agent development as a strategy shaper. *Autonomous Agents and Multi-Agent Systems*, *30*(3), 506–525.
19. Ernst, D., Stan, G. B., Goncalves, J., & Wehenkel, L. (2006). Clinical data based optimal STI strategies for HIV: A reinforcement learning approach. In *2006 45th IEEE conference on decision and control* (pp. 667–672). IEEE.
20. Garg, A. X., Adhikari, N. K., McDonald, H., Rosas-Arellano, M. P., Devereaux, P., Beyene, J., et al. (2005). Effects of computerized clinical decision support systems on practitioner performance and patient outcomes: A systematic review. *Jama*, *293*(10), 1223–1238.
21. Glass, A., McGuinness, D. L., & Wolverton, M. (2008). Toward establishing trust in adaptive agents. In *Proceedings of the 13th international conference on intelligent user interfaces* (pp. 227–236). ACM.
22. Greenwald, A., & Stone, P. (2001). Autonomous bidding agents in the trading agent competition. *IEEE Internet Computing*, *5*(2), 52–60. https://doi.org/10.1109/4236.914648.
23. Greydanus, S., Koul, A., Dodge, J., & Fern, A. (2017). Visualizing and understanding atari agents. arXiv preprint arXiv:1711.00138.
24. Hadfi, R., Ito, T. (2016a). Holonic multiagent simulation of complex adaptive systems. In *Workshop on MAS for complex networks and social computation (CNSC).*
25. Hadfi, R., & Ito, T. (2016b). Multilayered multiagent system for traffic simulation. In *International conference on autonomous agents and multi-agent systems (AAMAS), Singapore*, May 9–13, 2016.
26. Hart, S. G., & Staveland, L. E. (1988). Development of NASA-TLX (task load index): Results of empirical and theoretical research. *Advances in Psychology*, *52*, 139–183.
27. Hayes, B., & Shah, J. A. (2017). Improving robot controller transparency through autonomous policy explanation. In *Proceedings of the 2017 ACM/IEEE international conference on human–robot interaction* (pp. 303–312). ACM.
28. Hoffman, G. (2013). Evaluating fluency in human–robot collaboration. In *International conference on human–robot interaction (HRI), workshop on human robot collaboration* (Vol. 381, pp. 1–8).
29. Horvitz, E. (1999). Principles of mixed-initiative user interfaces. In *Proceedings of the SIGCHI conference on human factors in computing systems* (pp. 159–166). ACM.
30. Huang, S. H., Held, D., Abbeel, P., & Dragan, A. D. (2019). Enabling robots to communicate their objectives. *Autonomous Robots*, *43*(2), 309–326.
31. Khan, O., Poupart, P., Black, J., Sucar, L., Morales, E., & Hoey, J. (2011). Automatically generated explanations for markov decision processes. In *Decision theory models for applications in AI: Concepts and solutions* (pp. 144–163).
32. Khan, O. Z., Poupart, P., & Black, J. P. (2009). Minimal sufficient explanations for factored Markov decision processes. In *ICAPS*.
33. Kim, B., Khanna, R., & Koyejo, O. O. (2016). Examples are not enough, learn to criticize! criticism for interpretability. In *Advances in neural information processing systems* (pp. 2280–2288).
34. Kim, B., Rudin, C., & Shah, J. A. (2014). The Bayesian case model: A generative approach for case-based reasoning and prototype classification. In *Advances in neural information processing systems* (pp. 1952–1960).

35. Kosti, S., Sarne, D., & Kaminka, G. A. (2014). A novel user-guided interface for robot search. In *Proceedings of the international conference on intelligent robots and systems (IROS)* (pp. 3305–3310).
36. Lage, I., Lifschitz, D., Doshi-Velez, F., & Amir, O. (2019a). Exploring computational user models for agent policy summarization. In *Proceedings of the 28th international joint conference on artificial intelligence (IJCAI)*.
37. Lage, I., Lifschitz, D., Doshi-Velez, F., & Amir, O. (2019b). Toward robust policy summarization. In *Proceedings of the 18th international conference on autonomous agents and multi-agent systems (AAMAS)*.
38. Langley, P., Meadows, B., Sridharan, M., & Choi, D. (2017). Explainable agency for intelligent autonomous systems. In *AAAI* (pp. 4762–4764).
39. Lin, R., Kraus, S., Agmon, N., Barrett, S., & Stone, P. (2011). Comparing agents' success against people in security domains. In *Proceedings of the twenty-fifth AAAI conference on artificial intelligence*.
40. Lin, R., Kraus, S., Oshrat, Y., & Gal, Y. K. (2010). Facilitating the evaluation of automated negotiators using peer designed agents. In *Proceedings of the twenty-fourth AAAI conference on artificial intelligence*.
41. Lipton, Z. C. (2016). The mythos of model interpretability. arXiv preprint arXiv:1606.03490.
42. Lomas, M., Chevalier, R., Cross II, E. V., Garrett, R. C., Hoare, J., & Kopack, M. (2012). Explaining robot actions. In *Proceedings of the seventh annual ACM/IEEE international conference on human–robot interaction* (pp. 187–188). ACM.
43. Manisterski, E., Lin, R., & Kraus, S. (2008). Understanding how people design trading agents over time. In *Proceedings of 7th international joint conference on autonomous agents and multiagent systems (AAMAS)* (pp. 1593–1596).
44. Mash, M., Lin. R., & Sarne. D. (2014). Peer-design agents for reliably evaluating distribution of outcomes in environments involving people. In *Proceedings of the international conference on autonomous agents and multi-agent systems (AAMAS)* (pp. 949–956).
45. McGuinness, D. L., Glass, A., Wolverton, M., & Da Silva, P. P. (2007a). A categorization of explanation questions for task processing systems. In *ExaCt* (pp. 42–48).
46. McGuinness, D. L., Glass, A., Wolverton, M., & Da Silva, P. P. (2007b). Explaining task processing in cognitive assistants that learn. In *AAAI spring symposium: Interaction challenges for intelligent assistants* (pp. 80–87).
47. Myers, K. L. (2006). Metatheoretic plan summarization and comparison. In *ICAPS* (pp. 182–192).
48. Nikolaidis, S., & Shah, J. (2013). Human–robot cross-training: Computational formulation, modeling and evaluation of a human team training strategy. In *Proceedings of the 8th ACM/IEEE international conference on human–robot interaction* (pp. 33–40). IEEE Press.
49. Norman, D. A. (1983). Some observations on mental models. *Mental Models*, *7*(112), 7–14.
50. Olsen, D. R., & Goodrich, M. A. (2003). Metrics for evaluating human–robot interactions. In *Proceedings of PERMIS* (Vol. 2003, p. 4).
51. Ribeiro, M. T., Singh, S., & Guestrin, C. (2016a). Model-agnostic interpretability of machine learning. arXiv preprint arXiv:1606.05386.
52. Ribeiro, M. T., Singh, S., & Guestrin, C. (2016b). Why should i trust you?: Explaining the predictions of any classifier. In *Proceedings of ACM international conference on knowledge discovery and data mining* (pp. 1135–1144). ACM.
53. Sampedro, C., Rodriguez-Ramos, A., Bavle, H., Carrio, A., de la Puente, P., & Campoy, P. (2018). A fully-autonomous aerial robot for search and rescue applications in indoor environments using learning-based techniques. *Journal of Intelligent & Robotic Systems*. https://doi.org/10.1007/s10846-018-0898-1.
54. Scherer, J., Yahyanejad, S., Hayat, S., Yanmaz, E., Andre, T., Khan, A., Vukadinovic, V., Bettstetter, C., Hellwagner, H., & Rinner, B. (2015). An autonomous multi-UAV system for search and rescue. In *Proceedings of the first workshop on micro aerial vehicle networks, systems, and applications for civilian use, DroNet '15* (pp. 33–38). ACM, New York, NY, USA. https://doi.org/10.1145/2750675.2750683.
55. Seegebarth, B., Müller, F., Schattenberg, B., & Biundo, S. (2012). Making hybrid plans more clear to human users-a formal approach for generating sound explanations. In *Twenty-second international conference on automated planning and scheduling*.
56. Selten, R., Mitzkewitz, M., & Uhlich, G. (1997). Duopoly strategies programmed by experienced players. *Econometrica*, *65*(3), 517–555.
57. Sohrabi, S., Baier, J. A., & McIlraith, S. A. (2011). Preferred explanations: Theory and generation via planning. In *AAAI*.
58. Sreedharan, S., Srivastava, S., & Kambhampati, S. (2018). Hierarchical expertise level modeling for user specific contrastive explanations. In *IJCAI* (pp. 4829–4836).
59. Stone, P., Brooks, R., Brynjolfsson, E., Calo, R., Etzioni, O., Hager, G., Hirschberg, J., Kalyanakrishnan, S., Kamar, E., Kraus, S., Leyton-Brown, K., Parkes, D., William, P., AnnaLee, S., Julie, S., Milind, T., &

Astro, T. (2016). Artificial intelligence and life in 2030. One hundred year study on artificial intelligence: Report of the 2015–2016 study panel.

60. Stubbs, K., Hinds, P. J., & Wettergreen, D. (2007). Autonomy and common ground in human–robot interaction: A field study. *IEEE Intelligent Systems*, *22*(2), 42–50.
61. Sun, J., Li, B., Jiang, Y., & Wen, C. (2016). A camera-based target detection and positioning UAV system for search and rescue (SAR) purposes. *Sensors, 16*(11). https://doi.org/10.3390/s16111778. http://www.mdpi.com/1424-8220/16/11/1778.
62. Tomic, T., Schmid, K., Lutz, P., Domel, A., Kassecker, M., Mair, E., et al. (2012). Toward a fully autonomous UAV: Research platform for indoor and outdoor urban search and rescue. *IEEE Robotics Automation Magazine*, *19*(3), 46–56. https://doi.org/10.1109/MRA.2012.2206473.
63. Torrey, L., & Taylor, M. (2013). Teaching on a budget: Agents advising agents in reinforcement learning. In *Proceedings of the 2013 international conference on Autonomous agents and multi-agent systems* (pp. 1053–1060).
64. Urieli, D., & Stone, P. (2014). Tactex'13: A champion adaptive power trading agent. In *Proceedings of the twenty-eighth conference on artificial intelligence (AAAI'14)* (pp. 465–471).
65. Velagapudi, P., Wang, J., Wang, H., Scerri, P., Lewis, M., & Sycara, K. (2008). Synchronous vs. asynchronous video in multi-robot search. In *ACHI'08* (pp. 224–229).
66. Vellido, A., Martín-Guerrero, J. D., & Lisboa, P. J. (2012). Making machine learning models interpretable. *ESANN*, *12*, 163–172.
67. Wang, H., Kolling, A., Brooks, N., Owens, S., Abedin, S., Scerri, P., Lee, P., Chien, S. Y., Lewis, M., & Sycara, K. (2011). Scalable target detection for large robot teams. In *HRI'11* (pp. 363–370). https://doi.org/10.1145/1957656.1957792.
68. Wang, N., Pynadath, D. V., & Hill, S. G. (2016). The impact of pomdp-generated explanations on trust and performance in human–robot teams. In *Proceedings of the 2016 international conference on autonomous agents & multiagent systems* (pp. 997–1005).
69. Wang, H., Velagapudi, P., Scerri, P., Sycara, K., & Lewis, M. (2009). Using humans as sensors in robotic search. In *FUSION'09* (pp. 1249 – 1256).
70. Wellman, M., Greenwald, A., & Stone, P. (2007). *Autonomous bidding agents—Strategies and lessons from the trading agent competition*. Cambridge: MIT Press.
71. Yanco, H. A., & Drury, J. L. (2006). Rescuing interfaces: A multi-year study of human–robot interaction at the AAAI robot rescue competition. *Autonomous Robots*, *22*(4), 333–352. https://doi.org/10.1007/s10514-006-9016-5.
72. Yang, Z., Bai, S., Zhang, L., & Torr, P. H. (2018). Learn to interpret atari agents. arXiv preprint arXiv:1812.11276.
73. Zhang, Y., Sreedharan, S., Kulkarni, A., Chakraborti, T., Zhuo, H. H., & Kambhampati, S. (2017) Plan explicability and predictability for robot task planning. In *2017 IEEE international conference on robotics and automation (ICRA)* (pp. 1313–1320). IEEE.
74. Zhu, X. (2015). Machine teaching: An inverse problem to machine learning and an approach toward optimal education. In *AAAI* (pp. 4083–4087).
75. Zuckerman, I., Cheng, K. L., & Nau, D. S. (2018). Modeling agent's preferences by its designer's social value orientation. *Journal of Experimental & Theoretical Artificial Intelligence*, *30*(2), 257–277. https://doi.org/10.1080/0952813X.2018.1430856.