# Appendix

## Additional Demographic and Study Information

We report the additional demographic information that we collected from the customization session and the evaluation study.

### A. Robot Customization Session

**Demographics.** Participants were recruited from the local university student population through email, flyers, and word-of-mouth. A total of 25 participants were part of the study, with ages ranging from 19 to 43 (median 25); participants self-declared as men (13), women (10), and genderqueer, nonbinary, or declined to state (3, aggregated for privacy; some participants belonged to multiple groups). We recruited 13 participants who self-identified as LGBTQ+. Participants were Asian (13), Black (2), Latino (5), and White (6); some participants belonged to multiple groups. All participants were able to create signals they liked for all four categories, and all successfully interacted with the robot to collect all the items in the word search task.

**Additional Procedure Information.** The study took place in a conference room with a kitchen to reflect a realistic living environment. Participants entering the study were brought to a table in the middle of the room, with a clear view of a Kuri robot that was modified to have a screen and backpack.

First, the experimenter provided a ten-minute explanation of the study. In this explanation, participants were first introduced to the item-finding task, and were described each of the four signals in detail. The experimenter introduced the participant to the RoSiD interface and described how to use each part of the interface. Once the introduction concluded, participants were instructed to design each of the four signals in a randomized and counterbalanced order.

For each of the signals, participants were allowed to design any signal that they liked. There was no time limit, participants were able to continue customizing until they reached a finalized signal. Once they finished designing, they were instructed to tell the experimenter to move to the next signal. When all four signals were designed, the participant used the robot in it's intended use case of finding items.

The item-finding task aimed to simulate using the robot to find items while being distracted by other tasks. We achieved this by having the users engage in a word search. There were ten total words to find, but only seven of these words were listed on the actual word search. To find the final three words, users had to interact with the robot to find items around the room. When the robot returned these items to the user, the item had the word to find in the word search printed on a label attached to the item. The three items were: a stapler, a salt shaker, and a doorstop. The salt-shaker and doorstop were items that Kuri used the has-item signal for, because they were small enough to fit in the backpack on Kuri. The stapler was too big to fit in Kuri's backpack, and thus Kuri used the has-info signal to have the user stand up and walk over to the Kuri robot to pick up the item. The stapler was placed on a counter behind the participant, out of view from the table that the participant was facing.

Following the interaction, participants filled out the system usability scale. The experimenter then performed a semi-structured interview with the participant to understand their opinions on the design process. Participants then completed the study, and were compensated with an Amazon Gift Card sent to their email.

### B. Preference Evaluation Study

Participants were recruited from the local university student population through email, flyers, and word-of-mouth. A total of 42 participants were part of the study, with ages that ranged from 18 to 32 (median 24); participants self-declared as men (19), women (19), and genderqueer, nonbinary, or declined to state (4, aggregated for privacy; some participants belonged to multiple groups). There were 17 participants that self-identified as LGBTQ+. Participants were Asian (24), Black (1), Latino (7), Middle Eastern (3), and White (11) (some participants belonged to multiple groups). Participants rated their median familiarity with robotics as a 3 out of 9; a score of 1 corresponded with the term "novice" and a score of 9 corresponded with the term "expert".

## Additional Training Details

To encourage reproducibility, we provide the specifics of our training experiments.

### C. Training Feature Learning Models

We used the following encoder architectures for each modality. We used the transposed architecture for all self-supervised methods that required a decoder.

- Visual: the visual modality used a convolutional architecture that consisted of kernels with sizes: $(16, 16), (8, 8), (4, 4)$ followed by a three-layer MLP with hidden size 256. Each convolutional layer was followed by a batch norm and a leaky relu activation. Each MLP layer except the last was followed by a relu activation
- Auditory: the auditory modality used a convolutional architecture that consisted of kernels with sizes $(16, 16), (8, 8), (4, 4)$ followed by a three-layer MLP with hidden size 256. Each convolutional layer was followed by a batch norm and a leaky relu activation. Each MLP layer except the last was followed by a relu activation
- Kinetic: the kinetic modality used a recurrent architecture consisting of a bidirectional 2-layer GRU with a size 64 dimension hidden state.

We provide the pseudocode to train a network with the CLEA objective in Alg. 1. For all feature learning models we used the Adam optimizer with default learning rates. All feature learning models also used a batch size of 128. We trained a separate feature learning model for each signal that users designed for.

We selected hyperparameters for the networks using the query data collected in the robot customization study (Sec. IV) as a validation set. All our methods had three possible terms in

**Algorithm 1:** Contrastive Learning From Exploratory Actions

```
Given a list of robot trajectory datasets that all users
  saw over the course of the signal design process
  separated into explored and ignored data,
  $D = \{(\mathcal{D}_i^{ex.}, \mathcal{D}_i^{ig.})_{i=0}^{N}\}$, a learnable model that
  generates trajectory features, $\Phi$, and a hyperparameter
  for the contrastive margin, $\alpha$;
Initialize $\Phi$ to a random state (or to a pretrained
  network);
while not converged do
    $(\mathcal{D}^{ex.}, \mathcal{D}^{ig.}) \leftarrow$ sample item from D;
    // Sample anchor and positive from
       explored data
    if $Uniform(0, 1) < 0.5$ then
        $\xi_A \sim \mathcal{D}^{ex.}, \xi_P \sim \mathcal{D}^{ex.}, \xi_N \sim \mathcal{D}^{ig.}$;
    end
    // Sample anchor and positive from
       ignored data
    else
        $\xi_A \sim \mathcal{D}^{ig.}, \xi_P \sim \mathcal{D}^{ig.}, \xi_N \sim \mathcal{D}^{ex.}$;
    end
    $\mathcal{L}_1 = max(||\Phi(\xi_A) - \Phi(\xi_P)||_2^2 - ||\Phi(\xi_A) -
      \Phi(\xi_N)||_2^2 + \alpha, 0)$;
    $\mathcal{L}_2 = max(||\Phi(\xi_P) - \Phi(\xi_A)||_2^2 - ||\Phi(\xi_P) -
      \Phi(\xi_N)||_2^2 + \alpha, 0)$;
    update parameters of $\Phi$ to minimize $\mathcal{L}_1 + \mathcal{L}_2$
end
```

our loss function: the contrastive loss that we formulated, a reconstruction loss, or a KL-divergence loss comparing the batch distribution to a unit multivariate normal distribution. Only the contrastive loss and the KL-Divergence loss had tunable parameters. To select the margin for the contrastive loss, we performed a parameter sweep over $\alpha \in [.01, .1, .5, .9, 2, 5, 10.]$. We selected $\alpha = .1$ for the visual modality, $\alpha = .1$ for the auditory modality, and $\alpha = 2$ for the kinetic modality. To select the regularization term for the VAE, we performed a parameter sweep over $\beta \in [.01, .1, .5, .9, 2, 5, 10.]$. We selected $\beta = 1$ for the visual modality, $\beta = 10$ for the auditory modality, and $\beta = 10$ for the kinetic modality.

### D. Reward Training

We learned two forms of reward functions to evaluate CLEA. The first was a reward network that we evaluated with predicted choice accuracy, similar to Bobu et al. [4]. The second form of reward was a linear transformation of the features as in Bıyık et al. [8].

**Neural Network Reward.** We used the same reward network for all modalities. The network takes as input a $d$-dimensional vector. The network itself consists of two fully connected layers, each with 256 hidden units. Each layer is followed by a ReLU nonlinearity. We trained the network using cross entropy loss (Eq. 6).

We additionally encouraged the learned reward to not be too large by placing a L2-norm regularization on the predicted rewards, with a weight of .01 for all reward networks. We trained all reward networks for 60 epochs using the default settings for the Adam optimizer and a batch size of 16.

**Linear Reward.** The linear reward models as user's reward function as $R_H(\xi) = \omega \cdot \Phi(\xi)$, where the user's specific preference is represented by $\omega$. We learned a user's $\omega$ through pairwise choices. We adopted the Bradley-Terry preference model to model the probability of a user choosing $\xi_k$ from a query $Q = \{\xi_0, \xi_1, ...\xi_N\}$:

$$P(\xi_k|Q, \omega) = \frac{e^{\omega \cdot \Phi(\xi_k)}}{\sum_{i=0}^{N} e^{\omega \cdot \Phi(\xi_i)}} \tag{7}$$

To learn $\omega$, we apply Bayes' rule.

$$P(\omega|Q, \xi_k) \propto P(\xi_k|Q, \omega) \cdot P(\omega) \tag{8}$$

We assume a prior $\omega$ of a uniformly distributed unit ball, as in prior works [9]. We update our posterior after every observed choice to estimate the user's $\omega$.

## Extended Statistical Analysis

We report the full statistical analysis that we performed across algorithms to report effect sizes so that others may use these for power analyses.

### E. Completeness

We used a repeated measures ANOVA to evaluate completeness for each modality, using accuracy as the dependent measure and algorithm as the within-subjects factor. All assumptions were met. The choice of algorithm is significant for all modalities (visual: $p < .001, \eta^2 = .582$; auditory: $p < .001, \eta^2 = .532$; kinetic: $p = .008, \eta^2 = .247$). We performed post-hoc analysis using paired t-tests with a Bonferroni correction. Our analysis revealed that CLEA and CLEA+AE outperformed all other algorithms in the visual modality (all $p_{corr.} < .05$) and CLEA+AE outperformed all algorithms in the auditory modality (all $p_{corr.} < .05$). There were no significant differences between algorithms in the kinetic modality; Random, CLEA, and CLEA+AE empirically performed the highest.

### F. Simplicity

We performed a two-way repeated measures ANOVA with dimension and algorithm as within-subjects factors and AUC as the dependent measure. The feature space dimension and the feature learning algorithm were the within-subjects factor. We found that the choice of algorithm was significant for all modalities (visual: $p < .001, \eta^2 = .909$; auditory: $p < .001, \eta^2 = .502$; kinetic: $p < .001, \eta_p^2 = .530$). We used pairwise t-tests with Bonferroni corrections to assess all pairwise comparisons between the algorithms for learning features.

For the visual modality, we found that CLEA+AE features were the best on average, significantly outperforming all other

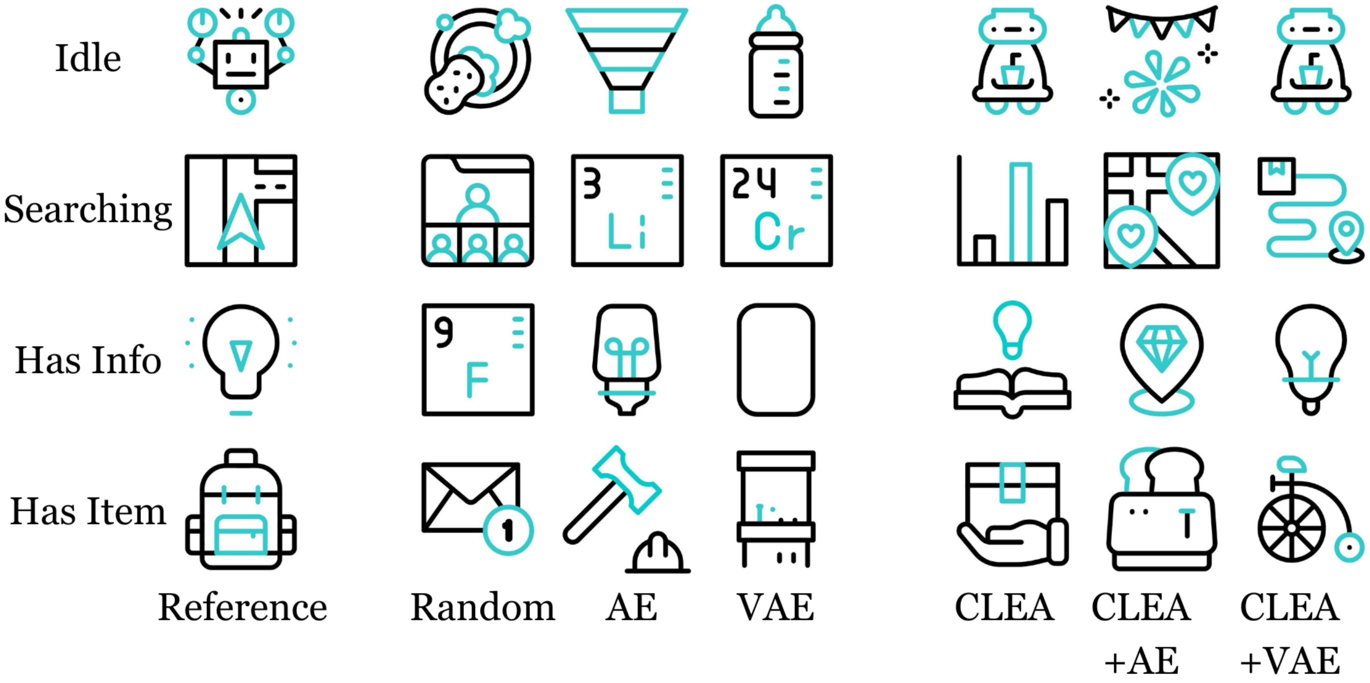

Fig. 8. Qualitative results. The leftmost image shows a reference image that users actually selected when designing signals for the robot. The other images show the next most similar image in the embedding space for each method. CLEA-based methods show more semantic similarity to the reference image than self-supervised approaches.

features across feature space dimensions (all $p_{corr.} < .001$). For the auditory modality, CLEA+VAE features were the best on average, significantly outperformed all algorithms across feature space dimensions (all $p_{corr.} < .001$). For the kinetic modality, CLEA features outperformed all algorithms across feature space dimensions except AE features (all other $p < .001$), however we note that CLEA has much higher performance than AE for lower-dimensional feature spaces.

### *G. Minimality*

We performed a repeated measures ANOVA using AUC as a dependent measure and the algorithm as the within-subjects factor and determined that the choice of algorithm was significant (visual: $p < .001, \eta^2 = .776$; auditory: $p < .001, \eta_p^2 = .685$; kinetic: $p < .001, \eta_p^2 = .658$) we used paired t-tests with bonferroni correction to assess all pairwise comparisons. In the visual modality, CLEA+AE significantly outperformed all other algorithms (all $p_{corr.} < .001$). In the auditory modality, CLEA+VAE significantly outperformed all other algorithms (all $p_{corr.} < .001$). In the kinetic modality CLEA outperformed all other algorithms (all $p_{corr.} < .003$).

### *H. Explainability*

We evaluated significance using a binomial test with continuity corrections [91] to determine if users selected CLEA-generated signals as their favorite signal significantly more often than random chance Participants selected the behaviors generated using CLEA features 16 times—significantly more often than chance, $p < .001$. All other algorithms showed no significant differences from random chance.

## Additional Experiments

We performed three additional experiments to evaluate CLEA. First, we qualitatively examined the structure of the feature space; second, we examined each method's robustness to injected noise; and third, we examined directly learning the user's reward from from the raw robot behaviors without learning a feature space.

### *I. Qualitative Results*

To illustrate the types of embeddings CLEA learns compared to the self-supervised approaches, we present examples from the visual modality in Fig. 8. We selected the most similar image based on the cosine similarity of the embeddings. We show that the embeddings learned by CLEA show more semantic similarity qualitatively than self-supervised models. These images were selected using the 8-dimensional embeddings for all models.

For the idle behavior, self-supervised approaches show similar structural composition to the image of a robot, but CLEA methods show other robots or show similar flashing motifs. For the searching behavior, self-supervised approaches show similar square compositions to the reference image, but CLEA methods maintain map-like images. For the has information signal, the autoencoder recovered a similar lightbulb, whereas CLEA embeddings show other information-related items like books, or a pin identifying where an object is. For the has item signal, self-supervised approaches are unrelated to the idea of possessing an item, whereas CLEA methods show containers.

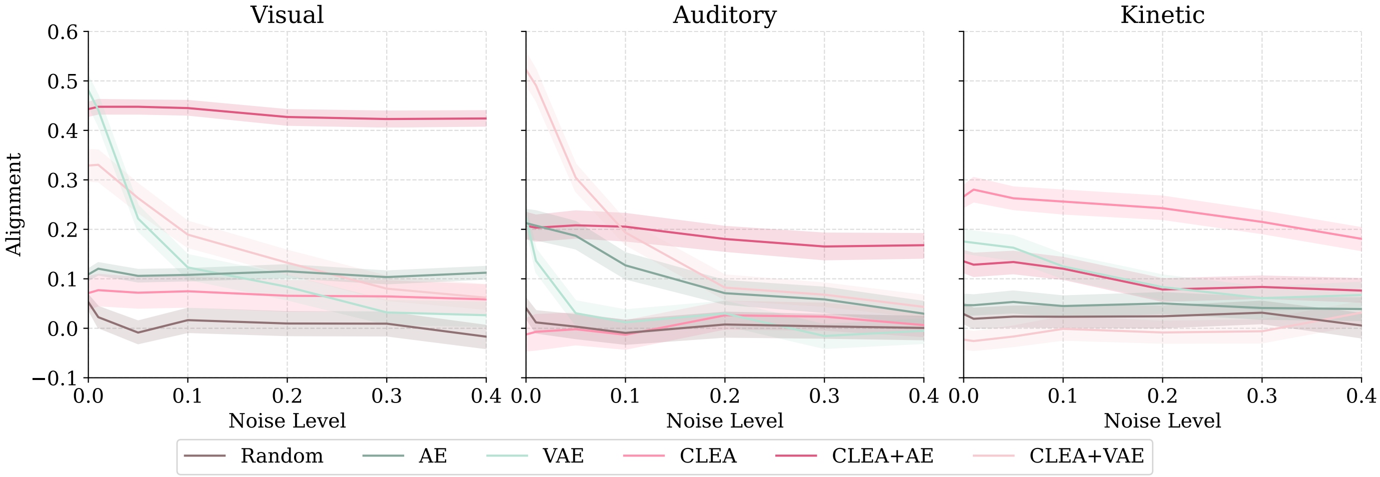

Fig. 9. Robustness results. Accuracy of a linear reward model across different levels of injected noise. Features using CLEA maintain higher performance across different noise levels. We find that CLEA+AE is the least sensitive to noise overall.

### *J. Robustness to Noise*

To evaluate robustness to noise, we again adopted a simple linear model, $R_H(\xi) = \omega \cdot \Phi(\xi)$. We estimated $\omega$ using bayesian inverse reward learning as in previous works [8, 9, 11]. When learning from the observed user queries, we add noise from a uniform Gaussian to simulate inaccuracies in features for new robot behaviors.

$$\Phi(\xi)' = \Phi(\xi) + \epsilon \cdot \mathcal{N}(0, 1) \tag{9}$$

where $\epsilon$ represents a scaling factor of the noise. We evaluated final alignment after 100 queries using these modified features to assess robustness to noise. We used 8-dimensional features, modified the noise parameter with the values $\epsilon \in [0, 0.01, 0.05, 0.1, 0.2, 0.3]$, and averaged over 60 trials for each participants to control for random effects. The results are illustrated in Fig. 9.

In the visual modality, we found that CLEA+AE was the most performant across all noise levels. In the auditory modality, CLEA+AE was the least sensitive to noise, but CLEA+VAE was more robust to noise level than a normal VAE, indicating that CLEA increases robustness. In the kinetic modality, CLEA was the most performant across noise levels.

### *K. Robustness to Sample Weighting*

Participants engaging in exploratory search may make more meaningful exploratory actions as they get closer to their end-signal. To evaluate if this is important in the learning process, we performed an additional experiment that weights exploratory actions at the end of the signal design process as more important than exploratory actions taken at the beginning of the signal design process. To weight the samples, we used the following equation based on the index $i$ of the sample ordered by time, and the total number of samples, $N$:

$$w(i) = \frac{i}{N} \tag{10}$$

We trained each of the CLEA algorithms again, multiplying the loss for each datapoint by this additional weight value. We compared the weighted models, CLEA (weighted), CLEA+AE (weighted), CLEA+VAE (weighted), with the unweighted models, CLEA, CLEA+AE, CLEA+VAE, for each of the four criteria: completeness, minimality, simplicity, and recoverability. We used the same evaluation processes as described in Sec. V-D.

TABLE II
COMPARISON OF COMPLETENESS, MEASURED WITH TEST PREFERENCE ACCURACY (TPA), FOR UNWEIGHTED AND WEIGHTED TRAINING. THERE WERE NO SIGNIFICANT DIFFERENCES IN TPA FOR ANY MODALITY OR METHOD (ALL P-VALUES $>$ .05).

| | | TPA<br>(unweighted) | TPA<br>(weighted) | p-value<br>(unc.) |
|---|---|---|---|---|
| Visual | CLEA | .955 | .949 | .553 |
| | CLEA+AE | .974 | .973 | .911 |
| | CLEA+VAE | .810 | .841 | .383 |
| Auditory | CLEA | .887 | .933 | .063 |
| | CLEA+AE | .949 | .935 | .295 |
| | CLEA+VAE | .800 | .767 | .429 |
| Kinetic | CLEA | .976 | .973 | .789 |
| | CLEA+AE | .979 | .973 | .525 |
| | CLEA+VAE | .936 | .934 | .904 |

**Completeness.** To evaluate the effect of completeness, we used pairwise t-tests to assess differences in TPA between the weighted and unweighted models. The results are shown in Table II. We observe no significant differences between any method or modality, indicating that there is no effect on feature completeness when reweighting samples.

**Minimality and Simplicity.** To evaluate the effect of minimality and simplicity, we present the results of AUC Alignment after 100 simulated pairwise queries. All values are significant because we can re-run simulations as many times as necessary, so we evaluate differences between the two algorithms by counting how many times the weighted version

TABLE III

Comparison of AUC Alignment across unweighted and weighted variants of the CLEA algorithms. Numbers that perform better are in bold. We find that there were 22 of 45 trials where the unweighted CLEA performed the best, and 23 of 45 trials where the weighted CLEA performed the best. These values do not significantly differ from random chance ($p > .05$).

| | | AUC Alignment (unweighted) | AUC Alignment (weighted) | AUC Alignment (unweighted) | AUC Alignment (weighted) | AUC Alignment (unweighted) | AUC Alignment (weighted) | AUC Alignment (unweighted) | AUC Alignment (weighted) | AUC Alignment (unweighted) | AUC Alignment (weighted) |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | Dimension | 8 | | 16 | | **32** | | **64** | | **128** | |
| Visual | CLEA | -.011 | **.011** | .288 | **.386** | **.304** | .239 | **.187** | .116 | **.118** | .054 |
| | CLEA+AE | .440 | **.506** | .411 | **.459** | .380 | **.418** | .298 | **.333** | **.390** | .341 |
| | CLEA+VAE | .257 | **.449** | .300 | **.422** | .379 | **.433** | **.317** | .222 | .156 | **.231** |
| Auditory | CLEA | .075 | **.119** | -.078 | **.177** | **.166** | .063 | **.155** | -.027 | **.101** | -.071 |
| | CLEA+AE | .337 | **.373** | .005 | **.225** | .189 | **.275** | .018 | **.087** | .229 | **.253** |
| | CLEA+VAE | **.524** | .279 | **.347** | .273 | **.303** | .156 | **.206** | .196 | **.176** | .141 |
| Kinetic | CLEA | **.289** | .265 | .329 | **.350** | .295 | **.310** | **.416** | .403 | **.405** | .335 |
| | CLEA+AE | .092 | **.266** | .209 | **.288** | **.309** | .272 | **.396** | .351 | **.335** | .320 |
| | CLEA+VAE | .000 | **.117** | **.265** | .178 | .161 | **.244** | **.400** | .379 | **.423** | .276 |

outperforms the weighted version for each trial. We use a binomial test to determine if the count of trial wins is significantly different from random chance, i.e., 50%. For **simplicity**, we examined AUC Alignment across all dimensions, and found that unweighted CLEA models outperform weighted CLEA models in 22 of 45 trials, which is not significantly different than random chance ($p = .500$). For **minimality**, we examine just the smallest feature space dimension. We find that unweighted CLEA outperforms weighted CLEA in 2 of 9 trials, which is not significantly different than random chance ($p = .0912$).

TABLE IV

Explainability

| | | Similarity (unweighted) | Similarity (weighted) | p-value (unc.) |
|---|---|---|---|---|
| Visual | CLEA | **.239** | .225 | **.003*** |
| | CLEA+AE | **.169** | .140 | **.001*** |
| | CLEA+VAE | .136 | **.141** | .486 |
| Auditory | CLEA | .248 | **.286** | **.001*** |
| | CLEA+AE | .250 | **.253** | .835 |
| | CLEA+VAE | .193 | **.259** | **.001*** |
| Kinetic | CLEA | .203 | **.204** | .896 |
| | CLEA+AE | .198 | **.202** | .213 |
| | CLEA+VAE | **.307** | .281 | .188 |

**Explainability.** To evaluate the effect of weighted training on explainability, we calculated 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) for both the weighted and unweighted CLEA variants. We present the results in Table IV. We found that there were only four significant differences across the three modalities. In two of the four differences, the unweighted version of CLEA showed a higher similarity, but this does not differ from random chance according to a binomial test ($p = .500$). Notably, the two instances where the unweighted CLEA training performed best were both in the Visual modality, and the two instances where weighted CLEA training performed best were in the Auditory modality. While we cannot draw any strong conclusions, these results may indicate that users explore different modalities in different ways. Future research may investigate reweighting strategies that capture these differences.

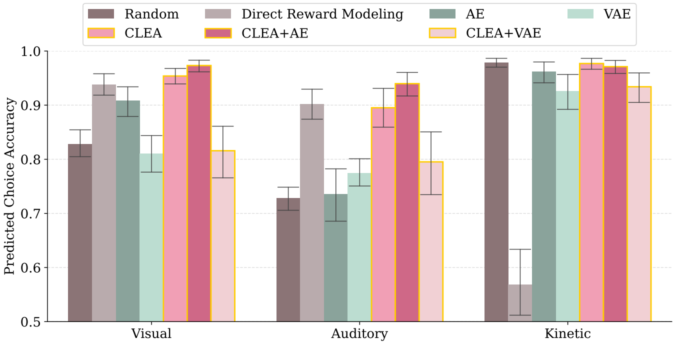

Fig. 10. Direct reward modeling results. We show the recoverability across the different methods of learning rewards, including directly learning rewards. We find that CLEA+AE outperforms all self-supervised methods and direct reward modeling.

**Summary.** Across the four evaluation criteria, we found that there was no clear benefit for using unweighted or weighted sampling techniques. This highlights that CLEA works well without additional engineered training techniques. This underscores the simplicity of this algorithm to leverage information from users' exploratory behaviors without having to explicitly model the user's search process.

### L. Comparison with Learning Rewards without Features

To validate that learning features is useful, we tested learning a user's reward function from raw inputs. While this is not scalable as each user is required to first perform ten ranking tasks, this approach is highly expressive because it uses a large network that take the raw data structures as input. We used the same approach as in Sec. D. The only change we made was to update the feature's model's weights during training. We started with a feature learning network with randomly initialized weights, compared to the rest of our evaluations, where the weights of the feature network were frozen. We call this method "Direct Reward Learning" and show the results in Fig. 10

Direct reward modeling showed improvement over self-supervised approaches, except for the kinetic modality. This modality is highly prone to over-fitting, and with the small individual preference datasets, direct reward modeling is not able to learn a generalizable reward function for the user. In the visual modality, CLEA+AE achieved a mean accuracy of .973, compared to .938 for Direct Reward Modeling. In the Auditory modality, CLEA+AE achieved .940 compared to .902 for Direct Reward Modeling. In the Kinetic modality, CLEA+AE achieved .971 compared to .569 for Direct Reward Modeling. This result underscores the utility of using human-generated data to learn features to both facilitate downstream preference learning and more easily scale to large numbers of users.