The Transfer of Human Trust in Robot Capabilities across Tasks

Harold Soh, Pan Shu, Min Chen, and David Hsu
Department of Computer Science, National University of Singapore

***Abstract*—Trust is crucial in shaping human interactions with one another and with robots. This work investigates how human trust in robot capabilities transfers across tasks. We present a human-subjects study of two distinct task domains: a Fetch robot performing household tasks and a virtual reality simulation of an autonomous vehicle performing driving and parking maneuvers. Our findings lead to a functional view of trust and two novel predictive models—a recurrent neural network architecture and a Bayesian Gaussian process—that capture trust evolution and transfer via latent task representations. Experiments show that the two proposed models outperform existing approaches when predicting trust across unseen tasks and participants. These results indicate that (i) a task-dependent functional trust model captures human trust in robot capabilities more accurately, and (ii) trust transfer across tasks can be inferred to a good degree. The latter enables trust-based robot decision-making for fluent human-robot interaction. In particular, our models can be used to derive robot policies that mitigate under-trust or over-trust by human teammates in collaborative multi-task settings.**

I. Introduction

As robots are increasingly deployed to our homes and workplaces, interactions between humans and robots are expected to become commonplace. Human trust is critical in shaping these interactions; for example, trust directly affects the degree of autonomy rendered to robots [28]. In recognition of its importance, there has been significant work into the conceptualization and measurement of human trust in robots and automation [19, 15, 3, 38].

Nevertheless, a key gap remains in understanding and predicting when people *transfer* trust in robots across tasks based on limited observations, or knowledge of robot capabilities. Understanding trust in the multi-task context is important as robots continue to transition from being single-purpose devices to multi-functional entities.

In this paper, we take a first step towards bridging this gap. We adopt the definition of trust as a psychological attitude [3] and focus on *trust in robot capabilities*, i.e., the belief in a robot's competence to complete a task. Capability is a primal factor in determining overall trust in robots and thus, the decision to rely on them [19].

We first present results from an human-subjects study ($n = 32$) in two separate domains: household tasks and autonomous driving. Using two different domains allowed us to validate the robustness of our findings to different contexts. We show that inter-task trust transfer depends on perceived task similarity, difficulty, and observed robot performance. These results are consistent across both domains, even though the robot and context in each were markedly different; in the household domain, we used a Fetch robot to navigate and pick and place different objects, whilst the driving domain involved a virtual reality (VR) simulation of an autonomous vehicle performing driving and parking maneuvers. To our knowledge, this is the first work showing concrete evidence for trust transfer across tasks in the context of human-robot interaction. We have made our data freely available online [1] to further trust research.

Based on our findings, we propose to formally conceptualize trust as a context-dependent *latent dynamic function*. This viewpoint is supported by prior socio-cognitive research showing trust depends on task properties and the agent to be trusted [3]. In this work, we focus on characterizing the *structure* of this "trust function" and its *dynamics*, i.e., how it changes with observations of robot performance across tasks. We develop two models: (i) a connectionist (recurrent neural) model based on recent advances in deep learning, and (ii) a Bayesian Gaussian process (GP) [26] model. The former is a purely data-driven approach which places light constraints on how trust evolves with observations. In comparison, the GP model explicitly encodes a specific assumption about how human trust evolves, i.e., via Bayes rule. Both models leverage latent task space representations learned using word vector descriptions of tasks, e.g., "Pick and place a glass cup". Experiments show both models accurately predict trust across unseen participants and unseen tasks.

Unlike prevailing approaches (e.g., [16, 36]), a key benefit of these trust models is that they are able to leverage inter-task structure and can be applied in situations where the agent (our robot) performs actions across many different tasks. As *predictive* models, they can be operationalized in decision-theoretic frameworks to calibrate trust during collaboration with human teammates [5, 35, 22]. Trust calibration is crucial for preventing over-trust that engenders unwarranted reliance in robots [27, 29], and under-trust that can cause poor utilization [17]. To summarize, this paper makes the following key contributions:

- A novel formalization of trust as a latent dynamic function, and efficient computational models that capture and predict human trust in robot capabilities across tasks;
- Empirical findings from a human subjects study showing the influence of three factors on human trust transfer across tasks, i.e., perceived task similarity, difficulty, and robot performance;
- A systematic evaluation showing the proposed methods outperform current methods, indicating the importance of modeling trust formation and transfer across tasks.

II. Background and Related Work

Research into trust in robots (and automation) is a large interdisciplinary endeavor spanning multiple fields including human-factors, psychology, and human-robot interaction. It is not possible to cover the breadth of this research here, so we focus on key concepts and computational models of trust.

Key Concepts and Definitions. Trust is a multidimensional concept, with many factors characterizing human trust in robots, e.g., the human's technical expertise and the complexity of the robot [15, 19, 11]. Of these factors, two of the most prominent are the performance and integrity of the machine. Similar to existing work in robotics [37, 36], we assume that robots are not intentionally deceptive and focus on characterizing trust based on robot performance. We view trust as a belief in the competence and reliability of another agent to complete a task.

Trust Measurement. Trust is a latent dynamic entity, which presents challenges for measurement [2]. Previous work has derived survey instruments and methods for quantifying trust, including time-varying and "area-under-the-curve" measures [8, 38]. In this paper, we use a self-reported measure of trust (similar to [36]) and Muir's questionnaire [19].

Computational Models of Trust. Previous work has explored explanatory models (e.g., [3, 17]) and predictive models of trust. Recent models have focused on dynamic modeling, for example, a recent predictive model—OPTIMo [36]—is a Dynamic Bayesian Network with linear Gaussian trust updates trained on data gathered from an observational study. OPTIMo was shown to outperform an Auto-Regressive and Moving Average Value (ARMAV) model [16], and stepwise regression [37]. Because trust is treated as "global" real-valued scalar in these models, they are appropriate when tasks are fixed (or have little variation). However, as our results will show, trust can differ substantially between tasks. As such, we develop models that capture both the dynamic property of trust and its variation across tasks. We leverage upon recurrent neural networks that have been applied to a variety of sequential learning tasks (e.g., [33]) and online Gaussian processes that have been previously used in robotics [32, 30, 31].

Application of Trust Models. Trust emerges naturally in collaborative settings. In human-robot collabation [23, 22], trust models can be used to enable more natural interactions. For example, Min *et al.* [5] proposed a decision-theoretic model that incorporates a predictive trust model, and showed that policies that took human trust into consideration led to better outcomes. The models presented in this work can be integrated into such frameworks to influence robot decision-making across different tasks.

III. Human Subjects Study

In this section, we describe our human subjects study, which was designed to evaluate if and when human trust transfers between tasks. Our general intuition was that human trust generalizes and evolves in a structured manner. We specifically hypothesized that:

- **H1:** Trust in the robot is more similar for tasks of the same category, compared to tasks in a different category.
- **H2:** Observations of robot performance have a greater affect on the *change in human trust* over similar tasks, i.e., in the same category, compared to dissimilar tasks.
- **H3:** Trust in a robot's ability to perform a task transfers more readily to easier tasks, compared to more difficult tasks.

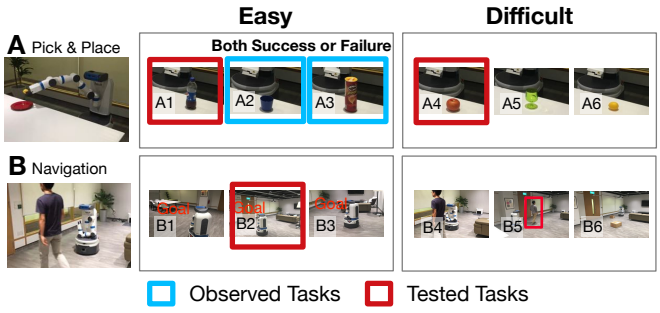
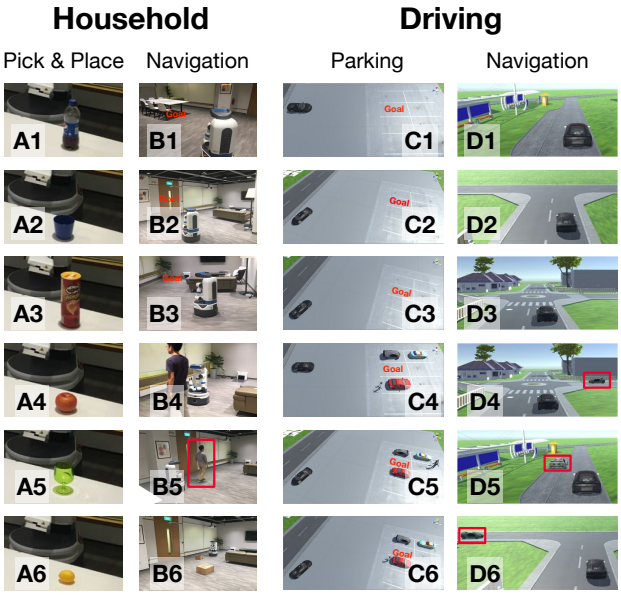


Fig. 1: Trust Transfer Experiment Design. Two categories of tasks were used: (A) picking and placing different objects, and (B) navigation in a room, potentially with people and obstacles. Participants were surveyed on their trust in the robot's ability to successfully perform three different tasks (red boxes) *before* and *after* being shown demonstrations of two tasks. The two observed tasks were always selected from the same cell (blue boxes; cell randomly assigned, with either both successes or both failures). The tested tasks were randomly selected from three different cells—the (i) same category and difficulty level, (ii) same category but different difficulty level, and (iii) different category but same difficulty level—compared to the observed tasks.

A. Experimental Design

An overview of our experimental design is shown in Fig. 1. We explored three factors as independent variables: task category, task difficulty, and robot performance. Each independent variable consisted of two levels: two task categories, easy/difficult tasks, and robot success/failure. We used tasks in two domains, each with an appropriate robot (Fig. 2):

- **Household**, which included two common categories of household tasks, i.e., picking and placing objects, and navigation in an indoor environment. The robot used was a real-world Fetch research robot with a 7-DOF arm, which performed *live* demonstrations of the tasks in a lab environment that resembles a studio apartment.
- **Driving**, where we used a Virtual Reality (VR) environment to simulate an autonomous vehicle (AV) performing tasks such as lane merging and parking, potentially with dynamic and static obstacles. Participants interacted with the simulation system via an Oculus Rift headset, which provided a first-person viewpoint from the driver seat.



Domain	Category	ID	Task
Household	A. Pick & Place	1	a bottle of soda
		2	a plastic cup
		3	a can of chips
		4	an apple
		5	a glass cup
		6	a lemon
	B. Indoor Navigation	1	to the table
		2	to the door
		3	to living room
		4	with people moving around
		5	following a person
		6	while avoiding obstacles
Driving	C. Parking	1	Forwards, empty lot (aligned)
		2	Backwards, empty lot (not aligned)
		3	Forwards, empty lot (not aligned)
		4	Backwards, with cars (aligned)
		5	Backwards, with cars (not aligned)
		6	Forwards, with cars (not aligned)
	D. Navigation	1	Lane merge
		2	T-junction
		3	Roundabout
		4	Roundabout with other cars
		5	Lane merge with other cars
		6	T-junction with other cars

Fig. 2: Tasks in the Household and Driving Domains.

The robots were different in both settings and there were no cross-over tasks; in other words, the same experiment was conducted independently in each domain with the same protocol. Obtaining data from two separate experiments enabled us to discern if our hypotheses held in different contexts.

In both domains, we developed pre-programmed success and failure demonstrations of robot performance. "Catastrophic" failures were avoided to mitigate complete distrust of the robot; for the household navigation tasks, the robot was programmed to fail by moving to the wrong location. For picking and placing, the robot failed to grasp the target object. The autonomous car failed to park by stopping too far ahead of the lot, and failed to navigate (e.g., lane merge) by driving off the road and stopping. More information about the failure conditions is available in the online supplementary material [1].

The primary dependent variables were the participants' subjective trust in the robot a's capability to perform specific tasks. Participants indicated their degree of trust given task x at time t, denoted as $\tau^a_{x,t}$, via a 7-point Likert scale in response to the agreement question: "*The robot is going to perform the task* $[x]$. *I trust that the robot can perform the task successfully*". From these task-dependent trust scores, we computed two derivative scores:

- **Trust Distance** $d_{\tau,t}(x, x') = |\tau^a_{x,t} - \tau^a_{x',t}|$, i.e., the 1-norm distance between scores for x and x' at time t.
- **Trust Change** $\Delta\tau^a_x(t_1, t_2) = |\tau^a_{x,t_1} - \tau^a_{x,t_2}|$, i.e., the 1-norm distance between the scores for x at t_1 and t_2.

As a general measure of trust, participants were also asked to complete Muir's questionnaire [19, 20] pre-and-post exposure to the robot demonstrations.

B. Robot Systems Setup

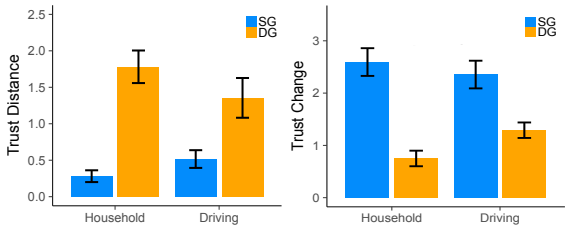
For both the Fetch Robot and Autonomous Driving simulator, we developed our experimental platforms using the Robot Operating System (ROS). On the Fetch robot, we used the MoveIt motion planning framework and the Open Motion Planning Library [34] to pick and place objects, and the ROS Navigation stack for navigation in indoor environments.

The VR simulation platform was developed using the Unity 3D engine. Control of the autonomous vehicle was achieved using the hybrid A* search algorithm [9] and a proportional-integral-derivative (PID) controller. More information about the setup is available in the online supplementary material [1].

C. Study Procedure

We recruited 32 individuals (Mean age: 24.09 years, $SD = 2.37$, 46% female) through an online advertisement and printed flyers on a university campus. After signing a consent form and providing standard demographic data, participants were introduced to the robot and continued with the experiment's four stages:

1) **Category and Difficulty Grouping:** To gain better control of the factors, participants were asked to group the 12 tasks evenly into the four cells shown in Fig. 1. As such, chosen observations matched a participant's own prior estimations.
2) **Pre-Observation Questionnaire:** Participants were asked to indicate their subjective trust on the three tested tasks using the measure instruments described above.
3) **Observation of Robot Performance:** Participants were randomly assigned to observe two tasks from a specific category and difficulty, and were asked to indicate their trust if the robot were to repeat the observed task.
4) **Post-Observation Questionnaire and Debrief:** Finally, participants were asked to re-indicate their subjective trust on the three tested tasks, answered consistency check questions, and underwent a short de-briefing.



(a) (b)

Fig. 3: (a) Trust distance between a given task and tasks in the same category group (SG) compared to tasks in a different category (DG). Trust in robot capabilities was significantly more similar for tasks in the same group. (b) Trust change due to observations of robot performance. Trust increased (or decreased) significantly more for SG versus DG.

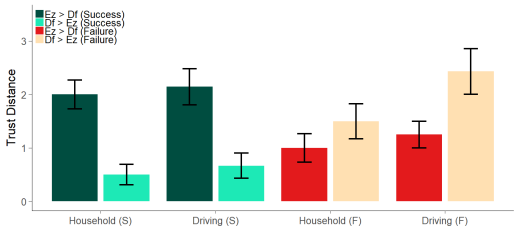


Fig. 4: Trust distance between the observed task and a more difficult task (Ez $\rightarrow$ Df) against when generalizing to a simpler task (Df $\rightarrow$ Ez). Participants who observed successful demonstrations of a difficult task trusted the robot to perform simpler tasks, but not vice-versa.

D. Results

Our first set of results are summarized in Fig. 3a. For the driving domain, one participant's results were removed due to a failure to pass attention/consistency check questions [1]. The figure clearly shows that tasks in the same category (SG) shared similar scores (supporting **H1**); the trust distances are significantly lower $(M = 0.28, SE = 0.081)$ compared to tasks in other categories (DG) $(M = 1.78, SE = 0.22)$, $t(31) = -5.82$, $p < 10^{-5}$ for the household tasks. Similar statistically significant differences are observed for the driving domain, $t(30) = -2.755$, $p < 10^{-2}$. This result implies that trust is not a single "scalar-valued" quantity, but varies across tasks.

Fig. 3b shows that the *change* in human trust due to performance observations of a given task was moderated by the perceived similarity of the tasks (**H2**). The trust change is significantly greater for SG than DG; $t(31) = 6.25$, $p < 10^{-6}$ for household and $t(30) = 3.46$, $p < 10^{-2}$ for driving. Note also that the trust change for DG was non-zero, $t(31) = 8.94, p < 10^{-6}$ for success and $t(31) = -8.35, p < 10^{-6}$ for failures respectively, indicating that trust transfers even between task categories, albeit to a lesser extent.

We analyzed the relationship between perceived difficulty and trust transfer (**H3**) by first splitting the data into two conditions: participants who received successful demonstrations, and those that observed failures (Fig. 4). For the success condition, the trust distance among the household tasks was significantly less for tasks perceived to be easier than the observed task $(M = 2.0, SE = 0.27)$, compared to tasks that were perceived to be more difficult $(M = 0.5, SE = 0.27)$, $t(14) = 4.58$, $p < 10^{-3}$. The hypothesis also holds in the driving domain, $M = 1.25$ $(SE = 0.25)$ v.s. $M = 2.43$ $(SE = 0.42)$, $t(14) = 3.6827$, $p < 10^{-3}$. For the failure condition, the results were not statistically significant at the $\alpha = 1\%$ level, but suggest that the effect was reversed; belief in robot *inability* would transfer more to difficult tasks compared to simpler tasks.

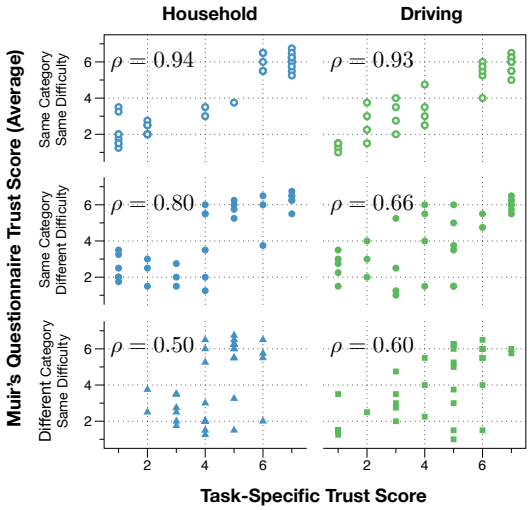


Fig. 5: Muir's Trust Score [19] vs Task-specific trust scores (for tested tasks). The scores are positively correlated across the three task types, but with different strengths; the general measure is less predictive of task-specific trust for tasks in different categories (Pearson correlation, $\rho = 0.5-0.6$) compared to tasks with same category and difficulty ($\rho = 0.93 - 0.94$).

Finally, in this study, we measured task-specific trust; a key question is how this trust differs from a "general" notion of trust in the robot as measured by Muir's questionnaire. Fig. 5 sheds light on this question; overall, task-specific and general trust are positively correlated but the degree of correlation depends greatly on the similarity of the task to previous observations. In other words, while general trust is predictive of task-specific trust, it does not capture the range or variability of human trust across multiple tasks.

E. Summary of Findings

Our main findings support the intuition that human trust transfers across tasks, but to different degrees. More specifically, similar tasks are more likely to share a similar level of trust (**H1**). Observations of robot performance changes trust both in the observed task, and also in similar yet unseen tasks (**H2**). Finally, trust transfer is asymmetric: positive trust transfers more easily to simpler tasks than to more difficult tasks (**H3**). These findings suggest that to infer human trust accurately in a collaboration across multiple tasks, robots should consider the similarity and difficulty of previous tasks.

IV. Computational Models for Trust Across Tasks

The results from our human subjects study indicate that it is inappropriate to model human trust in a robot a as a latent real-valued scalar $\tau_t^a \in \mathbb{R}$. In this work, we propose a richer model where trust is a latent *dynamic function* $\tau_t^a(\mathbf{x}) : \mathbb{R}^d \to [0, 1]$ that maps task features, $\mathbf{x}$, to real-values indicating trustworthiness of the robot to perform the task. This functional view of trust enables us to naturally capture trust differences across tasks, and can be extended to include other contexts, e.g., environmental or situational factors.

To model the dynamic nature of trust, we propose a Markovian function g that updates trust,

$$\tau_t^a = g(\tau_{t-1}^a, o_{t-1}^a) \tag{1}$$

where $o_{t-1}^a = (\mathbf{x}_{t-1}, c_{t-1}^a)$ is the observation of robot a performing a task with features $\mathbf{x}_{t-1}$ at time $t-1$ with performance outcome c_{t-1}^a. Here, we consider binary outcomes $c_{t-1}^a \in \{+1, -1\}$ indicating success and failure respectively, but differences in performance can be directly accommodated via "soft labels" $c_{t-1}^a \in [-1, +1]$ without significant modifications to the presented methods or alternatively, by using regression-based losses.

The principle challenge is then to determine appropriate forms for τ_t^a and g. In this work, we propose and evaluate two different approaches: (i) a connectionist approach utilizing a recurrent neural network (RNN), and (ii) a Bayesian approach where we model a probability distribution over latent functions via a Gaussian process. We describe both models in the following subsections.

A. Neural Trust Model

In our neural model, trust is modeled as a simple sigmoid function over latent task representations,

$$\tau_t^a(\mathbf{x}; \boldsymbol{\theta}_t) = \mathrm{sigm}(\boldsymbol{\theta}_t^\top f_z(\mathbf{x})) = \mathrm{sigm}(\boldsymbol{\theta}_t^\top \mathbf{z}), \tag{2}$$

and is fully parameterized by $\boldsymbol{\theta}_t$ given $\mathbf{z}$. This linear form has benefits: it is efficient to compute given $\mathbf{z}$ and is interpretable in that the latent task space $Z \subseteq \mathbb{R}^k$ can be examined, similar to other dot-product spaces, e.g., word embeddings [18]. From one perspective, Z can be seen as a "psychological task space" in which the similarities between tasks can be easily ascertained.

Task Projection. To give flexibility to the model, we project $\mathbf{x}$ into Z via a nonlinear function, specifically, a fully-connected neural network,

$$\mathbf{z} = f_z(\mathbf{x}) = \mathrm{NN}(\mathbf{x}; \theta_z) \tag{3}$$

where θ_z is the set of network parameters. Similarly, the robot's performance c^a is projected via a neural network, $\mathbf{c}^a = \mathrm{NN}(c^a; \theta_c^a)$. During trust updates, both the task and performance representations are concatenated, $\hat{\mathbf{z}}_i = [\mathbf{z}; \mathbf{c}^a]$, as input to the RNN's memory cells.

Trust Updating via Memory Cells. We model the trust update function g using a RNN with parameters θ_g,

$$\boldsymbol{\theta}_t = \mathrm{RNN}(\boldsymbol{\theta}_{t-1}, \hat{\mathbf{z}}_{t-1}; \theta_g). \tag{4}$$

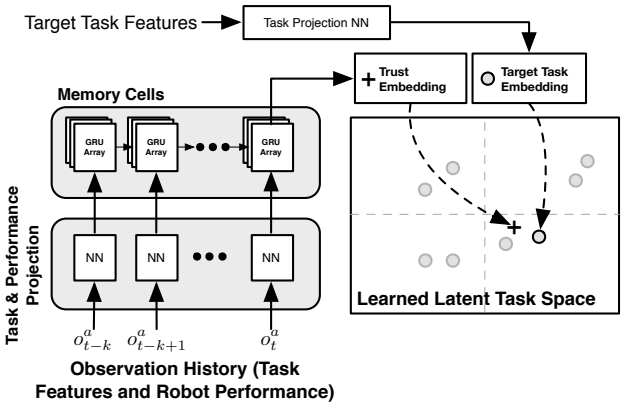


Fig. 6: A High-level Schematic of the Neural Trust Model. The trust vector is updated using GRU cells as memory of previously observed robot performance. The model uses feed-forward neural networks to project tasks into a dot-product space in which trust for a task can be efficiently computed.

In this work, we leverage on the Gated Recurrent Unit (GRU) [6], which is a variant of long short-term memory [12] with strong empirical performance [13]. In brief, the GRU learns to control two internal "gates"—the update and reset gates—that affect what it remembers and forgets. Intuitively, the previous hidden state is forgotten when the reset gate's value nears zero. As such, cells that have active reset gates have learnt to model short-term dependencies. In contrast, cells that have active update gates model long-term dependencies [6]. Our model uses an array of GRU cells that embed the interaction history up to time t as a "memory state" $\mathbf{h}_t$, which serves as our trust parameters $\boldsymbol{\theta}_t$.

More formally, a GRU cell k that has state $h_{t-1}^{(k)}$ and receives a new input $\hat{\mathbf{z}}_t$, is updated via

$$h_t^{(k)} = (1 - v_t^{(k)}) h_{t-1}^{(k)} + v_t^{(k)} \tilde{h}_t^{(k)}, \tag{5}$$

i.e., an interpolation of its previous state and a candidate activation $\tilde{h}_t^{(k)}$. This interpolation is affected by the update gate $v_t^{(k)}$, which is parameterized by matrices $\mathbf{W}_v$ and $\mathbf{U}_v$,

$$v_t^{(k)} = \mathrm{sigm}\big([\mathbf{W}_v \hat{\mathbf{z}}_t + \mathbf{U}_v \mathbf{h}_{t-1})]_k\big). \tag{6}$$

The candidate activation $\tilde{h}_t^{(k)}$ is given by

$$\tilde{h}_t^{(k)} = \tanh([\mathbf{W}\hat{\mathbf{z}}_t + \mathbf{U}(\mathbf{r}_t \odot \mathbf{h}_{t-1})]_k) \tag{7}$$

where $\odot$ denotes element-wise multiplication. The reset gate $r_t^{(j)} = [\mathbf{r}_t]_k$ is parameterized by two matrices $\mathbf{W}_r$ and $\mathbf{U}_r$,

$$r_t^{(k)} = \mathrm{sigm}\big([\mathbf{W}_r \hat{\mathbf{z}}_t + \mathbf{U}_r \mathbf{h}_{t-1})]_k\big) \tag{8}$$

Model Summary. The final architecture of our neural trust model is illustrated in Fig. 6. In summary, the model "compresses" the prior interaction history into a trust vector $\boldsymbol{\theta}_t$, which forms a hyperplane in a latent task space. Given latent task representations $\mathbf{z}$, trust predictions can be efficiently performed simply by taking sigmoids of dot-products in Z.

B. Bayesian Gaussian Process Trust Model

As an alternative to the purely data-driven approach embodied by the neural model, one can view trust formation as a cognitive process, specifically, human function learning [10]. We adopt a rational Bayesian framework, i.e., the human is learning about the capabilities of the robot f^a by combining prior beliefs about the robot with evidence (observations of performance) via Bayes rule,

$$p_t(f^a|o^a_{t-1}) = \frac{P(c^a_{t-1}|f^a, \mathbf{x}_{t-1})p_{t-1}(f^a)}{\int P(c^a_{t-1}|f^a, \mathbf{x}_{t-1})p_{t-1}(f^a)df^a}, \quad (9)$$

where p_t is the posterior distribution, $P(c^a_{t-1}|f^a, \mathbf{x}_{t-1})$ is the likelihood of observing the robot performance c^a_{t-1} given the task $\mathbf{x}_{t-1}$ and latent function f^a. Given that the robot is assumed not to be intentionally deceptive, the human estimates robot trustworthiness by integrating over the posterior:

$$\tau^a_t(\mathbf{x}_i) = \int P(c^a_i|f^a, \mathbf{x})p_t(f^a)df^a \quad (10)$$

Similar to prior work in human function learning [10], we place a Gaussian process (GP) prior over f^a,

$$p_0(f^a) = \mathcal{N}(m(\mathbf{x}), k(\mathbf{x}, \mathbf{x}')). \quad (11)$$

where $m(\cdot)$ is the prior mean function, and $k(\cdot, \cdot)$ is the kernel or covariance function. Note that the GP is completely parameterized by its mean and kernel functions.

Covariance Function. The kernel function is an essential ingredient for GPs and quantifies the similarities between inputs (tasks). Although task features are generally high dimensional (e.g., the word features used in our experiments), it is reasonable to expect tasks to live on a low-dimensional manifold, i.e., a psychological task space. With this in mind, we use a projection kernel:

$$k(\mathbf{x}, \mathbf{x}') = \exp(-(\mathbf{x} - \mathbf{x}')^\top \mathbf{M}(\mathbf{x} - \mathbf{x}')) \quad (12)$$

with a low rank matrix $\mathbf{M} = \mathbf{\Lambda}\mathbf{L}\mathbf{\Lambda}^\top$ where $\mathbf{\Lambda} \in \mathbb{R}^{d\times k}$ and $\mathbf{L}$ is a diagonal matrix of length-scales capturing axis-aligned relevances in the projected task space.

Capturing Prior Estimations of Task Difficulty and Initial Bias. As our studies have shown, perceived difficulty results in an asymmetric transfer of trust (**H3**), which presents a difficulty for standard zero/constant-mean GPs given symmetric covariance functions. To address this issue, we explore two different approaches:

1) First, the mean function is a convenient way of incorporating a human's prior estimation of task difficulty; tasks which are presumed to be difficult (beyond the robot's capability) will have low values. Here, we have used a data-dependent linear function, $m(\mathbf{x}) = \boldsymbol{\beta}^\top \mathbf{x}$ where $\boldsymbol{\beta}$ is learned along with other GP parameters.
2) A second approach is to use pseudo-observations $\mathbf{x}^+$ and $\mathbf{x}^-$ and associated f^a's to bias the initial model. Intuitively, $\mathbf{x}^+$ ($\mathbf{x}^-$) summarizes the initial positive (negative) experiences that the human may have had. Similar to $\boldsymbol{\beta}$, these parameters are learned during training.

Both approaches allow the GP to accommodate the aforementioned asymmetry; the evidence has to counteract the prior mean function or pseudo-observations respectively.

Observation Likelihood. Given binary success outcomes, an appropriate likelihood is the probit [21],

$$P(c^a_t|f^a, \mathbf{x}_t) = \Phi\left(\frac{c^a_t(f^a(\mathbf{x}_t) - m(\mathbf{x}_t))}{\sigma^2_n}\right) \quad (13)$$

where $\Phi(x) = \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{x} \exp\left(-\frac{t^2}{2}\right)dt$ is the CDF of the standard normal, and σ^2_n is the noise variance.

Trust Updates via Approximate Bayesian Inference Unfortunately, the Bayesian update (9) under the probit likelihood is intractable and yields a posterior process that is non-Gaussian. To address this problem and enable iterative trust updates, we employ approximate Bayesian inference: the posterior process is projected onto the closest GP as measured by the Kullback-Leibler divergence, $\mathrm{KL}(p_t||q)$, and q is our GP approximation [7]. Minimizing the KL divergence is equivalent to matching the first two moments of p_t and q, which can be performed analytically. The update equations in their natural parameterization forms are given by:

$$\mu_t(\mathbf{x}) = \boldsymbol{\alpha}_t^\top \mathbf{k}(\mathbf{x}) \quad (14)$$

$$k_t(\mathbf{x}, \mathbf{x}') = k(\mathbf{x}, \mathbf{x}') + \mathbf{k}(\mathbf{x})^\top \mathbf{C}_t \mathbf{k}(\mathbf{x}') \quad (15)$$

where $\boldsymbol{\alpha}$ vector and $\mathbf{C}$ are updated using:

$$\boldsymbol{\alpha}_t = \boldsymbol{\alpha}_{t-1} + \gamma_1(\mathbf{C}_{t-1}\mathbf{k}_t + \mathbf{e}_t) \quad (16)$$

$$\mathbf{C}_t = \mathbf{C}_{t-1} + \gamma_2(\mathbf{C}_{t-1}\mathbf{k}_t + \mathbf{e}_t)(\mathbf{C}_{t-1}\mathbf{k}_t + \mathbf{e}_t)^\top \quad (17)$$

where $\mathbf{k}_t = [k(\mathbf{x}_1, \mathbf{x}_t), ..., k(\mathbf{x}_{t-1}, \mathbf{x}_t)]$, $\mathbf{e}_t$ is the t^{th} unit vector and the scalar coefficients b_1 and b_2 are given by:

$$\gamma_1 = \partial_{f^a} \log \int P(c^a_t|f^a, \mathbf{x}_t)df^a = \frac{c^a_i \partial\Phi}{\sigma_x \Phi} \quad (18)$$

$$\gamma_2 = \partial^2_{f^a} \log \int P(c^a_t|f^a, \mathbf{x}_t)df^a = \frac{1}{\sigma^2_x}\left[\frac{\partial^2\Phi}{\Phi} - \left(\frac{\partial\Phi}{\Phi}\right)\right] \quad (19)$$

where $\partial\Phi$ and $\partial^2\Phi$ are the first and second derivatives of Φ evaluated at $\frac{c^a_i(\mu_t(\mathbf{x}) - m(\mathbf{x}))}{\sigma_\mathbf{x}}$. Given (16) and (17), predictions can be made with the probit likelihood (13) in closed-form:

$$\tau^a_t(\mathbf{x}) = \int P(c^a = 1|f^a, \mathbf{x})p_t(f^a)df^a = \Phi\left(\frac{\mu_t(\mathbf{x}) - m(\mathbf{x})}{\sigma_\mathbf{x}}\right) \quad (20)$$

where $\sigma_\mathbf{x} = \sqrt{\sigma^2_n + k_t(\mathbf{x}_i, \mathbf{x}_i)}$.

C. Relationships between the Neural and Bayesian Models

Although the neural and Bayesian models differ conceptually and in details, they both leverage upon the idea of a vector task space $Z \subseteq \mathbb{R}^k$ in which similarities—and thus, trust transfer—between tasks can be easily computed. For the RNN, Z is a dot-product space. For the GP, similarities are computed via the kernel function; the kernel linearly projects the task features into a lower dimensional space ($\mathbf{z} = \mathbf{\Lambda}\mathbf{x}$) where an anisotropic squared exponential kernel is applied.

Neither model attempts to represent exact trust processes in the human brain; rather, they are computational analogues. Both modeling approaches offer conceptual frameworks for capturing the *principles* of trust formation and transfer.

Both models assume Markovian trust updates and that trust summarizes past experience with the robot [5]. They differ principally in terms of the inherent flexibility of the trust updates. In the RNN model, the update parameters, i.e., the gate matrices, are learnt. As such, it is able to adapt trust updates to best fit the observed data. However, the resulting update equations do not lend themselves easily to interpretation. On the other hand, the GP employs fixed-form updates that are constrained by Bayes rule. While this can hamper predictive performance (humans are not thought to be fully Bayesian), the update is interpretable and may be more robust with limited data.

V. Experiments

Our experiments were designed to establish if the proposed trust models that incorporate inter-task structure outperform existing baseline methods. In particular, we sought to answer three questions:

- **Q1** Is it necessary to model trust transfer, i.e., do the proposed function-based models perform better than existing approaches when tested on unseen participants?
- **Q2** Do the models generalize to unseen tasks?
- **Q3** Is it necessary to model differences in initial bias, specifically perceptions of task difficulty?

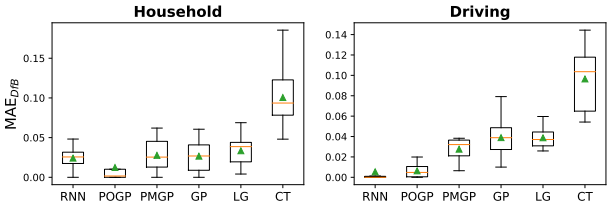
A. Experimental Setup

To answer these questions, we conducted two separate experiments. **Experiment E1** was a variant of the standard 10-fold cross-validation where we held-out data from 10% of the participants (3 people) as a test set. This allowed us to test each model's ability to generalize to unseen participants on the same tasks. To answer question **Q2**, we performed a leave-one-out test on the tasks (**Experiment E2**); we held-out all trust data associated with one task and trained on the remaining data. This process yielded 12 folds, one per task.

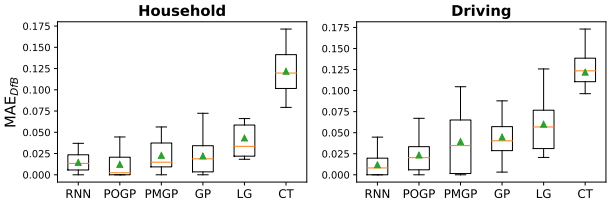
Trust Models. We evaluated 5 models in our experiments:

- **RNN**: The neural trust model (Sec. IV-A);
- **POGP**: The Bayesian GP trust model with prior pseudo-observations (Sec. IV-B);
- **PMGP**: The GP trust model with prior mean function (Sec. IV-B);
- **GP**: A constant-mean Gaussian process trust model;
- **LG**: A linear Gaussian trust model similar to the updates used in OPTIMo [36];
- **CT**: A baseline model with constant trust.

We implemented all the models using the PyTorch framework [24]. Source code for reproducing the following results and plots are available in the online supplementary material [1]. Preliminary cross-validation runs were conducted to find good parameters for the models. The RNN used a two layer fully-connected neural network with 15 hidden neurons and Tanh activation units to project tasks to a 30-dimensional latent task space (Eqn. (3)). The trust updates, Eqn. (4), were performed using two layers of GRU cells. A smaller 3-dimensional latent task space was used for the GP models. GP parameters were optimized during learning, except the length-scales matrix, which was set to the identity matrix $\mathbf{L} = \mathbf{I}$; fixing $\mathbf{L}$ resulted in a smoother optimization process.



(a) Experiment E1



(b) Experiment E2

Fig. 7: $\mathrm{MAE}_{\mathrm{DfB}}$ scores for experiments **E1** and **E2** with medians (lines) and means (triangles) shown. The RNN and POGP models achieve the best scores, indicating the importance of modeling trust transfer and prior bias.

Datasets, Model Training and Evaluation. The models were trained using the data collected in our human subjects study (Sec. III). The RNN and GP-based models were *not* given direct information about the difficulty and group of the tasks since this information is typically not known at test time. Instead, each task was associated with a 50-dimensional GloVe word vector [25] computed from the task descriptions in Fig. 2 (the average of all the word vectors in each description). Complete task descriptions are available online [1].

In these experiments, we predict how each individual's trust is dynamically updated. The tests are not conducted with a single "monolithic" trust model across all participants. Rather, training entails learning the latent task space and model parameters, which are shared among participants, e.g., $\boldsymbol{\beta}$ and $\boldsymbol{\Lambda}$ for the PMGP and the gate matrices for the GRU. However, each participant's model is updated *only* with the tasks and outcomes that the participant observes. Prediction and evaluation is carried out on both pre-and-post-update trust.

We applied maximum likelihood estimation (MLE) and optimized the Bernoulli likelihood of observing normalized trust scores (as soft labels). In more general settings where trust is *not* observed, the models can be trained using observed human actions, e.g., in [4]. We employed the Adam algorithm [14] for a maximum of 500 epochs, with early stopping using a validation set comprising 15% of the training data.

For both experiments, we computed two measures: the average Negative Log-likelihood (NLL) and Mean Absolute

Error (MAE). However, we observed that these scores varied significantly across the folds (each participant/task split). To mitigate this effect, we also computed relative Difference from Best (DfB) scores: $\mathrm{NLL}_{\mathrm{DfB}}(i,k) = \mathrm{NLL}(i,k) - \mathrm{NLL}^*(i)$, where $\mathrm{NLL}(i,k)$ is the NLL achieved by model k on fold i and $\mathrm{NLL}^*(i)$ is the best score among the tested models on fold i. $\mathrm{MAE}_{\mathrm{DfB}}$ is similarly defined.

B. Results

Results for **E1** are summarized in Tbl. Ia with boxplots of $\mathrm{MAE}_{\mathrm{DfB}}$ shown in Fig. 7a. In brief, the RNN and POGP outperform the other models on both datasets across the performance measures. The POGP makes better predictions on the Household dataset, whilst the RNN performed better on the Driving dataset. In addition, the GP achieves better or comparable scores to LG and CT. Taken together, these results indicate that the answer to **Q1** is in the affirmative: accounting for trust transfer between tasks leads to better trust predictions.

Next, we turn our attention to **E2**, which is potentially the more challenging experiment. The RNN and POGP again outperform the other models (see Tbl. Ib and Fig. 7b). Both models are able to make accurate trust predictions on *unseen* tasks (**Q2**), suggesting that (i) the word vectors are informative of the task, and (ii) the models learnt reasonable projections into the task embedding space.

Finally, to answer **Q3**, we examined the differences between the POGP, PMGP, and GP. The PO/PMGP achieved lower or similar scores to the GP model across the experiments and domains, indicating that difficulty modeling enabled better performance. The pseudo-observation technique POGP always outperformed the linear mean function approach PMGP, suggesting initial bias is nonlinearly related to the task features. Potentially, using a non-linear mean function may allow PMGP to achieve similar performance to POGP.

TABLE I: Model Performance on Held-out Participants (Experiments **E1** and **E2**). Average Negative log-likelihood (NLL) and Mean Absolute Error (MAE) scores shown with standard errors in brackets. Best scores in **bold**.

Models	Household		Driving	
	NLL	MAE	NLL	MAE
RNN	0.571 (0.023)	0.173 (0.010)	**0.549** (**0.024**)	**0.175** (**0.011**)
POGP	**0.558** (**0.027**)	**0.161** (**0.013**)	0.553 (0.025)	0.176 (0.012)
PMGP	0.577 (0.019)	0.176 (0.010)	0.567 (0.018)	0.197 (0.011)
GP	0.575 (0.023)	0.175 (0.013)	0.588 (0.022)	0.208 (0.012)
LG	0.578 (0.023)	0.182 (0.011)	0.584 (0.022)	0.208 (0.011)
CT	0.662 (0.029)	0.249 (0.016)	0.649 (0.017)	0.266 (0.010)

(a) Experiment E1

Models	Household		Driving	
	NLL	MAE	NLL	MAE
RNN	0.542 (0.012)	0.166 (0.006)	**0.531** (**0.016**)	**0.174** (**0.014**)
POGP	**0.542** (**0.014**)	**0.164** (**0.007**)	0.562 (0.018)	0.186 (0.008)
PMGP	0.564 (0.014)	0.174 (0.009)	0.574 (0.015)	0.202 (0.008)
GP	0.551 (0.010)	0.174 (0.009)	0.586 (0.013)	0.207 (0.009)
LG	0.568 (0.014)	0.195 (0.009)	0.584 (0.013)	0.222 (0.010)
CT	0.669 (0.008)	0.273 (0.007)	0.661 (0.013)	0.284 (0.005)

(b) Experiment E2

VI. Conclusion and Discussion: Summary, Limitations, and Future Work

In this paper, we took a first step towards conceptualizing and formalizing models for predicting human trust across tasks. We first presented findings from a human-subjects study in two separate domains (household tasks and autonomous driving) showing the effect of task similarity and difficulty on trust formation and transfer.

Based on these findings, we contributed two novel models that capture the form and evolution of trust as a latent function. Our experiments show that the function-based models achieved better predictive performance on both unseen participants and unseen tasks. These results indicate that (i) a task-dependent functional trust model more accurately reflects human trust across tasks, and (ii) it is possible to accurately predict trust transfer by leveraging upon a shared task space representation and update process. The comparable performance between the neural (RNN) and Bayesian (POGP) approaches suggests that Bayes rule well-captures the behavior of human trust updates, provided that initial bias is sufficiently modeled. The choice between a neural or Bayesian method would depend on the whether interpretability of the underlying trust update is required, and the quantity of data available. Further experimentation with a range of kernels may lead to better performance and shed light on underlying processes.

Formalizing trust as a function opens up several avenues for future research. In particular, we aim to more fully exploit this characterization by incorporating other contexts. Does trust transfer when the environment is substantially different or with a new, but similar, robot? Furthermore, proper design of experiments to elicit and measure trust is crucial; our experiments involved relatively short interactions with the robot and relied on subjective self-assessments. Future experiments could employ behavioral measures, such as operator take-overs, and longer-term interactions where trust is likely to play a more significant role.

Finally, it is important to investigate how these models can enhance human-robot interaction in multi-task settings. Embedding trust-transfer models in a decision theoretic framework (e.g., a POMDP [5]) would enable a robot to adapt its behavior according to a human teammate's trust, which can promote fluent long-term collaboration.

Acknowledgement

This work was supported by a NUS Office of the Deputy President (Research and Technology) Startup Grant. Thank you to Indu Prasad for her help with the data analysis.

REFERENCES

[1] Supplementary online material for predicting human trust in robot capabilities across tasks. https://github.com/crslab/human-trust-transfer.

[2] Deborah R Billings, Kristin E Schaefer, Jessie Y C Chen, and Peter A Hancock. Human-Robot Interaction: Developing Trust in Robots. *HRI '12*, pages 5–8, 2012.

[3] Christiano Castelfranchi and Rino Falcone. *Trust Theory: A Socio-Cognitive and Computational Model*. Wiley Publishing, 1st edition, 2010.

[4] Min Chen, Stefanos Nikolaidis, Harold Soh, David Hsu, and Siddhartha Srinivasa. The role of trust in decision-making for human robot collaboration. In *Workshop on Human-Centered Robotics, RSS*, 2017.

[5] Min Chen, Stefanos Nikolaidis, Harold Soh, David Hsu, and Siddhartha Srinivasa. Planning with trust for human-robot collaboration. In *HRI'18*, pages 307–315, New York, NY, USA, 2018. ACM. ISBN 978-1-4503-4953-6.

[6] Kyunghyun Cho, Bart van Merrienboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation. In *EMNLP*, pages 1724–1734, 2014.

[7] Lehel Csató and Manfred Opper. Sparse on-line gaussian processes. *Neural Comp.*, 14(3):641–668, March 2002.

[8] Munjal Desai, Poornima Kaniarasu, Mikhail Medvedev, Aaron Steinfeld, and Holly Yanco. Impact of robot failures and feedback on real-time trust. In *HRI '13*, pages 251–258, 2013.

[9] Dmitri Dolgov, Sebastian Thrun, Michael Montemerlo, and James Diebel. Path planning for autonomous vehicles in unknown semi-structured environments. *IJRR*, 29(5):485–501, 2010.

[10] Thomas L Griffiths, Christopher G Lucas, Joseph J Williams, and Michael L Kalish. Modeling human function learning with Gaussian processes. *NIPS 21*, pages 553–560, 2009.

[11] Peter A Hancock, Deborah R Billings, Kristin E Schaefer, Jessie YC Chen, Ewart J De Visser, and Raja Parasuraman. A meta-analysis of factors affecting trust in human-robot interaction. *Human Factors*, 53(5):517–527, 2011.

[12] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–80, 1997. ISSN 0899-7667.

[13] Rafal Jozefowicz, Wojciech Zaremba, and Ilya Sutskever. An empirical exploration of recurrent network architectures. In *ICML*, pages 2342–2350, 2015.

[14] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[15] John Lee and Neville Moray. Trust, control strategies and allocation of function in human-machine systems. *Ergonomics*, 35(10):1243–1270, 1992.

[16] John D Lee and Neville Moray. Trust, self-confidence, and operators' adaptation to automation. *Intl. J. of Human-Computer Studies*, 40(1):153–184, 1994.

[17] John D Lee and Katrina A See. Trust in automation: Designing for appropriate reliance. *Human Factors*, 46(1):50–80, 2004.

[18] Tomas Mikolov, Kai Chen, Greg S. Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space, 2013. URL http://arxiv.org/abs/1301.3781.

[19] Bonnie M Muir. Trust in automation: Part I. Theoretical issues in the study of trust and human intervention in automated systems. *Ergonomics*, 37(11):1905–1922, 1994.

[20] Bonnie M Muir and Neville Moray. Trust in automation. Part II. Experimental studies of trust and human intervention in a process control simulation. *Ergonomics*, 39(3):429–460, 1996.

[21] Radford M Neal. Monte carlo implementation of gaussian process models for bayesian regression and classification. *arXiv preprint physics/9701026*, 1997.

[22] Stefanos Nikolaidis, Swaprava Nath, Ariel D. Procaccia, and Siddhartha Srinivasa. Game-theoretic modeling of human adaptation in human-robot collaboration. In *HRI'17*, pages 323–331, New York, NY, USA, 2017. ACM. ISBN 978-1-4503-4336-7.

[23] Stefanos Nikolaidis, Yu Xiang Zhu, David Hsu, and Siddhartha Srinivasa. Human-robot mutual adaptation in shared autonomy. *arXiv preprint arXiv:1701.07851*, 2017.

[24] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. 2017.

[25] Jeffrey Pennington, Richard Socher, and Christopher D. Manning. Glove: Global vectors for word representation. In *EMNLP*, pages 1532–1543, 2014.

[26] Carl Edward Rasmussen and Christopher K. I. Williams. *Gaussian Processes for Machine Learning*. The MIT Press, 2006.

[27] Paul Robinette, Wenchen Li, Robert Allen, Ayanna M. Howard, and Alan R. Wagner. Overtrust of robots in emergency evacuation scenarios. In *HRI'16*, pages 101–108, 2016.

[28] Thomas B Sheridan and Robert T Hennessy. Research and modeling of supervisory control behavior. report of a workshop. Technical report, National Research Council Washington DC Committee on Human Factors, 1984.

[29] Indramani L. Singh, Robert Molloy, and Raja Parasuraman. Automation- induced "complacency": Development of the complacency-potential rating scale. *The Intl. J. of Aviation Psychology*, 3(2):111–122, 1993.

[30] Harold Soh and Yiannis Demiris. When and how to help: An iterative probabilistic model for learning assistance by demonstration. In *IROS'13*, pages 3230–3236, November 2013. doi: 10.1109/IROS.2013.6696815.

[31] Harold Soh and Yiannis Demiris. Spatio-temporal learning with the online finite and infinite echo-state gaussian processes. *IEEE Transactions on Neural Networks and Learning Systems*, 26, June 2014. doi: 10.1109/TNNLS.2014.2316291.

[32] Harold Soh and Yiannis Demiris. Learning Assistance by Demonstration: Smart Mobility With Shared Control and Paired Haptic Controllers. *Journal of Human-Robot Interaction*, 4(3): 76–100, 2015.

[33] Harold Soh, Scott Sanner, Madeleine White, and Greg Jamieson. Deep Sequential Recommendation for Personalized Adaptive User Interfaces. In *IUI '17*, pages 589–593, 2017. doi: 10.1145/3025171.3025207.

[34] Ioan A. Şucan, Mark Moll, and Lydia E. Kavraki. The Open Motion Planning Library. *IEEE Robotics & Automation Magazine*, 19(4):72–82, December 2012. http://ompl.kavrakilab.org.

[35] Ning Wang, David V. Pynadath, and Susan G. Hill. Trust calibration within a human-robot team: Comparing automatically generated explanations. In *HRI '16*, pages 109–116, 2016.

[36] Anqi Xu and Gregory Dudek. Optimo: Online probabilistic trust inference model for asymmetric human-robot collaborations. In *HRI*, pages 221–228. ACM, 2015.

[37] Anqi Xu and Gregory Dudek. Towards modeling real-time trust in asymmetric human–robot collaborations. In *Robotics Research*, pages 113–129. Springer, 2016.

[38] X Jessie Yang, Vaibhav V Unhelkar, Kevin Li, and Julie A Shah. Evaluating effects of user experience and system transparency on trust in automation. In *HRI*, pages 408–416, 2017.