Incorporating Logic in Online Preference Learning for Safe Personalization of Autonomous Vehicles

Ruya Karagulle
ruyakrgl@umich.edu
University of Michigan
Ann Arbor, Michigan, USA

Necmiye Ozay
necmiye@umich.edu
University of Michigan
Ann Arbor, Michigan, USA

Nikos Aréchiga
nikos.arechiga@tri.global
Toyota Research Institute
Los Altos, California, USA

Jonathan DeCastro
jonathan.decastro@tri.global
Toyota Research Institute
Cambridge, Massachusetts, USA

Andrew Best
andrew.best@tri.global
Toyota Research Institute
Los Altos, California, USA

ABSTRACT

Customizing autonomous vehicles to align with user preferences while ensuring safety may significantly impact their adoption. Collecting user preference data by asking a large number of comparison questions can be demanding. In this work, we use active learning along with temporal logic descriptions of constraints to enable safe learning of preferences with a reduced number of questions. We take a Bayesian inference approach combined with Weighted Signal Temporal Logic (WSTL), resulting in a WSTL formula that can rank signals based on user preferences and be used for correct-and-custom-by-construction control synthesis. Our method is practical for formulas and signals with various complexity since we compute STL-related values offline. We provide an upper bound for the number of answers in disagreement with user answers. We demonstrate the performance of our method both on synthetic data and by human subject experiments in an immersive driving simulator. We consider two driving scenarios, one involving a vehicle approaching a pedestrian crossing and the other with an overtake maneuver. Our results over synthetic experiments with ground truth weight valuation show that our query selection algorithm converges faster than random query selection. Human subject study results show an average agreement of 94% with user answers during training, and 79% during validation (which increases to 86% when restricted to high confidence results).

CCS CONCEPTS

• **Theory of computation** → **Modal and temporal logics**; • **Computing methodologies** → **Learning to rank**; • **Mathematics of computing** → *Bayesian computation.*

KEYWORDS

temporal logic, preference learning, autonomous driving

ACM Reference Format:
Ruya Karagulle, Necmiye Ozay, Nikos Aréchiga, Jonathan DeCastro, and Andrew Best. 2024. Incorporating Logic in Online Preference Learning for Safe Personalization of Autonomous Vehicles . In *27th ACM International Conference on Hybrid Systems: Computation and Control (HSCC '24), May 14–16, 2024, Hong Kong, Hong Kong*. ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3641513.3650129

1 INTRODUCTION

People have different comfort and performance preferences for autonomous driving. Customization of autonomous vehicles' driving styles is a critical aspect of enhancing user satisfaction. This can be done by leveraging learning from users' driving demonstrations or asking pairwise comparison questions over different driving behaviors. Surveys indicate that users' driving preferences are usually distinct from their driving styles [1], which puts emphasis on learning using pairwise comparison questions. However, we need to consider safety at all times while integrating preferences on autonomous vehicles. Exclusively relying on preferences may result in unsafe behaviors and potentially catastrophic failures. Therefore, our goal is to customize the autonomous vehicle behaviors within a well-defined safety rule set. Personalizing autonomous algorithms with safety guarantees will better improve the safety of the road, as users will be less likely to disable it, allowing for a decrease in accidents caused by human errors [2].

As discussed in [3], preference learning frameworks for safety-critical applications have three desirable properties: (i) *expressivity*: the model should be able to capture different preferences, (ii) *safety*: the model should never prefer a rule-violating behavior over a rule-satisfying one, (iii) *integration:* the final model should be easy to integrate into downstream controller synthesis tasks. One approach to represent safety guarantees is to use formal logic. Pairwise comparisons to learn rule-based or logical constructs for driving behaviors have been considered in [4] for capturing common sense behaviors, in [5] to tailor driving style classes, and in [3] for personalization. Helou et al. [4] conduct an extensive study with over 65 participants to create a consensus over reasonable driving behavior, Karlsson et al. [5] compute a defensiveness score from pairwise questions to better categorize the driving style of the user as defensive, neutral or aggressive. Karagulle et al. [3] do not use pre-defined driving styles but incorporate logical structure to solve the safety-guaranteed preference learning problem. They argue that drivers have a general rule set in their mind, but they

HSCC '24, May 14–16, 2024, Hong Kong, Hong Kong

ACM ISBN 979-8-4007-0522-9/24/05
https://doi.org/10.1145/3641513.3650129

assign different importance to subrules. All these approaches use offline data sets, where participants are asked a fixed set of comparison questions. Here, a question consists of a pair of driving behaviors/videos, and an answer indicates the user's preference within this pair. The authors in [3] acknowledge that the extensive number of questions posed to users was said to be cumbersome.

In this work, we introduce an active learning framework that integrates safety guarantees into preference learning. We incorporate temporal logic formalism into a Bayesian learning approach. We leverage human preference models from psychology and include Weighted Signal Temporal Logic (WSTL) satisfaction measure *weighted robustness* into the human preference mechanism. The framework provides specific benefits: (1) it is provably safe since a rule-violating signal can never have a greater weighted robustness value than a rule-satisfying one, (2) it is practical for many types of formulas and real-life applications, as we compute weighted robustness values offline, (3) it learns the preferences with a reduced number of questions with respect to offline learning methods by leveraging a greedy question-selection algorithm with expected information gain maximization. It is demonstrated that such greedy question selection methods select both informative and easy comparison questions when learning reward functions consistent with preferences [6]. Our framework uses a similar greedy approach to learn a weight valuation for a given WSTL formula in a way that more preferable behaviors satisfy this formula with higher weighted robustness values. Therefore, this formula can be readily used for controller synthesis to generate behaviors that are correct and maximally preferred by the user. We also establish a theoretical bound that quantifies the gap between the output weight valuation and an optimal weight valuation in terms of the maximum number of answers in agreement with user preferences.

To validate our approach, we conduct two sets of experiments: one employing synthetic data and another involving human subjects. Experiments with synthetic data aim to test the adaptability of the framework to different conditions. We also use these synthetic experiments to determine some of the hyper-parameters used in our method. Then, we conduct a human subject study, where the participants experience different driving behaviors in an immersive driving simulator. They are asked about their preferences in two different scenarios: one involving the vehicle approaching a pedestrian crossing and the other the vehicle performing an overtake maneuver. With these experiments, we demonstrate the ability of the proposed framework to capture human preferences.

2 LITERATURE REVIEW

Preference learning aims to predict a preference order for a set of options from an individual's preferences [7]. This is achieved by finding a utility function that ranks preferences in such a way that preferred items will have a greater value than their counterparts. Preference learning can be done offline or online. Training learning models in an online fashion has been studied for different reasons. One of the hypotheses supporting active learning is that selecting the next data point may help *learners* to perform better with less training data [8]. In the context of preference learning, active learning can ease the burden from *teachers* for answering comparison questions [9, 10].

In autonomous systems, active preference learning is used to infer system behaviors from expert choices [6, 11–13]. For instance, in inverse reinforcement learning, preferences are used to learn a reward function. Some active preference-based learning works propose different query selection methods. Sadigh et al. [12] introduce a volume removal method that maximizes the expected volume to be removed by answering a comparison pair. Biyik et al. [6] use maximum expected information gain to select the next query. Wilde et al. [13] solve the same problem by maximizing regret under some constraints. None of these works consider safety. Cosner et al. [14] propose a safety-aware method for preference-based learning using Control Barrier Functions, which encode state constraints.

For safety-critical applications, encoding rules as temporal logic formulas is a popular method in controller synthesis [15, 16], motion planning [17, 18] and learning applications [19, 20]. Temporal logics allow specifying complex rules beyond simple state constraints. We can additionally incorporate preferences into logic [21, 22]. Venkatesh [21] proposes a new temporal logic grammar that helps with expressing preferences over different formulas. Mehdipour et al. [22] propose a weighted extension to Signal Temporal Logic (WSTL), which gives different importance to subrules of a formula. They also propose a control synthesis method to be used with WSTL formulas when the weights are known. However, from an end-user perspective, it may be hard to interpret the weights and define preferences using weights in a temporal logic formula, so there needs to be an intermediate step to infer the weights based on user preferences. Works in [23, 24] present parametric extension to WSTL, namely Parametric Weighted Signal Temporal Logic (PWSTL). Karagulle et al. [3] propose an offline learning approach to learn the weights of a PWSTL formula from pairwise comparisons. Yet, it is indicated that the number of questions asked during training was cumbersome for users to answer. Active learning may help in reducing the number of questions and obtaining better performance, which motivates the current paper. To the best of our knowledge, our paper is the first work in the literature that solves the active preference learning problem with safety guarantees over complex logical structures.

3 PRELIMINARIES ON STL, WSTL, AND PWSTL

STL is a temporal logic proposed to reason about signals $\sigma : \mathbb{T} \to \mathbb{R}^n$, where $\mathbb{T}$ is the time domain and $\mathbb{R}^n$ is an n-dimensional real domain [25]. Domain $\mathbb{T}$ can be infinite $\mathbb{Z}_{\geq 0}$ or finite $[0, t_{final}] \subset \mathbb{Z}_{\geq 0}$. An STL formula ϕ is given by the syntax

$$\phi := \top \mid \pi \mid \neg\phi \mid \phi_1 \wedge \phi_2 \mid \phi_1 \mathcal{U}_{[a,b]} \phi_2,$$

where $\top$ is Boolean true, π is a predicate of the form $\pi(s(t)) := f_\pi(s(t)) \geq 0$ where $f_\pi : \mathbb{R}^n \to \mathbb{R}$, $\neg$ is the negation operator, $\wedge$ is the conjunction operator, and $\mathcal{U}_{[a,b]}$ is the (bounded) until operator, which is a temporal operator. We define the time interval for temporal operators with subscript $[a, b]$. We omit the subscript when the interval covers the total time length of the signal. Other common operators are derived in the usual way: disjunction is $\phi_1 \vee \phi_2 = \neg(\neg\phi_1 \wedge \neg\phi_2)$, (bounded) eventually is $\Diamond_{[a,b]}\phi = \top\mathcal{U}_{[a,b]}\phi$, and (bounded) always is $\Box_{[a,b]}\phi = \neg(\Diamond_{[a,b]}\neg\phi)$. The set of all well-formed STL formulas is denoted as $\Diamond$. If a signal s satisfies a formula ϕ at time t, it is shown as $s_t \models \phi$. If it violates at t, it is shown as $s_t \not\models \phi$. The qualitative semantics of STL can be found in [25]. STL has quantitative semantics as well, which is a measure of satisfaction or violation of the formula by the signal. As a quantitative semantics, we use the traditional robustness metric definition in [26]. Robustness metric $\rho : \mathbb{R}^n \times \mathcal{F} \times \mathbb{T} \rightarrow \mathbb{R}_e$ is defined recursively as:

$$\begin{aligned}
\rho(s, \top, t) &= \infty, \\
\rho(s, \pi, t) &= f_\pi(s(t)), \\
\rho(s, \neg\phi, t) &= -\rho(s, \phi, t), \\
\rho(s, \phi_1 \wedge \phi_2, t) &= \min\big(\rho(s, \phi_1, t), \rho(s, \phi_2, t)\big), \\
\rho(s, \phi_1 \mathcal{U}_{[a,b]} \phi_2, t) &= \max_{t' \in [t+a, t+b]} \big(\min\big(\rho(s, \phi_2, t'), \\
&\qquad \min_{t'' \in [t, t']} \rho(s, \phi_1, t'')\big)\big).
\end{aligned}$$

Robustness for derived operators can be defined similarly. The robustness metric ρ is *sound*, i.e., $\rho(s, \phi, t) > 0 \implies s_t \models \phi$ and $\rho(s, \phi, t) < 0 \implies s_t \not\models \phi$. Robustness at $t = 0$ is shown as $\rho(s, \phi)$. Note that for finite signals where $t_{final} < \infty$, time interval $[t+a, t+b]$ in temporal operators may exceed the time length of the signal. In this case, time interval can be taken as $[t + a, \min(t + b, t_{final})]$ assuming that $t + a \leq t_{final}$, and a slight modification gives finite semantics [27].

WSTL represents priorities and preferences in STL formulas [22]. While operators are interpreted as in STL, the syntax extends STL syntax as

$$\phi := \top \mid \pi \mid \neg\phi \mid \phi_1 \wedge^w \phi_2 \mid \phi_1 \mathcal{U}^{w^1, w^2}_{[a,b]} \phi_2,$$

where $w \in \mathbb{R}^2_+$ and $w^1, w^2 \in \mathbb{R}^{(b-a+1)}_+$ are the weights. The quantitative semantics of WTSL is called *weighted robustness*, denoted as $r : \mathbb{R}^n \times \mathcal{F} \times \mathbb{T} \rightarrow \mathbb{R}$. We adopt WSTL formalism with the following quantitative semantics, denoted $r(s, \phi, t)$:

$$\begin{aligned}
r(s, \top, t) &= \infty \\
r(s, \pi, t) &= \rho(s, \pi, t) \\
r(s, \neg\phi, t) &= -r(s, \phi, t), \\
r(s, \phi_1 \wedge^w \phi_2, t) &= \min\big(w_1 r(s, \phi_1, t), w_2 r(s, \phi_2, t)\big), \\
r(s, \phi_1 \mathcal{U}^{w^1, w^2}_{[a,b]} \phi_2, t) &= \max_{t' \in [t+a, t+b]} \Big(\min\Big(w^1_{t'-t-a+1} r(s, \phi_2, t'), \\
&\qquad w^2_{t'-t-a+1} \min_{t'' \in [t, t')} r(s, \phi_1, t'')\Big)\Big).
\end{aligned} \tag{1}$$

Derived operators have weighted robustness definitions as:

$$\begin{aligned}
r(s, \phi_1 \vee^w \phi_2, t) &= \max\big(w_1 r(s, \phi_1, t), w_2 r(s, \phi_2, t)\big), \\
r(s, \square^w_{[a,b]} \phi, t) &= \min_{t' \in [t+a, t+b]} \big(w_{t'-t-a+1} r(s, \phi, t')\big), \\
r(s, \diamond^w_{[a,b]} \phi, t) &= \max_{t' \in [t+a, t+b]} \big(w_{t'-t-a+1} r(s, \phi, t')\big).
\end{aligned} \tag{2}$$

As in STL robustness, $r(s, \phi)$ is weighted robustness at $t = 0$. Quantitative semantics defined in (1) and (2) is sound [3]. STL and WSTL formulas can be represented with a syntax tree [28]. Nodes of this syntax tree denote operators, and edges denote the connection between operators and operands. With a slight abuse of notation, the edge weights denote weights of the WSTL formula.

The most important subrule is the one that affects the final robustness value. Consider two different driving scenarios, where in the first one, vehicle needs to satisfy φ_1 and φ_2, that is, $\varphi = \varphi_1 \wedge^w \varphi_2$, and in the second one vehicles needs to avoid φ_3 or φ_4 at all times, that is $\varphi' = \neg(\varphi_3 \wedge^w \varphi_4)$. For the case when $r(s, \varphi_1) = r(s, \varphi_2)$, the subrule with smaller weight value affects $r(s, \varphi)$. However, when $r(s, \varphi_3) = r(s, \varphi_4)$, the subrule with greater weight value affects $r(s, \varphi')$. Therefore, the magnitude of weights is not easy to interpret but the order of weighted robustness values is interpretable. We define *root weights* as the weights associated with the weighted operator closest to the root of its syntax tree. For instance, for φ, the root of its syntax tree has the operator $\wedge^w$ and the root weights are w. For φ', the root of its syntax tree has $\neg$, which is not a weighted operator, so we look at its children until we find a weighted operator, which is again $\wedge^w$. Hence, root weights of φ' are w. One of the properties of WSTL formulas is root-layer homogeneity.

Lemma 1 (Root-layer homogeneity, Lemma 2 of [3]). *Let ϕ be a WSTL formula with root weights w, and $\tilde{w} = \alpha w$ be a valuation with $\alpha > 0$. We have $r(s, \phi_{\mathcal{W}=\tilde{w}}) = \alpha r(s, \phi_{\mathcal{W}=w})$.*

Finally, PWSTL is a parametric extension of WSTL. In WSTL, weights are unknown parameters (cf., [23]). We denote the set of unknown parameters as $\mathcal{W}$ and denote PWSTL formulas as $\phi_{\mathcal{W}}$. A PWSTL formula results in a WSTL formula $\phi_{\mathcal{W}=w}$ with the valuation w of parameters.

4 PROBLEM STATEMENT AND MOTIVATION

Given a driving scenario, we assume people follow a set of traffic rules, representable by an STL formula. Within all rule-abiding behaviors, different people might prefer different behaviors. Weights in the WSTL formalism can be used to express such differences. That is, we assume each user (possibly subconsciously) has a weight valuation w^H that captures their preferences and our goal is to learn this w^H. The weighted robustness of the WSTL formula with valuation w^H can be seen as a utility function that assigns greater robustness value to preferred signals than their non-preferred counterparts for all pairs. Consider that we have a candidate PWSTL formula $\phi_{\mathcal{W}}$ that specifies the scenario rule set. We denote the set of signals that we ask pairwise questions from as $\mathcal{S} = \{\sigma_i\}_{i=1}^K$. The set of all questions created from this set is $Q = \{q_{ij} = (\sigma_i, \sigma_j) : i \in \{1, K-1\}, j \in \{i+1, K\}\}$. The problem we aim to solve is:

Problem 1. *Given a formula $\phi_{\mathcal{W}}$, a set $\mathcal{S}$ of signals that satisfy ϕ, and the corresponding question set Q, find a valuation w^* for $\mathcal{W}$ by adaptively selecting questions from Q based on previous answers that "best" estimates w^H.*

A proper metric that measures the "best" estimate will be formalized in the next section.

5 METHODOLOGY

Problem 1 is a combinatorial active learning problem. To formalize it, we first present some simplifications to the problem and our modeling choice for human decision-making. This will allow us to recast the problem as a Bayesian active learning problem with noisy observations [29, 30]. We will then propose an algorithm to solve it.

As a first step, we sample a finite set Ω_W of M weight valuations from the intersection of a unit norm ball and the positive quadrant. Restricting the weight valuations to this region is without loss of generality as shown in [3]. Moreover, uniform random sampling of weights is shown to be sufficient in capturing human preferences in several driving scenarios [3], hence we will search for an estimate of w^H in Ω_W. We model the individual's weight valuation as a random variable W over Ω_W, with a known prior distribution. In particular, we use a uniform prior, i.e., $P(W = w) = \frac{1}{M}$ for all $w \in \Omega_W$.[1]

As mentioned earlier, a question consists of a signal pair $q_{ij} = (\sigma_i, \sigma_j)$ and an answer is a choice within this pair. We represent the answer for the question q_{ij} as a random variable A_{ij}, which takes values from $\mathcal{A} = \{0, 1\}$. Here, $A_{ij} = 0$ means choosing σ_i over σ_j, and $A_{ij} = 1$ means vice versa. We assume the answers to each question are independent of each other given W. Bradley-Terry model is commonly used to represent people's decision-making mechanism [31]. We incorporate our weighted robustness definition into this model:

$$\begin{aligned} P(A_{ij} = k \mid q_{ij}, w) &= \frac{(1-k)e^{r(\sigma_i, \phi_{\mathcal{W}=w})} + k e^{r(\sigma_j, \phi_{\mathcal{W}=w})}}{e^{r(\sigma_i, \phi_{\mathcal{W}=w})} + e^{r(\sigma_j, \phi_{\mathcal{W}=w})}} \\ &= \frac{e^{\Delta r^w(q_{ij})} + k(1 - e^{\Delta r^w(q_{ij})})}{1 + e^{\Delta r^w(q_{ij})}}, \end{aligned} \tag{3}$$

where $\Delta r^w(q_{ij}) = r(\sigma_i, \phi_{\mathcal{W}=w}) - r(\sigma_j, \phi_{\mathcal{W}=w})$. Note that the probability in (3) satisfies $P(k|q, w) \in [0, 1]$, and its magnitude depends on the weighted robustness difference of signals $\Delta r^w(q_{ij})$. We want this difference to represent the decisiveness level of the human on this question. If the value of $P(A_{ij} = k|q_{ij}, w)$ is close to 0 or 1, it reads the user has a strong opinion on their preferences. If it is close to 0.5, it reads the person cannot definitely decide on the preference. However, $\Delta r^w(q_{ij})$ depends also on weight value magnitudes. If weight values are too small, weighted robustness values will be small, and so do robustness differences. This may result in an unfair bias towards weight valuations with larger weights. Let $\Delta r = \min_{w \in \Omega_W} \min_{q \in Q} |\Delta r^w(q)|$ be the minimum of absolute values of weighted robustness difference for all pairs in Q and for all w in Ω_W. Given the root-layer homogeneity property in Lemma 1, we can lower bound Δr by scaling up root-layer weights, while keeping the preference ordering induced by the weights invariant. This gives us a hyper-parameter to adjust the human decisiveness level.

To attempt Problem 1, we need a question selection policy and a criterion to choose the best estimate in Ω_W based on human answers. A question selection policy π determines which question to pick from Q next based on the answers so far. If we restrict the number of questions to B, the policy π returns a sequence of questions and answers $\Gamma^\pi = (q_1^\pi, a_{q_1^\pi}), \ldots, (q_B^\pi, a_{q_B^\pi})$, which is a random variable. Let us define the entropy H of W given π:

$$H(W \mid \pi) \doteq \mathbb{E}_{\Gamma^\pi}[H(W \mid \Gamma^\pi)].$$

One common objective value is to maximize the information gain $I(\pi; W) = H(W) - H(W \mid \pi)$, which is equivalent to minimizing the conditional entropy $H(W \mid \pi)$ since the first term does not depend on the policy. So, the active question selection problem becomes that of minimizing $H(W \mid \pi) = -\sum_{w \in \Omega_W} P(w \mid \pi) \log(P(w \mid \pi))$ for a budget B of questions.

Problem 2. *Given a formula $\phi_{\mathcal{W}}$, a set $\mathcal{S}$ of signals that satisfy ϕ, the corresponding question set Q, and Ω_W, find an optimal policy π^* by solving*

$$\begin{aligned} \pi^* \in \arg\min_\pi \quad & H(W \mid \pi) \\ \textit{s.t.} \quad & |\gamma^\pi| \leq B. \end{aligned} \tag{4}$$

Problem 2 is generally hard [32, 33], but an effective greedy solution is commonly used, which we adopt in this work. At each step, we pose a question pair, we get the answer from the human, and we update our posterior distribution over weight valuations W accordingly. We represent all question-answer tuples until step N as $\bar{\gamma}_N := \{(q_i, a_i)\}_{i=1}^N$. Note that $\bar{\gamma}_N$ is a (partial) realization of the random variable Γ^π for the policy being used. Let $Q_N = \{q_i\}_{i=1}^N \subseteq Q$ be the set of questions we have already selected. By the human decision model assumption, answers do not depend on the order of questions, that is, $P(a|q, w, \bar{\gamma}_N) = P(a|q, w)$. We denote the probability distribution after N steps as $P(W|\bar{\gamma}_N)$. For the greedy query selection policy, we select the question that gives maximum expected information gain over weight valuations. It can be shown as $I(W; (q, a)|\bar{\gamma}_N) = H(W|\bar{\gamma}_N) - H(W|\bar{\gamma}_N \cup \{(q, a)\})$, hence it is the decrement that is achieved in the objective function in Equation (4) when choosing a single question q and getting the answer a, given the observations so far. Then, our greedy query selection policy solves

$$q^* \in \arg\max_{q \in Q \setminus Q_N} \mathbb{E}_A[I(W; (q, A) \mid \bar{\gamma}_N)], \tag{5}$$

at each iteration. When we substitute the entropy definition of the information gain, we note that $H(W|\bar{\gamma}_N)$ is the same for all questions. We have $P(w, a|q, \bar{\gamma}_N) = P(a|q, w)P(w|\bar{\gamma}_N)$. Therefore, the optimization Equation (5) can be written as

$$q^* \in \arg\max_{q \in Q \setminus Q_N} \sum_{\substack{a \in \mathcal{A} \\ w \in \Omega_W}} P(a|q, w)P(w|\bar{\gamma}_N) f_{\bar{\gamma}_N}(a, q, w).$$

with $f_{\bar{\gamma}_N}(a, q, w) = \log\left(\frac{P(a|q,w)P(w|\bar{\gamma}_N)}{\sum_{w \in \Omega_W} P(a|q,w)P(w|\bar{\gamma}_N)}\right)$. We know $P(a|q, w)$ from the human model (3) and discuss how to obtain the conditional probabilities $P(w|\bar{\gamma}_N)$ next.

Note that the set of remaining questions shrinks with every step. Once the user provides an answer a^* to the question, we update the posterior of weight valuations using Bayes' Rule for each $w \in \Omega_W$:

$$P(w|\bar{\gamma}_{N+1}) = \frac{P(a^*|q^*, w)P(w|\bar{\gamma}_N)}{\sum_{w \in \Omega_W} P(a^*|q^*, w)P(w|\bar{\gamma}_N)}. \tag{6}$$

This probability represents the probability of w being the correct valuation. Therefore, to determine the best estimate for w^H, we use *the most likely valuation*

$$w^* = \arg\max_{w \in \Omega_W} P(w|\bar{\gamma}_{N+1}), \tag{7}$$

where $\bar{\gamma}_{N+1} = \bar{\gamma}_N \cup \{(q^*, a^*)\}$ is the selection criterion.

Problem 2 includes question limit B. Another natural limit to terminate asking questions is to ask all questions in Q, which can happen if $B > |Q|$. We also add a third termination condition that depends on the posterior probability of the most likely valuation. If the framework is confident enough that the most likely valuation represents the human's answers, then we can terminate early. Therefore, the third termination condition is $P(w^*|\bar{\gamma}_k) \geq P_{thresh}$.

[1] Through the rest of the paper, we use uppercase letters for random variables and lowercase for their values/realizations. For a random variable V, we often shorten $P(V = v)$ as $P(v)$ when it is clear from the context.

The workflow can be summarized in Algorithm 1. Lines 1-3 correspond to initialization, lines 6-7 select the query, line 8 sets the user answer, lines 9-10 update the probability distribution, and line 11 picks the most likely valuation.

Algorithm 1 An algorithm for active preference learning of WSTL formula

```
Input $\mathcal{S} = \{\sigma_1, \sigma_2, \dots, \sigma_K\}$
 1: $\Omega_W \leftarrow M$ Uniform Samples
 2: $Q \leftarrow \{(\sigma_i, \sigma_j) : i \in \{1, \dots, K-1\}, j \in \{i+1, K\}\}$
 3: $k \leftarrow 0, \bar{\gamma}_k = \emptyset, P(w|\bar{\gamma}_k) = \frac{1}{M}, \quad \forall w \in \Omega_W$
 4: while $P(w^*) < P_{thres}$ or $k \leq B$ or $Q \neq \emptyset$ do
 5:     $q^* \leftarrow$ Equation (5)
 6:     $Q \leftarrow \text{pop}(Q, q^*)$
 7:     $a^* \leftarrow \gamma_H(q^*)$
 8:     $\bar{\gamma}_{k+1} \leftarrow \bar{\gamma}_k \cup \{(q^*, a^*)\}$
 9:     $P(w|\bar{\gamma}_{k+1}) \leftarrow$ Equation (6)
10:     $w^* \leftarrow$ Equation (7)
11:     $k \leftarrow k + 1$
12: end while
Output $w^*$
```

6 MODEL ANALYSIS AND THEORETICAL GUARANTEES

To understand how a valuation differs from another, we compare their preferences in pairs. We assume that there exists a hypothetical person whose internal weight valuation is w. This person should pick a deterministic answer to questions. To determine these potential answers that are tied to w, we compute the maximum likelihood estimate of answers to a question, $a_q^w = \arg\max_{a \in \mathcal{A}} P(a|q, w)$. For the sake of simplicity, we say that a_q^w is the answer that the weight valuation w would pick. Then, we compute *agreement* on answers between the two weight valuations given a question set. The agreement between w_1 and w_2 over set Q_k, denoted $\text{agr}(w_1, w_2, Q_k)$, is computed as

$$\text{agr}(w_1, w_2, Q_k) = \frac{\sum_{q \in Q_k} \mathbb{1}(q)_{a_q^{w_1} = a_q^{w_2}}}{|Q_k|},$$

where $\mathbb{1}(\cdot)$ is the indicator function. The agreement is the percentage of questions on which realizations parameterized by two weight valuations have the same answer. *User agreement* of valuation w is the agreement between w and a partial realization $\bar{\gamma}_k$ over Q_k, denoted as $\text{agr}_H(w, \bar{\gamma}_k)$,

$$\text{agr}_H(w, \bar{\gamma}_k) = \frac{\sum_{q_i \in Q_k} \mathbb{1}(q)_{a_{q_i}^w = a_i}}{|Q_k|}.$$

Our method aims to find a weight valuation that best represents the user. A suitable valuation should match user preferences while considering potential inconsistencies due to reasons discussed earlier. Otherwise, we may overfit the training data. Thus, we do not want to output a valuation that necessarily gives the maximum agreement on seen questions and answers. However, we still need to keep the agreement as a metric in our decision mechanism. We can leverage our human preference model to quantify the trade-off between agreement and generalizability. Putting a lower bound to the minimum of absolute values of robustness differences Δr, we provide a bound for agreement between the most likely weight valuation and a valuation having the maximum agr_H.

Theorem 1. *Assume that $P(a|q, w) \in [0, u] \cup [1-u, 1]$ where $0 \leq u \leq 0.5$ for all answers to questions in Q and for all w in Ω_W. Let $\bar{\gamma}_L$ be a partial realization, and Q_L be the set of questions we have already selected. Let w^* be the most likely valuation after L questions and $\bar{w}$ be a weight valuation with the maximum user agreement, that is, $\bar{w} \in \arg\max_{w \in \Omega_W} agr_H(w, \bar{\gamma}_L)$. After asking L questions, the agreement $agr(w^*, \bar{w}, Q_L) \geq 1 - \frac{\log(1-u)}{\log(u)}$.*

Proof. To prove the lower bound on agreement, we examine the worst-case scenario. Assume that $\bar{w}$ picks all answers in agreement with human answers but stays at the least decisive side, that is $P(a|q, \bar{w}) = 1-u$, for all $(q, a) \in \bar{\gamma}_L$. Other weight valuations in Ω_W pick $L-N$ answers in agreement with human answers while staying around the most decisive side $P(a|q, w) \approxeq 1$, and pick N answers in disagreement with human answers while staying at the least decisive side $P(a|q, w) = u$. After L questions, the posterior of $\bar{w}$ is[2] $P(\bar{w}|\bar{\gamma}_L) = \frac{(1-u)^L}{(1-u)^L + (M-1)u^N}$, and the posterior of other weight valuations $w \in \Omega_W \setminus \bar{w}$ is $P(w|\bar{\gamma}_L) = \frac{u^N}{(1-u)^L + (M-1)u^N}$. If we are to choose a weight valuation $w^* \in \Omega_W \setminus \bar{w}$, we have $P(w^*|\bar{\gamma}_L) \geq P(\bar{w}|\bar{\gamma}_L)$. Thus, we have $u^N \geq (1-u)^L$, and $N \leq L\frac{log(1-u)}{log(u)}$. Agreement between w^* and $\bar{w}$ over Q_L is $\text{agr}(w^*, \bar{w}, Q_L) = \frac{L-N}{L} \geq 1 - \frac{log(1-u)}{log(u)}$. □

The bound u represents the trust we have in human answers. Please note that as u approaches to 0, N approaches to 0 as well. We can define this probabilistic bound in terms of the weighted robustness difference of signals.

Lemma 2. *If $\Delta r \geq \Delta r^*$, then $P(a|q, w) \in \left[0, \frac{e^{-\Delta r^*}}{1+e^{-\Delta r^*}}\right] \cup \left[\frac{1}{1+e^{-\Delta r^*}}, 1\right]$ for all questions, answers, and weight valuations in Ω_W.*

The proof directly follows Equation (3). Note that by using root-layer homogeneity, we scale up root-layer weights and set Δr^*. This step is added after line 1 in Algorithm 1. With Theorem 1, we put a bound to the disagreement level that the most likely weight valuation can have.

Another point to look at is the performance of selecting questions greedily, as is done in our algorithm, compared to the optimal solution of Problem (4). While the greedy algorithms are known to be widely effective in practice [32], it is interesting to understand if they can perform poorly in some extreme cases or whether one can provide performance guarantees under certain conditions. Prior work has shown that when the answers are deterministic, i.e., when $u = 0$, information gain is adaptive submodular [29], which guarantees that the greedy approach provides a constant factor sub-optimal solution to Problem (4). However, $u = 0$ is not a good model for human decision-making as it does not take into account the uncertainty in users' answers. On the other hand, when using the

[2] Initializing prior distribution uniformly, the probability update after k questions is $P(w|\bar{\gamma}_k) = \frac{\prod_{((q_{ij}, a_{ij}) \in \bar{\gamma}_k)} P(a_{ij}|q_{ij}, w)}{\sum_{w' \in \Omega_W} \prod_{((q_{ij}, a_{ij}) \in \bar{\gamma}_k)} P(a_{ij}|q_{ij}, w)}$.

(a) Pedestrian Scenario: Vehicle approaching an intersection with a stop sign while a pedestrian is crossing. The traffic rule says to stop before the stop sign and keep a safe distance from the pedestrian.

(b) Overtake scenario: Vehicle completing an overtaking maneuver on the highway. There is a speed limit enforced and the ego vehicle must keep a safe distance from the lead vehicle.

Figure 1: Experiment scenarios simulated with CARLA.

human model in Equation (3), we are in the noisy Bayesian learning setting for which adaptive submodularity-based guarantees are no longer valid for our cost function [29, 33]. On the other hand, we can still provide suboptimality guarantees for this approach using the results in [32] if we were to allow repetition of questions. In this setting, the suboptimality gap increases proportional to the minimum squared total variation between the distributions $P(A|q, w)$ for a given q and different w, which essentially is an indication of how informative the answers to a question are for identifying the underlying weight valuation.

7 EXPERIMENTS

In this section, we present an empirical analysis of the framework's performance. Following this, we share the results of a human-subject study conducted using an immersive driving simulator. Our studies involve two distinct driving scenarios.

Driving Scenarios: The first scenario is an intersection with a stop sign wherein a pedestrian is crossing the crosswalk. An illustrative screenshot is shown in Figure 1a. The candidate rule set for this behavior can be expressed as $\phi_p = \phi_p^r \wedge \phi_p^c$, where $\phi_p^r = \Box(d \geq 2) \wedge \Diamond\Box_{[0,1]}(v = 0)$ denotes the traffic rule, and $\phi_p^c = \Box(a \leq 10 \wedge \dot{a} \leq 30)$ denotes comfort related specifications. Variable d represents the distance of the ego vehicle to the pedestrian, v, a, and $\dot{a}$ denote the ego vehicle's speed, acceleration, and jerk, respectively. The comfort rule is trivially true for all vehicles behaving reasonably, it is included to increase preference capturing power of the formula by adding more weights to the weight set. Note that $\rho(s, \phi) \leq 0$ for all signals in the signal domain, since $v = 0$ predicate only holds with a robustness value of 0 or is violated with a negative robustness value. Therefore, after ensuring that all signals in $\mathcal{S}$ satisfy ϕ, we treat the speed predicate as Boolean with quantitative robustness value infinity, it is practically removed from the robustness computation of a series of conjunctions as in [3].

The second scenario involves overtaking behavior on a highway. A screenshot is shown in Figure 1b. The candidate rule set is defined as $\phi_o = \phi_o^r \wedge \phi_o^c$, where $\phi_o^r = \Box(d \geq 2)$, and $\phi_o^c = \Box(d_{lat} \geq 0) \wedge \Box(d_{long} \geq 0) \wedge \Box(v_{rel} \geq 0)$. Variable d, d_{lat}, and d_{long} denote total distance, lateral distance, and longitudinal distance from the ego's longitudinal axis to the lead vehicle, respectively. Variable v_{rel} is the relative speed of the ego vehicle with respect to the lead vehicle. Note that ϕ_o^c is trivially true for all vehicles that complete a safe overtaking. These sub-formulas in ϕ_o^c increase flexibility to capture underlying preferences.

Simulator: The simulator shown in Figure 2 utilizes a 6-degree of freedom motion base, 250° projection system, and the CARLA simulator for physics modeling [34] along with ROS2. For both scenarios “Town 5" of CARLA is used.

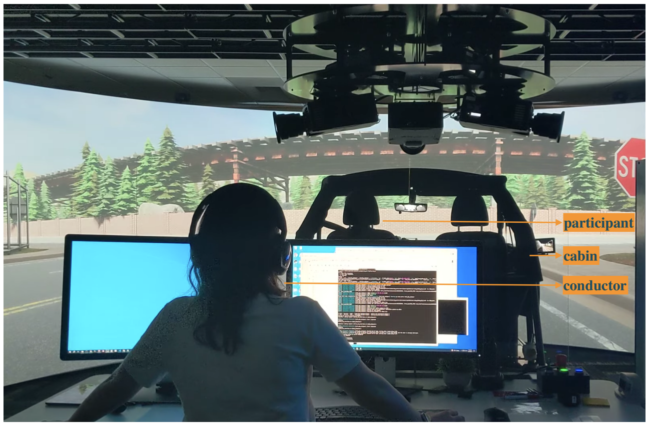

Figure 2: An instance from the human subject studies with the simulator running the pedestrian scenario.

Trajectory generation: Trajectory generation is completed using the simulator when the motion base is activated. To generate naturalistic and distinguishable trajectories, we relied on professional racing drivers. Drivers are prompted to drive each scenario with only simple stylistic cues such as “aggressive” or “cautious”, and the rest is left to their expertise. For both scenarios, we produced 17 trajectories that satisfy ϕ_p or ϕ_o. These trajectories can be replayed using CARLA, and related signals are used in the framework to generate pairwise comparison questions.

7.1 Synthetic Experiments

In this series of experiments, we aim to evaluate the framework's performance under various conditions. For synthetic experiments, we randomly select a weight valuation as the correct valuation w^H of a human. This valuation is then utilized to generate responses to questions, mimicking how a human with the underlying answering mechanism would answer. When we work with noise-free answers, for a question pair q if $a_q^{w^H} = k$, then the answer provided by this hypothetical human is also k.

The query selection performance. In this analysis, we include the correct weight valuation in the sample set Ω_W. Thus, under the assumption of no noise in answers, we know that there is at least one weight valuation in the sample set that has 100% agreement with the hypothetical person. With confidence level $P_{thresh} = 99\%$, and no limit on the number of questions we can ask, we want to assess how often we converge to the correct weight valuation, and how many questions it takes to converge to the correct weight valuation when questions are selected randomly. We complete 100 runs, where in each run, we select a random correct valuation in Ω_W, run our framework with the information gain query selection method, and then run it with random question selection.

Table 1: Convergence analysis: our query method vs. random selection. Convergence rate (CR): the percentage of correct valuation being in the most likely valuation. Mean, std, and median indicate average, standard deviation, and median question numbers for algorithm termination. 's(i)' is for the pedestrian scenario, and 's(ii)' is for the overtake. 'Ours' refers to our framework, and 'Random' to random selection.

Method	CR		Mean		Std		Median	
	s (i)	s (ii)	s (i)	s (ii)	s (i)	s (ii)	s (i)	s (ii)
Ours	98%	100%	18.1	10.4	29.80	0.68	10	10
Random	98%	100%	37.02	26.07	33.58	12.65	25.5	24

Statistics for this experiment are presented in Table 1. In both scenarios, results highlight the effectiveness of query selection based on information gained in reducing the length of the questionnaire. We observe a few instances with a high number of questions causing a shift in the mean to greater values. However, the median stays remarkably low, at just 10 questions for both scenarios. Overall, the median number of questions required for convergence is $2-3$ times lower than random selection.

The effect of different probabilistic bounds u. In Theorem 1, we establish a lower bound for the agreement between the most likely valuation and a valuation that maximizes user agreement. When we include the correct weight valuation into the sample set Ω_W, this theorem provides a lower bound for the user agreement of the most likely valuation. According to the theorem, if $u \approxeq 0$, $1 - \log(1-u)/\log(u) = 1$ and if $u = 0.5$, $1 - \log(1-u)/\log(u) = 0$. In this set of experiments, we aim to assess how varying values of u impact the user agreement of the most likely weight valuation and its generalizability. We set the limit for the number of questions to 20, and confidence threshold $P_{thres} = 99\%$.

Figure 3 illustrates the results. Notably, the user agreement in the training data is almost always 100%. This result is expected for lower values of u (higher values of Δr^*) since likelihood function values from the Bradley-Terry model are close to 0 or 1. Consequently, for a question, when a realization of weight valuation picks an answer in disagreement, its posterior becomes negligible. After 20 questions, all valuations that share the same answers for these questions end up with almost equal posterior probabilities. Therefore, the most likely valuation is one of the many valuations that answer training set questions in the same manner as the correct valuation. However, here, the challenge is for unseen questions. The most likely valuation performs poorly in the agreement for all questions, indicating overfitting to training data. With increasing u, we observe that the overall agreement reaches 100% in both scenarios, and we converge to the correct valuation. The convergence region is depicted by the pink region in the figure. Additionally, in the range of $u \in [0.32, 0.46]$, the number of questions required is less than 12. As u continues to increase, the number of questions needed for convergence increases. Based on this analysis, we choose u in $[0.32, 0.46]$ for all experiments.

Resilience to noisy answers. In this set of experiments, we analyze the framework's resilience to inconsistent answers. We will use the Bradley-Terry model to determine noisy answers. In this experiment, we limit the number of questions to 12. We also set the confidence level to $P_{thres} = 99\%$, and $u = 0.4$. The answer mechanism works in the following way: for a question q, we pick a random number $v \in [1-u, 1]$, if $\max(P(A=0|q, w^H), P(A=1|q, w^H)) > v$, we select the answer that w^H would give. Otherwise, we randomly assign an answer to this question. In essence, this approach simulates the notion that people are more likely to stick to their answers on questions for which they have strong opinions, while otherwise, it may result in inconsistent responses. With this setup, we complete 100 runs.

Table 2: Resilience to noisy answers: "CR" is the convergence percentage to the correct valuation. "Train Agreement" and "Overall Agreement" denote agreement between the most likely and the correct valuation (when not converged to) for the training and overall question sets. "Min" and "Max" are minimum and maximum values within the respective sets.

Scenario	CR	Train Agreement		Overall Agreement	
		Min	Max	Min	Max
Pedestrian	96%	91.66%	100%	88.97%	100%
Overtake	97%	100%	100%	79.47%	100%

Table 2 shows the performance of the framework with noisy answers. In both scenarios, noisy answers have a diminishing effect on the convergence rate. However, even when convergence to the correct valuation is not achieved, the framework's agreement performance remains promising, especially in the overall question set. For both scenarios, the most likely valuation consistently attains an agreement performance of at least 91.67% even with a question limit of 12. This indicates that in a real-life study where people

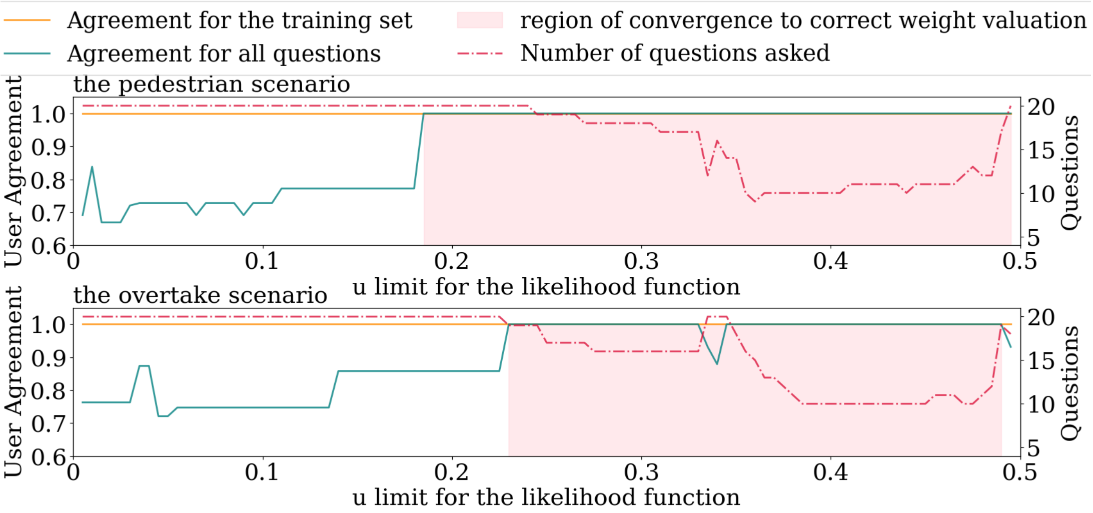

Figure 3: The effect of different u limits over the performance of the most likely valuation. The orange plot is the user agreement in the training set. Theorem 1 puts a lower bound to this value. The teal plot is the user agreement in all questions. The pink region shows the u interval that the framework converges to the correct weight valuation.

would give inconsistent answers, we expect that the most likely scenario can be generalized to unseen questions.

Performance analysis when the correct valuation is out of sample set. In the final set of synthetic experiments, we assess the performance when the correct valuation is not inside the sample set Ω_W. That means, we cannot converge to the correct valuation, but we can assess the agreement performance of the most likely valuation over the training and overall question set. In this setup, we set $u = 0.4$, question limit to 20, and confidence threshold to $P_{thres} = 99\%$. We complete 100 runs.

Table 3: Out-of-sample performance: performance analysis when the correct valuation is not in the sample set. "# Qs" denotes the average number of questions needed to terminate, "Mean" and "Std" are the average agreement and standard deviation in agreement for the given set.

Scenario	# Qs	Train Agreement		Overall Agreement	
		Mean	Std	Mean	Std
Pedestrian	13.36	95.32%	6.59%	88.20%	8.53%
Overtake	10.91	97.55%	4.97%	82.88%	6.64%

Table 3 presents statistics for out-of-sample experiment. We can see that, with small standard deviation values, we can obtain agreement close to 90% for the overall question set. Therefore, even when the correct valuation is not in the sample set, with weight valuations in the sample set, the outcome can give promising agreement values for unseen questions.

All four synthetic experiments demonstrate that in a real-life study, setting a reasonably small question number limit, with an assumption on the likelihood probability, even when the participant is giving inconsistent answers, we can output a valuation that can generalize over unseen questions. Now, we will continue with the human subject study.

7.2 Human Subject Study

The goal of the study is to assess the preference-capturing performance of the framework over participants, who can potentially give inconsistent answers. The human subject study was conducted with 11 participants. One participant's overtake data was discarded, giving 21 total completed cases. Participants have an average age of 29.77 and gender percentage is 45.4% female, 54.6% male.

We use the driving simulator described above with participants in the driver's seat of the cabin while the motion base remains deactivated. An instance from the study is shown in Figure 2. We present them with a series of questions based on their prior answers, where we show two trajectories played sequentially. Participants can replay any trajectory as many times as needed. We limit the number of questions to 12, set the probabilistic bound $u = 0.36$, and set a confidence threshold 99%, based on findings from synthetic experiments. This study was approved by IRB with protocol number 20221727.

We refer to questions we use to infer a weight valuation as *training questions*. We then pose three more *validation questions* to assess the success of the most likely valuation. The weighted robustness of the WSTL formula with the most likely weight valuation serves as a ranking function for trajectories. In the validation questions, participants compare the trajectory with the highest weighted robustness value to (i) the lowest-value trajectory, (ii) the median-value trajectory, and (iii) a randomly selected one. Depending on the number of replays, a study per scenario takes less than 20 minutes to complete.

Figure 4 shows the performance results for each participant, revealing a correlation between the agreement in training and validation questions and the confidence level of the most likely valuation. Across all scenarios, we find that four cases have a confidence level

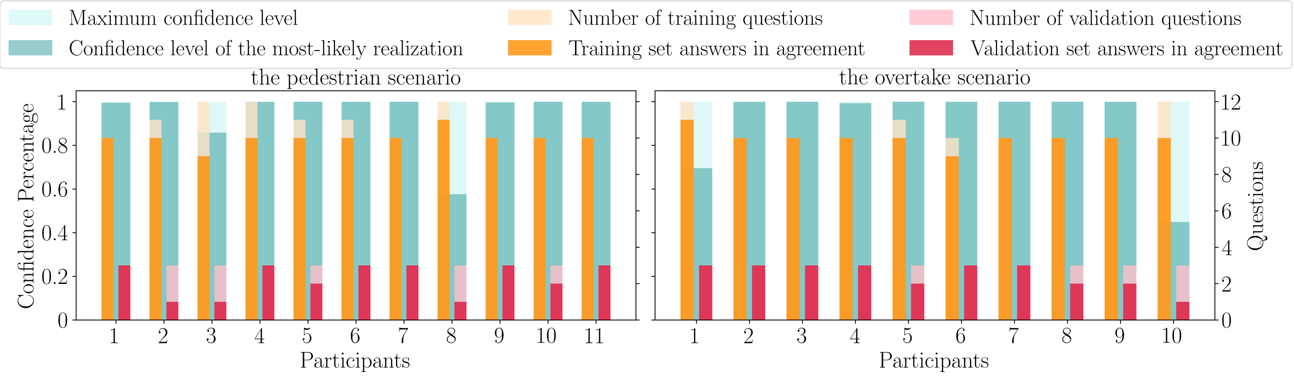

Figure 4: Human subject study results. Teal bars show the confidence level of the most likely valuation, and pale turquoise bars show the maximum level. Pale orange bars show the number of training questions and orange bars show the number of answers that the most likely valuations give in agreement with user answers. Pale pink bars represent validation questions (three for all), and pink bars show the user agreement of the most likely valuation over validation questions.

less than the threshold. Among these instances, three cases have less than 34% user agreement in the validation set, covering 75% of instances with validation agreement below 34%. On the flip side, in 57% of cases across all scenarios, the most likely valuation manages to achieve 100% user agreement in validation questions. Moreover, in 72.7% cases where we have full user agreement in training questions, we also have full agreement in validation questions. When the confidence level is above the threshold, the framework's ability to capture preferences and its generalizability performance show promise. In fact, our results exceed 80% user agreement in training and validation sets for 16 out of 21 cases. When all participants are considered, we achieve 79% agreement in the validation set. The performance on the validation set further increases up to 86% agreement when only the cases where the most likely weight valuation has confidence level above the threshold are considered. Hence, higher confidence in the most likely weight valuation can yield better generalization.

Finally, we investigate if signal pairs that are often selected as questions in the study have any common properties. For a subset of six participants, we collected their question logs: question pairs posed in order, and replay requests for trajectories in each question pair. We observe that some pairs are posed to more participants than others, whereas some pairs are never chosen. In Figure 5, we show the distance to pedestrian signals d of three different trajectories. This signal is one of the propositions in the pedestrian formula. Pair $(2, 10)$ is posed to three participants, which makes this pair one of the most repeated ones. Moreover, this pair is always posed in the first half of the study. Pair $(0, 10)$ is posed to two participants, whereas pair $(0, 2)$ is never chosen. A quick observation reveals that signals 0 and 2 follow a similar pattern, while signal 10 is fairly distinguishable from others. Moreover, when we take the Euclidean distance of these pairs separately, including acceleration and jerk signals, we saw that the distance between pair $(0, 2)$, $\|\sigma_0 - \sigma_2\| = 78.75$, whereas $\|\sigma_0 - \sigma_{10}\| = 210.75$, and $\|\sigma_2 - \sigma_{10}\| = 168.67$. As the query selection method is expected to choose informative and distinguishable signals, this observation is in line with the claims of the framework.

We also observe replay requests throughout the study. Our initial premise anticipates an inverse correlation between replay times and confidence levels as well as agreements: if a participant asks for replays many times, this may imply that they are hesitant in their choices, and thus prone to give inconsistent answers. However, our findings are inconclusive for this claim. As an example, in the pedestrian scenario, Participant 7 completes their study without any replay requests, while Participant 11 asks for replays five times. Nonetheless, both participants have high confidence levels for their most likely valuation, and their training and validation set agreements are 100% as shown in Figure 4.

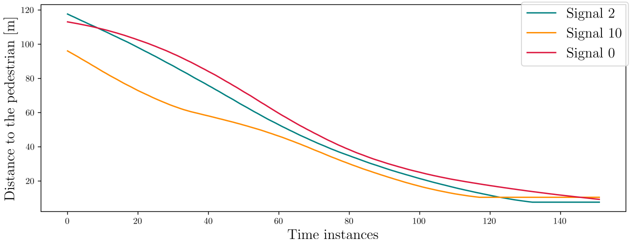

Figure 5: Some examples of distance to the pedestrian signal *d*, extracted from the simulations used in questions for the pedestrian scenario.

By learning the weights of a WSTL formula, we are essentially learning a personalized robustness metric. We finally make a few observations on the rankings induced by the learned robustness metrics. As d values in Figure 5 and Euclidean distance values show, signals 0 and 2 exhibit similar patterns. However, similarities in one signal channel do not usually mean similar preferences when they are compared to other signals, or equal preferences when they are compared to each other. For instance, Participant 3 ranks signal 0 as the second highest. However, the same participant ranks signal 2 as the eleventh. On the other hand, for Participant 6, signal 2 is the fifth signal, whereas signal 0 is the thirteenth. Recall that question pair $(0, 2)$ was never chosen. This shows us that using temporal logic can help infer more complex characteristics over signals than Euclidean distance. As another interesting data point, we see that signal 14 is ranked fourth for Participant 11. However, even adding a small Gaussian noise to signal 14 makes this signal violate the STL formula ϕ_p, thus not to be chosen over any rule-satisfying signal for safety. Recall that the soundness property of WSTL guarantees that a rule-violating signal can never have a positive robustness value. That is, the noisy signal is the last in the rank order as its weighted robustness value is negative whereas all others are positive. This shows that using logical structures benefits the preference learning framework by making it more responsive to safety violations due to noise or other deviations.

8 CONCLUSION AND FUTURE WORK

In this work, we present an active approach for customizing autonomous vehicles to align with user preferences while ensuring safety. Offline learning methods may require large training datasets, which can be impractical to gather from one person for personalization. To mitigate this challenge, we leverage active Bayesian inference and incorporate Weighted Signal Temporal Logic (WSTL), which yields a WSTL formula that can be used in correct-and-custom-by-construction control synthesis. Its adaptability to formulas and signals of varying complexity and length, enabled by the offline computation of STL-related values, makes this method practical in complex and real-life situations. We provide a theoretical bound for the agreement level of the most likely valuation. While our work focuses on the autonomous driving application, the methodology is general and can be readily applied in other safety-critical cyber-physical-human systems that can benefit from personalization.

In both sets of experiments, our findings highlight the success of the query selection algorithm over random query selection. Notably, in synthetic experiments, our algorithm not only converges to an optimal weight valuation within our search set with a reduced questionnaire length but also exhibits promising performance in capturing user preferences during training and in generalizability.

As a future work, an implementation of this framework into controller-synthesis algorithms like the one proposed in [35] would show us potential challenges in connecting two problems. An open direction leveraging the complete correct-and-custom-by-construction controller synthesis pipeline is to close the loop of the personalization framework by automating and including the trajectory generation step into it. This requires almost real-time controller synthesis using WSTL constraints. Another problem to consider is reasoning about preferences over a wider set of scenarios based on learning results over a smaller but significant scenario set. Driving scenarios often consist of common patterns that may be leveraged for generalization to a common driving behavior scheme.

ACKNOWLEDGMENTS

Toyota Research Institute (TRI) provided funds for this work. Part of this work was completed when the first author was an intern at TRI.

REFERENCES

[1] C. Basu, Q. Yang, D. Hungerman, M. Sinahal, and A. D. Draqan, "Do you want your autonomous car to drive like you?" in *2017 12th ACM/IEEE Intl. Conf. on Human-Robot Interaction (HRI*, 2017, pp. 417–425.

[2] M. Hasenjäger and H. Wersing, "Personalization in advanced driver assistance systems and autonomous vehicles: A review," in *2017 IEEE 20th Intl. Conf. on Intelligent Transportation Systems (ITSC)*, 2017, pp. 1–7.

[3] R. Karagulle, N. Aréchiga, A. Best, J. DeCastro, and N. Ozay, "A preference learning approach to develop safe and personalizable autonomous vehicles," 2023.

[4] B. Helou, A. Dusi, A. Collin, N. Mehdipour, Z. Chen, C. Lizarazo, C. Belta, T. Wongpiromsarn, R. D. Tebbens, and O. Beijbom, "The reasonable crowd: Towards evidence-based and interpretable models of driving behavior," in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS).* IEEE, 2021, pp. 6708–6715.

[5] J. Karlsson, S. van Waveren, C. Pek, I. Torre, I. Leite, and J. Tumova, "Encoding human driving styles in motion planning for autonomous vehicles," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, 2021, pp. 1050–1056.

[6] E. Biyik, M. Palan, N. C. Landolfi, D. P. Losey, and D. Sadigh, "Asking Easy Questions: A User-Friendly Approach to Active Reward Learning," in *Proceedings of the Conference on Robot Learning.* PMLR, May 2020, pp. 1177–1190, iSSN: 2640-3498.

[7] J. Fürnkranz and E. Hüllermeier, *Preference learning.* Springer Berlin Heidelberg, 2011.

[8] B. Settles, *Active Learning*, ser. Synthesis Lectures on Artificial Intelligence and Machine Learning. Springer International Publishing, 2012.

[9] R. Holladay, S. Javdani, A. Dragan, and S. Srinivasa, "Active Comparison Based Learning Incorporating User Uncertainty and Noise," in *RSS Workshop on Model Learning for Human-Robot Communication*, Jun. 2016.

[10] L. Maystre and M. Grossglauser, "Just Sort It! A Simple and Effective Approach to Active Preference Learning," in *Proceedings of the 34th International Conference on Machine Learning.* PMLR, Jul. 2017, pp. 2344–2353, iSSN: 2640-3498.

[11] B. Eric, N. Freitas, and A. Ghosh, "Active Preference Learning with Discrete Choice Data," in *Advances in Neural Information Processing Systems*, vol. 20. Curran Associates, Inc., 2007.

[12] D. Sadigh, A. Dragan, S. Sastry, and S. Seshia, "Active Preference-Based Learning of Reward Functions," in *Robotics: Science and Systems XIII.* Robotics: Science and Systems Foundation, Jul. 2017.

[13] N. Wilde, D. Kulić, and S. L. Smith, "Active Preference Learning using Maximum Regret," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Oct. 2020, pp. 10 952–10 959, iSSN: 2153-0866.

[14] R. Cosner, M. Tucker, A. Taylor, K. Li, T. Molnar, W. Ubelacker, A. Alan, G. Orosz, Y. Yue, and A. Ames, "Safety-aware preference-based learning for safety-critical control," in *Proc. of The 4th Annual Learning for Dynamics and Control Conf.*, vol. 168. PMLR, 2022, pp. 1020–1033.

[15] P. Nilsson, O. Hussien, A. Balkan, Y. Chen, A. D. Ames, J. W. Grizzle, N. Ozay, H. Peng, and P. Tabuada, "Correct-by-construction adaptive cruise control: Two approaches," *IEEE Trans. on Control Systems Technology*, vol. 24, no. 4, pp. 1294–1307, 2016.

[16] L. Lindemann and D. V. Dimarogonas, "Control barrier functions for signal temporal logic tasks," *IEEE Control Systems Letters*, vol. 3, no. 1, pp. 96–101, 2019.

[17] G. E. Fainekos, A. Girard, H. Kress-Gazit, and G. J. Pappas, "Temporal logic motion planning for dynamic robots," *Automatica*, vol. 45, no. 2, pp. 343–352, 2009.

[18] A. Linard, I. Torre, B. Ermanno, A. Sleat, I. Leite, and J. Tumova, "Real-time rrt* with signal temporal logic preferences," in *Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2023.

[19] R. Karagulle, N. Aréchiga, J. DeCastro, and N. Ozay, "Classification of driving behaviors using stl formulas: A comparative study," in *Formal Modeling and Analysis of Timed Systems.* Springer Intl. Publishing, 2022, p. 153–162.

[20] D. Neider and I. Gavran, "Learning linear temporal properties," in *2018 Formal Methods in Computer Aided Design (FMCAD)*, 2018, pp. 1–10.

[21] G. Venkatesh, "Temporal Logic with Preferences and Reasoning About Games," in *Proof, Computation and Agency: Logic at the Crossroads*, ser. Synthese Library, J. van Benthem, A. Gupta, and R. Parikh, Eds. Springer Netherlands, 2011, pp. 241–258.

[22] N. Mehdipour, C.-I. Vasile, and C. Belta, "Specifying user preferences using weighted signal temporal logic," *IEEE Control Systems Letters*, vol. 5, no. 6, pp. 2006–2011, 2021.

[23] R. Yan, A. Julius, M. Chang, A. Fokoue, T. Ma, and R. Uceda-Sosa, "Stone: Signal temporal logic neural network for time series classification," in *2021 Intl. Conf. on Data Mining Workshops (ICDMW)*, 2021, pp. 778–787.

[24] N. Fronda and H. Abbas, "Differentiable inference of temporal logic formulas," *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, vol. 41, no. 11, pp. 4193–4204, 2022.

[25] O. Maler and D. Nickovic, "Monitoring temporal properties of continuous signals," in *Formal Techniques, Modelling and Analysis of Timed and Fault-Tolerant Systems*, Y. Lakhnech and S. Yovine, Eds. Springer Berlin Heidelberg, 2004, pp. 152–166.

[26] A. Donzé and O. Maler, "Robust satisfaction of temporal logic over real-valued signals," in *Formal Modeling and Analysis of Timed Systems.* Springer Berlin Heidelberg, 2010, pp. 92–106.

[27] G. De Giacomo and M. Y. Vardi, "Linear temporal logic and linear dynamic logic on finite traces," in *Proc. of the Twenty-Third Intl. Joint Conf. on Artificial Intelligence*, 2013, p. 854–860.

[28] X. Li, G. Rosman, I. Gilitschenski, C.-I. Vasile, J. A. DeCastro, S. Karaman, and D. Rus, "Vehicle trajectory prediction using generative adversarial network with temporal logic syntax tree features," *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 3459–3466, 2021.

[29] D. Golovin, A. Krause, and D. Ray, "Near-optimal bayesian active learning with noisy observations," *Advances in Neural Information Processing Systems*, vol. 23, 2010.

[30] M. Naghshvar, T. Javidi, and K. Chaudhuri, "Noisy bayesian active learning," in *2012 50th Annual Allerton Conference on Communication, Control, and Computing (Allerton).* IEEE, 2012, pp. 1626–1633.

[31] R. A. Bradley and M. E. Terry, "Rank analysis of incomplete block designs: I. the method of paired comparisons," *Biometrika*, vol. 39, no. 3/4, pp. 324–345, 1952.

[32] Y. Chen, S. H. Hassani, A. Karbasi, and A. Krause, "Sequential information maximization: When is greedy near-optimal?" in *Conference on Learning Theory.* PMLR, 2015, pp. 338–363.

[33] E. Bıyık, "Learning preferences for interactive autonomy," Ph.D. dissertation, Stanford University, 2022.

[34] A. Dosovitskiy, G. Ros, F. Codevilla, A. Lopez, and V. Koltun, "CARLA: An open urban driving simulator," in *Proceedings of the 1st Annual Conference on Robot Learning*, 2017, pp. 1–16.

[35] G. A. Cardona, D. Kamale, and C.-I. Vasile, "Mixed integer linear programming approach for control synthesis with weighted signal temporal logic," in *Proceedings of the 26th ACM International Conference on Hybrid Systems: Computation and Control*, 2023, pp. 1–12.