Generating Impact and Critique Explanations of Predictions made by a Goal Recognizer

Jair da Silva Ferreira Jr.
Dept. of Data Science and AI
F. of Information Technology
Monash University, Australia
jair.dasilvaferreirajunior@monash.edu

Ingrid Zukerman
Dept. of Data Science and AI
F. of Information Technology
Monash University, Australia
ingrid.zukerman@monash.edu

Cecile Paris
CSIRO Data61, Australia
cecile.paris@data61.csiro.au

Enes Makalic
Dept. of Data Science and AI
F. of Information Technology
Monash University, Australia
enes.makalic@monash.edu

Mor Vered
Dept. of Data Science and AI
F. of Information Technology
Monash University, Australia
mor.vered@monash.edu

Abstract

In this paper, we generate two types of explanations, *Impact* and *Critique*, of predictions made by a Goal Recognizer (GR) – a system that infers agents' goals from observations. Impact explanations describe the top-predicted goal(s) and the main observations that led to these predictions. Critique explanations augment these explanations with evidence that challenges the GR's predictions if so warranted.

Our user study compares users' goal recognition accuracy for Impact and Critique explanations, and users' views about these explanations, under three prediction-correctness conditions: correct, partially correct and incorrect. Our results show that (1) users stick with a GR's predictions, even when a Critique explanation highlights its flaws; yet (2) Critique explanations are deemed better than Impact explanations in most respects.

1 Introduction

Goal recognition is a field that infers agents' goals from observations.[1] Sample applications include games (Singh et al., 2020) and autonomous vehicles (Yurtsever et al., 2020). Goal Recognizer (GR) predictions are characterized by the existence of a partial order between observations, i.e., in general, certain actions can be performed only after some other actions, which differs from feature independence or correlational dependencies between features in Machine Learning (ML). In this paper, we offer two approaches for generating explanations of the predictions of GRs; and compare their effect on users' goal recognition accuracy (how accurately they assign likelihoods to goals) and users' views of the explanations.

[1]We focus on *keyhole recognition*, where agents' goals are inferred from unobtrusive observations (Carberry, 2001).

Our explanation-generation approach follows Biran and McKeown (2017)'s human-centered view, whereby an explanation is "not about the model, but about the evidence that led to the prediction". Our explanations are aimed at non-expert users wishing to understand the reasons for a prediction. Specifically, we offer domain-agnostic algorithms for generating two types of explanations:

- *Impact* explanations describe the top-predicted goal(s) and the most influential observations that led to these predictions.
- *Critique* explanations, inspired by (Kim et al., 2016), augment Impact explanations with evidence that challenges a GR's predictions if so warranted. These explanations differ markedly from Impact explanations when the predictions are not entirely plausible.

XAI systems generally do not vary the type of explanation they generate, irrespective of whether the ML model produced correct or wrong predictions, leaving it to users to discern between these options. In contrast, the Critique explanations generated in this research differ for predictions that seem plausible and predictions that require further scrutiny. In our evaluation, we distinguish between three levels of GR prediction correctness (correct, partially correct and incorrect) and three explanatory conditions (prediction only, Impact and Critique), and examine their influence on users' goal recognition accuracy and opinions about the explanations. For the latter we employ six explanatory attributes from (Hoffman et al., 2018; Maruf et al., 2024).

Our results show that users stick with the GR's prediction, even when a Critique explanation highlights its flaws; yet Critique explanations are deemed better than Impact ones in most respects.

Proceedings of the 18th International Natural Language Generation Conference, pages 554–575
October 29–November 2, 2025.

Our main contributions are: (1) two explanation types with algorithms that select goals and observations to be mentioned; (2) an evaluation approach that distinguishes between different levels of prediction correctness; and (3) results about the effect of explanations on users' goal recognition accuracy and about users' explanation-type preferences.

This paper is organized as follows. Section 2 discusses related work. Our demonstration domain appears in Section 3, and our explanation-generation algorithms in Section 4. Section 5 describes our user study, followed by its results in Section 6. Section 7 presents concluding remarks.

2 Related Work

XAI is a subfield of AI that studies the generation of explanations for the predictions made by ML models. XAI has recently gained popularity owing to the success of ML models and the opacity of powerful ML models such as neural networks.

We focus on *Explainable Goal Recognition* (*XGR*) which generates explanations for GR predictions (Alshehri et al., 2023). Even though the number of GR applications is growing, e.g., traffic monitoring (Pynadath and Wellman, 2013), human-robot interaction (Buerkle et al., 2021), games (Singh et al., 2020) and autonomous vehicles (Yurtsever et al., 2020), generating explanations for GR predictions has gathered relatively little attention. A notable exception is the autonomous vehicle domain, where several models have been developed to explain the predictions of GRs regarding the intentions of other vehicles (Albrecht et al., 2021; Hanna et al., 2021; Brewitt et al., 2021, 2023). These XGR models leverage GR transparency to extract intuitive explanations from a GR's internal features — an approach that makes sense in this high-stakes domain, but may not be feasible for every domain. Hence, it seems advisable to develop model-agnostic XGR approaches which treat GRs as a black box.

Goal recognition resembles time-series classification (TSC) in that both process observations over time, and yield a distribution over predicted outcomes. However, unlike goal recognition, an entire distribution must be explained for TSC, which is often done using heatmaps (Theissler et al., 2022).

Feature attribution methods explain how individual features influence ML model predictions by distributing importance across features (Ribeiro et al., 2016; Lundberg and Lee, 2017), which aligns with human-centered XAI goals (Biran and McKeown, 2017). While some feature-attribution methods can handle correlational dependencies between features (Aas et al., 2021; Janizek et al., 2021), the dependencies in planning, and hence in goal recognition, are structural (e.g., preconditions, goal hierarchies). This motivates our Critique method, which explicitly accounts for such dependencies.

Alshehri et al. (2023, 2025) adapted feature attribution to goal recognition, identifying one key observation supporting or opposing a goal G or G' to address *"Why G?"* or *"Why not G'?"* questions respectively; they also generate counterfactual explanations that suggest alternative actions leading to different predictions. Our work differs from Alshehri et al. (2023, 2025)'s in several key respects: (1) we automatically identify top-ranked goals (Section 4.1), removing the need for users to specify goals initially; (2) we avoid the potential introduction of bias during observation selection, and determine which observations to mention – often more than one (Section 4.2); and (3) we introduce evidence-based Critique explanations that highlight potential inconsistencies between the GR's predictions and domain knowledge (Sections 4.3 and 4.4).

3 Domain and Dataset

We use the *Rovers* dataset (Pereira and Meneguzzi, 2017) to demonstrate our algorithms. This dataset contains (1) a set of missions (goals) attempted by planetary rovers, which require different types of data from specific *waypoints* (locations); (2) rovers with possibly different capabilities (e.g., collecting rocks or soil, traveling between specific waypoints), who collaborate to achieve a mission; and (3) a sequence of observations (which may miss some entries) of the actions performed by the rovers and their waypoints. Given a sub-sequence of observations, the GR predicts which mission(s) are being attempted by the rovers.

Table 1 displays three segments that describe a problem instance: *Missions' Specifications*, *Rovers' Capabilities* and *Observations*. For example, according to the Missions segment, Mission M_1 requires data about rock samples from waypoints 2 and 6, and about soil samples from waypoints 2, 3 and 5; and according to the Rovers segment, Rover0 can collect soil samples only, starts at waypoint 4, and can navigate between waypoint 0 and 1 and between waypoint 0 and 7 among others. The Observations segment shows ten observations made so far, e.g., Rover1 navigated to waypoint 2 (Obs. 1), where it sampled rock (Obs. 2), and then

Table 1: Scenario from our user study: six missions, three rovers, and ten observations. Missions require rock and/or soil data from specific waypoints. Rovers can sample rock and/or soil, start at a specific waypoint, and navigate between waypoints — denoted as $origin \leftrightarrow (dest_1, \ldots, dest_n)$.

Missions' Specifications

Mission Id	Rock Wpts.	Soil Wpts.	Mission Id	Rock Wpts.	Soil Wpts.
M_1	2, 6	2, 3, 5	M_4	2	2, 4, 5, 7
M_2	2, 6	2, 5, 6	M_5	0, 6	2, 5, 7
M_3	6, 7	2, 5, 7	M_6	2, 6, 7	2, 6

Rovers' Capabilities

Rover	Coll.	Init. Wpt.	Waypoint Navigability
Rover0	Soil only	4	$0 \leftrightarrow (1,7); 1 \leftrightarrow (0,4,6);$ $2 \leftrightarrow (3,4,5); 3 \leftrightarrow (2,5,7);$ $4 \leftrightarrow (1,2); 5 \leftrightarrow (2,3);$ $6 \leftrightarrow (1,7); 7 \leftrightarrow (0,3)$
Rover1	Rock only	5	$0 \leftrightarrow (4,6); 1 \leftrightarrow 3;$ $2 \leftrightarrow (5,6); 3 \leftrightarrow (1,4,5,6);$ $4 \leftrightarrow (0,3,7); 5 \leftrightarrow (2,3);$ $6 \leftrightarrow (0,2,3); 7 \leftrightarrow 4$
Rover2	Soil only	2	$0 \leftrightarrow (3,5); 1 \leftrightarrow 4; 2 \leftrightarrow 6;$ $3 \leftrightarrow 0; 4 \leftrightarrow (1,5);$ $5 \leftrightarrow (0,4,6); 6 \leftrightarrow (2,5)$

Observations

1-Rover1 nav. Wpt. $5 \rightarrow 2$	6-Rover2 nav. Wpt. $6 \rightarrow 5$
2-Rover1 samp. rock Wpt. 2	7-Rover0 samp. soil Wpt. 2
3-Rover1 sent rock Wpt. 2	8-Rover0 sent soil Wpt. 2
4-Rover2 nav. Wpt. $2 \rightarrow 6$	9-Rover0 dropped sample
5-Rover0 nav. Wpt. $4 \rightarrow 2$	10-Rover0 nav. Wpt. $2 \rightarrow 3$

Table 2: GR predictions for the problem in Table 1 and explanations for the predictions. Impact information is blue shaded and critique information is yellow shaded.

GR's Predictions with their Probabilities

$M_2 = 0.237$	$M_6 = 0.225$	$M_1 = 0.184$
$M_4 = 0.177$	$M_3 = 0.099$	$M_5 = 0.079$

Impact Explanation

According to the AI, **Missions #2** and **#6** are the most likely (probabilities of 23.7% and 22.5%, respectively). The observation that most influenced this result was:

- (#1) "*Rover1* navigated from *Waypoint5* to *Waypoint2*", which increased the probabilities of **Missions #2** and **#6** by about 10% and 8%, respectively.

Critique Explanation

According to the AI, **Missions #2** and **#6** are the most likely (probabilities of 23.7% and 22.5%, respectively). The observation that most influenced this result was:

- (#1) "*Rover1* navigated from *Waypoint5* to *Waypoint2*", which increased the probabilities of **Missions #2** and **#6** by about 10% and 8%, respectively.

Up to now:

- Even though **Missions #2** and **#6** are the most likely missions, a few observations do **not** contribute to any of their requirements. For example: *Observation #10* "*Rover0* navigated from *Waypoint2* to *Waypoint3*" contributes to *"send soil data from Waypoint3"* and *"send soil data from Waypoint7"*, which are not required by **Missions #2** and **#6**; and
- Even though **Missions #1** and **#4** are **not** the most likely missions, all the observations contribute to their requirements. For example: *Observation #10* (above) contributes to requirements of **Missions #1** and **#4**.

sent data about this rock sample (Obs. 3).

Table 2 shows the predictions generated by the GR in (Vered et al., 2018) for this instance with their probabilities, and the corresponding Impact and Critique explanations. The Impact explanation lists the top-ranked missions with their probabilities, and presents the observation(s) that had the largest influence on these probabilities – only one observation in this example (blue shaded). The Critique explanation reiterates this information, and presents two types of information that are incongruent with the GR's prediction (yellow shaded): (i) challenges to the predicted goals (M_2 and M_6), and (ii) support for goals that are *not* predicted (M_1 and M_4).

4 Generating Explanations

To generate Impact explanations, we select top-predicted goals and identify observations with the highest impact on these predictions. These are then fed into explanatory templates. Additionally, Critique explanations require evidence that supports or challenges a GR's predictions. This evidence is obtained from a domain model. Here, we provide details about these steps.

4.1 Selecting top-ranked goals

Our approach to selecting goals for inclusion in an explanation balances conciseness and completeness: mentioning every goal may lead to verbose explanations, while focusing only on the highest-probability goal may omit important information.

Computing confidence intervals (CIs) for goal probabilities plays a central role in our approach. However, GRs do not provide these intervals. To overcome this limitation, we introduce a novel technique leveraging the Dirichlet distribution – a multivariate continuous distribution widely used to model proportions and compositional data (Ng et al., 2011). Estimating the Dirichlet parameters from goal probabilities enables us to calculate robust CIs, which we use to discriminate between higher-probability goals to include in an explanation and lower-probability goals to exclude.

Goals to be included are determined by two criteria: (i) they have the highest probability after the last observation; or (ii) their probability is

greater than or equal to the lower bound of the CI for the highest-probability goal. The idea behind criterion (ii) is that if the probability of a not-top-ranked goal falls within the CI of a top-ranked goal, their probabilities are statistically indistinguishable. This suggests that the not-top-ranked goal is a plausible alternative given the GR's uncertainty. As an example, consider the final goal probability distribution in the GR's Predictions segment in Table 2, where Mission M_2 satisfies criterion (i). Assuming the estimated CI for M_2 is $[0.222, 0.244]$, M_6 is included in the explanation, as it meets criterion (ii), unlike M_1, M_3, M_4 and M_5, which are excluded.

Using the Dirichlet distribution to model goal probabilities. We apply the Fixed-point Iteration algorithm (Minka, 2000) to estimate the parameters of a Dirichlet distribution from the goal probability distributions generated by the GR. That is, given k observations and n goals, the GR generates k distributions of these goals – one distribution after each observation. These distributions are used to estimate a Dirichlet parameter vector of size n. We then draw a large number of goal probability-distribution samples (e.g., 1000) from the Dirichlet distribution specified by this vector, and determine the CI for each goal from its probabilities across these samples — the lower/upper bounds for each goal correspond to its minimum/maximum drawn probabilities. Details about the calibration of this algorithm appear in Appendix A.

4.2 Identifying high-impact observations

We identify high-impact observations by first quantifying the contribution of each observation, and then selecting those with the highest contributions.

Calculating the contribution of an observation to a goal. We define $C(o_t, G)$, the contribution of an observation o_t to goal G, as the change in the probability of G after observing o_t at time t, weighted by a factor $\gamma \in (0, 1]$ that discounts earlier observations (Davison and Hirsh, 1998).

$$C(o_t, G) = [\Pr(G \mid \Omega_t) - \Pr(G \mid \Omega_{t-1})] \times \gamma^{|\Omega_T|-t}$$

where $\Omega_T = (o_1, \ldots, o_T)$ is the observation sequence seen so far up to time T, and Ω_t is a subsequence of Ω_T that ends in observation o_t.

In this formulation, the exponent of γ is higher for earlier observations than for later ones, leading to lower contributions of earlier observations. For example, if $T = 10$, $\gamma^{10-3} = \gamma^7$ for o_3, while $\gamma^0 = 1$ for o_T. Also, higher/lower γs lead to higher/lower contributions of earlier observations. Additional details appear in Appendix B.

To identify the most influential observations, we apply a clustering algorithm over all unique positive observation contributions, for each top-ranked goal separately. We employ a clustering algorithm because it handles cases where several observations have high but slightly different contributions, while a threshold-based approach may arbitrarily include or exclude observations of interest. We have chosen the Ckmeans.1d.dp algorithm (Song and Zhong, 2020), which is guaranteed to find the optimal cluster configuration (that minimizes the sum of squares of within-cluster distances) for one-dimensional variables such as ours. The algorithm is run for k clusters, where $k = 2, \ldots, |C_G^+| - 1$, and C_G^+ is the set of all unique positive contributions for goal G; the clustering solution with the highest average Silhouette score (Rousseeuw, 1987) is selected. This solution comprises k_{best} clusters of observations, and we select the cluster containing observations that have the highest contribution to a particular top-ranked goal. The generated explanation incorporates observations from the highest-impact cluster for each top-ranked goal – usually five observations or less.

4.3 Collecting domain evidence

Critique explanations need domain information in order to present evidence for or against the GR's predictions. To obtain this information, we (i) represent the GR problem with a separate classical planning domain model (Ghallab et al., 2016), which updates its state from observed actions; and (ii) use an AI planner (Hoffmann and Nebel, 2001) to derive plans that achieve goal requirements.

Our approach to collecting evidence hinges on the idea that an agent performs actions that reduce the cost of achieving at least one requirement of their goal. For instance, when a rover collects rock from waypoint 1, it reduces the remaining cost of all the missions that require these data. Hence, this rock-collection action is *positive evidence* for these missions, and *negative evidence* for missions that do not require rock data from waypoint 1.

To find these pieces of evidence, we connect each observation to at least one goal requirement. This is done by using the separate domain model to derive a *dependency graph* that links the initial state of the model, the observed actions and *achieved* goal requirements or *potential* goal requirements (whose cost has decreased). Figure 1 illustrates this graph for the example in Table 1 — achieved requirements are represented by purple

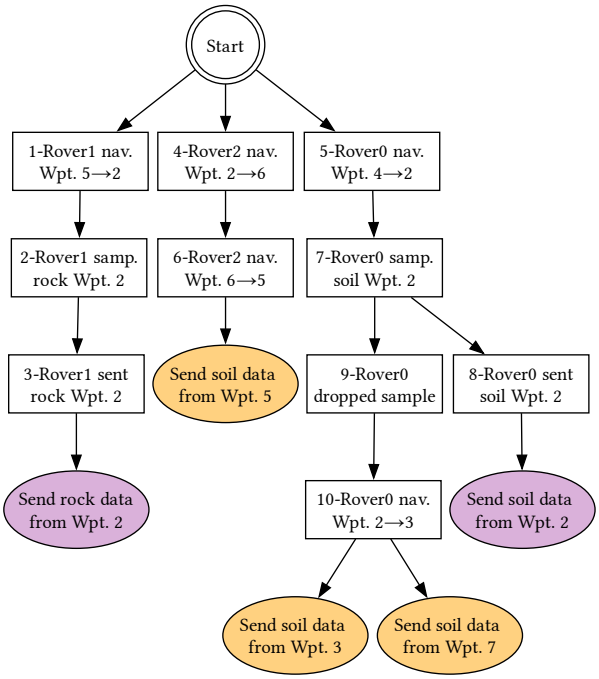


Figure 1: Dependency graph for the example in Table 1. Rectangles denote observations, with leading digits indicating observation numbers from Table 1; purple ovals denote achieved goal requirements; orange ovals denote potential requirements; and edges denote dependencies.

ovals, and potential requirements by orange ovals. The generation of the dependency graph is detailed in Algorithm 1, Appendix C, and described below.

The domain model determines which goal requirements have been achieved by particular observed actions, e.g., Obs. 3 (*Rover1 sent rock Wpt. 2*) achieves requirement "*send rock data from Wpt. 2*". To identify potential goal requirements, we calculate the cost (in number of steps) of plans generated by our planner before and after each node that is not followed by another observation or a goal requirement. For example, before completing the construction of the dependency graph in Figure 1, Obs. 6 (*Rover2 nav. Wpt. 6→5*) is such a node. This observation reduces the cost of a plan for requirement "*send soil data from Wpt. 5*", as the rover must be in waypoint 5 to collect a soil sample from there. We can therefore designate this goal requirement as potential, and directly link it with Obs. 6.

Upon completion of the graph-generation process, all the observations are connected to an achieved or potential goal requirement. Table 3 illustrates four observation sequences with requirement types to which they are linked, and associated missions (the extraction of observation sequences from the dependency graph is detailed in Algorithm 2, Appendix C). The link between Obs. 10 in Seq. 4 and the potential requirements of Missions M_1 and M_4 is presented as evidence in the final sentence of the Critique explanation in Table 2 (M_3 and M_5 are omitted due to their low probability). It is worth noting that according to Figure 1, Obs. 5 and 7 appear in two different observation sequences, but they are assigned to Seq. 3 (Table 3), which achieves a goal requirement.

Table 3: Observation sequences, linked goal requirements derived from the dependency graph in Figure 1, and supported missions.

#	Obs. Seq.	Rover	Req. Type	Goal Requirement	Supported Missions
1	1, 2, 3	1	Achiev.	Send rock data from Wpt. 2	M_1, M_2 M_4, M_6
2	4, 6	2	Potent.	Send soil data from Wpt. 5	M_1, M_2, M_3 M_4, M_5
3	5, 7, 8	0	Achiev.	Send soil data from Wpt. 2	M_1, M_2, M_3 M_4, M_5, M_6
4	9, 10	0	Potent.	Send soil data from Wpt. 3	M_1
				Send soil data from Wpt. 7∗	M_3, M_4, M_5

(∗ Rover0 can go from Wpt.3 to Wpt. 7, Table 1)

4.4 Using templates to realize explanations

We employ three templates in our explanations: *Prediction*, *Effect* and *Analysis*. Impact explanations use the first two templates, yielding the first and second paragraph (blue shaded) in Table 2; and Critique explanations use all three templates.

The Prediction template (Figure 2) presents the probabilities of the top-ranked goals after the most recent observation, and the Effect template (Figure 3) describes the contributions of observations to the probabilities of these goals.

To generate Critique explanations, we identify goals supported by all the observations, and contrast them with the top-ranked GR-predicted goals. To this effect, we determine how the observations in each observation sequence are connected to a goal requirement. For instance, looking at Mission M_1 (Table 1), its rock and soil requirements at waypoint 2 have been achieved (Seq. 1 and 3 in Table 3). In addition, Obs. 4 and 6 are linked to a potential soil requirement at waypoint 5 (Seq. 2), and Obs. 9 and 10 are linked to a potential soil requirement at waypoint 3 (Seq. 4).

We define three types of relationships between the GR's top-ranked goals and those supported by all observations: *Agreement* ($\mathcal{G}_{\text{agree}}$) – supported

According to the AI, goal(s) $G_1, G_2, \ldots$ is/are the most likely (probability(ies) of $\Pr(G_1 \mid \Omega_T), \Pr(G_2 \mid \Omega_T), \ldots$ [respectively]).

Figure 2: Prediction template. G_i is a top-ranked goal and $\Pr(G_i \mid \Omega_T)$ is its probability.

The observation(s) that most influenced this result are:
- o_1, which increased the probability of $G_1, G_2, \ldots$ by about $\Delta \Pr(G_1, o_1), \Delta \Pr(G_2, o_1), \ldots$;
- o_2, which increased the probability of $G_1, G_2, \ldots$ by about $\Delta \Pr(G_1, o_2), \Delta \Pr(G_2, o_2), \ldots$;
- $\ldots$

Figure 3: Effect template, where o_t is a high-impact observation for goal G_i and $\Delta \Pr(o_t, G_i)$ is the increase in G_i's probability after o_t is observed.

goals that are top-ranked by the GR; *Commission* ($\mathcal{G}_{\text{commit}}$) – top-ranked goals that are *not* supported by all observations; and *Omission* ($\mathcal{G}_{\text{omit}}$) – supported goals that are *not* top ranked by the GR.
$\mathcal{G}_{\text{agree}} = \{\text{top-ranked goals}\} \cap \{\text{supported goals}\}$;
$\mathcal{G}_{\text{commit}} = \{\text{top-ranked goals}\} - \{\text{supported goals}\}$;
$\mathcal{G}_{\text{omit}} = \{\text{supported goals}\} - \{\text{top-ranked goals}\}$.

Each of these relationships leads to an eponymous section of a critique. The Analysis template (Figure 4) presents these sections together with one or more examples. The examples in the Commit section link observations $\{o_{\text{commit}}\}$ to requirements $\{req_{\text{commit}}\}$ that are *not* required by goals $\mathcal{G}_{\text{commit}}$; and the examples in the Omit section link observations $\{o_{\text{omit}}\}$ to requirements of goals $\mathcal{G}_{\text{omit}}$. The type of the link is conveyed through *contrib_verb*, which is "contributes to" for potential goal requirements or "achieved" for achieved requirements. The examples are selected algorithmically based on the type of relation, the type of link, the recency of the linked observations and the presence of these examples elsewhere in the critique (Appendix D). The Analysis template yields the yellow shaded Critique explanation in Table 2.

5 Experimental Setup

We conducted a user study to address the following research questions:

RQ1. Do explanations help users attain a better goal recognition accuracy than no explanations? If so, which explanation type is better?

RQ2. Which independent variables influence users' performance?

RQ3. What are users' views of Impact and Critique explanations? We consider the extent to which an explanation is liked (Maruf et al., 2024), and five

Up to now:
- (**Agree**) All the observations *contrib_verb* requirements of $\mathcal{G}_{\text{agree}}$, which are the most likely goal(s).
- (**Commit**) Even though $\mathcal{G}_{\text{commit}}$ is/are the most likely goal(s), one/a few/several observation(s) do(es) **not** contribute to any of its/their requirements. For example: Observation o_{commit} *contrib_verb* to $\{req_{\text{commit}}\}$, which is/are **not** required by goal(s) $\mathcal{G}_{\text{commit}}$.
- (**Omit**) Even though $\mathcal{G}_{\text{omit}}$ is/are **not** the most likely goal(s), all the observations *contrib_verb* to its/their requirements. For example: Observation o_{omit} *contrib_verb* to requirement(s) of goal(s) $\mathcal{G}_{\text{omit}}$.

Figure 4: Analysis template presenting agreement, commission and omission relations with examples.

explanatory attributes: completeness; presence of misleading or irrelevant information; and helpfulness for understanding the AI's reasoning, assessing the likelihood of the goals and judging when to trust the AI (Hoffman et al., 2018).

5.1 User study design

Our design comprises two experiments: *between-subjects* – one group of participants saw only Impact explanations, and another group saw only Critique explanations; and *within-subject (Combined)* – each participant saw an Impact explanation followed by a Critique explanation. We conducted both types of experiments for the following reason. Within-subject experiments usually yield stronger results, but seeing Critique explanations after Impact explanations may influence users' views of both explanations as the experiment progresses.

Independent variables. Our experiment has four intrinsic independent variables, viz *explanatory condition* (None = prediction only, Impact, Critique), GR *prediction correctness* (correct, partially correct, incorrect), *scenario group* (A or B), and *order of scenario presentation* (1st, 2nd, 3rd); and two extrinsic independent variables, viz *time spent* on each explanatory condition (minutes) and participants' score on the *Need for Cognition Scale* (*NCS*) (Cacioppo et al., 1984). We used two scenario groups, each comprising three scenarios, to determine the effect of the actual scenario on the dependent variables. The NCS assesses people's enjoyment of thinking (5-point Likert scale: 1=extremely uncharacteristic to 5=extremely characteristic); we used the six questions chosen by Lins de Holanda Coelho et al. (2020), which yield total scores between 6-30 (Appendix E).

Scenario characteristics and groups. The scenarios from our domain resemble that in Table 1,

and are of similar complexity: each scenario involves three rovers, eight waypoints, and six possible missions – each with five requirements; the number of observed actions per scenario ranges from nine to eleven, and completing a mission requires twenty actions. To increase task complexity, each scenario includes two high-probability missions that are not distinguishable based on the observations. Each scenario group has one correct scenario (two correctly predicted missions), one partially correct scenario (one correct and one wrong prediction), and one incorrect scenario (two wrongly predicted missions).

5.2 The experiment

Our survey was implemented in the Qualtrics survey platform and conducted on Connect – a Cloud Research platform (Litman and Robinson, 2020). After signing a consent form, participants filled a demographic and ML expertise questionnaire, and the NCS questionnaire. They then saw a description of the Rovers space exploration domain, a brief account of the GR's output and an overview of the study (Figure 6, Appendix G). Next, a competency test about the study and the domain was given, where at least 3 out of 5 answers had to be correct in order to continue. The retained participants proceeded to the body of the survey, which consists of a demonstration scenario and three actual scenarios. Each participant was randomly assigned to scenario group A or B; all participants within the same group saw the same three scenarios, but scenario order was randomized for each participant.

Each scenario began with a problem description (mission requirements, waypoints, rover capabilities and the rovers' observed actions); an attention-check question was included to identify unreliable responses (Figure 7, Appendix G). Participants then rated the likelihood of each mission without access to the GR's prediction. Next, they were shown the GR's prediction and asked to reassess the mission likelihoods.

From this point, the between-subjects and within-subject arms of the experiment diverge, but each arm displays scenarios with three different levels of prediction correctness.

Between-subjects. Participants saw either an Impact or a Critique explanation, then reassessed mission likelihoods (Figures 8 and 9, Appendix G), and rated explanatory attributes.

Within-subject. Participants saw an Impact explanation and reassessed mission likelihoods, followed by a Critique explanation and another reassessment. They then rated explanatory attributes for both explanations, shown in randomized left-right order (Figure 10, Appendix G).

Following best practice recommendations in (van der Lee et al., 2021), all the assessments were on a 7-point Likert scale: 1= 'Extremely unlikely' to 7 = 'Extremely likely' for mission probabilities; and 1 = 'Strongly disagree' to 7 = 'Strongly agree' for statements about explanatory attributes.

5.3 Participants

Upon recruitment, participants were randomly assigned to one cohort (between-subjects Impact or Critique, or within-subject Combined). The participants retained after the competency test (178 out of 188) spent 35 minutes on the rest of the experiment on average. Responses were validated based on answers to the attention questions and time spent on each scenario, yielding 141 valid surveys, for which participants were paid $10-$12 USD.

Table 6 (Appendix H) shows descriptive statistics for the retained participants per cohort. There were 46-48 participants in each cohort, and they had similar demographic characteristics, ML expertise and NCS scores. To validate cohort similarity, we compared the NCS scores of each pair of six groups (3 cohorts $\times$ 2 A/B scenario groups). We used the Kruskal-Wallis H test, which yielded no statistically significant differences ($H(df = 5) = 4.7571, p\text{-}value = 0.4462$).

5.4 Dependent variables and statistical models

Dependent variables. We define three dependent variables to measure participants' performance: L_{cor} and L_{incor} – the average likelihood assigned by a participant to correct and incorrect goals respectively; and $A_{\mathcal{G}}$ – the agreement between a participant's ranking of the goals and that of the GR. An effective explanation should lead participants to increase the likelihoods of correct goals and decrease the likelihoods of incorrect goals, provided these likelihoods are not already maximal for correct goals (=7) or minimal for incorrect goals (=1). This, in turn, should yield a reranking of the goals.

$$L_{\text{cor}} = \frac{1}{|\mathcal{G}_{\text{cor}}|} \sum_{G \in \mathcal{G}_{\text{cor}}} \textit{Likely}_G$$

$$L_{\text{incor}} = \frac{1}{|\mathcal{G}_{\text{incor}}|} \sum_{G \in \mathcal{G}_{\text{incor}}} \textit{Likely}_G$$

$$A_{\mathcal{G}} = 1 - \frac{1}{(|\mathcal{G}|^2/2)} \sum_{i=1}^{|\mathcal{G}|} |R_P[i] - R_{GR}[i]|$$

where $\mathcal{G}$ is the set of all goals (missions), $\mathcal{G}_{\text{cor}}$ and $\mathcal{G}_{\text{incor}}$ are the sets of correct ground-truth goals (two missions) and incorrect goals (four missions) respectively, and $\textit{Likely}_G$ is the likelihood assigned

by a participant to goal G. We use averages, rather than sums, because in general, there are more incorrect than correct goals, and we want to avoid overwhelming the results for correct goals. R_P and R_{GR} are vectors, such that $R_P[i]$ and $R_{GR}[i]$ are the ranks of goal G_i according to participants' likelihoods and GR's probabilities respectively.

L_{cor} and L_{incor} range from 1 to 7, where a higher L_{cor} denotes better decisions, while a higher L_{incor} denotes worse decisions. A_G, which was adapted from the Spearman Footrule distance (Diaconis and Graham, 1977), ranges from 0 (no similarity) to 1 (identical ranks).

Statistical models. We used linear mixed-effects models to examine how the independent variables influence the dependent variables.

For L_{cor} and L_{incor}, the fixed effects correspond to all the independent variables except presentation order, which had no effect on task performance. For A_G, the fixed effects correspond to all the independent variables except NCS score and presentation order, which had no effect on agreement with the GR. In total, we fitted three distinct models (formulas and details appear in Appendix F).

6 Results

We applied the following procedures to adjust statistical significance for multiple comparisons. For RQ1 and RQ2, we used Tukey's Honestly Significant Difference (HSD) (Tukey, 1949) for comparisons among estimated means. For RQ3, we used the Holm-Bonferroni correction (Holm, 1979). Results with $0.05 < p\text{-}value < 0.1$ after adjustment are designated as trends.

RQ1 and RQ2. We compare participants' goal recognition accuracy under Impact, Critique and None explanatory conditions for the three correctness levels of GR predictions. The only statistically significant results were for the average likelihoods assigned to correct goals (L_{cor}) for the Critique explanatory condition compared to None (between-subjects Critique cohort and within-subject cohort) for the correct and partially correct scenarios (first two rows of Table 4). No statistically significant differences were found between the likelihoods assigned to correct (not predicted) goals (L_{cor}) for the incorrect scenario (last row of Table 4); the likelihoods assigned to incorrect goals (L_{incor}) or participant-GR rank agreements (A_G) for the different scenarios and explanatory conditions; or between Impact explanations and the other explanatory conditions across our metrics. Tables 7 and 8 (Appendix H) respectively display complete results and contrasts between means.

Table 4: L_{cor} results for the Critique vs None explanatory conditions in the between subjects (Critique cohort) and within subject experiments: mean (standard deviation); statistically significant differences ($p\text{-}value < 0.05$) are **boldfaced** and trends ($0.05 < p\text{-}value < 0.1$ after adjustment) are ***italicized***.

Correctness level	**Between subjects** None	Critique	**Within subject** None	Critique
Correct	5.09 (1.59)	**5.56 (1.55)**	5.43 (1.27)	**5.80 (1.08)**
Part. correct	4.60 (1.60)	***4.98 (1.41)***	4.90 (1.15)	**5.26 (1.06)**
Incorrect	4.23 (1.20)	4.23 (1.49)	4.61 (0.96)	4.79 (1.21)

Table 5: Estimated slope parameters for significant results derived from the linear mixed-effects models: statistically significant differences ($p\text{-}value < 0.05$) are **boldfaced** and trends ($0.05 < p\text{-}value < 0.1$ after adjustment) are ***italicized***.

Estimated Slopes - Time				
Metric	Cohort	Expl. cond.	Estimate	p-value
L_{cor}	Between-Critique	Critique	0.213	**0.047**
L_{cor}	Within	Critique	0.232	**0.002**
L_{cor}	Between-Impact	None	0.127	**0.007**
L_{incor}	Within	None	0.106	***0.076***
Estimated Slopes - NCS				
Metric	Cohort	Expl. cond.	Estimate	p-value
L_{cor}	Between-Critique	–	0.097	**0.001**
L_{incor}	Between-Critique	–	0.040	***0.057***

It is worth noting that Critique explanations for correct predictions only add one (essentially summary) sentence to the corresponding Impact explanations: "All the observations contribute to Missions X and Y, which are the most likely missions". Thus, it is surprising that Critique explanations statistically significantly increased L_{cor} for the correct and partially correct scenarios, while Impact explanations had no statistically significant effects.

We also control for the interaction between *time spent* and *NCS score* and each explanatory condition, and for the effect of *scenario group* (presentation order had no effect, Section 5.4). Table 5 displays the slope and interactions for the independent variables that had statistically significant effects, viz *time spent* and *NCS score*. The *time spent* variable interacts with explanatory condition, yielding the following statistically significant effects: (1) every additional minute spent on a Critique explanation increased L_{cor} – between-subject Critique and within-subject cohorts (first segment of Table 5); and (2) every additional minute spent on the None condition increased both L_{cor} and L_{incor} (wrong direction) – between-subject Impact and

within-subject cohorts (second segment of Table 5). The NCS score had a similar effect in the between subjects Critique cohort, with an additional unit of NCS score statistically significantly increasing both L_{cor} and L_{incor} under all explanatory conditions (no interactions) (last segment of Table 5).

RQ3. We compared participants' views about our explanations in terms of six explanatory attributes: liking an explanation; completeness; presence of extraneous information; and helpfulness for understanding the AI's reasoning, assessing goal likelihood, and judging when to trust the AI (Section 5). We used the Wilcoxon rank-sum test to compare the views of the Critique and Impact cohorts of the between-subjects experiment, and the Wilcoxon signed-rank test for the within-subject experiment (Table 9, Appendix H shows detailed results).

In the between-subjects experiment, Critique explanations were deemed more helpful than Impact explanations for assessing mission likelihood and judging when to trust the GR for partially correct GR predictions, and more complete for incorrect predictions (*p-value* ~ 0.05). However, when participants in the within-subject experiment directly compared the two types of explanations, the Critique explanations were deemed better than Impact explanations in terms of all explanatory attributes (*p-value* < 0.05) except for presence of extraneous information, for which they were equivalent. These findings, like those in (Zukerman and Maruf, 2024), are not surprising, as when each cohort saw only one type of explanation, it was deemed adequate, but when the explanations were seen side-by-side, Critique explanations were preferred. This result may be explained by the observations in (Lombrozo, 2016), whereby users generally prefer longer explanations (Critique explanations are markedly longer than the corresponding Impact explanations for partially correct and incorrect predictions), though in our case, the content of these explanations was not absorbed.

7 Discussion

We presented domain-agnostic algorithms that generate Impact and Critique explanations for predictions made by a GR. Our algorithms treat the GR as a black box, but derive information from domain knowledge to generate critiques of predictions.

As mentioned in Section 1, XAI systems generally do not vary the type of explanation they generate, irrespective of whether the ML model produced correct or wrong predictions. In contrast, our approach distinguishes between explanations of predictions that seem plausible and predictions that require further scrutiny.

Our algorithms were tested in several domains (e.g., Blocks world and Sokoban), demonstrating their generalizability. However, our user study was restricted to the Rovers domain, owing to budgetary constraints. In this study, we evaluated our explanations in terms of their effect on users' goal recognition accuracy, considering the influence of five other independent variables; and assessed these explanations in terms of six explanatory attributes.

Critique explanations led participants to increase the likelihood of correct goals for correct and partially correct predictions, but did not affect the likelihood assigned to incorrect goals. Also, additional *time spent* increased the likelihood assigned to correct goals for Critique explanations, while a higher *NCS score* increased the likelihood assigned to correct and incorrect goals for these explanations. Despite their lack of impact with respect to incorrect goals, users viewed Critique explanations favourably, when compared directly with Impact explanations, in terms of five explanatory attributes. This result calls into question evaluations that are based solely on such attributes, without considering task performance.

The results of our user study could be affected by our Critique explanations not being critical enough, or by the crowdworkers we recruited not being sufficiently engaged with the experiment (or by recruiting crowdworkers at all (Reiter, 2025)). Unfortunately, we could not recruit real users who would be engaged with our Rovers domain, which is a well-known problem in NLG evaluation. To address this limitation, we are supplementing our study with think-aloud sessions involving a small group of highly engaged participants.

Finally, it is worth noting that explanations may become unwieldy when many goals or observations are selected to be mentioned. This suggests another avenue of investigation, which involves interacting with users (Maruf et al., 2023; Miller, 2023).

Acknowledgments

This research was supported in part by a Monash University Graduate Scholarship and by a CSIRO/Data61 Supplementary Postgraduate Research Scholarship. Ethics approval for the user study was obtained from Monash University Human Research Ethics Committee (ID-36832).

References

K. Aas, M. Jullum, and A. Løland. 2021. Explaining individual predictions when features are dependent: More accurate approximations to Shapley values. *Artificial Intelligence*, 298:103502.

S.V. Albrecht, C. Brewitt, J. Wilhelm, B. Gyevnar, F. Eiras, M. Dobre, and S. Ramamoorthy. 2021. Interpretable goal-based prediction and planning for autonomous driving. In *Proceedings of the 2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1043–1049, Xi'an, China. IEEE.

A. Alshehri, A. Abdulrahman, H. Alamri, T. Miller, and M. Vered. 2025. Towards explainable goal recognition using weight of evidence (woe): A human-centered approach. *Journal of Artificial Intelligence Research*, 82:2535–2594.

A. Alshehri, T. Miller, and M. Vered. 2023. Explainable goal recognition: a framework based on weight of evidence. In *Proceedings of the International Conference on Automated Planning and Scheduling*, volume 33, pages 7–16, Prague, Czech Republic.

O. Biran and K. McKeown. 2017. Human-centric justification of Machine Learning predictions. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence*, IJCAI 2017, pages 1461–1467, Melbourne, Australia.

C. Brewitt, B. Gyevnar, S. Garcin, and S.V. Albrecht. 2021. GRIT: fast, interpretable, and verifiable goal recognition with learned decision trees for autonomous driving. In *Proceedings of the 2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1023–1030, Prague, Czech Republic. IEEE.

C. Brewitt, M. Tamborski, C. Wang, and S.V. Albrecht. 2023. Verifiable goal recognition for autonomous driving with occlusions. In *Proceedings of the 2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 11210–11217, Detroit, Michigan.

A. Buerkle, W. Eaton, N. Lohse, T. Bamber, and P. Ferreira. 2021. EEG based arm movement intention recognition towards enhanced safety in symbiotic human-robot collaboration. *Robotics and Computer-Integrated Manufacturing*, 70:102137.

J.T. Cacioppo, R.E. Petty, and C.F. Kao. 1984. The efficient assessment of need for cognition. *Journal of Personality Assessment*, 48(3):306–307. PMID: 16367530.

S. Carberry. 2001. Techniques for plan recognition. *User Modeling and User-Adapted Interaction*, 11:31–48.

B.D. Davison and H. Hirsh. 1998. Predicting sequences of user actions. In *Notes of the AAAI/ICML 1998 workshop on predicting the future: AI approaches to time-series analysis*, pages 5–12, Madison, Wisconsin.

P. Diaconis and R.L. Graham. 1977. Spearman's Footrule as a measure of disarray. *Journal of the Royal Statistical Society: Series B (Methodological)*, 39(2):262–268.

M. Ghallab, D. Nau, and P. Traverso. 2016. *Automated Planning and Acting*, 1st edition. Cambridge University Press.

J.P. Hanna, A. Rahman, E. Fosong, F. Eiras, M. Dobre, J. Redford, S. Ramamoorthy, and S.V. Albrecht. 2021. Interpretable goal recognition in the presence of occluded factors for autonomous vehicles. In *Proceedings of the 2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 7044–7051, Prague, Czech Republic. IEEE.

R.R. Hoffman, S.T. Mueller, G. Klein, and J. Litman. 2018. Metrics for explainable AI: Challenges and prospects. *arXiv preprint arXiv:1812.04608*.

J. Hoffmann and B. Nebel. 2001. The FF planning system: Fast plan generation through heuristic search. *Journal of Artificial Intelligence Research*, 14:253–302.

S. Holm. 1979. A simple sequentially rejective multiple test procedure. *Scandinavian Journal of Statistics*, 6(2):65–70.

J. D. Janizek, P. Sturmfels, and S. Lee. 2021. Explaining explanations: Axiomatic feature interactions for deep networks. *Journal of Machine Learning Research*, 22(1).

B. Kim, R. Khanna, and O.O. Koyejo. 2016. Examples are not enough, learn to criticize! criticism for interpretability. *Advances in neural information processing systems*, 29.

G. Lins de Holanda Coelho, P.H.P Hanel, and L.J. Wolf. 2020. The Very Efficient Assessment of Need for Cognition: Developing a Six-Item Version. *Assessment*, 27(8):1870–1885.

L. Litman and J. Robinson. 2020. *Conducting online research on Amazon Mechanical Turk and beyond*. Sage Publications.

T. Lombrozo. 2016. Explanatory preferences shape learning and inference. *Trends in Cognitive Sciences*, 20(10):748–759.

S.M. Lundberg and S-I. Lee. 2017. A unified approach to interpreting model predictions. In *Proceedings of the 31st Annual Conference on Neural Information Processing Systems*, NIPS'17, pages 4768–4777, Long Beach, California.

S. Maruf, I. Zukerman, E. Reiter, and G. Haffari. 2023. Influence of context on users' views about explanations for decision-tree predictions. *Computer Speech & Language*, 81:101483.

S. Maruf, I. Zukerman, X. Situ, C.L. Paris, and G. Haffari. 2024. Generating simple, conservative and unifying explanations for logistic regression models. In *Proceedings of the 17th International Conference on Natural Language Generation*, INLG 2024, pages 103–120, Tokyo, Japan.

T. Miller. 2023. Explainable AI is dead, long live explainable AI! hypothesis-driven decision support using evaluative AI. In *Proceedings of the 2023 ACM Conference on Fairness, Accountability, and Transparency*, FAccT '23, pages 333–342, New York, New York. ACM.

T. Minka. 2000. Estimating a Dirichlet distribution. Technical report, Massachusetts Institute of Technology, Boston, Massachusetts.

K.W. Ng, G.-L. Tian, and M.-L. Tang. 2011. *Dirichlet and related distributions: Theory, methods and applications*. Wiley Series in Probability and Statistics. John Wiley & Sons, Ltd, Hoboken, N.J.

R.F. Pereira and F. Meneguzzi. 2017. Goal and plan recognition datasets using classical planning domains. Zenodo.

D.V. Pynadath and M.P. Wellman. 2013. Accounting for context in plan recognition, with application to traffic monitoring. *arXiv preprint arXiv:1302.4980*, abs/1302.4980.

E. Reiter. 2025. We should evaluate real-world impact. *Computational Linguistics*, pages 1–13.

M.T. Ribeiro, S. Singh, and C. Guestrin. 2016. "Why should I trust you?": Explaining the predictions of any classifier. In *Proceedings of the ACM/SIGKDD Conference on Knowledge Discovery and Data Mining*, KDD'16, pages 1135–1144, San Francisco, California.

P.J. Rousseeuw. 1987. Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *Journal of computational and applied mathematics*, 20:53–65.

R. Singh, T. Miller, J. Newn, E. Velloso, F. Vetere, and L. Sonenberg. 2020. Combining gaze and AI planning for online human intention recognition. *Artificial Intelligence*, 284:103275.

M. Song and H. Zhong. 2020. Efficient weighted univariate clustering maps outstanding dysregulated genomic zones in human cancers. *Bioinformatics*, 36(20):5027–5036.

A. Theissler, F. Spinnato, U. Schlegel, and R. Guidotti. 2022. Explainable AI for time series classification: A review, taxonomy and research directions. *IEEE Access*, 10:100700–100724.

J.W. Tukey. 1949. Comparing individual means in the analysis of variance. *Biometrics*, 5(2):99–114.

C. van der Lee, A. Gatt, E. van Miltenburg, and E.J. Krahmer. 2021. Human evaluation of automatically generated text: Current trends and best practice guidelines. *Computer Speech & Language*, 67:1–24.

M. Vered, R.F. Pereira, M.C. Magnaguagno, G.A. Kaminka, and F. Meneguzzi. 2018. Towards online goal recognition combining goal mirroring and landmarks. In *Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems*, AAMAS '18, page 2112–2114, Stockholm, Sweden.

E. Yurtsever, J. Lambert, A. Carballo, and K. Takeda. 2020. A survey of autonomous driving: Common practices and emerging technologies. *IEEE access*, 8:58443–58469.

I. Zukerman and S. Maruf. 2024. Communicating uncertainty in explanations of the outcomes of machine learning models. In *Proceedings of the 17th International Conference on Natural Language Generation*, INLG 2024, pages 30–46, Tokyo, Japan.

A Calibrating the Fixed-point Iteration Algorithm

We employ two hyper-parameters to adjust the operation of the CI estimation algorithm: the temporal window percentage $w_p \in (0, 1]$, and the confidence level $c_p \in (0, 1]$.

- w_p is the proportion of the most recent goal probability distributions used to estimate the parameter of the Dirichlet distribution. For example, if $w_p = 0.4$ and the GR provides goal probability distributions for ten observations, only the four most recent ones will be used.

  The Fixed-point Iteration algorithm may not converge if there is not enough variability in the probability distributions within the specified temporal window (w_p). This may happen when the number of observations is small, i.e., at the beginning, or if w_p itself is small. In this case, we iteratively expand the window to include one additional probability distribution and then re-execute the Fixed-point Iteration algorithm. If convergence is still not achieved after using all available distributions, we simply select the goal(s) with the highest probability.

- c_p is the confidence level parameter that defines the range of the goal's CI in terms of percentiles: $l_p = (1 - c_p)/2$ is the lower percentile, and $u_p = 1 - l_p$ is the upper percentile. For example, if $c_p = 0.9$, the CI for each goal will span from the 5th percentile ($l_p = 0.05$) to the 95th percentile ($u_p = 0.95$) of the samples drawn for that goal from the estimated Dirichlet distribution. This yields the CI $[0.222, 0.244]$ for Mission M_2 in Table 2.

Together, w_p and c_p control the goal selection sensitivity to observation recency and fluctuations in the GR's goal probabilities. The calibration of w_p and c_p depends on the domain, the outputs of the GR, and the desired maximum number of goals in an explanation. A low w_p emphasizes recent observations over older ones, while a low c_p minimizes the impact of goal probability variations within the window defined by w_p. In addition, higher w_p and c_p values generally increase the number of top-ranked goals. For example, in the Rovers domain, where the scenarios have up to 36 observations, when $w_p < 0.25$ and $0.5 \leq c_p < 0.8$, one goal was selected in 95.6% of the cases, and at most two goals in 100%. In contrast, when $0.75 \leq w_p$ and $0.8 \leq c_p$, one goal was selected in 46.2% of the cases, at most two goals in 66.6%, and at most three in 85.1%.

Finally, it is worth noting that the computational cost of our method, which depends on the number of observations and the Dirichlet estimation of the CIs, is low. For example, in the Rovers domain, CI estimation ranged from 4 ms for fewer than 15 observations to 10 ms for 35–48 observations (on an Intel i7 laptop with one core).

B Window-based Calculation of the Contributions of Observations

Our formulation also includes a configurable parameter, calibrated experimentally, that sets the size of a window of observations to which the discount is applied simultaneously. Specifically, the observation sequence is divided into windows of size W, where W is in the range $[1, \ldots, |\Omega_T|]$, and the discount is kept constant within each window. Thus, the temporal discount factor is defined as:

$$\gamma^{\lfloor \frac{|\Omega_T|-t}{W} \rfloor}$$

C Algorithms

Algorithm 1 enhances a dependency graph derived from the classical domain model by linking observations to goal requirements. Each link indicates whether an observation node that is not followed by another observation node directly achieves a requirement or reduces the cost of achieving it. In the former case, the requirement is classified as *achieved*, and in the latter case, as *potential*.

Algorithm 2 receives the graph and the links generated by Algorithm 1, and extracts sequences that connect all the observations to the goal requirements in those links.

For brevity, the pseudocode for the auxiliary functions used in Algorithm 1 has been omitted. An overview of these functions follows.

- COST receives a plan, returning the cost to execute the plan.
- GETADJREQS receives a dependency graph and a node, returning the set of goal requirements that are adjacent (i.e., directly connected) to the node.
- GETOBSERVATION receives a node of a dependency graph, returning the observation represented by the node.
- ISLEAF receives a dependency graph and a node, returning *true* if the node has no outgoing edges, and *false* otherwise.

- IsObservation receives a node of a dependency graph, returning *true* if the node is associated with an observation, and *false* otherwise.
- Plan receives an environment state and a goal requirement, returning a sequence of actions that achieve the requirement starting from the environment state.
- StateBef and StateAft receive an observation, respectively returning the environment state immediately before and after the observation is perceived.
- Time receives an observation, returning the time it was perceived.

Algorithm 2 also uses auxiliary functions whose pseudocode has been omitted. An overview of these functions is presented below.

- AddPlans receives a dependency graph and the links returned by Algorithm 1, returning a dependency graph with actions from the derived plans that achieve a potential goal requirement.
- Cost – same as for Algorithm 1.
- FindObsNodes receives a dependency graph, returning all observation nodes from the graph.
- GetAdjReqs – same as for Algorithm 1.
- GetAdjObs receives a dependency graph and a node, returning the observations adjacent to that node.
- GetLastObsNode receives a set of observation nodes, returning the last (i.e., most recent) observation node in that set.
- GetObsFromNodes receives a set of observation nodes, returning the set of observations associated with these nodes.
- GetReqFromNode receives a node of a dependency graph, returning the requirement represented by the node.
- IsGoalReqNode receives a node of a dependency graph, returning *true* if the node represents a goal requirement, and *false* otherwise.
- PathExists receives a dependency graph, a source node and a target node, returning *true* if there is a path connecting the source node to the target node, and *false* otherwise.
- SortByAchievAndSeqAsc receives an associative array of observation sequences and links to goal requirements, returning the same type of array sorted first by achievement (i.e., achieved requirements first), and next in ascending order of sequence length.

Algorithm 1 Generate links

```
1: ▷ Inputs: Dgraph is a dependency graph; GReq is the
     set of all goal requirements; and s_cur is the current
     state of the environment.
     Output: L is an associative array of goal requirements
     and links. ◁
2: function GenerateLinks(Dgraph, GReq, s_cur)
3:     ▷ Find goal requirements that were not achieved ◁
4:     GReq_¬achv ← ∅
5:     for all req ∈ GReq do
6:         if req ∉ s_cur then
7:             GReq_¬achv ← GReq_¬achv ∪ {req}
8:     L ← ∅
9:     for all node ∈ Dgraph do
10:        if ¬IsObservation(node) then continue
11:        ▷ Discard node if it is not a leaf and its obser-
             vation does not achieve any requirements ◁
12:        GReq^o_achv ← GetAdjReqs(Dgraph, node)
13:        if ¬IsLeaf(Dgraph, node) and GReq^o_achv =
           ∅ then continue
14:        o ← GetObservation(node)
15:        ▷ Link observation o to its achieved goal require-
             ments or to its best potential requirements. ◁
16:        if GReq^o_achv = ∅ then
17:            for all req ∈ GReq_¬achv do
18:                π_before ← Plan(StateBef(o), req)
19:                π_after ← Plan(StateAft(o), req)
20:                Δcost ← Cost(π_after) − Cost(π_before)
21:                l ← ⟨o, req, π_before, π_after, Δcost⟩
22:                L[req] ← SelectBest(L[req], l)
23:        else
24:            π_before ← π_after ← ∅; Δcost ← 0
25:            for all req ∈ GReq^o_achv do
26:                l ← ⟨o, req, π_before, π_after, Δcost⟩
27:                L[req] ← l
28:    return L

    ▷ Returns the best of links l_a and l_b
29: function SelectBest(l_a, l_b)
30:    if l_a = ∅ then return l_b
31:    else if l_b = ∅ then return l_a
32:    if Cost(l_a[π_after]) < Cost(l_b[π_after]) then
33:        return l_a
34:    else if Cost(l_a[π_after]) > Cost(l_b[π_after]) then
35:        return l_b
36:    if l_a[Δcost] < l_b[Δcost] then return l_a
37:    else if l_a[Δcost] > l_b[Δcost] then return l_b
38:    if Time(l_a[o]) > Time(l_b[o]) then return l_a else
       return l_b
```

- SortObsByTimeAsc receives a set of observations, returning these observations is ascending time order (i.e., earliest first).

Algorithm 2 Extract observation sequences and associate them with links.

▷ *Inputs: $Dgraph$ is a dependency graph; and $\mathcal{L}$ is an associative array of goal requirements and links. Output: $\mathcal{A}$ is a set of associations.* ◁
function GENERATESEQUENCES($Dgraph, \mathcal{L}$)
  $Dgraph_\pi \leftarrow$ ADDPLANS($Dgraph, \mathcal{L}$)
  $nodes_\Omega \leftarrow$ FINDOBSNODES($Dgraph_\pi$)
  $GReq_O \leftarrow \varnothing$ ▷ *Associative array of observation nodes and reachable goal requirement nodes*
  for all $node_{\text{req}} \in Dgraph_\pi$ **do**
    if ¬ISGOALREQNODE($node_{\text{req}}$) **then** continue
    $GReq_O[node_{\text{req}}] \leftarrow \varnothing$
    for all $node_o \in nodes_\Omega$ **do**
      if PATHEXISTS($Dgraph_\pi, node_o, node_{\text{req}}$) **then**
        $GReq_O[node_{\text{req}}] \leftarrow GReq_O[node_{\text{req}}] \cup \{node_o\}$
  $\mathcal{OL} \leftarrow \varnothing$ ▷ *Associative array of candidate observation sequences and links*
  for all $\langle node_{\text{req}}, nodes_o \rangle \in GReq_O$ **do**
    if $nodes_o = \varnothing$ **then** continue
    $lastN_o \leftarrow$ GETLASTOBSNODE($nodes_o$)
    $O_{\text{adj}} \leftarrow$ GETADJOBS($Dgraph_\pi, lastN_o$)
    $GReq_{\text{adj}} \leftarrow$ GETADJREQS($Dgraph_\pi, lastN_o$)
    if $O_{\text{adj}} \neq \varnothing$ and $GReq_{\text{adj}} = \varnothing$ **then** continue
    ▷ *Associate the observation sequence with links to the goal requirement node* ◁
    $\mathcal{O}_\Phi \leftarrow$ GETOBSFROMNODES($nodes_o$)
    $sorted\mathcal{O}_\Phi \leftarrow$ SORTOBSBYTIMEASC($\mathcal{O}_\Phi$)
    $req \leftarrow$ GETREQFROMNODE($node_{\text{req}}$)
    $\mathcal{OL}[sorted\mathcal{O}_\Phi] \leftarrow \mathcal{OL}[sorted\mathcal{O}_\Phi] \cup \{\mathcal{L}[req]\}$
  ▷ *Update the sequences in $\mathcal{OL}$ to ensure that observations connected to an achieved requirement are in just one sequence* ◁
  $O_{\text{singleL}} \leftarrow \varnothing$ ▷ *Observations connected to a single achieved goal requirement*
  $\mathcal{A} \leftarrow \varnothing$ ▷ *A set of associations*
  $\mathcal{OL} \leftarrow$ SORTBYACHIEVANDSEQASC($\mathcal{OL}$)
  for all $\langle \mathcal{O}_\Phi, \mathcal{L}_\mathcal{O} \rangle \in \mathcal{OL}$ **do**
    $\mathcal{O} \leftarrow \mathcal{O}_\Phi \setminus O_{\text{singleL}}$
    if $\mathcal{O} = \varnothing$ **then** continue
    if $|\mathcal{L}_\mathcal{O}| = 1$ **then**
      $l \leftarrow \mathcal{L}_\mathcal{O}[1]$
      if COST($l[\pi_{\text{after}}]$) $= 0$ **then**
        $O_{\text{singleL}} \leftarrow O_{\text{singleL}} \cup \mathcal{O}$
    ▷ *Add links in $\mathcal{L}_\mathcal{O}$ to the set of achieved or potential requirements for sequence $\mathcal{O}$* ◁
    $\mathcal{L}^{\text{achv}}_\mathcal{O} \leftarrow \mathcal{L}^{\text{pot}}_\mathcal{O} \leftarrow \varnothing$
    for all $l \in \mathcal{L}_\mathcal{O}$ **do**
      if COST($l[\pi_{\text{after}}]$) $= 0$ **then**
        $\mathcal{L}^{\text{achv}}_\mathcal{O} \leftarrow \mathcal{L}^{\text{achv}}_\mathcal{O} \cup \{l\}$
      else if $j[\Delta cost] < 0$ **then**
        $\mathcal{L}^{\text{pot}}_\mathcal{O} \leftarrow \mathcal{L}^{\text{pot}}_\mathcal{O} \cup \{l\}$
    $a \leftarrow \langle \mathcal{O}, \mathcal{L}^{\text{achv}}_\mathcal{O}, \mathcal{L}^{\text{pot}}_\mathcal{O} \rangle$
    $\mathcal{A} \leftarrow \mathcal{A} \cup \{a\}$
  return $\mathcal{A}$

D Generating Examples for Critique Explanations

Let $\mathcal{G}_{\text{commit}}$ be the set of top-ranked goals that were **not** supported by all the observations. To select sample observations for the *Commission* section of the Analysis template, we apply the following procedure.

1. Find all observations that do **not** contribute to at least one goal in $\mathcal{G}_{\text{commit}}$;
2. Sort the selected observations in descending order of the number of goals not in $\mathcal{G}_{\text{commit}}$ to which they contribute, and then in descending order of recency (most recent first);
3. Let o_{commit} be the first observation in the sorted list, and add it as an example;
4. Let $\mathcal{G}_{\text{address}}$ be all the goals in $\mathcal{G}_{\text{commit}}$ that are not supported by o_{commit};
5. Exit if $G_{\text{address}} = \mathcal{G}_{\text{commit}}$. Otherwise, redefine $\mathcal{G}_{\text{commit}} = \mathcal{G}_{\text{commit}} \setminus \mathcal{G}_{\text{address}}$, and go to Step 1.

Let $\mathcal{G}_{\text{omit}}$ be the set of goals that were **not** top-ranked and are supported by all the observations. To select sample observations for the *Omission* section of the Analysis template, we apply the following procedure.

1. If $\mathcal{G}_{\text{commit}} \neq \varnothing$, then use the observations from the *Commission* section to save space, and exit;
2. Sort all observations in descending order of the number of goals not in $\mathcal{G}_{\text{omit}}$ to which they contribute, and then in descending order of recency (most recent first);
3. Let o_{omit} be the first observation in the sorted list, select it as an example, and exit.

E Need for Cognition Scale

The Need for Cognition Scale (NCS), developed by Cacioppo et al. (1984), consists of 18 statements that measure an individual's tendency to engage in and enjoy cognitive activities. The answers are given on a 5-point Likert scale, where 1 indicates that the stated behaviour is extremely uncharacteristic for the user, and 5 indicates that it is extremely characteristic. In our user study, we employed the six-item version in Figure 5, which was extracted by Lins de Holanda Coelho et al. (2020) from the original 18 statements. Participants' NCS score is the sum of the ratings in their answers, with the scores of the "negative" questions (3 and 4) reversed, which yields total scores between 6-30.

1. I would prefer complex to simple problems.
2. I like to have the responsibility of handling a situation that requires a lot of thinking.
3. Thinking is not my idea of fun.
4. I would rather do something that requires little thought than something that is sure to challenge my thinking abilities.
5. I really enjoy a task that involves coming up with new solutions to problems.
6. I would prefer a task that is intellectual, difficult, and important to one that is somewhat important but does not require much thought.

Figure 5: Six statements from the Need for Cognition Scale – answers are on a 5-point Likert scale (1=extremely uncharacteristic to 5=extremely characteristic).

F Statistical Models

Linear mixed-effects models were employed to investigate the influence of the independent variables on the dependent variables. Models for the three dependent variables (L_{cor}, L_{incor} and A_G) were fitted for each of the three cohorts, but we obtained only three distinct models.

For L_{cor} and L_{incor}, we fitted the model $L \sim ncs + t \times ec + (pc/sg) \times ec + (pc \mid partic)$ where:

- ec represents explanatory condition, pc prediction correctness, sg scenario group, t the mean-centered time spent on each explanatory condition, ncs the mean-centered NCS score, and $partic$ represents individual participants;
- L represents either L_{cor} or L_{incor};
- the times symbol ($\times$) indicates both the main effects of the variables and their interaction. For example, $t \times ec$ translates to $t + ec + t : ec$;
- the slash symbol ($/$) indicates that the variable on the right is *nested* within the variable on the left. For example, pc/sg means $pc + pc : sg$, where sg is nested within pc. Specifically, a GR problem for a *partially correct prediction / Scenario group A* differs from a GR problem for an *incorrect prediction / Scenario group A*.
- the pipe symbol ($|$) indicates a random effect of the variable on the left, grouped by the variable on the right. For example, $(pc \mid partic)$ means a random intercept and slope for pc that varies for each participant.

The fixed effects of the model are: ec, pc, sg, t and ncs (presentation order was excluded, as it had no effect on task performance). Since the time spent depends on the explanatory condition, we included the interaction term $t \times ec$. Additionally, we modeled the interaction between explanatory condition and scenario group nested within prediction correctness, $(pc/sg) \times ec$, allowing the influence of the GR prediction to vary for each explanatory condition, prediction correctness and the nested scenario group.

The random effects structure of the model includes a random slope for prediction correctness within participants, denoted $(pc \mid partic)$. This structure was designed to capture individual variability in how participants responded to prediction correctness. However, for the L_{cor} dependent variable and the Impact cohort, this approach led to a singular model, indicating insufficient variance to support the specified random effects. To address this issue, that structure was simplified to $(1 \mid partic)$ just for the L_{cor} dependent variable and the Impact cohort, effectively removing random slopes for prediction correctness, and modeling just random intercepts by participant.

For A_g, we fitted the model $A_g \sim t \times (pc/sg) + (pc/sg) \times ec + (1 \mid partic)$ ncs was removed, as it did not significantly influence participant-GR agreement. Additionally, random slopes for prediction correctness were excluded due to singular model convergence issues. Instead, we modeled just random intercepts by participant: $(1 \mid partic)$.

G Screenshots from the Experiment

Figure 6 displays a screenshot of the introduction to the experiment, and Figure 7 shows a scenario in the Rovers domain. Figures 8 and 9 display the screens for the explanatory conditions Impact and Critique respectively. Finally, Figure 10 shows the screen where users in the within-subject cohort rate the explanatory attributes of Impact and Critique explanations.

You'll now receive background information about the experiment.

Please read it carefully, as a short questionnaire will follow.

The Rovers Space Exploration Domain

Pretend you work at a space exploration agency, managing rovers on a distant planet. Communication with the rovers is not in real-time, so they are programmed to autonomously plan and execute missions, and collaborate. Missions consist of tasks that collect rocks and/or soil on the planet and send results of the analysis of these samples back to Earth.

In addition to these research data, rovers log every activity they execute. Research and log data are sent to you. It is your job to analyse, validate and share the research data with the appropriate groups of researchers. However, due to a malfunction, you received only part of the activity logs, and the information about which mission is being executed was lost. **Your task is to infer which mission is being executed based on the partial activity logs you received.** Note that several missions may have similar likelihoods based on the partial activity log.

The agency has given you an AI system to help you identify the most likely mission(s). You will also receive additional explanations about the AI's findings from a dedicated explainer system.

The goal of this study is to determine the effectiveness of these explanations.

About the AI Systems

The findings of the AI systems consist of probabilities assigned to each mission. The AI systems estimate these probabilities by analysing the activity logs to calculate the degree of completion of the missions. These probabilities are updated when new actions are received.

Please, be aware that the AIs are **not perfect** and may not always accurately determine the most likely mission(s). Each problem scenario will feature a different AI.

About the Study

In this study, you will be shown **three** problem scenarios, where rovers execute actions to accomplish one of **six** missions. Each scenario will be analysed by a different AI system.
For each of these scenarios, you will have to **pretend you are a space exploration agency employee**.
You will then be asked to do the following:

- Assess the likelihood of each mission being the one executed by the rovers.
- Read the findings of each AI system, determine if the findings make sense to you, and provide a new assessment of the likelihood of the missions.
- Read generated by an explainer system. information about how the rovers' actions affect the findings of the AI. Use this information to assess the likelihood of the missions again.
- Rate the explanations along several criteria, such as completeness and clarity.

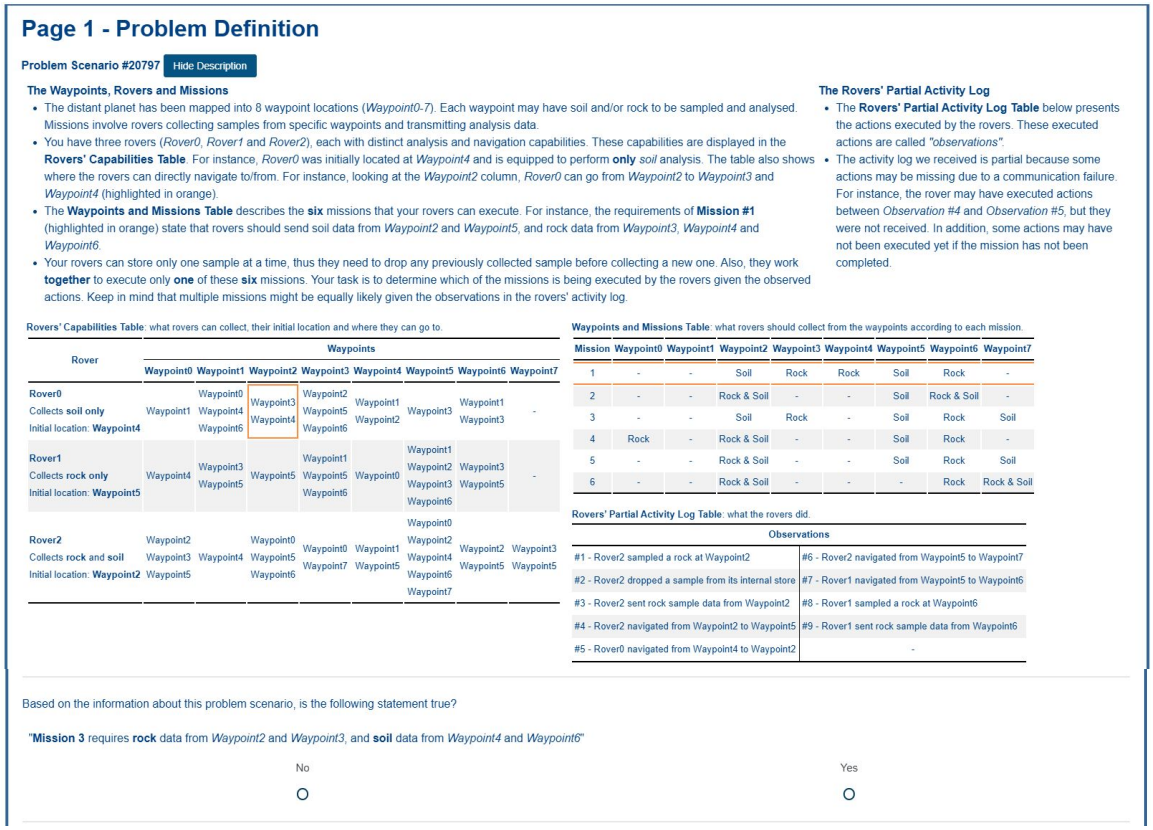
While assessing the likelihood of the missions, keep in mind that the rovers will only perform actions to achieve the requirements of the mission they are executing. For example, the rovers will never collect a rock sample if sending rock data is not part of their mission.

Important notes:

- While taking part in this study, **do not** use generative AIs, such as ChatGPT, Copilot and Gemini. Generative AI is not allowed in this task as we are testing **people's understanding of our explanations**. If we identify that generative AI has been used, you will not be paid.
- If you need to take a break during the experiment, please do so **between** problem scenarios.

Now please fill out a short questionnaire based on the background information you have just received and a problem scenario. Your responses will help us decide if you have been able to develop a basic understanding of this experiment and can proceed further.

Figure 6: Narrative about the Rovers space exploration domain; account of the output of the AI system and an overview of the study.



Page 1 - Problem Definition

Problem Scenario #20797 Hide Description

The Waypoints, Rovers and Missions

- The distant planet has been mapped into 8 waypoint locations (*Waypoint0-7*). Each waypoint may have soil and/or rock to be sampled and analysed. Missions involve rovers collecting samples from specific waypoints and transmitting analysis data.
- You have three rovers (*Rover0*, *Rover1* and *Rover2*), each with distinct analysis and navigation capabilities. These capabilities are displayed in the **Rovers' Capabilities Table**. For instance, *Rover0* was initially located at *Waypoint4* and is equipped to perform **only** *soil* analysis. The table also shows where the rovers can directly navigate to/from. For instance, looking at the *Waypoint2* column, *Rover0* can go from *Waypoint2* to *Waypoint3* and *Waypoint4* (highlighted in orange).
- The **Waypoints and Missions Table** describes the **six** missions that your rovers can execute. For instance, the requirements of **Mission #1** (highlighted in orange) state that rovers should send soil data from *Waypoint2* and *Waypoint5*, and rock data from *Waypoint3*, *Waypoint4* and *Waypoint6*.
- Your rovers can store only one sample at a time, thus they need to drop any previously collected sample before collecting a new one. Also, they work **together** to execute only **one** of these **six** missions. Your task is to determine which of the missions is being executed by the rovers given the observed actions. Keep in mind that multiple missions might be equally likely given the observations in the rovers' activity log.

The Rovers' Partial Activity Log

- The **Rovers' Partial Activity Log Table** below presents the actions executed by the rovers. These executed actions are called *"observations"*.
- The activity log we received is partial because some actions may be missing due to a communication failure. For instance, the rover may have executed actions between *Observation #4* and *Observation #5*, but they were not received. In addition, some actions may have not been executed yet if the mission has not been completed.

Rovers' Capabilities Table: what rovers can collect, their initial location and where they can go to.

Rover	Waypoints							
	Waypoint0	Waypoint1	Waypoint2	Waypoint3	Waypoint4	Waypoint5	Waypoint6	Waypoint7
Rover0 Collects **soil only** Initial location: **Waypoint4**	Waypoint1	Waypoint0 Waypoint4 Waypoint6	Waypoint3 Waypoint4	Waypoint2 Waypoint5 Waypoint6	Waypoint1 Waypoint2	Waypoint3	Waypoint1 Waypoint3	-
Rover1 Collects **rock only** Initial location: **Waypoint5**	Waypoint4	Waypoint3 Waypoint5	Waypoint5	Waypoint1 Waypoint5 Waypoint6	Waypoint0	Waypoint1 Waypoint2 Waypoint3 Waypoint6	Waypoint3 Waypoint5	-
Rover2 Collects **rock and soil** Initial location: **Waypoint2**	Waypoint2 Waypoint3 Waypoint5	Waypoint4	Waypoint0 Waypoint5 Waypoint6	Waypoint0 Waypoint7	Waypoint1 Waypoint5	Waypoint0 Waypoint2 Waypoint4 Waypoint6 Waypoint7	Waypoint2 Waypoint5	Waypoint3 Waypoint5

Waypoints and Missions Table: what rovers should collect from the waypoints according to each mission.

Mission	Waypoint0	Waypoint1	Waypoint2	Waypoint3	Waypoint4	Waypoint5	Waypoint6	Waypoint7
1	-	-	Soil	Rock	Rock	Soil	Rock	-
2	-	-	Rock & Soil	-	-	Soil	Rock & Soil	-
3	-	-	Soil	Rock	-	Soil	Rock	Soil
4	Rock	-	Rock & Soil	-	-	Soil	Rock	-
5	-	-	Rock & Soil	-	-	Soil	Rock	Soil
6	-	-	Rock & Soil	-	-	-	Rock	Rock & Soil

Rovers' Partial Activity Log Table: what the rovers did.

Observations	
#1 - Rover2 sampled a rock at Waypoint2	#6 - Rover2 navigated from Waypoint5 to Waypoint7
#2 - Rover2 dropped a sample from its internal store	#7 - Rover1 navigated from Waypoint5 to Waypoint6
#3 - Rover2 sent rock sample data from Waypoint2	#8 - Rover1 sampled a rock at Waypoint6
#4 - Rover2 navigated from Waypoint2 to Waypoint5	#9 - Rover1 sent rock sample data from Waypoint6
#5 - Rover0 navigated from Waypoint4 to Waypoint2	-

Based on the information about this problem scenario, is the following statement true?

"**Mission 3** requires **rock** data from *Waypoint2* and *Waypoint3*, and **soil** data from *Waypoint4* and *Waypoint6*"

No ○ Yes ○

Figure 7: Description of a GR problem; and attention-check question about the description.

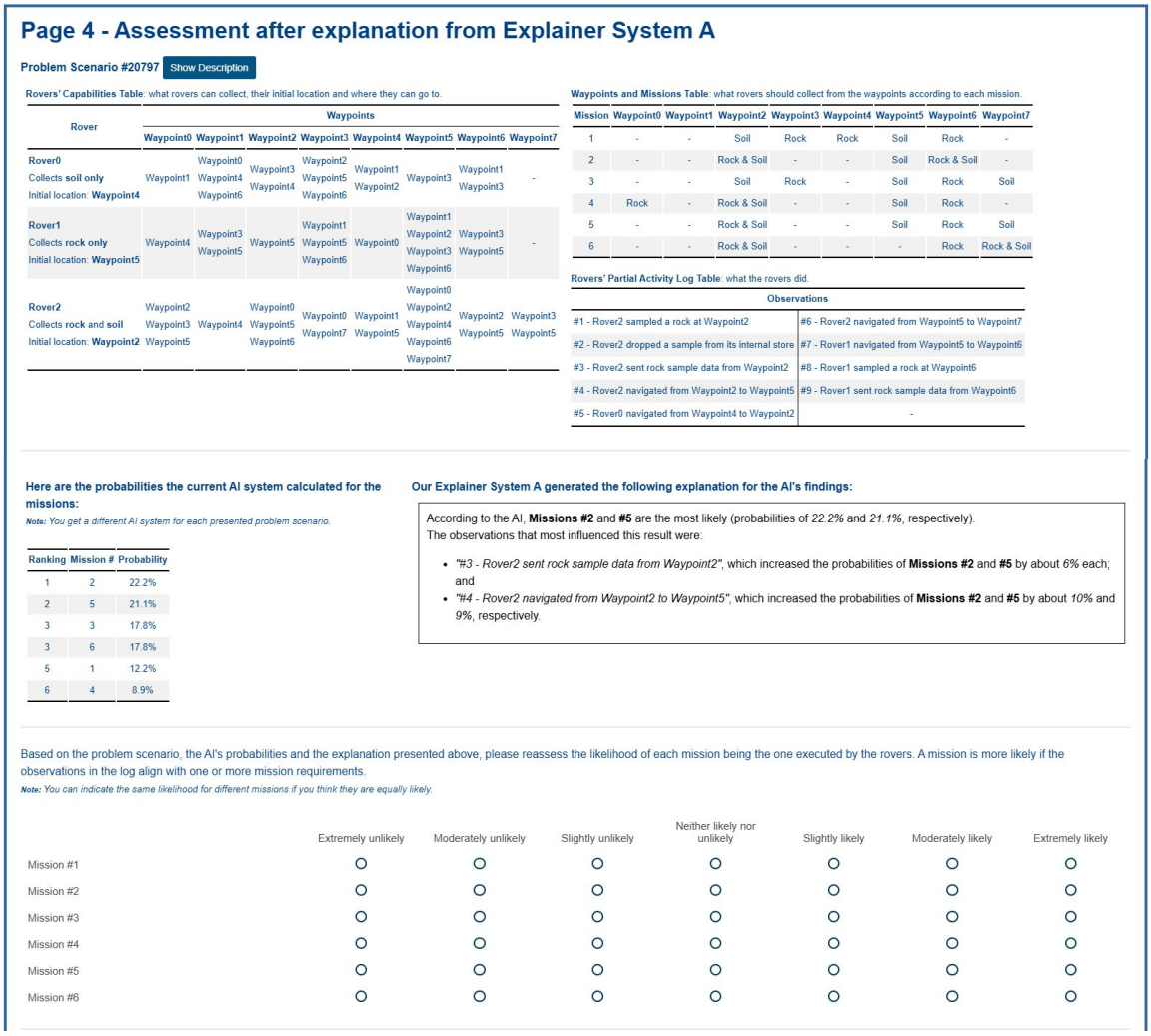


Figure 8: GR problem elements: rover capabilities, waypoints and observations; prediction of the GR alongside an Impact explanation about the prediction; and question about the likelihood of the missions after presenting the GR's prediction and the explanation.

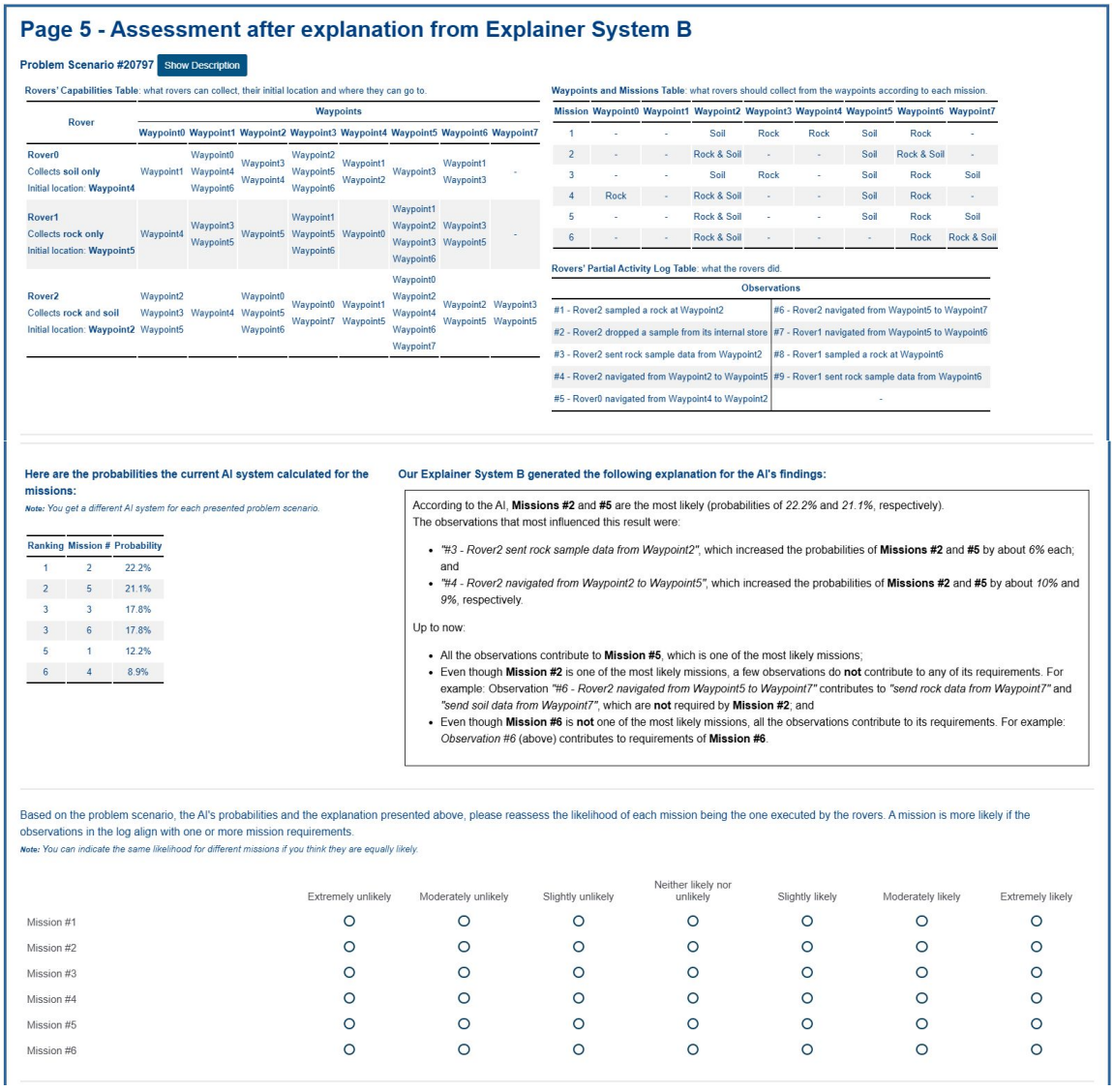


Figure 9: GR problem elements: rover capabilities, waypoints and observations; prediction of the GR alongside a Critique explanation about the prediction; and question about the likelihood of the missions after presenting the GR's prediction and the explanation.

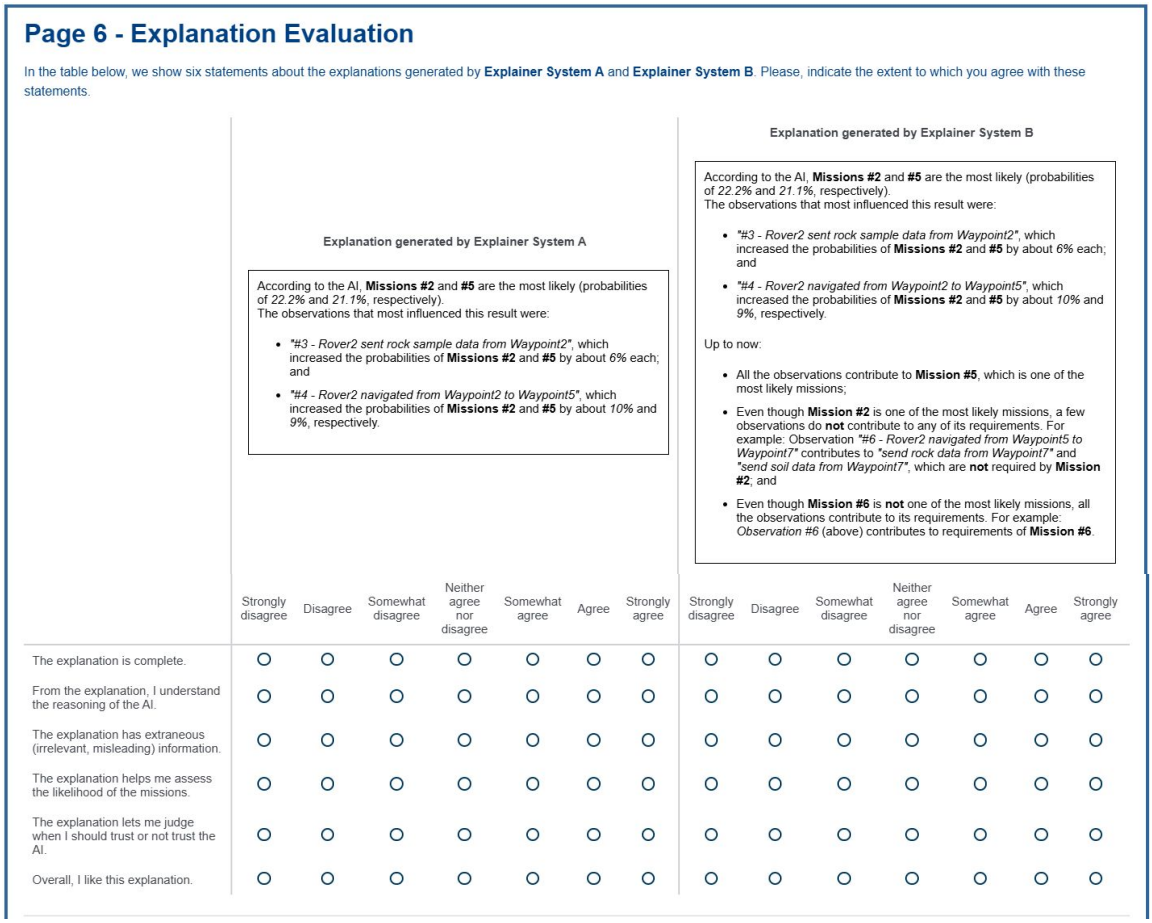


Figure 10: Questionnaire for rating explanatory attributes of Impact and Critique explanations in the within-subject experiment.

H Experimental Results

In Tables 7-9, statistically significant differences (*p-value* < 0.05) are **boldfaced**, and trends ($0.05 <$ *p-value* < 0.1 after adjustment for multiple comparisons) are ***italicized***.

- Table 6 shows descriptive statistics for our three cohorts: between subjects Impact and Critique, and within subject (Combined).
- Table 7 displays means (standard deviations) of our three metrics (L_{cor}, L_{cor} and $A_{\mathcal{G}}$) for three explanatory conditions in two experiments (between subjects and within subject) across three levels of correctness (correct, partially correct and incorrect).
- Table 8 displays the effects (i.e., contrasts) of statistically significant results from Table 7, where a contrast is the estimated mean difference between two explanatory conditions. For example, the first row states that, for the within-subject cohort and Correct prediction level, the estimated mean value of the L_{cor} variable is 0.529 higher (*p-value* < 0.05) for the Critique condition than for the None condition.
- Table 9 shows the means (standard deviations) of participants views about six explanatory attributes.

Table 6: Descriptive statistics for the Impact, Critique and Combined groups (number of participants) – two options with the most participants; and NCS score (on a 5-point Likert scale).

Attribute	**Option**	**Between subjects**		**Within subject**
		Impact (47)	Critique (46)	Combined (48)
Gender	Male / Female	24 / 23	27 / 19	30 / 16
Age	25-34 / 35-44	14 / 17	21 / 14	16 / 16
Ethnicity	Caucasian	30	27	27
English proficiency	High	46	46	47
Education	Bachelor / Some college, no degree	23 / 9	20 / 11	21 / 10
ML expertise	Medium / Low	29 / 13	33 / 7	31 /10
NCS score	Mean (std. dev.) [range: 6-30]	22.60 (5.83)	23.35 (4.99)	22.25 (5.75)

Table 7: L_{cor}, L_{incor} and $A_{\mathcal{G}}$ for three explanatory conditions (None, Impact and Critique) grouped by prediction correctness level for the between-subjects and within-subject experiments: mean (standard deviation); statistically significant differences of Impact/Critique explanations vs None (*p-value* < 0.05) are **boldfaced** and trends ($0.05 <$ *p-value* < 0.1 after adjustment) are ***italicized***.

Metric	**Correctness**	**Between subjects**				**Within subject (Combined)**			
		None	vs Impact	None	vs Critique	None	vs Impact	Critique	
L_{cor}	Correct	5.23 (1.48)	5.64 (1.35)	5.09 (1.59)	**5.56 (1.55)**	5.43 (1.27)	5.67 (0.96)	**5.80 (1.08)**	
	Part. Correct	4.66 (1.47)	4.71 (1.35)	4.60 (1.60)	***4.98 (1.41)***	4.90 (1.15)	5.12 (1.08)	**5.26 (1.06)**	
	Incorrect	4.48 (1.19)	4.36 (1.24)	4.23 (1.20)	4.23 (1.49)	4.61 (0.96)	4.51 (0.97)	4.79 (1.21)	
L_{incor}	Correct	3.49 (0.96)	3.38 (1.00)	3.11 (1.03)	3.13 (0.95)	4.16 (1.09)	4.16 (1.08)	4.15 (1.28)	
	Part. Correct	3.74 (0.89)	3.84 (0.95)	3.54 (0.95)	3.43 (0.89)	3.82 (1.07)	3.86 (1.09)	3.75 (1.14)	
	Incorrect	3.85 (1.07)	4.00 (1.07)	3.86 (0.93)	3.80 (0.95)	4.16 (1.09)	4.16 (1.08)	4.15 (1.28)	
$A_{\mathcal{G}}$	Correct	0.71 (0.29)	0.76 (0.24)	0.71 (0.26)	0.75 (0.24)	0.73 (0.23)	0.78 (0.18)	0.76 (0.18)	
	Part. Correct	0.66 (0.25)	0.69 (0.25)	0.67 (0.24)	0.68 (0.20)	0.63 (0.24)	0.69 (0.22)	0.64 (0.20)	
	Incorrect	0.69 (0.24)	0.68 (0.23)	0.65 (0.22)	0.66 (0.22)	0.69 (0.21)	0.73 (0.20)	0.66 (0.22)	

Table 8: Estimated mean contrasts for significant results derived from the linear mixed-effects models: statistically significant differences (*p-value* < 0.05) are **boldfaced** and trends ($0.05 <$ *p-value* < 0.1 after adjustment) are ***italicized***.

			Estimated Mean Contrasts						
Metric	Cohort	Correctness	Contrast	Estimate	Std. error	DF	t-value	p-value	95% CI
L_{cor}	Within	Correct	Critique vs None	0.529	0.17	462.7	3.07	**0.012**	[0.08, 0.97]
L_{cor}	Within	Part. Correct	Critique vs None	0.449	0.17	460.4	2.68	**0.038**	[0.02, 0.88]
L_{cor}	Between–Critique	Correct	Critique vs None	0.623	0.23	299.9	2.66	**0.022**	[0.07, 1.17]
L_{cor}	Between–Critique	Part. Correct	Critique vs None	0.447	0.21	293.2	2.08	***0.095***	[-0.06, 0.95]

Table 9: Participants' views about Impact and Critique explanations in terms of six explanatory attributes grouped by prediction correctness level for the between-subjects and within-subject experiments: mean (standard deviation); statistically significant differences (*p-value* < 0.05) are **boldfaced** and trends ($0.05 <$ *p-value* < 0.1 after adjustment) are ***italicized***.

Explanatory Attribute	**Correctness**	**Between-subjects**			**Within-subject**		
		Impact	Critique	p-value	Impact	Critique	p-value
The explanation is complete	Correct	4.53 (1.61)	4.50 (1.59)	1.000	4.35 (1.48)	4.71 (1.58)	0.153
	Part. correct	4.15 (1.56)	4.70 (1.41)	0.332	3.92 (1.57)	5.48 (1.32)	**7.97E-06**
	Incorrect	3.62 (1.65)	4.50 (1.52)	***0.051***	3.56 (1.75)	5.52 (1.38)	**1.28E-05**
Helps me assess mission likelihood	Correct	4.91 (1.25)	5.04 (1.19)	1.000	4.96 (1.30)	5.08 (1.38)	0.365
	Part. correct	4.40 (1.57)	5.24 (1.06)	**0.045**	4.65 (1.52)	5.52 (1.11)	**0.001**
	Incorrect	4.11 (1.72)	4.84 (1.37)	0.216	4.46 (1.46)	5.42 (1.38)	**4.27E-04**
Helps me judge when to trust or not the AI	Correct	4.60 (1.42)	4.61 (1.37)	1.000	4.48 (1.41)	4.60 (1.32)	0.365
	Part. correct	4.28 (1.48)	5.00 (1.26)	***0.054***	4.42 (1.40)	5.33 (1.08)	**5.72E-04**
	Incorrect	4.40 (1.53)	4.67 (1.33)	1.000	4.54 (1.40)	5.19 (1.18)	**0.020**
Contains misleading or irrelevant info	Correct	3.06 (1.45)	2.89 (1.30)	1.000	3.50 (1.64)	3.15 (1.50)	0.363
	Part. correct	3.55 (1.53)	3.72 (1.38)	0.617	3.21 (1.64)	3.50 (1.69)	0.146
	Incorrect	3.55 (1.50)	3.46 (1.39)	1.000	3.33 (1.62)	3.73 (1.65)	0.229
Helps me understand the AI's reasoning	Correct	4.87 (1.41)	5.06 (1.39)	1.000	4.90 (1.37)	5.06 (1.45)	0.365
	Part. correct	4.30 (1.68)	4.83 (1.25)	0.617	4.52 (1.38)	5.40 (1.52)	**0.001**
	Incorrect	4.02 (1.70)	4.65 (1.55)	0.216	4.23 (1.50)	5.48 (1.40)	**6.71E-05**
I like this explanation	Correct	4.68 (1.56)	4.70 (1.35)	1.000	4.40 (1.45)	4.77 (1.60)	0.216
	Part. correct	4.34 (1.55)	4.70 (1.38)	0.617	3.98 (1.73)	5.58 (1.47)	**3.87E-05**
	Incorrect	3.66 (1.70)	4.41 (1.63)	0.213	4.02 (1.69)	5.35 (1.36)	**2.55E-04**