HIGHLIGHTS: Summarizing Agent Behavior to People

Dan Amir
The Hebrew University of Jerusalem
dan.amir@mail.huji.ac.il

Ofra Amir
Technion - Israel Institute of Technology
oamir@technion.ac.il

ABSTRACT

People increasingly interact with autonomous agents. This paper introduces and formalizes the problem of automatically generating a summary of an agent's behavior with the goal of increasing people's familiarity with the agent's capabilities and limitations. In contrast with prior approaches which developed methods for explaining a single decision made by an agent, our approach aims to provide users with a summary that describes the agent's behavior in different situations. We hypothesize that reviewing such summaries could help people in tasks such as choosing between agents or determining the level of autonomy to grant to an agent. We develop "HIGHLIGHTS", an algorithm that produces a summary of an agent's behavior by extracting important trajectories from simulations of the agent. We conducted a human-subject experiment to evaluate whether HIGHLIGHTS summaries help people assess the capabilities of agents. Our results show that participants were more successful at evaluating the capabilities of agents when presented with HIGHLIGHTS summaries compared to baseline summaries, and rated them as more helpful. We also explore a variant of the HIGHLIGHTS algorithm which aims to increase the diversity of states included in the summary, and show that this modification further improves people's ability to assess agents' capabilities.

KEYWORDS

Strategy summarization, Explainable AI

ACM Reference Format:
Dan Amir and Ofra Amir. 2018. HIGHLIGHTS: Summarizing Agent Behavior to People. In *Proc. of the 17th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2018), Stockholm, Sweden, July 10–15, 2018*, IFAAMAS, 9 pages.

1 INTRODUCTION

From cleaning robots to self-driving cars, autonomous and semi-autonomous agents are becoming increasingly prevalent [21]. People's understanding of such agents' behaviors can increase their trust in the agents and their ability to collaborate with them [5, 8]. An understanding of an agent's behavior could also support people in tasks such as choosing between alternative agents and determining when the agent can be trusted with performing a task autonomously and when the user's attention is needed. For example, if a user can anticipate the behavior of a self-driving car in different scenarios, she could be more prepared to take control in situations where the car might not perform well on its own.

While prior work has suggested ways to explain individual decisions of an agent to a person [11, 12], these approaches do not convey a "global" view of an agent's policy. Similarly, recent methods for interpretable machine learning [7, 24] typically explain a single decision made by a model, e.g. by presenting a simplified model which justifies decisions in a certain region in the space [17]. In this paper, we introduce the problem of providing users with a summary of an agent's behavior. This approach aims to provide users with an overview of the agent's global strategy rather than explaining specific decisions after the fact.

A trivial way of communicating an agent's behavior is to show past executions or simulations. This approach, however, has important drawbacks. First, many of the situations an agent encounters might be uninteresting to a person (e.g., a self-driving car stuck in traffic for an hour). Second, reviewing long execution traces will require a person to spend a significant amount of time, and people might give up early, or not pay attention, potentially missing important states. Therefore, we seek solutions that extract *effective* summaries which show the actions taken by the agent in key scenarios. Such summaries can reduce the human effort required to review the agent's behavior, while still providing sufficient information about its capabilities. We note that this is analogous to the approach taken in many settings in which people need to assess the performance of other people. For example, sports scouting agencies typically prepare videos that include highlights from players' games to demonstrate their skills[1].

We developed "HIGHLIGHTS", an algorithm that extracts important states from an execution trace of an agent in an online manner. Intuitively, a state is important if different actions in that states can lead to substantially different outcomes for the agent. For example, deciding which turn to take when driving in a city will not be considered important if taking the next turn will result in a similar arrival time; deciding whether to exit a highway will be considered more important, as missing the exit can result in a significant delay. Our approach assumes that HIGHLIGHTS has access to the agent's strategy which is described using a Markov Decision Process (MDP) policy, and quantifies the importance of states based on the agent's Q-values. To provide more context to the user, rather than showing important states in isolation, the algorithm extracts a trajectory that includes neighboring states and composes a summary of the agent's behavior from these trajectories.

We used HIGHLIGHTS to create summaries of agents playing Mrs. Pacman [18] and evaluated these summaries in a human-subject experiment. We compared HIGHLIGHTS summaries with two baselines. One baseline generated summaries by extracting random trajectories of the agent, which will, on average, include states that are more likely to be encountered. The other baseline generated

Proc. of the 17th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2018), M. Dastani, G. Sukthankar, E. André, S. Koenig (eds.), July 10–15, 2018, Stockholm, Sweden.

[1] e.g., https://www.youtube.com/watch?v=gX3e0UM-OeM. We note that while such scouting videos are often biased to showcase only successful actions, we intend that summaries of agent behavior will include states that demonstrates their behavior in different states of interest, whether successful or not.

summaries by extracting the first trajectories the agent encountered, which is akin to having a user watch the agent until she runs out of time. In the experiment, participants were shown summaries of different Pacman agents which varied in their performance, and were asked to select an agent to play on their behalf. They were also asked to rate the helpfulness of different summaries for evaluating an agent's capabilities. Our results show that HIGHLIGHTS led to improved objective performance of participants: they were significantly more likely to choose the better performing agent when the HIGHLIGHTS summaries were shown. HIGHLIGHTS summaries were also rated as more helpful by the study participants.

One limitation of the HIGHLIGHTS algorithm is that it does not consider the diversity of states in the summary, and therefore if important states are similar to each other, the summary will consist of similar trajectories, thus conveying less new information to users. To mitigate this problem, we developed a variant of the HIGHLIGHTS algorithm which, in addition to state importance, takes into consideration the similarity of the state to other states in the summary. This extension further improved participants' ability to assess the performance of different agents.

The contributions of the paper are threefold: (1) we introduce and formalize the problem of summarizing an agent's behavior to people; (2) we develop HIGHLIGHTS and HIGHLIGHTS-DIV, algorithms that automatically extract summaries of an agent's policy, and (3) we conduct human-subject experiments, showing that summaries generated by HIGHLIGHTS and HIGHLIGHTS-DIV were preferred by participants and improved their ability to assess the capabilities of agents compared to the baseline summaries.

2 RELATED WORK

Myers [15] developed methods for summarizing Hierarchical task networks (HTNs) to help people review and compare plans. These summaries showed features such as role allocation and subtasks in the plan. In contrast to this work, our goal is to summarize general agent strategies rather than a hierarchical plan toward completing a specific goal. Other works attempted to explain MDP-based plans [19] and recommendations given by MDP-based intelligent assistants [6, 11, 12]. The problem we address differs from the problem of generating explanations for specific decisions, as rather than explaining an action taken (or a suggested action), we aim to describe *which* actions will be taken by the agent in *different states*.

The human-robot interaction literature has developed methods for helping people get insight into a robot's behavior. For example, Lomas et al. [14] developed a system that enables a user to query robots about their behavior (e.g., "why did you turn left here?"). Brooks et al. [3] developed a system that visualizes and explains past actions of a robot. In other work, animation techniques of anticipation and reaction were used to help people predict what a robot will do next [22]. Hayes & Shah [9] drew an analogy between reviewing a robot's behavior to software debugging and developed methods that enable users to query the agent's behavior in different states and request explanations. Nikolaidis et al. [16] proposed a cross-training approach where the human and the agent switch roles in simulation to develop a better understanding of their teammate. Our work introduces a new approach which lets users review automatically generated summaries exemplifying the agent's behavior, without requiring them to manually specify states of interest or work with the agent directly. Our approach is complementary to the above approaches, and could be used in conjunction with them.

Last, the growing literature on interpretable machine learning [7, 24] introduced methods for algorithms and models more transparent to users. Simlarly to the methods for explaining MDP decisions, these approaches typically explain a one-shot decision (e.g. classification of a particular sample). This is done in different ways, e.g., by showing a simpler model which explains decisions in a particular region of the space [17]. Some methods aim to generate a user-understandable decision-making model more generally (e.g., using a prototype-based classification model [13]), but these do not address sequential decision-making settings and do not explicitly describe behavior in different scenarios.

3 SUMMARIZING AGENT BEHAVIORS

Our formalization of the summarization problem assumes that the agent uses a Markov Decision Process (MDP), where A is the set of actions available to the agent, S is the set of states, $R\colon S \times A \rightarrow \mathbb{R}$ is the reward function, which maps each state and action to a reward, and 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. The agent has a policy π which specifies which action to take in each of the states.

We formalize the problem of summarizing an agent's behavior as follows: from execution traces of an agent, choose a set $T = \langle t_1, ..., t_k \rangle$ of trajectories to include in the summary, where each trajectory is composed of a sequence of l consecutive states and the actions taken in those states $\langle (s_i, a_i), ..., (s_{i+l-1}, a_{i+l-1}) \rangle$. We consider trajectories rather than single states because seeing what action was taken by the agent in a specific state might not be meaningful without a broader context (e.g., watching a self-driving car for one second will not reveal much useful information). Because it is infeasible that people will be able to review the behavior of an agent in all possible states, we assume a limited budget k for the size of the summary, such that $|T| = k$. This budget limits the amount of time and cognitive effort that a person needs to invest in reviewing the agent's behavior. We discuss alternative formulations of the summarization problem in Section 8.

There are several factors that could be considered when deciding which states to include in a summary, such as the effect of taking a different action in that state, the diversity of the states that are included in the summary and the frequency at which states are likely to be encountered by the agent. In this paper, we focus on the first factor, which we refer to as the "importance" of a state. Intuitively, a good summary should provide a person reviewing the summary with a sense of the agent's behavior in states that the person considers important (e.g., when making a mistake would be very costly). The importance of states included in the summary could substantially affect the ability of a person to assess an agent's capabilities. For example, imagine a summary of self-driving car that only shows the car driving on a highway with no interruptions. This summary would provide people with very little understanding of how the car might act in other, more important, scenarios (e.g., when another car drives into its lane, when there is road construction). In contrast, a summary showing the self-driving car in a range on more interesting situations (e.g., overtaking another car, breaking when a person enters the road) would convey more useful information to people reviewing it.

In Section 4 we describe an algorithm that generates summaries based on this state importance criteria. We then extend the algorithm to also take into consideration the diversity of the states included in the summary (described in Section 4.1). That is, instead of considering each state in isolation when deciding whether to include it in the summary, the decision will also depend on the other states that are currently included in the summary. We discuss other possible desired summary properties in Section 8.

4 THE "HIGHLIGHTS" ALGORITHM

We developed HIGHLIGHTS, an algorithm that generates a summary of an agent's behavior from simulations of the agent in an online manner. HIGHLIGHTS uses the notion of state *importance* [23] to decide which states to include in the summary. 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. Formally, the importance of a state, denoted $I(s)$, is defined as:

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

This measure has been shown to be useful for choosing teaching opportunities in the context of student-teacher reinforcement learning [1, 23]. We note, however, that this measure has significant limitations (e.g., sensitivity to the number of possible actions) which we discuss in Section 8.

Before providing a detailed pseudo-code of the algorithm, we describe its operation at a high-level. HIGHLIGHTS generates a summary that includes trajectories that captures the most important states that an agent encountered in a given number of simulations. To do so, at each step it evaluates the importance of the state and adds it to the summary if its importance value is greater than the minimal value currently represented in the summary (replacing the minimal importance state). To provide more context to the user, for each such state HIGHLIGHTS also extracts a trajectory of states neighboring it and the actions taken in those states.

A pseudo-code of the HIGHLIGHTS algorithm is given in Algorithm 1. Table 1 summarizes the parameters of the algorithm. HIGHLIGHTS takes as input the policy of the agent π which is used to determine the agent's actions in the simulation and state importance values, the budget for the number of trajectories to include in the summary (k) and the length of each trajectory surrounding a state (l). Each such trajectory includes both states preceding the important state and states that were encountered immediately after it. The number of subsequent states to include is determined by the $statesAfter$ parameter (the number of preceding states can be derived from this parameter and l). We also specify the number of simulations that can be run (*numSimulations*), and the minimal "break" interval between trajectories (*intervalSize*) which is used to prevent overlaps between trajectories. HIGHLIGHTS outputs a summary of the agent's behavior, which is a set of trajectories (T).

Parameter	Description (value used in experiments)
k	Summary budget, i.e., number of trajectories (5)
l	Length of each trajectory (40)
$numSimulations$	The number of simulations run by HIGHLIGHTS (50)
$intervalSize$	Minimal number of states between two trajectories in the summary (50)
$statesAfter$	Number of states following s to include in the trajectory (10)

Table 1: Parameters of the HIGHLIGHTS algorithm and the values assigned to them in the experiments (in parentheses).

The algorithm maintains two data structures: T is a priority queue (line 2), which will eventually hold the trajectories chosen for the summary; t is a list of state-action pairs (line 3), which holds the current trajectory the agent encounters. The procedure runs simulations of the agent acting in the domain. At each step of the simulation, the agent takes an action based on its policy and advances to a new state (line 8). That state-action pair is added to the current trajectory (line 11). If the current trajectory reached its maximal length, the oldest state in the trajectory is removed (lines 9-10). HIGHLIGHTS computes the importance of s based on the Q-values of the agent itself, as defined in Equation 1 (line 14).

If a sufficient number of states were encountered since the last trajectory was added to the summary, state s will be considered for the summary (the $c == 0$ condition in line 17). s will be added to the summary if one of two conditions hold: either the size of the current summary is smaller than the summary size budget, or the importance of s is greater than the minimal importance value of a state currently represented in the summary (line 17). If one of these conditions holds, a trajectory corresponding to s will be added to the summary. The representation of a trajectory in the summary (a $summaryTrajectory$ object) consists of the set of state-action pairs in the trajectory (which will be presented in the summary), and the importance value I_s based on which the trajectory was added (such that it could be compared with the importance of states encountered later). This object (st) is initialized with the importance value (line 20) and is added to the summary (line 21), replacing the trajectory with minimal importance if the summary reached the budget limit (lines 18-19). Because the trajectory will also include states that follow s, the final set of state-action pairs in the trajectory is updated later (lines 15-16). Last, we set the state counter c to the interval size, such that the immediate states following s will not be considered for the summary. At the end of each simulation, the number of runs is incremented (line 24). The algorithm terminates when it reaches the specified number of simulations.

We chose to implement HIGHLIGHTS as an online algorithm because it is less costly, both in terms of runtime and in terms of memory usage. In addition, such an algorithm can be incorporated into the agent's own learning process without additional cost. The algorithm can be easily generalized to work offline.

4.1 Considering State Diversity

Because HIGHLIGHTS considers the importance of states in isolation when deciding whether to add them to the summary, the produced summary might include trajectories that are similar to each other. This could happen in domains in which the most important scenarios tend to be similar to each other. To mitigate this problem, we developed a simple extension to the HIGHLIGHTS algorithm, which we call HIGHLIGHTS-DIV. Similarly to HIGHLIGHTS, this algorithm also determines which states to include in the summary based on their importance. However, it also attempts

Algorithm 1: The HIGHLIGHTS algorithm.

```
Input: $\pi, k, l, numSimulations, intervalSize, statesAfter$
Output: $T$
$runs = 0$
$T \leftarrow PriorityQueue(k, importanceComparator)$
$t \leftarrow$ empty list
$c = 0$
while ($runs < numSimulations$) do
    $sim = InitializeSimulation()$
    while ($!sim.ended()$) do
        $(s, a) \leftarrow sim.advanceState(\pi)$
        if ($|t| == l$) then
            $t.remove()$
        $t.add((s, a))$
        if ($c > 0$) then
            $c = c - 1$
        $I_s \leftarrow computeImportance(\pi, s)$
        if ($IntervalSize - c == statesAfter$) then
            lastSummaryTrajectory.setTrajectory(t)
        if $((|T| < k)$ or $(I_s > minImportance(T)))$ and
         $(c == 0))$ then
            if $|T| == k$ then
                T.pop()
            $st \leftarrow$ new $summaryTrajectory(I_s)$
            $T.add(st)$
            $lastSummaryTrajectory \leftarrow st$
            $c = intervalSize$
    runs = runs+1
```

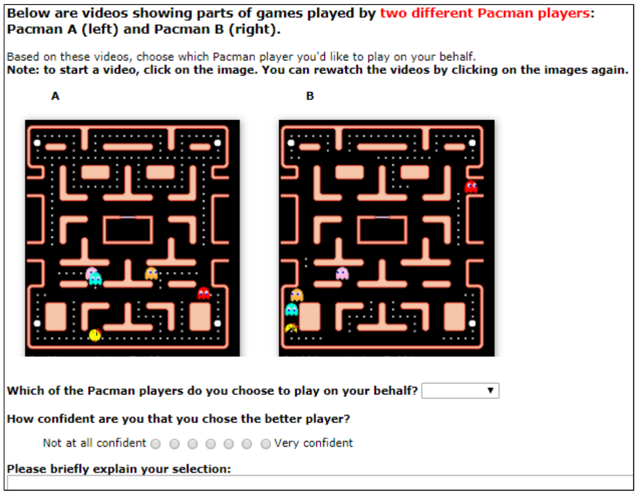
to avoid including a very similar set of states in the summary, thus potentially utilizing the summary budget more effectively.

HIGHLIGHTS-DIV takes into consideration the diversity of states in the following way: when evaluating a state s, it first identifies the state most similar to s that is currently included in the summary[2], denoted s'. Then, instead of comparing the importance of a state to the minimal importance value that is currently included in the summary, HIGHLIGHTS-DIV compares I_s to $I_{s'}$. If I_s is greater than $I_{s'}$, the trajectory which includes s' in the summary will be replaced with the current trajectory (which includes s). This approach allows less important states to remain represented in the summary (because they will not be compared to some of the more important states that differ from them), potentially increasing the diversity of trajectories in the summary and thus conveying more information to users.

5 EMPIRICAL METHODOLOGY

Empirical Domain. To evaluate HIGHLIGHTS and HIGHLIGHTS-DIV, We generated summaries of agents playing the Mrs. Pacman game [18]. Figure 1 shows a screen from the experiment which includes snapshots of the Pacman maze used in our experiments. This game configuration includes two types of food pellets: regular pellets (small dots) are worth 10 points each and power pellets (larger dots) are worth 50 points each. After eating a power pellet, ghosts become edible for a limited time period. Pac-Man receives 200 points for each ghost it eats. Ghosts chase Pac-Man with 80% probability and otherwise move randomly. In each state, Pac-Man has at most four moves (right, left, up or down). Important states in the game include situations where Pacman is very close to ghosts (e.g., the state shown for the Pacman game on the right side in Figure 1) or when Pacman has an opportunity to eat a power pellet, or a ghost.

Due to the large size of the state space, we used the high-level 7-feature representation from Torrey & Taylor's [23] implementation. Q-values are defined as a weighted function of the feature values, i.e., $Q(s, a) = \omega_0 + \sum_i \omega_i \cdot f_i(s, a)$



Below are videos showing parts of games played by two different Pacman players: Pacman A (left) and Pacman B (right).

Based on these videos, choose which Pacman player you'd like to play on your behalf.
Note: to start a video, click on the image. You can rewatch the videos by clicking on the images again.

Which of the Pacman players do you choose to play on your behalf? ▾

How confident are you that you chose the better player?

Not at all confident ○ ○ ○ ○ ○ ○ ○ Very confident

Please briefly explain your selection:

Figure 1: A screenshot of the agent selection task.

Experimental Conditions. In addition to generating summaries using the two versions of the HIGHLIGHTS algorithm, we also generated summaries using two baseline methods:

- *First*: a summary is generated from the first k trajectories Pacman encounters. This baseline is akin to having a user watch the agent act (e.g., watching a video of an autonomous vehicles driving) until she runs out of time.
- *Random*: a summary is generated by sampling k trajectories uniformly from the agent's execution trace. With this baseline states that are more frequently encountered are more likely to be selected to the summary.

The parameter values for the algorithms used to generate the Pacman summaries are listed in Table 1. All summaries included five trajectories ($k = 5$), each showing 40 neighboring states (l=40). They enforced a gap of 50 states before considering a state for inclusion in the summary (i.e., $intervalSize = 50$). To present the summaries to users, we generated video-clips (GIF files) showing the trajectories that were chosen for the summary[3]. We note that the summaries shown to participants did not include the current score

[2] We assume that distance metric to compare states can be defined. This can be done in many domains, e.g., by computing Euclidean distance if states are represented by feature vectors.

[3] See example summary video here: https://goo.gl/79dqsd.

of the agent (see Figure 1) to ensure that participants' evaluation will be based on the observed behavior of the Pacman player rather than its score.

We generated three different agents by varying their training period. The lowest performing agent was trained for 200 episodes (scores 2262 points on average), the medium agent for 400 episodes (scores 2732 points on average), and the highest performing agent was trained for 2000 episodes (scores 3826 points on average). Henceforth, we refer to these agents as the *200E*, *400E* and *2000E* agent, respectively. Generating agents of varying performance enabled us to have a ground truth when asking participants to assess the agents' performance. The summaries were generated after the agents were fully trained and reflect the final policies of the agents.

We conducted two experiments. Experiment 1 compared the HIGHLIGHTS algorithm with the two baseline methods. Experiment 2 compared the HIGHLIGHTS-DIV algorithm with the basic HIGHLIGHTS algorithm and the *Random* baseline. We used the same procedure in both experiments.

Procedure. Participants were first shown a tutorial explaining the rules of the Pacman game. They then had to pass a quiz ensuring they read and understood the rules. Next, they were asked to complete two different tasks (described next). Participants received a base payment of $1.5, and could earn a bonus of up to $0.9 (explained in task 1). We used a within-subject study design, such that all participants evaluated all summary methods.

Task 1: Agent Selection. In the first task, participants were shown pairs of summaries of two *different* Pacman agents, produced by the *same* summary method (e.g., a HIGHLIGHTS summary of the *200E* agent and a HIGHLIGHTS summary of the *400E*). They were asked to choose the agent they would like to play on their behalf. Participants were also asked to explain their selection and to rate their confidence in their decision on a 7-point Likert scale (1 - not at all confident to 7 - very confident). Overall, there were 9 such pairs (3 agent levels X 3 summarization methods). An example agent selection task is shown in Figure 1. The ordering of pairs to compare as well as which summary was shown on the left and which on the right were randomized. Participants were given a bonus of 10 cents for each correct agent selection, such that they had a monetary incentive to select the better performing agent.

The different agent comparisons differed in the difficulty of identifying the better agent: the *200E* and *400E* agents had the most similar performance, resulting in a *high-difficulty* comparison; the *200E* and *2000E* agents differed most substantially in their performance, resulting in a *low-difficulty* comparison; we refer to the comparison of the *400E* and *2000E* agents as the *medium-difficulty* comparison, as the differences in the agents' policies were more substantial than for the *200E* and *400E* agents, but less substantial than for the *200E* and *2000E* agents.

Task 2: Summary Preferences. While the first task measured participants' objective ability to identify the better agent, in the second task we elicited participants' subjective opinions about the helpfulness of different summaries. They were again shown pairs of summaries. This time the two summaries were of *the same* agent (participants were told it was the same agent), but were generated by a *different* summary method (e.g., comparing a HIGHLIGHTS summary of the 200E agent with a *Random* summary of the 200E agent). Participants were asked to rate which of the summaries they find more helpful for assessing the capabilities of the Pacman agent using a 7-point Likert scale (1 - video A is more helpful, 7 - video B is more helpful). They were also asked to provide a short explanation for their preference.

To maintain a reasonable experiment length and because we were primarily interested in the usefulness of HIGHLIGHTS summaries, in this task participants only made 4 comparisons (2 of the 3 agents, comparing HIGHLIGHTS summaries with each of the baseline summaries). The ordering of pairs to compare as well as their location on the screen (left or right) were randomized.

Evaluation Metrics and Analyses. The analysis of task 1 evaluated participants' correctness rate when selecting Pacman agents with each summary method. We analyzed the data using a logistic regression, controlling for the comparison type (200E vs. 400E agents, 400E vs. 2000E agents or 200E vs. 2000E agents). Since we used a within-subject design, we ran a repeated measures logistic regression. We also compared participants' confidence in making these selections. Confidence ratings were analyzed using an ordinal logistic regression, again controlling for the comparison type. For the fitted regression models, we report the significance of the coefficients as well as the odds ratio values (*OR*), which can be interpreted as effect sizes. Values between 1.5 and 3 are interpreted as a small effect, between 3 and 5 as medium, and above 5 as large [2, 4].

When analyzing task 2, we compared the helpfulness ratings given to the summaries. We normalized the preferences such that 7 always means "HIGHLIGHTS is more helpful" and 1 means "[other method] is more helpful". That is, a rating greater than 4 indicates a preference for HIGHLIGHTS. We analyzed these ratings using the non-parametric Wilcoxon rank sum test.[4]

To account for multiple hypotheses testing, we adjusted p-values with the Holm's sequentially rejective Bonferroni procedure [10, 20]. We report raw p-values, but in all cases we state significant differences, the adjusted p-values were also smaller than 0.05.

6 EXPERIMENT 1: BASIC HIGHLIGHTS

Experiment 1 compared summaries generated by the basic HIGHLIGHTS algorithm, which only considers state importance, with summaries generated with the *Random* and *First* baselines. 40 participants were recruited through Amazon Mechanical Turk (23 female, Mean age = 35.35, STD = 10.4).

Agent Selection Results. As shown in Figure 2, participants were more likely to choose the better performing agents when shown summaries generated by the HIGHLIGHTS algorithm compared to the baselines. The analysis shows statistically significant and substantial differences in performance when comparing HIGHLIGHTS to *First* ($\chi^2 = 49.79, p < 1^{-10}, OR = 12.09$) and when comparing HIGHLIGHTS to *Random* ($\chi^2 = 6.93, p = 0.001, OR = 2.38$).

When comparing HIGHLIGHTS and *First*, we found a significant difference for all three agent comparison types (low, medium and high difficulty). When comparing HIGHLIGHTS to *Random*, we observed a significant difference only for the *medium-difficulty*

[4] We used Wilcoxon rank sum as the scale was ordinal and the data was not normally distributed. However, we obtain similar results when using standard t-test.

comparison (*400E* vs. *2000E* agents). This makes sense as the difference between the *200E* and *400E* agents is relatively small, making the comparison hard with any summary. The difference between the *200E* vs. *2000E* agents is very substantial, making it easier to identify the better agent even with random trajectories. Interestingly, for the low-difficulty comparison, the summaries generated by *First* were particularly misleading. We hypothesize the reason for this is that participants saw the *2000E* agent taking more risks initially, as participants' explanations often referred to the *2000E* agent behavior as risky, e.g. "Player B [2000E] made some risky turns which will end his play before Player A [200E]."

We observed different types of explanations provided by participants. Some explanations referred directly to the capabilities demonstrated by the agent in the summary, e.g. "B [2000E] seems like they are better at actually eating the ghosts". Other participants noted the state of the board shown in different summaries, e.g. "B has more of the screen cleared". Some explanations described the general behavior of the agent, e.g. "Pacman B seems to be effective at routing" or how Pacman's strategy compares to their own strategy, e.g. "He went the way I would have." Last, some explanations referred to specific events, e.g. "Pacman a looked like it was about to be cornered."

The type of explanation provided often depended on the summary method used and the difficulty of the agent comparison. Participants' explanations when shown HIGHLIGHTS summaries for the low-difficulty and medium-difficulty comparisons typically referred to specific capabilities they observed, e.g. "Player A [2000E] is eating ghost so earning more points." We observed fewer such explanations for the high-difficulty comparison, e.g. "Player B has eaten one power pill which means he's had the chance to go after the ghosts (for more points) at least once. Also seems to have eaten more dots on the whole than Player A." Explanations for this comparison more often pointed to the state of the board or provided a general impression of the agent's behavior. For the *First* summaries, participants typically conveyed their general impression of the agent's behavior. Explanations for the *Random* baseline were similar to those given in the HIGHLIGHTS condition for the low-difficulty comparison, but tended to refer to more general agent behaviors for the medium- and high-difficulty comparisons (the analysis of these summaries is more difficult as each participant could observe a different *Random* summary).

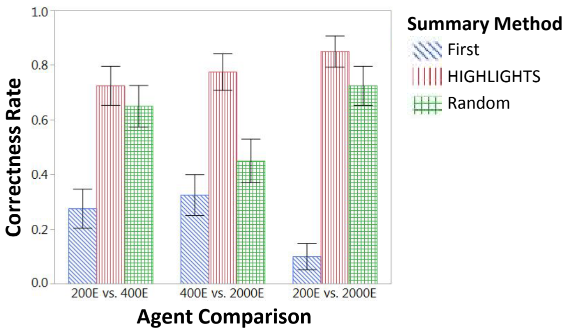


Figure 2: Correctness rate of participants in the agent selection task (Experiment 1).

Figure 3 shows the distribution of participants' confidence ratings when choosing an agent to play on their behalf. Participants were more confident in their choice of agents when presented with HIGHLIGHTS summaries than they were when presented with summaries generated by *First*. The differences were significant for the medium-difficulty ($\chi^2 = 22.04, p < 0.001, OR = 7.44$) and low-difficulty ($\chi^2 = 13.84, p < 0.001, OR = 3.64$) comparisons. We observed mixed results when comparing participants' confidence when reviewing HIGHLIGHTS and *Random* summaries. When presented with the low-difficulty comparison, participants were significantly more confident when shown *Random* summaries ($\chi^2 = 6.96, p = 0.008, OR = 2.214$) When making the low-difficulty comparisons, participants were significantly more confident with HIGHLIGHTS summaries ($\chi^2 = 5.819, p = 0.016, OR = 2.3$). Interestingly, we found no difference in confidence for the medium-difficulty comparison, although participants performed significantly better with HIGHLIGHTS summaries in this agent comparison task. This suggests that people's confidence might not correlate with their actual ability to assess agents' capabilities. We hypothesize that a reason for this is that they only get to review a short summary, and they might think it was sufficient because they are unaware of the information that was *not* included in the summary.

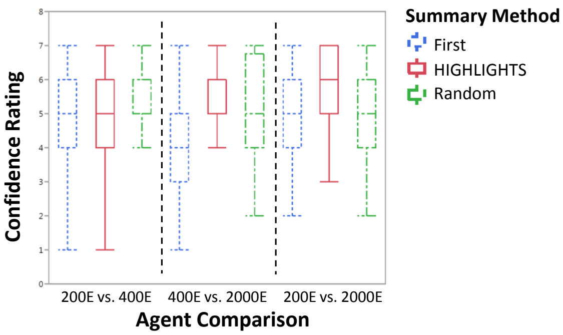


Figure 3: Participants' confidence rating on a scale of 1–7 when selecting an agent (Experiment 1).

Summary Preferences. The distribution of participants' subjective preferences ratings for the different summaries are shown in Figure 4. Recall, that a rating closer to 7 means they stated that the summary generated by HIGHLIGHTS was more helpful in assessing the agent's capability, while a rating closer to 1 indicates that they found the other summary (generated by either *First* or *Random*) as more helpful. That is, ratings greater than 4 indicate a preference for HIGHLIGHTS. The ratings are shown separately for each type of agent for which summaries were presented.

On average, participants preferred summaries generated by HIGHLIGHTS over summaries generated by *First* ($Median = 6$) and summaries generated by *Random* ($Median = 5$). The only statistically significant differences in preferences were for the highest performing agent (*2000E*). The ratings were significantly greater than 4 both when comparing HIGHLIGHTS with *First* ($Median = 7, p < 0.001$) and when comparing HIGHLIGHTS with *Random* ($Median = 6, p = 0.009$). We attribute this stronger preference to the greater difference between summaries generated by different methods when

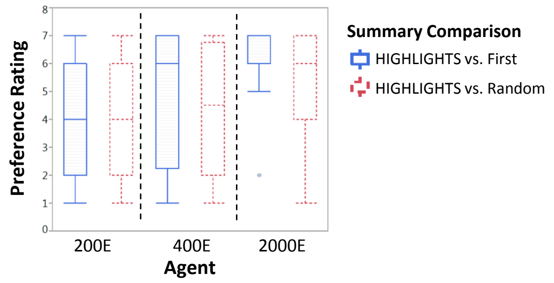


Figure 4: Participants' preference rating on a scale of 1–7, where 7 = "HIGHLIGHTS is more helpful" (Experiment 1).

considering agents that have more capabilities. For example, the highest performing Pacman agent was able to both escape ghosts and eat the power pellets which let it eat ghosts. The HIGHLIGHTS summary for this agent included trajectories demonstrating this capability, while *First* summaries did not show it, and only some *Random* summaries did. Participants often referred to the skills demonstrated in the summary when explaining their ratings of these summaries, e.g. "A [HIGHLIGHTS] was better for showing how good they were at leading the ghosts." In contrast, the lowest performing agent did not have many capabilities, and therefore there was less difference between the summaries generated by the different methods. Another possible explanation for this difference is that because the *200E* and *400E* agents are less trained, their Q-values are less accurate, making their judgment of state importance inferior, and thus potentially making the summaries less useful.

7 EXPERIMENT 2: HIGHLIGHTS-DIV

Experiment 2 compared summaries generated by the HIGHLIGHTS-DIV algorithm with summaries generated by the basic HIGHLIGHTS algorithm and summaries generated by the *Random* baseline (which significantly outperformed the *First* baseline in Experiment 1). We recruited 48 participants through Amazon Mechanical Turk (25 female, Mean age = 36, STD = 11.6).

Agent Selection Results. Figure 5 shows the correctness rates obtained by participants in experiment 2 for each of the summary methods. We begin by comparing HIGHLIGHTS summaries and HIGHLIGHTS-DIV summaries, which is the main focus of this experiment. When making the high-difficulty comparison (*200E* vs. *400E* agents), participants were more likely to identify the superior agent when presented with HIGHLIGHTS-DIV summaries ($\chi^2 = 7.16, p = 0.007, OR = 4.2$). We note that participants' performance when presented with HIGHLIGHTS summaries was lower than that of participants in Experiment 1 for the high-difficulty agent comparison. However, since we used a within-subject design, if participants were less attentive, they should also be less successful when presented with HIGHLIGHTS-DIV summaries. The difference between HIGHLIGHTS and HIGHLIGHTS-DIV remains significant even when aggregating the HIGHLIGHTS data from both experiments. We did not find significant differences for the medium- and low-difficulty agent comparisons.

When comparing the performance of participants when presented with *Random* summaries with their performance when presented with HIGHLIGHTS or HIGHLIGHTS-DIV summaries, we did not observe an interaction effect between summary and comparison type. Therefore, we fit a single model for each summary comparison (*Random* vs. HIGHLIGHTS and *Random* vs. HIGHLIGHTS-DIV). We found a statistically significant effect for both comparisons, with participants being less successful when presented with *Random* summaries (*Random* vs. HIGHLIGHTS: $\chi^2 = 20.686, p < 0.001, OR = 3.06$; *Random* vs. HIGHLIGHTS-DIV: $\chi^2 = 27.28, p < 0.001, OR = 5.15$). These results reinforce the results of Experiment 1 which showed improved performance with HIGHLIGHTS summaries compared to *Random* summaries. Moreover, while in Experiment 1 we found a significant difference only for the difficult agent comparison, here we observed significant differences for all agent comparisons.[5] When aggregating the data from both experiments for HIGHLIGHTS and *Random*, we find significant differences for all agent comparisons, strengthening the conclusions from Experiment 1.

The explanations given by participants for their agent choices were similar to those given by participants in Experiment 1. When making the high-difficulty comparison with the HIGHLIGHTS-DIV summaries, participants often referred to the fact that the summary of the *200E* agent included a trajectory were Pacman was eaten, e.g. "Player A [200E] gets caught by the ghosts at least once so it looks like Player B [400E] might have the better game", or to the fact that the *400E* agent was shown eating ghosts, e.g. "Player B [400E] was able to eat some power pills and blue ghosts to gain more points." Explanations for the other comparisons were similar to those given when presented with HIGHLIGHTS summaries described in Section 6.

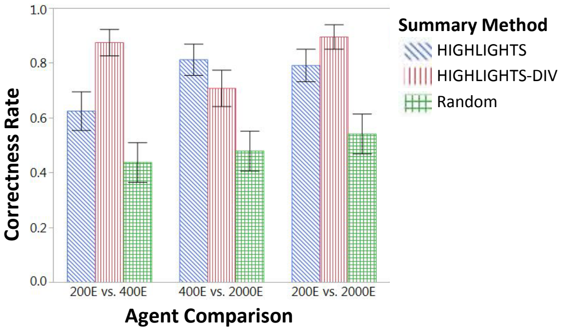


Figure 5: Correctness rate of participants in the agent selection task (Experiment 2).

As in Experiment 1, we observed mixed results in terms of participants' confidence alignment with their actual performance. For the high-difficulty and low-difficulty agent comparisons, Participants' confidence ratings were significantly higher when presented with HIGHLIGHTS-DIV summaries compared to their confidence when reviewing HIGHLIGHTS summaries (high-difficulty: $\chi^2 = 17.84, p < 0.001, OR = 3.68$; low-difficulty: $\chi^2 = 11.14, p = 0.001, OR = 2.82$). We did not observe a statistically significant difference for the medium-difficulty comparison. The higher confidence when shown HIGHLIGHTS-DIV summaries is in line with

[5] It is reasonable to observe different performance levels with *Random* summaries, as different participants could observe different *Random* summaries.

participants' objective performance for the high-difficulty comparison. We did not find any differences in confidence ratings between HIGHLIGHTS and *Random*, despite the significantly higher objective performance of participants when shown HIGHLIGHTS summaries. When comparing participants' confidence in the HIGHLIGHTS-DIV and *Random* conditions, we found a statistically significant difference for the low-difficulty agent comparison ($\chi^2 = 16.24, p < 0.001, OR = 3.72$).

Summary Preferences. When presented with summaries of the *400E* agent, participants preferred the HIGHLIGHTS-DIV summary over the HIGHLIGHTS summary, though this preference was only marginally significant ($Median = 3, p = 0.1$). When explaining their ratings of these summaries participants often referred to skills demonstrated in the summary, e.g. "The types of explanations provided were similar to those given in Experiment 1 and often referred to the skills demonstrated in the summary, e.g. "It [HIGLIGHTS-DIV] shows how well the player baited the ghosts and then was able to snack on them." We found no difference in preferences between the two summary methods when reviewing summaries of the *2000E* agent ($Median = 4$). When comparing participants' preferences for HIGHLIGHTS and *Random* summaries, we observed similar results to those obtained in Experiment 1 ($Median = 5$). As in Experiment 1, the difference was significant only when comparing summaries of the *2000E* agent ($Median = 5, p = 0.05$). We did not directly compare HIGHLIGHTS-DIV with *Random* summaries, but the results suggest a preference for HIGHLIGHTS-DIV summaries (as they were preferred over HIGHLIGHTS summaries, which were preferred over *Random* summaries).

8 DISCUSSION & FUTURE WORK

With the growing use of intelligent agents, it is important to provide ways for people become more familiar with the the behaviors of such agents, their capabilities and limitations. This paper proposes a new approach for increasing the familiarity of users with agents – generating summaries of agent behaviors. Our results show that presenting people with "highlights" summaries of an agent behavior can help people evaluate the capabilities of different agents. These results provide initial evidence for the potential usefulness of the proposed approach. We next discuss several limitations of the developed HIGHLIGHTS algorithms and possible ways to address them, as well as additional directions for future work.

The algorithms we described can be improved in several ways. First, the importance measure we used is sensitive to the distribution of Q-values in different states (e.g., it might not make sense in domains where there are many possible actions at any given state, because agents will never consider the worst action). In future work we will define more sophisticated importance measures, e.g., by considering the variance in Q-values or regret values. In addition, because the importance assessment was based on the agent's own Q-values, different agents might consider different states as important, and in particular low-quality agents might not be able to recognize states that people will consider important. To mitigate this problem, we will explore ways of assessing importance that do not rely solely on the judgment of the agent itself. For example, aggregating importance values of different agents. In addition, importance could be computed for an entire trajectory rather than for a specific state. Similarly, the diversity of the summary can also be computed based on complete trajectories. We note that while our approach assumed an MDP representation for importance computations, similar notions could also be defined for other decision-making models. For example, with hierarchical plans it might be possible to define a measure that assesses the impact of an action on the ability to achieve a goal.

While considering importance and diversity criteria already improved people's ability to evaluate the performance of different agents compared to the baselines, there are other criteria that should be taken into consideration when generating summaries. For example, in our experiments, participants sometimes referred to specific events when justifying their choice of agents. To ensure that people do not overweight or underweight specific events, the likelihood of encountering states should be reflected in the summary and conveyed to users. In addition, we hypothesize that different summaries may be effective in different contexts. For instance, if the goal of the user is to compare two agents, summaries highlighting states in which their actions differ might be more helpful than summaries that produced for each of the agents separately. Evaluation criteria for summaries can also be extended to include additional metrics such as the ability of people to predict an agent's actions and their ability to collaborate with the agent on a task.

Our formulation of the summary generation problem assumed a limited budget for the number of trajectories that can be included in the summary. A different way of approaching strategy summarization is framing it as an optimization problem where the goal is to create a minimal summary that satisfies certain criteria (e.g., with respect to coverage of the state space). We will explore such formulations in future work.

The presentation of summaries is likely to depend on the characteristics of different domains. In the Pacman domain used in our study, presenting a video-clip of the agent was appropriate for conveying the agent's behavior, and showing trajectories that include neighboring states provided people with sufficient context for assessing the agent's actions. This approach could apply more generally to domains where there is a physical agent (e.g., a robot or a self-driving car), but may not be appropriate for some virtual agents (e.g., a personal assistant). In the latter domains, different visualization methods of states will likely be required (e.g., showing some feature-based representation of a state).

Finally, while automatically generated summaries can provide users with a basic overview of an agent's behavior, in some situations users may require more detailed information, tailored to their needs. To this end, we plan to design collaborative interfaces that let people adjust summaries and explore the behavior of agents in different states. This is particularly important as our experiment showed that people's confidence did not always correlate with the correctness of their assessments, highlighting the importance of providing users with more information about the summaries they observe and more ways to explore them.

ACKNOWLEDGMENTS

We thank Avi Segal, Ken Arnold and Sebastian Gehrmann for their feedback on drafts of this paper.

REFERENCES

[1] Ofra Amir, Ece Kamar, Andrey Kolobov, and Barbara Grosz. 2016. Interactive Teaching Strategies for Agent Training. *IJCAI* (2016).

[2] Michael Borenstein, Larry V Hedges, Julian Higgins, and Hannah R Rothstein. 2009. Converting among effect sizes. *Introduction to meta-analysis* (2009), 45–49.

[3] Daniel J Brooks, Abraham Shultz, Munjal Desai, Philip Kovac, and Holly A Yanco. 2010. Towards State Summarization for Autonomous Robots.. In *AAAI Fall Symposium: Dialog with Robots*, Vol. 61. 62.

[4] Henian Chen, Patricia Cohen, and Sophie Chen. 2010. How big is a big odds ratio? Interpreting the magnitudes of odds ratios in epidemiological studies. *Communications in StatisticsâĂŤSimulation and Computation®* 39, 4 (2010), 860–864.

[5] Sandra Devin and Rachid Alami. 2016. An implemented theory of mind to improve human-robot shared plans execution. In *Human-Robot Interaction (HRI), 2016 11th ACM/IEEE International Conference on*. IEEE, 319–326.

[6] Thomas Dodson, Nicholas Mattei, and Judy Goldsmith. 2011. A natural language argumentation interface for explanation generation in Markov decision processes. *Algorithmic Decision Theory* (2011), 42–55.

[7] Finale Doshi-Velez and Been Kim. 2017. Towards a rigorous science of interpretable machine learning. (2017).

[8] Alyssa Glass, Deborah L McGuinness, and Michael Wolverton. 2008. Toward establishing trust in adaptive agents. In *Proceedings of the 13th international conference on Intelligent user interfaces*. ACM, 227–236.

[9] Bradley Hayes and Julie A Shah. 2017. Improving Robot Controller Transparency Through Autonomous Policy Explanation. In *Proceedings of the 2017 ACM/IEEE International Conference on Human-Robot Interaction*. ACM, 303–312.

[10] S. Holm. 1979. A simple sequentially rejective multiple test procedure. *Scandinavian Journal of Statistics* 6, 65-70 (1979), 1979.

[11] O Khan, Pascal Poupart, J Black, LE Sucar, EF Morales, and J Hoey. 2011. Automatically generated explanations for Markov decision processes. *Decision Theory Models for Applications in Artificial Intelligence: Concepts and Solutions* (2011), 144–163.

[12] Omar Zia Khan, Pascal Poupart, and James P Black. 2009. Minimal Sufficient Explanations for Factored Markov Decision Processes.. In *ICAPS*.

[13] Been Kim, Cynthia Rudin, and Julie A Shah. 2014. The bayesian case model: A generative approach for case-based reasoning and prototype classification. In *Advances in Neural Information Processing Systems*. 1952–1960.

[14] Meghann Lomas, Robert Chevalier, Ernest Vincent Cross II, Robert Christopher Garrett, John Hoare, and Michael Kopack. 2012. Explaining robot actions. In *Proceedings of the seventh annual ACM/IEEE international conference on Human-Robot Interaction*. ACM, 187–188.

[15] Karen L Myers. 2006. Metatheoretic Plan Summarization and Comparison.. In *ICAPS*. 182–192.

[16] Stefanos Nikolaidis and Julie Shah. 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*. IEEE Press, 33–40.

[17] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. 2016. Model-agnostic interpretability of machine learning. *arXiv preprint arXiv:1606.05386* (2016).

[18] Philipp Rohlfshagen and Simon M Lucas. 2011. Ms pac-man versus ghost team cec 2011 competition. In *Evolutionary Computation (CEC), 2011 IEEE Congress on*. IEEE, 70–77.

[19] Bastian Seegebarth, Felix Müller, Bernd Schattenberg, and Susanne Biundo. 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*.

[20] Juliet P. Shaffer. 1995. Multiple Hypothesis-Testing. *Annual Review of Psychology* 46 (1995), 561–584.

[21] Peter Stone, Rodney Brooks, Erik Brynjolfsson, Ryan Calo, Oren Etzioni, Greg Hager, Julia Hirschberg, Shivaram Kalyanakrishnan, Ece Kamar, Sarit Kraus, Kevin Leyton-Brown, David Parkes, Press William, Saxenian AnnaLee, Shah Julie, Tambe Milind, and Teller Astro. 2016. Artificial intelligence and life in 2030. *One Hundred Year Study on Artificial Intelligence: Report of the 2015-2016 Study Panel* (2016).

[22] Leila Takayama, Doug Dooley, and Wendy Ju. 2011. Expressing thought: improving robot readability with animation principles. In *Proceedings of the 6th international conference on Human-robot interaction*. ACM, 69–76.

[23] Lisa Torrey and Matthew Taylor. 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*. International Foundation for Autonomous Agents and Multiagent Systems, 1053–1060.

[24] Alfredo Vellido, José David Martín-Guerrero, and Paulo JG Lisboa. 2012. Making machine learning models interpretable.. In *ESANN*, Vol. 12. 163–172.