Contrastive Learning from Exploratory Actions: Leveraging Natural Interactions for Preference Elicitation

Anonymous Author(s)
Affiliation
email address

***Abstract*—People have a variety of preferences for how robots behave. In order to understand and reason about these preferences, robots aim to learn a reward function that describes how aligned their behaviors are with a user's preferences. Good representations of a robot's behavior can significantly reduce the time a user needs to spend teaching the robot their preferences, making the robot easier to use and facilitating adoption. Specifying these representations—what "features" of the robot's behavior matter to users—remains a difficult problem; Features learned from raw data lack semantic meaning and features learned from user data require users to engage in tedious labeling processes. Our key insight is that users tasked with customizing a robot automatically engage in *exploratory search*; they explore behaviors that they find interesting and ignore behaviors that are irrelevant. We describe these exploration behaviors as *exploratory actions* and identify them as a novel data source that can be leveraged to facilitate the feature learning process for robots. We propose *constrastive learning from exploratory actions* (CLEA) by defining a loss function that learns features from such exploratory actions. We collected exploratory actions from users performing an open-ended signal design activity with a Kuri robot, and evaluated CLEA features through a second user study with a different set of users. CLEA features outperformed self-supervised features when learning user preferences in four ways; CLEA features were more complete, simple, minimal, and explainable.**

***Index Terms*—Preference Learning, Signal Design, Multimodal Learning**

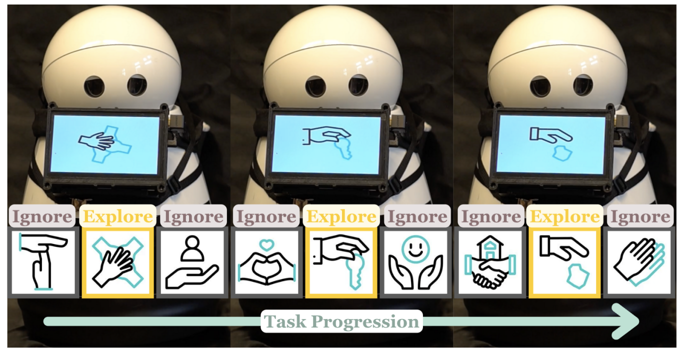

Fig. 1. **Example exploratory search process**. Users engaging in *exploratory search* test out different robot behaviors to learn what the robot is capable of and what they prefer the robot to do.

I. Introduction

People have a variety of preferences for how robots should behave based on many contextual factors, but those contextual factors are often unknown to the designers of robotic systems before a robot is deployed. Consider a wheeled robot that helps users find misplaced items in their home. One user may be a long-time dog owner and thus interpret this interaction as similar to playing fetch. That user might expect the behavioral aspects of the robot to be dog-like. For example, the robot may move erratically as if following a scent, bark when it has found an item, and emote to portray happiness having completed its command. Another user, in contrast, may be more familiar with smart devices and expect the interaction to be purely functional. That user might instead expect the robot to move and scan the room methodically, chime when it finds an item, and immediately return the item to the requester.

Deploying a robot that behaves in only one way cannot satisfy both of these users. Thus, users must be able to *customize* robot behaviors to align with their preferences. Several works view the problem of aligning the robot with the user's preferences as modeling a user's internal reward function, which can be addressed with inverse reinforcement learning [1, 2]. In this context, the reward function takes in numerical "features" of the robot's behavior, e.g., a score of how dog-like or machine-like the behavior is, and output a single value that corresponds to how good that behavior is for the user. How these features are defined heavily influences how effectively a robot can adapt to a specific user. Features can be learned directly from the robot behaviors through self-supervised techniques like autoencoders (AEs) and variational autoencoders (VAEs). While these methods result in features that are physically representative of the robot's behaviors, they may not align with the features people actually care about. The most effective way to learn user-aligned features is by leveraging user-generated data [3]. However, collecting such data typically requires a user to engage in a data-labelling process known as a *proxy task* before the user can engage in the actual task of customizing the robot [4–6].

Our goal in this work is to learn features for robot behaviors that are aligned with user preferences, but do not require users to engage in unrelated proxy tasks. To accomplish this, we identify a new form of user-collected data that is generated during the robot customization process. We collected this data by recruiting users to customize behaviors for a Mayfield Kuri robot that helped them locate items around a room. Participants used the RoSiD interface to design state-expressive signals [7], which allowed them to search through

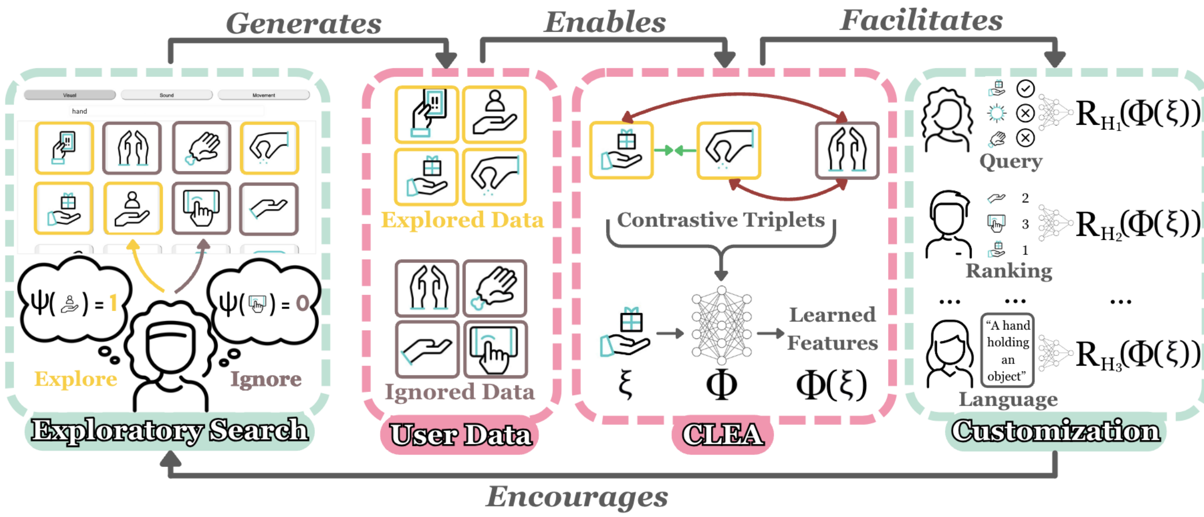

Fig. 2. **CLEA: Contrastive Learning from Exploratory Actions**. Users engage in exploratory search to select their preferred robot behaviors. We automatically generate data from their exploratory actions to learn features that facilitate future interactive learning processes. Our contributions are highlighted in pink, and the enabling work that CLEA supports in highlighted in green.

thousands of example robot behaviors. Participants automatically performed *exploratory actions* by selecting behaviors that appeared appealing to them while ignoring those they thought were irrelevant, as illustrated in Fig. 1.

Our key insight is that we can use these user exploratory actions to learn features for robot behaviors that are both aligned with the features that users care about and do not require users to complete proxy tasks before customizing the robot. We view users performing exploratory actions as engaging in an intuitive reasoning process, and model this using a contrastive loss to train feature-generating networks. We call this framework *contrastive learning from exploratory actions* (CLEA) and provide an overview in Fig. 2.

We show that CLEA learns features that are more effective for eliciting a user's preferences than the state-of-the-art self-supervised learning techniques, offering a scalable and user-friendly approach to personalizing robot behaviors. In Sec. IV, we collected a training set of exploratory actions from 25 participants specifying their signaling preferences to learn CLEA features. In Sec. V, We evaluated the generalizability of those features with a testing set from 42 naïve participants. We found that CLEA features outperformed self-supervised features for robot signaling along four criteria [3]: they contained information relevant to elicit user preferences, required fewer user interactions to elicit preferences, captured information in fewer dimensions, and demonstrated properties that indicate compatibility with explanatory methods.

II. Related Work

Eliciting User Preferences through Interaction. Users must be able to communicate their preferences to the robot so the robot can learn these preferences. Previous works in preference learning identified several interactions that allow users to specify their preferences for robot behaviors, including behavior comparisons [8–10], behavior rankings [11–13], binary rewards [14], corrections [15–18], natural language [19–22], facial expression [23, 24], and demonstrations [25–29]. Those interactions require different skills and provide varying levels of information on the user's true preferences [30–32]. Additionally, all of them assume that there is a numerical representation of robot behavior that encapsulates the features that users care about. Thus, having meaningful numerical features is necessary to allow users of different skill levels to teach robots in a variety of ways.

Learning Representations for Eliciting Preferences. There are three popular approaches for representing features of robot behaviors: hand-crafted features, features learned from modeling robot behaviors, and features learned from user interactions. Hand-crafted features are based on an engineer's intuitions of what is meaningful for users [1, 15, 33, 34]. Such features can speed up the preference learning process because they are meaningful to users, but they can also be difficult to design and can lead to incomplete feature spaces that limit the range of preferences that can be captured [3].

In contrast, features learned from modeling the robot's behaviors require less engineering effort, but are still incomplete. Such techniques learn features with little human input through self-supervised learning [35–37] or weakly-supervised learning [38–40]. While these algorithms result in features that describe the underlying behaviors well and do not require extensive data collection from users, the resulting feature spaces are not semantically meaningful to users [4].

Learning feature spaces from human input can result in more complete feature spaces of robot behaviors. A user may manually select features [41, 42], physically move compliant robots to demonstrate behaviors [43], provide demonstrations for use in multi-task learning [44, 45] and meta-learning frameworks [46, 47], or answer trajectory similarity queries [4]. These methods focus on developing *proxy tasks* to learn features that are aligned with user preferences [3], however these tasks require conscious effort from the user.

We emphasize that such proxy tasks are not necessarily aligned with the users' goal of customizing a robot's behaviors, and thus users may be unmotivated to perform the tasks [48, 49]. In contrast, *exploratory search* provides an interaction that allows the user to achieve their primary goal of robot behavior customization. This work identifies that *exploratory search* additionally generates data that can be used to learn robot feature spaces in place of proxy-task data.

Exploratory Search. Work in human-computer interaction (HCI) distinguishes two interactions with databases: *information retrieval* and *exploratory search* [50]. *Information retrieval* [51, 52] refers to an interaction with a data system wherein the user knows exactly what they need to find—the user's reward is known. In *exploratory search*, the exact goal is unknown ahead of time because the user is unfamiliar with the search topic and how the goal can be achieved [50]. While many previous works implicitly assume that users know what the robot is capable of doing and present preference learning as an information retrieval problem [1, 52], recent work has identified that reformulating robot learning as an exploratory search interaction is useful for novice robot users [7].

Exploratory search interfaces encourage users to generate more search data by allowing them to inspect, save, and filter items in large databases [53–55]. By aggregating and scaling these search data across millions of users, HCI researchers can learn fine-grained profiles of user behaviors [56, 57].

In this work, we examine the effectiveness of exploratory actions for learning features of robot behaviors that users care about. We frame an exploratory action as an intuitive reasoning process where a user quickly evaluates if a robot behavior is somewhat aligned with their preferences. If it is, they select that behavior to perform a more in-depth evaluation on the physical robot. These perceptual processes are often modeled with triplet losses both to capture how people make intuitive decisions and to aggregate individual differences across user populations [4, 58–61]. We use this insight from prior work to learn features of robot behaviors that people care about from the novel data source of exploratory actions.

III. Learning Features from Exploratory Actions

In this section, we formalize our approach that leverages exploratory actions to learn features of robot behaviors.

A. Preliminaries

We consider robot behaviors as trajectories in a fully-observed deterministic dynamical system. We denote a behavior as $\xi \in \Xi$, which represents a series of states and actions: $\xi = (s_0, a_0, s_1, a_1, ..., s_T, a_T)$. These states and actions are abstractly defined; they can be videos (behaviors in image-space), audio (behaviors in frequency-space), or movements (behaviors in joint-space). We assume that all behaviors $\xi \in \Xi$ accomplish the task without resulting in errors, allowing users to specify based on user preferences rather than the behavior's ability to achieve a goal [4, 26, 62]. While generating Ξ is not the focus of this work, it can be completed through several techniques, such as collecting demonstrations [26], performing quality diversity optimization [63], and diversely combining motion primitives [64].

We model a user's preference as a reward function over robot behaviors that maps the space of behaviors to a real value: $R_H : \Xi \mapsto \mathbb{R}$. The user's reward function is not directly observable, but can be inferred through interaction. Our goal is to learn a reward function from user interactions, R_H, that maximizes the likelihood of the user performing the observed interactions. Higher values of R_H for a particular behavior implies that the behavior is more preferred by the user.

Because the state space of robot behaviors can be very large [65, 66], directly learning R_H from state-action sequences is intractable. To make reward learning tractable, several works [1–3] assume that there exists a function Φ that maps from the state-action space to a lower dimensional *feature space*–a real vector of dimension d: $\Phi : \Xi \mapsto \mathbb{R}^d$. This assumption allows us to learn $R_H(\Phi(\xi))$ from fewer user interactions.

B. Contrastive Learning from Exploratory Actions

To learn a Φ, we leverage interaction data that we collected through the robot customization process. Users naturally engaged in *exploratory search* when they were presented with many robot behaviors they could choose from to customize the robot.

We formalize exploratory search as presenting a dataset of behaviors to the user: $\mathcal{D}_i = \{\xi_0, \xi_1, ..., \xi_N\}$ where each ξ_i is sampled from the full database of behaviors Ξ. In our case, ξ_i is a video, a sound, or a head movement, but this definition extends to other behaviors such as robot gaits, or robot arm movements. This dataset can be generated using various methods, including keyword search [67], collaborative filtering [68], and faceted search [69]. Users can view brief summaries of each behavior in the dataset to determine if the behavior is relevant.

We mathematically model the user's internal reasoning process when making an exploratory action with the function $\psi : \mathcal{D} \mapsto \{0, 1\}$. If the user performs an exploratory action on a behavior ξ_j from the dataset $\mathcal{D}_i$, then $\psi(\xi_j) = 1$. If the user does not perform an exploratory action on a behavior ξ_k from the dataset $\mathcal{D}_i$, then $\psi(\xi_k) = 0$. We use this definition of an exploratory action to partition $\mathcal{D}_i$ into two sets:

$$\mathcal{D}_i^{ex.} := \{\xi \in \mathcal{D}_i | \psi(\xi) = 1\}; \mathcal{D}_i^{ig.} := \{\xi \in \mathcal{D}_i | \psi(\xi) = 0\} \tag{1}$$

For example, if a user is initially presented with $\mathcal{D}_0 = \{\xi_A, \xi_B, \xi_C, \xi_D\}$, and they choose ξ_B and ξ_D to execute on the robot, the explored dataset is $\mathcal{D}_0^{ex.} = \{\xi_B, \xi_D\}$ and the ignored dataset is $\mathcal{D}_0^{ig.} = \{\xi_A, \xi_C\}$. In our data collection study, $|\mathcal{D}_i| \approx 100$ to allow users to meaningfully search through behaviors.

A common way to model and aggregate diverse internal reasoning processes, such as ψ, across a population of users is to use a triplet loss [4, 58–61]. We adopt this loss function and generate triplets of behaviors from on the explored and ignored subsets. The triplets are formed by sampling two behaviors at random from one subset and one behavior from the other subset: $(\xi_1^{\mathcal{D}_i^{ex.}}, \xi_2^{\mathcal{D}_i^{ex.}}, \xi_1^{\mathcal{D}_i^{ig.}})$ or conversely $(\xi_1^{\mathcal{D}_i^{ig.}}, \xi_2^{\mathcal{D}_i^{ig.}}, \xi_1^{\mathcal{D}_i^{ex.}})$. The triplet loss encourages features from the same subset to be more similar to each other than features from opposite subsets, according to any metric function. We select the Euclidean distance, $d(\xi_i, \xi_j) = ||\Phi(\xi_i) - \Phi(\xi_j)||_2^2$, as our metric because other works found that Euclidean distances are an appropriate metric for modeling perceptual processes [4, 58]:

$$\mathcal{L}_{trip.}(\xi_A, \xi_P, \xi_N) = \max\left[d(\xi_A, \xi_P) - d(\xi_A, \xi_N) + \alpha, 0\right] \tag{2}$$

We refer to ξ_A as the anchor example, ξ_P as the positive example, ξ_N as the negative example, and $\alpha \geq 0$ as the margin of separation between positive and negative examples. In our case, the anchor and positive example are interchangeable as they are both from the same unordered set, so we formulate the triplet loss to be symmetric:

$$\mathcal{L}_{sym.}(\Phi) = \mathcal{L}_{trip.}(\xi_A, \xi_P, \xi_N) + \mathcal{L}_{trip.}(\xi_P, \xi_A, \xi_N) \tag{3}$$

We formulate the CLEA loss as the sum of this symmetric triplet loss across all of the datasets presented to all the users in the signal design study:

$$\mathcal{L}_{CLEA}(\Phi) = \sum_{i=0}^{|\mathcal{D}_{pop.}|} \sum_{(\xi_A, \xi_P, \xi_N) \sim \mathcal{D}_i} \mathcal{L}_{sym.}(\xi_A, \xi_P, \xi_N) \tag{4}$$

where $\mathcal{D}_{pop}$ represents the set of all datasets presented to the population of users that performed exploratory actions. We learn features that minimize this loss to create a feature space for robot behaviors that is consistent with the variations in the population's preferences.

C. Learning Preferences from Rankings.

We evaluated CLEA through behavior rankings, as in previous works [12]. We presented each user with a set of behaviors to rank, referred to as a query, $Q = \{\xi_0, \xi_1, ...\xi_N\}$. The user then ordered these options from their least favorite behavior to their most favorite behavior by creating a mapping $\sigma : \{0, 1, ..., N\} \mapsto \{0, 1, ..., N\}$ such that $\sigma(Q) := \xi_{\sigma(0)} \prec \xi_{\sigma(1)} \prec ... \prec \xi_{\sigma(N)}$. The notation $\xi_i \prec \xi_j$ denotes that behavior ξ_j is preferred over ξ_i.

We interpreted this ranking as a collection of pairwise comparisons, as in previous works [11]. We adopted the Bradley-Terry preference model [70] to model the probability that the user chooses behavior ξ_j from the pair of behaviors (ξ_i, ξ_j) based on the feature space mapping Φ (i.e., learned with CLEA or other self-supervised objectives):

$$P(\xi_i \prec \xi_j | R_H) = \frac{e^{R_H(\Phi(\xi_j))}}{e^{R_H(\Phi(\xi_i))} + e^{R_H(\Phi(\xi_j))}} \tag{5}$$

To learn the user's reward function, R_H, we maximize the probability of all pairwise comparisons induced by the rankings the user performed. We construct a dataset containing all the users rankings, $\mathcal{D}_{pref.} = \{(Q_0, \sigma_0), (Q_1, \sigma_1), ...(Q_K, \sigma_K)\}$. We minimize the total cross entropy loss summed over all pairwise comparisons across all rankings:

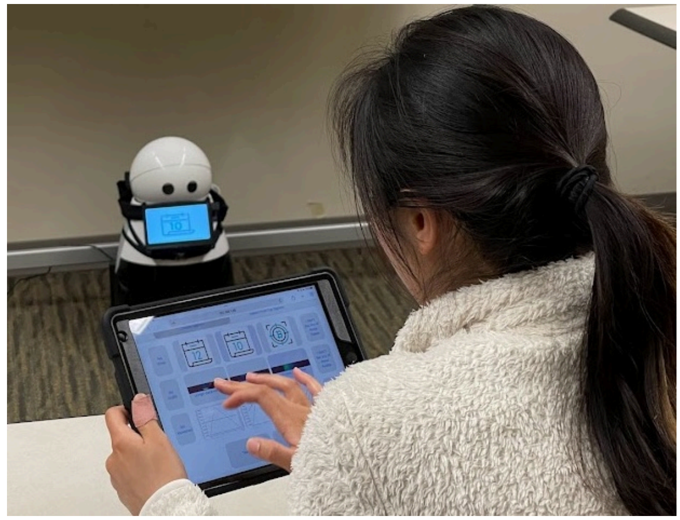

Fig. 3. **Customization session setup.** Participant designing signals for the modified Kuri robot using the query-based interface.

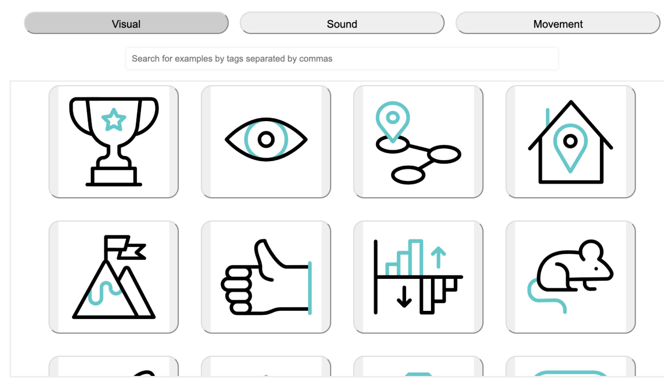

Fig. 4. **Exploratory search interface.** The exploratory search interface the participants used to design robot signals. Participants could explore visual, auditory, and kinetic robot behaviors by scrolling through the behaviors and by typing in search terms to filter results.

$$\mathcal{L}(R_H) = \sum_{(Q,\sigma) \in \mathcal{D}_{pref}} \sum_{i=0}^{|Q|-1} \sum_{k=i+1}^{|Q|} -\log P(\xi_{\sigma(i)} \prec \xi_{\sigma(k)} | R_H) \tag{6}$$

R_H can be any computational model that can update its parameters to minimize a loss function. In this work, we used both neural networks and linear models to approximate R_H to compare with prior works, however other techniques such as gaussian processes [71] are possible.

IV. Collecting Exploratory Actions from a Robot Customization Session

In this section, we describe our methodology for collecting user exploratory actions in a free-form customization session involving a Kuri robot performing an item-finding task.

A. Procedure

We adapted a signal design task from previous work [7] to collect *exploratory action* data from users designing multimodal signals for a robot to express the robot's state while assisting the user in an item-finding task. Robot signaling presents a wide array of unconstrained and diverse user preferences, as opposed to purely functional tasks, where users tend to converge to a small set of effective robot behaviors [9, 71]. Additionally, signaling does not require experimenters to "engineer" user responses by telling them what their preference should be [4].

We recruited participants to design signals for a robot that assists them in finding items around a room. This study received ethical approval from our university's Institutional Review Board. Participants designed four state-expressive signals for a Mayfield Kuri robot [72]. We chose Kuri because it was designed to be a low-cost consumer product capable of natural interaction with non-expert users. For this study, we modified Kuri with an added screen and backpack to hold items, as shown in Fig. 3.

The participants were tasked with designing the following four signals: (1) idle, indicating that the robot is ready to receive a command; (2) searching, indicating that the robot is actively looking for a requested item; (3) has-item, indicating that the robot has the requested item in its backpack; and (4) has-information, indicating that the robot found the requested item but will need the user's assistance to procure the item.

Each signal consisted of three separate modalities: (1) **visual**: video played on the robot's screen; (2) **auditory**: sound played through the robot's speaker; and (3) **kinetic**: robot head movement. These three modalities correspond to common data structures in robotics [4, 23, 73]: the visual modality is a sequence of images, the auditory modality is a spectrogram, and the kinetic modality is a sequence of joint angles. These robot behaviors pulled from a database of 5,192 videos, 867 sounds, and 2,125 head movements[1].

To engage in customizing the robot's signaling behaviors, users were presented with the RoSiD interface [74]. This interface allowed users to specify preferences for the robots in two ways. In the *query-based* interaction for customizing robots, shown in Fig. 3, participants were presented with a set of three options and chose their favorite among them. In the *exploratory search* interaction, shown in Fig. 4, participants were presented with up to one hundred options which they could filter, scroll through, and test on the physical robot. We designed the *exploratory search* interface to allow users to explore robot behaviors without having to repeatedly answer questions about rating the specific behaviors, which leads to user fatigue and decreased motivation [75]. Participants were allowed to freely choose either the *query-based* interface or the *exploratory search* interface to design signals for the robot. Participants were instructed to end the signal design when they were satisfied with their signal.

After a participant designed all four signals, they interacted with the robot to complete an item-finding task, where the robot was piloted by an experimenter. The robot used the signals the participant designed to complete the interaction and allow the user to evaluate their design choices. Following the interaction, participants were interviewed to determine how they liked to design signals. The interview data are outside of the scope for this paper. See Appendix A for more details.

B. Data Collection Results

In total, 25 participants (10 women, 3 non-binary, 12 men; see Appendix A for more demographic information) were recruited to design signals for the robot. Participants were compensated with US$ 20 Amazon gift cards.

In total, participants spent an average of 415 seconds using the query-based interface and 654 seconds using the exploratory search interface. Each participant performed an average of 15.32 query interactions and 72.72 *exploratory actions* across the four signals. Query interactions took an average of 27 seconds for a user to produce, compared to an average of 9 seconds for exploratory actions. We define *exploratory actions* as the robot behaviors from the exploratory search interface that the user chose to evaluate on the real robot because they appeared appealing, compared to the behaviors that the user ignored because they seemed irrelevant. These exploratory actions encompass **all** actions the user took to explore appealing robot behaviors, including examined behaviors that were seemingly unrelated to the specific signal. These unrelated explorations were generally actions users took to learn more about the robot's capabilities.

We use the data we collected from all users performing exploratory actions to learn features of robot behaviors according to Eq. 4. These features can be used for downstream preference learning tasks, such as behavior comparisons [8–10] or behavior rankings [11–13]. To select hyperparameters, we used the data we collected from the *query-based* interface. For more training details see Appendix C. Our previous work showed preliminary efficacy for CLEA by evaluating with data from the *query-based* interface using leave-one-out cross-validation [76]. We expand on those results by recruiting a new set of participants performing behavior rankings to evaluate CLEA on a new user population, detailed in the next section.

V. User Study Evaluation

To evaluate the efficacy of learning feature spaces for robot behaviors, we conducted an experiment with a new set of participants. The participants ranked behaviors to generate individual datasets that we could use to quantitatively test different feature-learning algorithms.

A. Manipulated Variables

To evaluate the effectiveness of learned features using automatically collected data, we evaluated seven total algorithms for learning feature spaces. The first baseline was: (1) ***Random***, a randomly-initialized neural network that projects each behavior to a vector. Random networks can be effective feature learners, as they cannot overfit to data or learn spurious correlations. (2) ***Pretrained***, a large pre-trained neural network to generate features. We used the vision-language foundation

[1] The full dataset of behaviors is available to view on our project page: https://interaction-lab.github.io/CLEA/

model X-CLIP [77] to create features from videos of the visual and kinetic modalities. We used the audio foundation model AST [78] to create features for the auditory modality. We also evaluated two self-supervised baselines: (3) ***AE***, an autoencoder that uses a self-supervised loss to learn features that reconstruct the behavior, (4) ***VAE***, a variational autoencoder that uses a self-supervised loss to both reconstruct and standardize the distribution that the features come from. The AE and VAE methods use the latent space of these models as features. All of these self-supervised losses can also be combined with CLEA, so we evaluate the following as our proposed algorithms: (5) ***CLEA***, (6) ***CLEA+AE***, and (7) ***CLEA+VAE***. For all algorithms, we learned separate feature spaces for each of the three signal modalities: visual, auditory, and kinetic. The size of each feature space was a 128-dimension vector, which was sufficient to capture diverse preferences for complex behaviors [79]. Additional information on the training processes is presented in Appendix C.

B. Procedure

To evaluate the generalizability of the features we learned, we collected ranking data from a separate set of 42 new participants (19 women, 4 non-binary, 19 men; more details in Appendix A). Each participant completed ten behavior ranking trials for a particular modality and signal, with each ranking consisting of five robot behaviors.

The five behaviors we presented to the user for each ranking were selected based on the final customized signals in the customization session described in Sec. IV, because previous work has shown that using other users' preferences is a good initialization for new users [7, 26, 62, 80]. To generate each of these five behaviors, we first randomly sampled a customized behavior from the customization session and then sampled one of the six feature-learning algorithms. Then, we calculated the behavior in the full database of behaviors that minimized the feature distance to the custom behavior we sampled, according to the feature space we sampled.

In order to fully evaluate the proposed algorithms, we must know the user's overall favorite behavior to use as a ground-truth preference. To achieve this, the fifth and tenth ranking used the top-ranked signals from the previous ranking trials to create a "super ranking". The highest-ranked behavior in the final ranking trial represents the participant's overall favorite behavior.

C. Hypotheses

A survey by Bobu et al. [3] identified four criteria that constitute good representations for downstream preference-learning tasks: (1) ***Completeness***, the ability of a representation to capture a user's true preferences, (2) ***Simplicity***, the ability to recover user preferences from linear transformations of the representations, (3) ***Minimality***, the ability of a representation to exist in low-dimensional spaces, and (4) ***Explainability***, the ability of a representation to be compatible with existing explainability tools. We adopt this framework for our analysis and present our additional results in Appendix H.

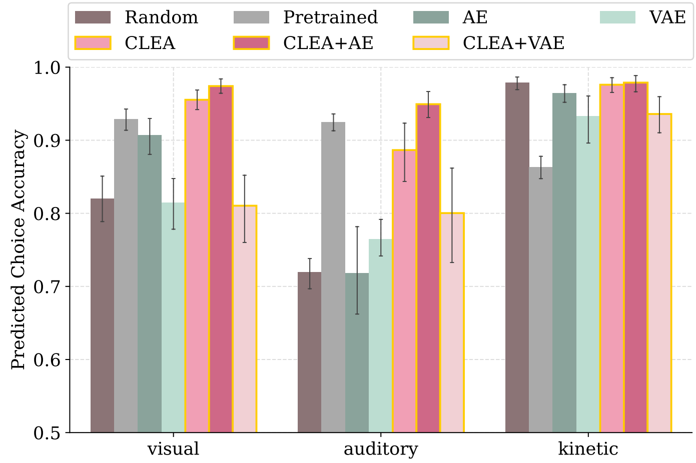

Fig. 5. **Completeness results.** Across three modalities, feature spaces using CLEA are able to accurately predict user preferences. Error bars show mean standard error across participants.

We evaluate these four criteria with the data collected from the 42 participants in Sec. V-B. We split the data from each participant into 70% for training our reward models, and 30% for evaluating our reward models.

Based on this framework, we tested four hypotheses comparing CLEA features to self-supervised features:

- **(H1)** Exploratory actions reflect user preferences, so the most **complete** features will leverage CLEA;
- **(H2)** Exploratory actions align with preference teaching tasks, so the most **simple** features will leverage CLEA;
- **(H3)** Exploratory actions efficiently express user preferences, so the most **minimal** features will leverage CLEA; and
- **(H4)** Exploratory actions are semantically meaningful, so the most **explainable** features will leverage CLEA.

D. Results

Evaluating Completeness. Completeness refers to the feature's ability to capture all relevant information to understand how a user ranks robot behaviors. To evaluate completeness, we aim to learn a **neural network** reward model that can accurately model the choices participants made during the ranking experiment. We quantitatively measure this with the *test preference accuracy* (TPA) metric [4], which measures the accuracy of a trained model to predict the user's choice in an unseen test set. To predict user choices, we used the 128-dimensional feature spaces for the six algorithms as input to a neural network that estimated the participant's internal reward. This reward network that consisted of two fully connected layers with hidden dimensions of 256 units to output a single value. The training objective maximized the probabilities of the selected behaviors in the training set using Eq. 5.

The TPA of all three modalities is shown in Fig. 5. We show that CLEA+AE had the highest TPA in modeling participant's choices in the Visual and Auditory modality, determined by t-tests (all $p < .05$). In the Kinetic modality, CLEA+AE, CLEA, and Random were tied for the highest TPA, but had a higher TPA than the other methods ($p < .05$). CLEA features contain ***complete*** information to model user preferences, supporting **H1**. For an extended statistical anaylsis, see Appendix E.

TABLE I

Simplicity results. For each modality, we found the area under the curve (AUC) of the alignment metric over 100 pairwise comparisons across feature dimensionalities. Asterisks indicate best-performing algorithm within each dimension (all $p < .05$).

	Visual					Auditory					Kinetic				
Dimension	8	16	32	64	128	8	16	32	64	128	8	16	32	64	128
Random	.005	.024	.018	.013	.004	-.001	.134	-.001	.000	.000	.003	.179	.091	.187	.272
Pretrained	.127	.056	.047	.042	.035	.267	.022	.021	.025	.025	.051	.042	.051	.057	.055
AE	.024	.014	.011	.014	.007	.038	.065	.042	.080	.015	.014	.227	**.330***	.321	.154
VAE	.269	**.335***	.247	.180	.033	.234	.174	.117	.083	.077	.207	.251	.192	.203	.346
CLEA	.012	.261	.245	.090	.044	.058	.002	.142	.113	.046	**.284***	.224	.255	.345	.217
CLEA+AE	**.315***	.219	.330	.163	**.275***	.260	.023	.141	.015	.140	.079	.208	.192	.207	.147
CLEA+VAE	.196	.295	**.376***	**.293***	.147	**.438***	**.343***	**.236***	**.198***	**.175***	.009	**.260***	.165	**.373***	**.377***

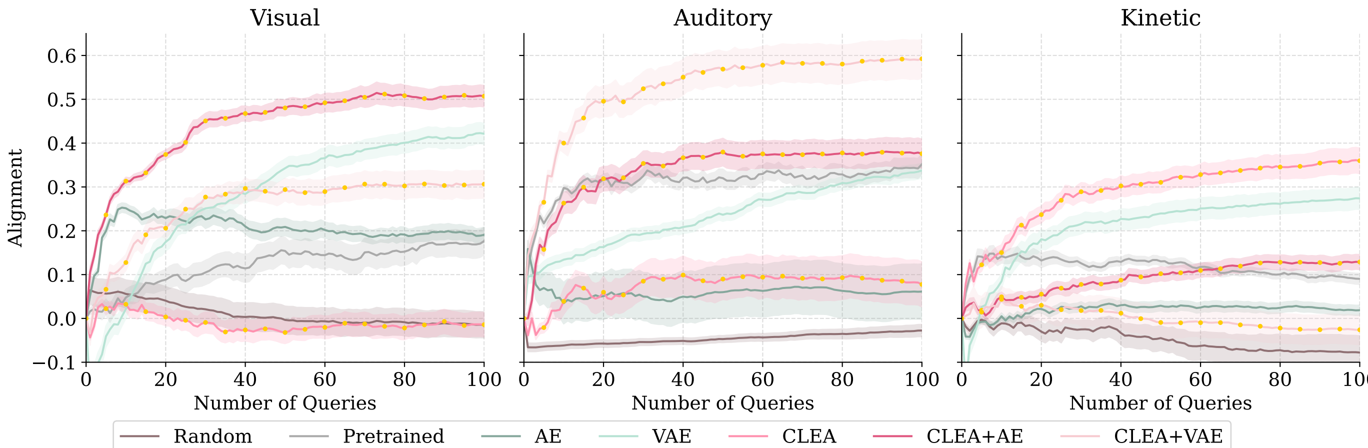

Fig. 6. **Minimality results.** Alignment of a linear reward model across numbers of pairwise comparisons for the smallest sized feature space. Shaded region indicates mean standard error.

Evaluating Simplicity and Minimality. A feature space is ***simple*** if it can model a user's preference with a **linear** reward model, and it is ***minimal*** if the dimensionality of the feature space that is used as input to this reward model is small while still accurately capturing user preferences [3]. Simple linear reward models are practically useful compared to complex neural network reward models because linear models are easier to store, interpret, and compare [81]. To quantify both minimality and simplicity, we used the area under the curve of alignment (AUC Alignment) [8, 9, 12] over 100 pairwise queries. We evaluated AUC Alignment across feature spaces of five dimensions: 8, 16, 32, 64, and 128. The simple linear model to estimate a user's reward was described as $R_H(\xi) = \omega \cdot \Phi(\xi)$. We estimated ω using Bayesian inverse reward learning, as in previous works [8, 9, 11].

To calculate AUC Alignment, we sequentially updated the estimate of the user's preference $\omega_{est.}$ after observing each ranking action they made, decomposed into pairwise queries. We calculated the alignment of the user's true preference ω_{true} and estimated preference, $\omega_{est.}$, following the equation, $\frac{1}{M}\sum_{\omega_{est.}\sim\Omega}\frac{\omega_{true}\cdot\omega_{est.}}{||\omega_{true}||_2\cdot||\omega_{est.}||_2}$, from prior work [8, 9]. We set the user's true preference, ω_{true}, as the vector corresponding to the user's top-ranked signal. We used the area under the curve of alignment (AUC Alignment) [8, 9, 12] over the number of queries as the metric to assess simplicity and minimality. A higher AUC Alignment indicates that we learned the user's preference more accurately and with fewer queries. We show the alignment curve in Fig. 6.

To evaluate ***simplicity***, we compared AUC Alignment of our simple model across all five feature space dimensions to show that a simple model effectively models preferences for all dimensions; the results are shown in Table I. We observed that across all modalities, a CLEA-based feature space has the highest AUC Alignment in 13 of the 15 experiments, with CLEA+VAE being the best on 10 of these experiments. Overall, training with the CLEA objective resulted in ***simple*** representations that were useful across different sized dimensions, supporting **H2**. Further analysis is in Appendix F.

To evaluate ***minimality***, we compared AUC Alignment for only the 8-dimensional feature space to determine if CLEA can model user preferences for low-dimensional feature spaces. The results are shown in Fig. 6. We found that CLEA+AE has the highest AUC Alignment in the Visual modality, CLAE+VAE has the highest AUC Alignment in the Auditory modality, and CLEA has the highest AUC Alignment in the Kinetic modality. We conclude that using a loss function that includes the CLEA objective results in features that ***minimally*** elicit user preferences, supporting **H3**. For additional analysis see Appendix G.

Evaluating Explainability. A common explainability technique for neural systems is explanation by example [82, 83]. In this framing, the user is presented with training examples,

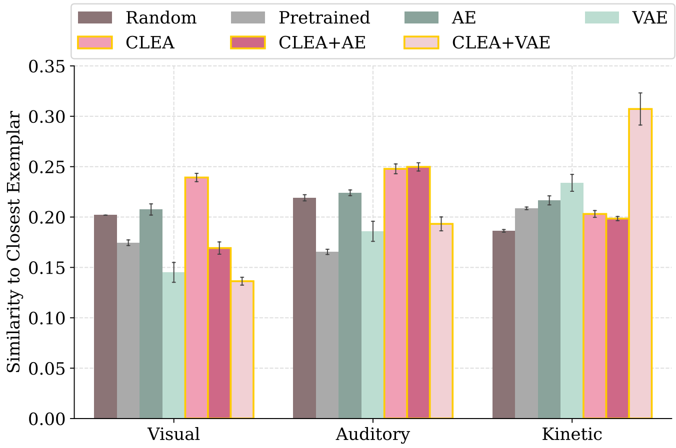

Fig. 7. **Explainability Results.** We examined the similarity between top-ranked signals from the ranking user study and the nearest exemplar signal from the robot customization session. We found that CLEA-based feature spaces have higher similarities, facilitating explanation by example.

called exemplars, that are near unseen test samples to provide an interpretation of learned feature spaces. This approach has been use to explain feature spaces used for clustering [84], image recognition [85], and language [86]. In this work, we quantitatively evaluate explainability of the feature space by measuring the cosine similarity of the top-ranked signals from the ranking study (Sec. V-B) to their nearest exemplar from the customization session (Sec. IV). We show the similarity scores for each modality in Fig. 7.

We found that the CLEA feature space resulted in the highest similarity between top-ranked signals and the nearest exemplars for both the visual and auditory modality. The CLEA+VAE feature space demonstrated the highest similarity in the Kinetic modality. These results indicate that CLEA-based feature spaces are more conducive to explanation by example than self-supervised feature spaces, thus CLEA results in the most ***explainable*** feature spaces, supporting **H4**.

VI. Discussion and Limitations

Our results demonstrate the efficacy of learning useful features for eliciting preferences by leveraging data from natural user interface interaction. By incorporating exploratory search concepts into interfaces for teaching robots, we can scale data collection while also providing users with an intrinsically motivating task. We performed evaluations across three modalities of state-expressive signals in robots—visual, auditory, and kinetic—and found that using CLEA significantly increased performance in all modalities and evaluation criteria.

CLEA can be readily combined with any algorithm that learns feature spaces for eliciting preferred robot behaviors by learning a lower-dimensional embedding of the behaviors. We demonstrated CLEA's use with self-supervised losses; it can also be used with other methods of learning feature spaces such as trajectory similarity queries [4], multi-task learning [44, 45], and labelled behaviors [87, 88].

We developed an algorithm for learning feature spaces using exploratory search actions. This requires users to be able to briefly review many behaviors at a high level before deciding which of these behaviors might be relevant and warrant further exploration. We used visual summaries for all modalities–a video frame for the visual modality, a spectrogram for the auditory modality, and a graph of joint angles for the kinetic modality. These were interpretable by the participants, but they may not be the most effective means of representing the underlying behaviors. Users may be better at interpreting natural language descriptions [89], tags that describe the behavior [54], or animated gifs [90]. Future work can explore how robot behaviors can be summarized to non-expert users in ways that allow them to most efficiently search through robot behavior options. Understandable summaries are especially needed for users to perform exploratory search with more complex robot behaviors, such as different gaits for quadruped robots or dexterous manipulation skills for high degree of freedom manipulators.

The robot behaviors that users explored often appeared unrelated to the user's personal preferences, but these seemingly random explorations were still indicative of other users' preferences, as demonstrated by the transferability of CLEA to distinct populations. We also assumed that users would be motivated to perform this exploratory search since they were unfamiliar with what the robot was capable of [50]. If users are already familiar with a particular robot, they may not be motivated to perform exploratory actions because they have already found their preferred robot behaviors. This familiarity could decrease the efficacy of CLEA as a framework for learning feature spaces in already adopted robots.

We also found that additional loss terms, such as the reconstruction term and the KL-divergence term, were only sometimes helpful for training with CLEA, depending on the underlying data structures used. We found that the additional reconstruction loss and KL-Divergence loss were often helpful in video and sound data structures but could hinder preference learning for joint state sequences. This effect is due to the social interpretations interfering with these additional terms. For joint state behaviors, gestures portraying fear and excitement have very similar joint states, but vastly different social interpretations. CLEA aims to separate these features, while the reconstruction term brings them together. While the optimal loss terms can only be determined experimentally, we presented a set of metrics for evaluation that can systematically determine these terms, in accordance with the qualities of good feature spaces [3].

Conclusion. We present *contrastive learning from exploratory actions* (CLEA), an algorithm to leverage a novel data source of interactions that users automatically perform when teaching robot systems. We showed that CLEA can be used to learn feature spaces that reflect underlying personal preferences, represent robot behaviors in low-dimensional vectors, quickly elicit user preferences, and are explainable.

References

[1] A. Y. Ng and S. Russell, "Algorithms for inverse reinforcement learning." in *Icml*, vol. 1, 2000, p. 2.

[2] P. Abbeel and A. Y. Ng, "Apprenticeship learning via inverse reinforcement learning," in *Proceedings of the twenty-first international conference on Machine learning*, 2004, p. 1.

[3] A. Bobu, A. Peng, P. Agrawal, J. A. Shah, and A. D. Dragan, "Aligning human and robot representations," in *Proceedings of the 2024 ACM/IEEE International Conference on Human-Robot Interaction*, 2024, pp. 42–54.

[4] A. Bobu, Y. Liu, R. Shah, D. S. Brown, and A. D. Dragan, "Sirl: Similarity-based implicit representation learning," in *Proceedings of the 2023 ACM/IEEE International Conference on Human-Robot Interaction*, 2023, pp. 565–574.

[5] K. Lee, L. Smith, and P. Abbeel, "Pebble: Feedback-efficient interactive reinforcement learning via relabeling experience and unsupervised pre-training," *arXiv preprint arXiv:2106.05091*, 2021.

[6] M. Yang and O. Nachum, "Representation matters: Offline pretraining for sequential decision making," in *International Conference on Machine Learning*. PMLR, 2021, pp. 11 784–11 794.

[7] N. Dennler, D. Delgado, D. Zeng, S. Nikolaidis, and M. J. Matarić, "The rosid tool: Empowering users to design multimodal signals for human-robot collaboration," in *Experimental Robotics: The 18th International Symposium*. Springer, 2023.

[8] E. Bıyık, M. Palan, N. C. Landolfi, D. P. Losey, and D. Sadigh, "Asking easy questions: A user-friendly approach to active reward learning," *arXiv preprint arXiv:1910.04365*, 2019.

[9] D. Sadigh, A. D. Dragan, S. Sastry, and S. A. Seshia, "Active preference-based learning of reward functions," 2017.

[10] S. Lu, M. Zheng, M. C. Fontaine, S. Nikolaidis, and H. Culbertson, "Preference-driven texture modeling through interactive generation and search," *IEEE transactions on haptics*, vol. 15, no. 3, pp. 508–520, 2022.

[11] D. Brown, W. Goo, P. Nagarajan, and S. Niekum, "Extrapolating beyond suboptimal demonstrations via inverse reinforcement learning from observations," in *International conference on machine learning*. PMLR, 2019, pp. 783–792.

[12] V. Myers, E. Biyik, N. Anari, and D. Sadigh, "Learning multimodal rewards from rankings," in *Conference on robot learning*. PMLR, 2022, pp. 342–352.

[13] L. Chen, R. Paleja, and M. Gombolay, "Learning from suboptimal demonstration via self-supervised reward regression," in *Conference on robot learning*. PMLR, 2021, pp. 1262–1277.

[14] C. Celemin and J. Ruiz-del Solar, "An interactive framework for learning continuous actions policies based on corrective feedback," *Journal of Intelligent & Robotic Systems*, vol. 95, pp. 77–97, 2019.

[15] A. Bajcsy, D. P. Losey, M. K. O'Malley, and A. D. Dragan, "Learning from physical human corrections, one feature at a time," in *Proceedings of the 2018 ACM/IEEE International Conference on Human-Robot Interaction*, 2018, pp. 141–149.

[16] M. Li, A. Canberk, D. P. Losey, and D. Sadigh, "Learning human objectives from sequences of physical corrections," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2021, pp. 2877–2883.

[17] Y. Cui and S. Niekum, "Active reward learning from critiques," in *2018 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2018, pp. 6907–6914.

[18] D. P. Losey and M. K. O'Malley, "Including uncertainty when learning from human corrections," in *Conference on Robot Learning*. PMLR, 2018, pp. 123–132.

[19] A. Tabrez, J. Kawell, and B. Hayes, "Asking the right questions: Facilitating semantic constraint specification for robot skill learning and repair," in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2021, pp. 6217–6224.

[20] S. S. Raman, V. Cohen, E. Rosen, I. Idrees, D. Paulius, and S. Tellex, "Planning with large language models via corrective re-prompting," in *NeurIPS 2022 Foundation Models for Decision Making Workshop*, 2022.

[21] J. X. Liu, Z. Yang, I. Idrees, S. Liang, B. Schornstein, S. Tellex, and A. Shah, "Grounding complex natural language commands for temporal tasks in unseen environments," in *Conference on Robot Learning*. PMLR, 2023, pp. 1084–1110.

[22] J. Wu, R. Antonova, A. Kan, M. Lepert, A. Zeng, S. Song, J. Bohg, S. Rusinkiewicz, and T. Funkhouser, "Tidybot: Personalized robot assistance with large language models," *Autonomous Robots*, vol. 47, no. 8, pp. 1087–1102, 2023.

[23] Y. Cui, Q. Zhang, B. Knox, A. Allievi, P. Stone, and S. Niekum, "The empathic framework for task learning from implicit human feedback," in *Conference on Robot Learning*. PMLR, 2021, pp. 604–626.

[24] M. Stiber, R. H. Taylor, and C.-M. Huang, "On using social signals to enable flexible error-aware hri," in *Proceedings of the 2023 ACM/IEEE International Conference on Human-Robot Interaction*, 2023, pp. 222–230.

[25] S. Nikolaidis and J. Shah, "Human-robot cross-training: computational formulation, modeling and evaluation of a human team training strategy," in *2013 8th ACM/IEEE international conference on human-robot interaction (HRI)*. IEEE, 2013, pp. 33–40.

[26] S. Nikolaidis, R. Ramakrishnan, K. Gu, and J. Shah, "Efficient model learning from joint-action demonstrations for human-robot collaborative tasks," in *Proceedings of the tenth annual ACM/IEEE international conference on human-robot interaction*, 2015, pp. 189–196.

[27] B. D. Ziebart, A. L. Maas, J. A. Bagnell, and A. K. Dey, "Maximum entropy inverse reinforcement learning." in *Aaai*, vol. 8. Chicago, IL, USA, 2008, pp. 1433–1438.

[28] Z. Cao and D. Sadigh, "Learning from imperfect demonstrations from agents with varying dynamics," *IEEE Robotics and Automation Letters*, vol. 6, no. 3, pp. 5231–5238, 2021.

[29] D. Hughes, A. Agarwal, Y. Guo, and K. Sycara, "Inferring non-stationary human preferences for human-agent teams," in *2020 29th IEEE International Conference on Robot and Human Interactive Communication (RO-MAN)*. IEEE, 2020, pp. 1178–1185.

[30] T. Fitzgerald, P. Koppol, P. Callaghan, R. Q. J. H. Wong, R. Simmons, O. Kroemer, and H. Admoni, "Inquire: Interactive querying for user-aware informative reasoning," in *6th Annual Conference on Robot Learning*, 2022.

[31] E. Bıyık, D. P. Losey, M. Palan, N. C. Landolfi, G. Shevchuk, and D. Sadigh, "Learning reward functions from diverse sources of human feedback: Optimally integrating demonstrations and preferences," *The International Journal of Robotics Research*, vol. 41, no. 1, pp. 45–67, 2022.

[32] Y. Cui, P. Koppol, H. Admoni, S. Niekum, R. Simmons, A. Steinfeld, and T. Fitzgerald, "Understanding the relationship between interactions and outcomes in human-in-the-loop machine learning," in *International Joint Conference on Artificial Intelligence*, 2021.

[33] D. Hadfield-Menell, S. Milli, P. Abbeel, S. J. Russell, and A. Dragan, "Inverse reward design," *Advances in neural information processing systems*, vol. 30, 2017.

[34] J. Huang, R. M. Aronson, and E. S. Short, "Modeling variation in human feedback with user inputs: An exploratory methodology," in *Proceedings of the 2024 ACM/IEEE International Conference on Human-Robot Interaction*, 2024, pp. 303–312.

[35] D. Brown, R. Coleman, R. Srinivasan, and S. Niekum, "Safe imitation learning via fast bayesian reward inference from preferences," in *International Conference on Machine Learning*. PMLR, 2020, pp. 1165–1177.

[36] M. Laskin, A. Srinivas, and P. Abbeel, "Curl: Contrastive unsupervised representations for reinforcement learning," in *International conference on machine learning*. PMLR, 2020, pp. 5639–5650.

[37] D. Ha and J. Schmidhuber, "Recurrent world models facilitate policy evolution," *Advances in neural information processing systems*, vol. 31, 2018.

[38] Y. Wang, Z. Sun, J. Zhang, Z. Xian, E. Biyik, D. Held, and Z. Erickson, "Rl-vlm-f: Reinforcement learning from vision language foundation model feedback," *arXiv preprint arXiv:2402.03681*, 2024.

[39] C. Chamzas, M. Lippi, M. C. Welle, A. Varava, A. Marino, L. E. Kavraki, and D. Kragic, "State representations in robotics: Identifying relevant factors of variation using weak supervision," *Robot Learn. Wksp. at NeurIPS*, 2020.

[40] J. Brehmer, P. De Haan, P. Lippe, and T. S. Cohen, "Weakly supervised causal representation learning," *Advances in Neural Information Processing Systems*, vol. 35, pp. 38 319–38 331, 2022.

[41] K. Bullard, S. Chernova, and A. L. Thomaz, "Human-driven feature selection for a robotic agent learning classification tasks from demonstration," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2018, pp. 6923–6930.

[42] M. Cakmak and A. L. Thomaz, "Designing robot learners that ask good questions," in *Proceedings of the seventh annual ACM/IEEE international conference on Human-Robot Interaction*, 2012, pp. 17–24.

[43] A. Bobu, M. Wiggert, C. Tomlin, and A. D. Dragan, "Feature expansive reward learning: Rethinking human input," in *Proceedings of the 2021 ACM/IEEE International Conference on Human-Robot Interaction*, 2021, pp. 216–224.

[44] K. Nishi and M. Shimosaka, "Fine-grained driving behavior prediction via context-aware multi-task inverse reinforcement learning," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 2281–2287.

[45] J. Yamada, K. Pertsch, A. Gunjal, and J. J. Lim, "Task-induced representation learning," in *10th International Conference on Learning Representations, ICLR 2022*. International Conference on Learning Representations, ICLR, 2022.

[46] M. L. Schrum, E. Hedlund-Botti, N. Moorman, and M. C. Gombolay, "Mind meld: Personalized meta-learning for robot-centric imitation learning," in *2022 17th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*. IEEE, 2022, pp. 157–165.

[47] M. L. Schrum, E. Hedlund-Botti, and M. Gombolay, "Reciprocal mind meld: Improving learning from demonstration via personalized, reciprocal teaching," in *Conference on Robot Learning*. PMLR, 2023, pp. 956–966.

[48] R. Thompson, "Extending the technology acceptance model with motivation and social factors," *AMCIS 1998 Proceedings*, p. 254, 1998.

[49] Y. Lee, J. Lee, and Y. Hwang, "Relating motivation to information and communication technology acceptance: Self-determination theory perspective," *Computers in Human Behavior*, vol. 51, pp. 418–428, 2015.

[50] G. Marchionini, "Exploratory search: from finding to understanding," *Communications of the ACM*, vol. 49, no. 4, pp. 41–46, 2006.

[51] M. Sanderson and W. B. Croft, "The history of information retrieval research," *Proceedings of the IEEE*, vol. 100, no. Special Centennial Issue, pp. 1444–1451, 2012.

[52] A. Singhal, "Modern information retrieval: A brief overview," *IEEE Data Eng. Bull.*, vol. 24, no. 4, pp. 35–43, 2001.

[53] G. Allen, B. L. Peterson, D. K. Ratakonda, M. N. Sakib, J. A. Fails, C. Kennington, K. L. Wright, and M. S. Pera, "Engage!: co-designing search engine result pages to foster interactions," in *Interaction Design and Children*, 2021, pp. 583–587.

[54] J. C. Chang, N. Hahn, A. Perer, and A. Kittur, "Searchlens: Composing and capturing complex user interests for exploratory search," in *Proceedings of the 24th International Conference on Intelligent User Interfaces*, 2019, pp. 498–509.

[55] B. Rahdari, P. Brusilovsky, D. Babichenko, E. B. Littleton, R. Patel, J. Fawcett, and Z. Blum, "Grapevine: A profile-based exploratory search and recommendation system for finding research advisors," *Proceedings of the Association for Information Science and Technology*, vol. 57, no. 1, p. e271, 2020.

[56] J. He, P. Qvarfordt, M. Halvey, and G. Golovchinsky, "Beyond actions: Exploring the discovery of tactics from user logs," *Information Processing & Management*, vol. 52, no. 6, pp. 1200–1226, 2016.

[57] Y. Yang and P. Zhai, "Click-through rate prediction in online advertising: A literature review," *Information Processing & Management*, vol. 59, no. 2, p. 102853, 2022.

[58] Ç. Demiralp, M. S. Bernstein, and J. Heer, "Learning perceptual kernels for visualization design," *IEEE transactions on visualization and computer graphics*, vol. 20, no. 12, pp. 1933–1942, 2014.

[59] R. Hadsell, S. Chopra, and Y. LeCun, "Dimensionality reduction by learning an invariant mapping," in *2006 IEEE computer society conference on computer vision and pattern recognition (CVPR'06)*, vol. 2. IEEE, 2006, pp. 1735–1742.

[60] E. Hoffer and N. Ailon, "Deep metric learning using triplet network," in *Similarity-based pattern recognition: third international workshop, SIMBAD 2015, Copenhagen, Denmark, October 12-14, 2015. Proceedings 3*. Springer, 2015, pp. 84–92.

[61] F. Radlinski and T. Joachims, "Query chains: learning to rank from implicit feedback," in *Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining*, 2005, pp. 239–248.

[62] H. Nemlekar, N. Dhanaraj, A. Guan, S. K. Gupta, and S. Nikolaidis, "Transfer learning of human preferences for proactive robot assistance in assembly tasks," in *Proceedings of the 2023 ACM/IEEE International Conference on Human-Robot Interaction*, 2023, pp. 575–583.

[63] B. Tjanaka, M. C. Fontaine, J. Togelius, and S. Nikolaidis, "Approximating gradients for differentiable quality diversity in reinforcement learning," in *Proceedings of the Genetic and Evolutionary Computation Conference*, 2022, pp. 1102–1111.

[64] Z. Wang, C. R. Garrett, L. P. Kaelbling, and T. Lozano-Pérez, "Active model learning and diverse action sampling for task and motion planning," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2018, pp. 4107–4114.

[65] N. Robinson, B. Tidd, D. Campbell, D. Kulić, and P. Corke, "Robotic vision for human-robot interaction and collaboration: A survey and systematic review," *ACM Transactions on Human-Robot Interaction*, vol. 12, no. 1, pp. 1–66, 2023.

[66] K. Arulkumaran, M. P. Deisenroth, M. Brundage, and A. A. Bharath, "Deep reinforcement learning: A brief survey," *IEEE Signal Processing Magazine*, vol. 34, no. 6, pp. 26–38, 2017.

[67] A. Chapman, E. Simperl, L. Koesten, G. Konstantinidis, L.-D. Ibáñez, E. Kacprzak, and P. Groth, "Dataset search: a survey," *The VLDB Journal*, vol. 29, no. 1, pp. 251–272, 2020.

[68] Y. Koren, S. Rendle, and R. Bell, "Advances in collaborative filtering," *Recommender systems handbook*, pp. 91–142, 2021.

[69] K.-P. Yee, K. Swearingen, K. Li, and M. Hearst, "Faceted metadata for image search and browsing," in *Proceedings of the SIGCHI conference on Human factors in computing systems*, 2003, pp. 401–408.

[70] 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.

[71] E. Bıyık, N. Huynh, M. J. Kochenderfer, and D. Sadigh, "Active preference-based gaussian process regression for reward learning and optimization," *The International Journal of Robotics Research*, vol. 43, no. 5, pp. 665–684, 2024.

[72] R. Team, "Kuri," Aug 2018. [Online]. Available: https://robotsguide.com/robots/kuri

[73] U. Yoo, Z. Lopez, J. Ichnowski, and J. Oh, "Poe: Acoustic soft robotic proprioception for omnidirectional end-effectors," *arXiv preprint arXiv:2401.09382*, 2024.

[74] N. Dennler, C. Ruan, J. Hadiwijoyo, B. Chen, S. Nikolaidis, and M. Matarić, "Design metaphors for understanding user expectations of socially interactive robot embodiments," *ACM Transactions on Human-Robot Interaction*, vol. 12, no. 2, pp. 1–41, 2023.

[75] M. Schrum, M. Ghuy, E. Hedlund-Botti, M. Natarajan, M. Johnson, and M. Gombolay, "Concerning trends in likert scale usage in human-robot interaction: Towards improving best practices," *ACM Transactions on Human-Robot Interaction*, vol. 12, no. 3, pp. 1–32, 2023.

[76] N. S. Dennler, S. Nikolaidis, and M. Mataric, "Using exploratory search to learn representations for human preferences," in *Companion of the 2024 ACM/IEEE International Conference on Human-Robot Interaction*, 2024, pp. 392–396.

[77] Y. Ma, G. Xu, X. Sun, M. Yan, J. Zhang, and R. Ji, "X-clip: End-to-end multi-grained contrastive learning for video-text retrieval," in *Proceedings of the 30th ACM International Conference on Multimedia*, 2022, pp. 638–647.

[78] Y. Gong, Y.-A. Chung, and J. Glass, "Ast: Audio spectrogram transformer," *arXiv preprint arXiv:2104.01778*, 2021.

[79] T. Chen, H. Yin, Y. Zheng, Z. Huang, Y. Wang, and M. Wang, "Learning elastic embeddings for customizing on-device recommenders," in *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, 2021, pp. 138–147.

[80] N. Dennler, C. Yunis, J. Realmuto, T. Sanger, S. Nikolaidis, and M. Matarić, "Personalizing user engagement dynamics in a non-verbal communication game for cerebral palsy," in *2021 30th IEEE International Conference on Robot & Human Interactive Communication (RO-MAN)*. IEEE, 2021, pp. 873–879.

[81] C. J. Reed, X. Yue, A. Nrusimha, S. Ebrahimi, V. Vijaykumar, R. Mao, B. Li, S. Zhang, D. Guillory, S. Metzger, K. Keutzer, and T. Darrell, "Self-supervised pretraining improves self-supervised pretraining," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2022, pp. 2584–2594.

[82] R. Caruana, H. Kangarloo, J. D. Dionisio, U. Sinha, and D. Johnson, "Case-based explanation of non-case-based learning methods." in *Proceedings of the AMIA Symposium*. American Medical Informatics Association, 1999, p. 212.

[83] N. Papernot and P. McDaniel, "Deep k-nearest neighbors: Towards confident, interpretable and robust deep learning," *arXiv preprint arXiv:1803.04765*, 2018.

[84] B. Kim, C. Rudin, and J. Shah, "The bayesian case model: a generative approach for case-based reasoning and prototype classification," in

Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2, ser. NIPS'14. Cambridge, MA, USA: MIT Press, 2014, p. 1952–1960.

[85] L. A. Hendricks, Z. Akata, M. Rohrbach, J. Donahue, B. Schiele, and T. Darrell, "Generating visual explanations," in *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part IV 14*. Springer, 2016, pp. 3–19.

[86] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," *Advances in neural information processing systems*, vol. 26, 2013.

[87] S. Nair, E. Mitchell, K. Chen, S. Savarese, and C. Finn, "Learning language-conditioned robot behavior from offline data and crowd-sourced annotation," in *Conference on Robot Learning*. PMLR, 2022, pp. 1303–1315.

[88] S. Sontakke, J. Zhang, S. Arnold, K. Pertsch, E. Bıyık, D. Sadigh, C. Finn, and L. Itti, "Roboclip: One demonstration is enough to learn robot policies," *Advances in Neural Information Processing Systems*, vol. 36, 2024.

[89] S. Li, T. Xiao, H. Li, B. Zhou, D. Yue, and X. Wang, "Person search with natural language description," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 1970–1979.

[90] S. Bakhshi, D. A. Shamma, L. Kennedy, Y. Song, P. De Juan, and J. Kaye, "Fast, cheap, and good: Why animated gifs engage us," in *Proceedings of the 2016 chi conference on human factors in computing systems*, 2016, pp. 575–586.

[91] S. Wallis, "Binomial confidence intervals and contingency tests: mathematical fundamentals and the evaluation of alternative methods," *Journal of quantitative linguistics*, vol. 20, no. 3, pp. 178–208, 2013.