Learning Task Preferences from Real-World Data

Daphne Chen

CMU-RI-TR-23-49

July 27, 2023

The Robotics Institute
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA

Thesis Committee:
Reid Simmons, *chair*
Henny Admoni
Jean Oh
Michelle Zhao

Submitted in partial fulfillment of the requirements for the degree of Master of Science in Robotics.

To my parents, my greatest source of inspiration.

Abstract

In order to provide personalized assistance that is capable of adapting to the needs of unique individuals, it is necessary to understand peoples' preferences for different tasks. Robot assistance often assumes a static model of the individual, while in the real world, people have different capabilities and needs that may change over time. Learning an individual's task preferences enables the agent to detect when the individual has deviated from their usual behavior, and subsequently understand how to proactively provide assistance when needed. Our work proposes an approach to learn peoples' preferences for commonplace real-world meal preparation tasks from few demonstrations. We provide two learning methods – mixture-of-experts and meta learning – that condition a model on an individual's preferences and determine the next step towards completing the task sequence. We evaluate our methods in an in-person user study and data collection with a diverse population of users and real-world kitchen environment on two different tasks. The results highlight the importance of incorporating a representation of users' implicit preferences into personalized predictive models of their behavior.

Acknowledgments

The list of people to whom I owe the utmost gratitude is too long to enumerate here, but I would be remiss to not begin it with those who have supported me most since the beginning – to my parents, thank you for everything.

I also express my sincere thanks to my advisor, Reid Simmons, whose thorough questioning was key to making this thesis possible. I am very grateful to my other committee members, Henny Admoni and Jean Oh, for bringing their different perspectives to this work. To my undergraduate advisor, Sonia Chernova, thank you for sparking my interest in robotics and encouraging me to consider graduate school.

I am incredibly fortunate to have had incredible collaborators and mentors, especially Michelle Zhao for providing constant support on every front – from technical to emotional and many other aspects in between. Special thanks as well to Gavin Zhu for contributing to a fun class project that shaped the earliest version of a user study for this work.

To my MSR cohort: I have countless fond memories of the past two years together, and I will miss every one of you beyond words. Special thanks to a few particular people who kept me sane: Mateo Guaman Castro, Swapnil Pande, Rohan Deshpande – thank you all. The Taco Tuesdays, late nights in TCS, weekly trips to Ascend, and spontaneous jam sessions will forever remain in my memory.

To the broader AI-CARING community, thank you for supporting this work and for providing many avenues for real-world impact and collaboration.

I also want to thank Yonatan Bisk for the fruitful discussions around formalizing preference learning, and to helping me make sense of ambiguous future plans.

Thanks as well to members of the broader CMU HRI community – Aaron Steinfeld and Liz Carter, your help with navigating IRB was crucial to making this work possible!

Funding

This work was supported by the AI-CARING Institute (NSF grant number 53668.1A.1043222).

Contents

When this dissertation is viewed as a PDF, the page header is a link to this Table of Contents.

List of Figures

List of Tables

Chapter 1

Introduction

Imagine a scenario where you come back home after a long day at work, head to your kitchen, and start to prepare dinner for yourself. You reach for a box of pasta to make one of your usual meals, and a robot assistant is already preparing a pot of water to boil on the stove. Later, after the meal is cooked, the robot has set the dinner table exactly where you usually sit, in a manner that's just how you like it. This imagined scenario illustrates a future that we hope is possible for intelligent robot assistants – one where seamless interaction in line with our preferences is commonplace and natural.

In order for intelligent agents to effectively and intuitively interact with humans in the real world, it is necessary for them to be able to act in line with individuals' unique preferences. Of particular importance are assistive applications, where understanding an individual's typical habits, behaviors, and preferences is essential to (1) personalize a model to reflect peoples' varying needs and abilities or (2) determine when critical mistakes or deviations occur.

Preferences are as diverse and complex as the humans who exhibit them – for example, they can range from discrete attributes such as deliberately choosing a certain object over another, or be as nuanced as someone's current mental state/mood. Thus in this work, we constrain the definition of preferences to *temporal* ones, meaning an individual's chosen order of actions.

In recent years, preference learning has become an exciting new area of research, with a number of efforts dedicated to learning from human feedback [6], preference inference for assistance [32], and learning representations for implicit preferences [3]. However, many of these approaches are not grounded in the real world, and test on toy tasks and environments that have limited nuance or range in individuals' preferences. They also often involve repeatedly querying the user, or require the user to assign a score to an agent's actions, both of which can be noisy, low signal, or unintuitive [4].

To address these gaps, we present a method to learn peoples' preferences from few demonstrations by clustering them from a latent space via an autoencoder, and pair this with a mixture of experts algorithm to predict the user's next action in a manner that is conditioned on their preference type. We additionally employ a meta learning model to adapt to new preference types. Instead of continually seeking feedback from the user, we aim to learn preferences implicitly by observing the human carrying out the task in a natural setting, such as their own home or kitchen.

Furthermore, it is challenging to find task datasets in human-centric, real-world environments. Such datasets are often small, lack annotations, or have limited instances per person – thus making it challenging to learn individuals' preferences. To address these issues with lack of viable datasets, we present a two-fold solution: firstly, we provide a data augmentation approach to generate large-scale simulated data that reflects an underlying probability distribution seen in real-world data. Secondly, we contribute a large-scale, real-world dataset of 17 people performing 2 different meal preparation tasks, annotated with actions, timestamps, and including self-reported preferences. The resulting 170-video dataset will be provided as an open-source contribution.

To address the limitations in existing methods for preference learning, we present two contributions. We first develop a method to learn implicit preference types via an autoencoder. Secondly, we propose two modeling approaches – a mixture-of-experts and a meta-learning algorithm – to respectively learn a preference-based model and enable adaptation to new preference types. Each component is evaluated on both the real-world dataset and a larger-scale simulated dataset of diverse task demonstrations.

We find that the autoencoder enables us to learn human-interpretable preference groups in both datasets. Furthermore, our results suggest that a model conditioned on preference type via mixture-of-experts is able to achieve better predictive accuracy than the baseline model. Lastly, we highlight suggestions for promising future applications in assistive technology and human-robot interaction.

1.1 Learning Preferences for Personalized Assistance

In order for assistive agents to effectively operate with humans, it is important that they are capable of learning in a manner that aligns with the preferences of unique individuals. Furthermore, to be practical in the real world, assistive agents should be able to learn from few initial observations so as to minimize the burden on the user. Thus we first describe a technique to learn preferences in an unsupervised manner from real-world human data, and subsequently propose a method that learns preferences via a mixture-of-experts or meta learning algorithm.

Understanding individual preferences is key to providing assistance to a diverse population of users, especially in highly sensitive applications such as caregiving. For example, consider a scenario in which a robot must learn when to intervene if a user performs an unexpected action within a daily routine. In such a setting, it is not always appropriate to assume that the optimal behavior is the human's desired behavior [5], especially if the optimal model is static, and thus cannot adapt to the human's preferences. However, learning a model for assistance that incorporates the user's unique preferences and abilities remains a challenging problem. We focus on household tasks in order to constrain the definition of task preferences and ground this work in a natural setting for user assistance.

1.2 Learning Preferences to Monitor Health

The population of older adults worldwide is rapidly increasing, and according to the World Health Organization, is expected to reach 78 million individuals by 2030 [19]. In order to support this, one promising avenue is to develop technology that enables older adults and their caretakers to maintain independent lives in their own homes by providing assistance that detects changes in their behavior.

Among older adults, those who experience mild cognitive impairment (MCI) face more challenges in performing household tasks independently. For example, people with MCI may find difficulty in remembering the next step of a meal preparation task, or make critical mistakes that endanger themselves or their caretakers. Such mistakes such as forgetting to turn off a stove or neglecting appliances could be more easily noticeable if there is an accurate model of the person's typical routines and preferences within their home.

1.3 Real-World Datasets

There are a number of simulators for realistic environments, such as Isaac Sim, AI2-THOR, and Virtual Home, but these present setbacks for learning preferences in a human-centric manner. Although Isaac Sim contains the ability to simulate humans, it lacks human-centric environments such as households or kitchens [18]. AI2-THOR has a variety of human-centric environments to choose from – including kitchens, bedrooms, bathrooms and living rooms – but it contains a limited action space for interacting with the environment [16]. Virtual Home lacks a first-person interface, which is crucial to support interaction with the simulator in a user study setting [21]. Additionally, each of these simulators experience the general issue of low fidelity in comparison to real world environments. These factors motivate the need to carry out real-world evaluation especially when human-centric factors, such as individuals' preferences, are a key component.

However, the majority of real-world datasets and user studies will still face the problem of *scale* of data. Thus this work additionally includes an approach to generate simulated data at larger scale than previously described datasets. The resulting simulated dataset aims to be an approximation to augment the underlying real-world task dataset.

1.4 Research Contributions

- We provide a method for generating large, customizable quantities of simulated data that approximates real-world data, and can be used to augment existing datasets.
- We run an in-person data collection initiative, resulting in a real-world task

dataset comprised of 17 individuals performing 2 meal preparation tasks over 5 instances each. We plan to release the resulting dataset of 170 videos as an open-source contribution. Data from each individual is annotated with their self-reported preferences for ground truth evaluation.

- We present an approach for learning implicit preference types via an autoencoder combined with clustering on the latent space.
- We develop two models: (1) a mixture-of-experts model conditioned on preference type and (2) a meta learning model that learns over a broad distribution of different preference types, and evaluate the predictive accuracy of each algorithm on simulated and real-world data.

Chapter 2

Background

2.1 Defining Preferences

The concept of preferences is broad, and many current definitions of preferences are highly task-dependent [29]. Generally, preferences can be thought of as an individual's usual manner of performing a task within the set of possible task execution types.

2.2 Types of Preferences

Object-Based Preferences This category covers an individual's preferences for specific objects in the environment. This may include particular task-dependent tools, utensils, or ingredients that are available to the individual.

Example: someone may prefer to use chopsticks instead of a spatula to cook scrambled eggs.

Action-Based Preferences This category encompasses an individual's preference for taking a specific action out of a set of other viable actions that would also complete the task.

Example: someone may prefer to do their daily commute by biking instead of taking public transit.

Style-Based Preferences This category includes personality-based stylistic tendencies for a task.

Example: someone may drive more aggressively than others in the general population.

Temporal Preferences In a sequential or hierarchical task, an individual may prefer to perform certain steps in a particular order. These are perhaps best demonstrated in longitudinal interactions, such as day-long household routines.

Example 1: someone may prefer to add peanut butter before jelly when making a PB&J sandwich, instead of vice versa.

Example 2: (in the context of assistive robotics) someone may usually make their coffee before eating breakfast; thus making coffee may be indicative of the robot proactively retrieving breakfast ingredients.

2.3 Preference Learning

Preference learning refers to the concept of computing a representation of an individual's default manner for performing a task. By learning a representation that reflects peoples' preferences, we propose that it is possible to also predict their actions in a manner that is more accurate than a general-purpose baseline model for the task.

However, we acknowledge that the definition of preferences is very broad, and defining preferences as the subject of a learning problem is a challenging topic in itself. We dedicate a section to an ontology of other preference types that may be applied to future work in Section 2.2.

Among the many different types of preferences, in this work we choose to focus on preferences as *temporal* patterns of behavior, i.e. the predominantly-chosen order of the user's actions while performing a step-by-step task, such as a daily routine or recipe. Although the other types of preferences are also important to study, we select temporal preferences due to the intended downstream applications of this work and the nature of publicly-available datasets, which are further described in Section 2.6.1.

2.3.1 Related Work

Previous approaches to preference learning use trajectory preference queries for feedback, and often have low signal to noise ratio, thus requiring several hundred to several thousand trajectory segments per task per human evaluator [6, 28]. Some works have tried to address this by optimizing for adaptation by incrementally updating learned preferences via maximum a posteriori (MAP) estimate, inferred from physical human perturbations [2]. However, this approach also requires repeated querying of the human.

Other methods use a similarity-based approach to determine which of two compared trajectories an individual prefers [3]. However, this approach uses binary comparisons – i.e., the user is presented with two different instances of how the task may be performed and asked to select which one they prefer. This provides sparse information with limited nuance, and also places burden on the user to continually specify their preferences.

Additionally, past works on preference learning have been tested using games, toy problems, or in simulation, which limits the types of preferences and intuitiveness for the participant [31]. Work by Fitzgerald et al. suggests that different query types such as demonstrations, corrections, or preference queries have varying levels of utility for getting feedback from people in interactive settings [9].

2.4 Mixture-of-Experts

A mixture-of-experts (MoE) model consists of a number of ”experts”, each of which is a different network, combined with a belief weighting mechanism which is used to update the model and subsequently select an ”expert” to process each input. This method was first introduced in 1991 by Jacobs et al. as a supervised learning procedure that learns how to handle a subset of a complete set of training cases [14]. MoE can be thought of as a divide-and-conquer approach to learning, which has been noted to have advantages over deep learning methods in that it is interpretable and less computationally complex [15].

2.4.1 Related Work

Prior work has shown that by transforming action sequences into a lower-dimensional space, one can recognize strategies without any prior knowledge of the possible strategy types – a technique also known as strategy matching [33].

In this work, we apply strategy matching to a new task domain, and additionally extend upon prior work by using an autoencoder instead of a Hidden Markov Model in order to learn preference types.

2.5 Meta Learning

Meta learning is a machine learning method where a model learns a better initial parametrization given multiple training samples, also known as ”learning to learn” [8]. It was first introduced in 1987 by Schmidhuber as self-referential learning, where a model receives its own weights as inputs and predicts an update for these weights [23]. In the meta learning paradigm, a model is able to leverage all training data and is optimized to adapt with few inputs, a method also known as few shot learning [20, 27]. By using meta learning, we aim to train a single model that is sensitive to changes in the input such that few gradient updates can more substantially correct predictions in the direction of gradient loss. We apply this technique to develop a model optimized for rapid adaptation in order to improve personalized predictions for assistance in household tasks.

2.5.1 Related Work

Prior work in meta learning has focused primarily on classification problems such as multi-class image recognition, object detection, and segmentation [13]. The introduction of model-agnostic meta learning (MAML) [8] has increased the applications for meta learning towards other domains such as regression and reinforcement learning problems. In this work, we use MAML to formulate our approach as a classification problem with the action space of the task as the different classes that the model can predict.

2.6 Real-World Datasets

In this work, our criteria for choosing a dataset was that it must contain several instances of one task, rather than a few instances of many tasks, in order to place the emphasis on learning user preferences within a single activity. Additionally, it must contain multiple instances per individual, in order to evaluate preference alignment. We specifically looked for datasets that encompass meal preparation tasks because recipes typically follow a predictable structure in their number and order of steps. Despite this, many meal preparation tasks can also be completed with sufficient variation such that distinctive temporal preferences may arise without diverging from the end goal. For example, within a recipe for preparing vegetable stew, an individual may prefer to peel all vegetables, then cut all vegetables, and finally add them to the pot together; others may prefer to process each ingredient individually, i.e. first peel/cut/add the carrots, then peel/cut/add the potatoes, and lastly peel/cut/add the onions.

For these reasons, we chose the 50 Salads Dataset [25] as the basis for one of the tasks. The 50 Salads Dataset contains RGB-D videos of 25 people performing 2 instances of the same salad-preparation task, yielding a total of 50 videos. The videos include timestamped annotations, accelerometer data, and depth maps. To create the initial dataset for this study, we extracted the raw text annotations for each video and used these to represent each instance of the task.

However, a limitation of this dataset is its small size relative to datasets typically used for training models. This problem is not unique to this domain, as labeled real-world data is often difficult to find, expensive to collect, or time-consuming to annotate. Thus in the following chapter we outline our method for addressing this by developing a generative model of activity sequences to yield a larger, simulated dataset for the same task.

2.6.1 Related Work

Most of the relevant datasets that we surveyed optimize for breadth of the data rather than depth. For example, the Ego4D dataset is large at 3670 hours of video, but it lacks multiple task instances from single individuals, thus is not ideal for learning someone’s preferences [11]. Other household task datasets such as Carnegie Mellon University Multimodal Activity (CMU-MMAC) [17] and EPIC-KITCHENS [7] also have this limitation.

Others face the issue of lacking ground truth labels for evaluating a model. For example, the YouCook2 dataset contains videos sourced from YouTube including 89 different recipes with an average of 22 videos per recipe, but the videos are not annotated timestep by timestep [34]. Finally, other task datasets, such as ”Something-something”, only have short snippets of an activity, such as an instance of someone picking up a mug or closing a door, rather than a full end-to-end sequence of someone performing a household routine from start to finish [10].

Chapter 3

Methodology

3.1 Overview

In this section, we describe two categories of contributions: data-based and modeling-based approaches. Firstly, we present a method for augmenting data and generating a simulated dataset, in order to address some of the issues around lack of large, viable datasets for preference learning. Secondly, we detail our collection of a real-world task dataset comprised of 17 individuals performing two different meal preparation tasks in a real kitchen.

In our modeling-based approaches, we describe a representation learning method that uses an autoencoder to identify different preference types within a task dataset. Lastly, we present two different learning approaches – mixture-of-experts and meta learning – for action prediction that is conditioned on preference type.

3.2 Creating a Simulated Dataset

Real-world annotated data is notoriously difficult, time-consuming, and expensive to collect, even more so for human-centric task data. To mitigate this problem, we formulate a method to generate simulated data based on the sequences seen in an underlying task dataset. As the first step to generating simulated data, we formulate a Markov-model based approach to approximate the transition probability distribution of the original dataset.

We use the 50 Salads Dataset as the base dataset in this work. This dataset contains 50 instances of annotated videos where 25 people prepare 2 salads each. Before further using the data, we apply initial preprocessing in order to format the sequences in a consistent manner for the simulated data generation algorithm. We extract the annotations corresponding to the low-level activity for each sequence in the dataset, merge the pre-, core-, and post- phases into a single annotation, and eliminate adjacent duplicate annotations. The preprocessed data was organized into a dictionary of sequences mapped to participant ID and formatted into a JSON file.

In order to closely reproduce the natural variation seen within the original dataset and in the real-world population, we approach the simulated data generation problem through the lens of a Markov process. We first create a probability matrix based on the transition counts between each possible pair of annotations, or actions, in the sequences of the original dataset. The transition probability at the (i, j)-th index represents the probability of performing action j following action i. These counts are then normalized so that the corresponding probabilities for each annotation sum to 1.

Using the action transition matrix, sequences are probabilistically generated via sampling such that they are complete (i.e., achieve the task) and valid. To enforce the validity of each generated sequence and avoid extraneous repetition, we use a constraint tree to eliminate implausible transitions (e.g. mixing an ingredient before it has been added to the bowl). Our task-based constraint tree allows for incomplete or invalid sequences to be pruned, while including probabilistically unlikely sequences to account for a broad range of preference types in the simulated dataset. High-level pseudocode for this algorithm is provided in Algorithm 1.

Using this method, we create an augmented dataset containing simulated sequences for the salad preparation task, which serves as our augmented training set for the subsequent learning algorithms. Section 3.4 details the analysis of the dataset to identify patterns of preferences within these tasks.

Additionally, it is important to note that this approach produces sequences that are not seen in the original dataset. Furthermore, because of the constraints that we apply within the simulated dataset generational algorithm, the task's probability distribution of is shifted from that of the underlying data. We mention these limitations as a consideration for future applications of this work. Due to the inherent limitations of simulated data, we also collect a real-world dataset described in Section 3.3 that addresses the drawbacks (detailed in Section 2.6.1) of other datasets for preference learning.

Algorithm 1 Simulated dataset generation algorithm

```
Assume: Transition probability matrix $T$ from dataset $D$ with annotation set $A$
sequence = [start_token]

while end of sequence not end_token do:
    next_a = randomly sample $A$ weighted by $T$
    if next_a not valid, randomly sample until next_a is valid
    append next_a to sequence
return sequence
end while
```

3.3 Real-World Dataset Collection

As previously described in Section 2.6.1, it is challenging to find high-quality existing task datasets in human-centric environments. Despite starting with the 50 Salads Dataset, we found that this dataset was not ideal for learning preferences at the individual user level, because it contains only two instances per participant. We ran our own real-world data collection initiative with 17 participants performing 2 different meal preparation tasks in a real kitchen.

Our goal was for the tasks to encompass something familiar such that most people would have prior knowledge and preferences for the task, and be able to perform the task naturally with minimal instruction. We also chose the tasks to be simple enough for a 1-hour session per participant, yet complex enough to possess innate variation in personal preferences.

Based on these requirements, we selected salad preparation (similar to that of the 50 Salads Dataset) and peanut butter and jelly sandwich preparation as the two tasks for the data collection. Each participant performed each task 5 times, resulting in a total of 170 videos. The data was subsequently annotated with the high-level actions performed at each step of the task. An in-depth report of the data collection initiative is described in Chapter 4.

3.4 Preference Type Analysis

We formulate the concept of learning preference types as a mapping from a high to lower-dimensional representation of task demonstrations. In order to determine the high-level strategies in which individuals perform each of the two tasks, we implemented two unsupervised learning methods described below, both of which aim to learn a lower-dimensional representation of preference types from a large dataset of task demonstrations.

The high-level idea is to learn a representation as clusters within the latent space of the task sequences. The resulting clusters were analyzed via manually inspecting their constituent sequences and further validated by sampling the top 5 sequences closest to the centroid within each cluster to determine a predominant preference type.

3.4.1 Autoencoder

The goal of this component is to learn a low-dimensional representation of the task in latent space, then cluster into distinct preference groups. The key idea of an autoencoder is to perform dimensionality reduction by training a neural network to reconstruct its input [22]. Using an implementation proposed by Zhao as an extension to prior work [33], we trained an autoencoder to assign each task sequence to a latent variable K. Given sequential data, or task demonstrations, as input, the autoencoder reconstructs the original sequence after summarizing the data as a fixed-length vector in selected number of dimensions [30]. For our autoencoder we use mean squared error (MSE) loss, the Adam optimizer with learning rate 0.001, and a $2D$ latent space.

We applied K-Means clustering to the latent space in order to give rise to groups of distinct preferences. Finally, we used the elbow method with silhouette score to determine the optimal number of clusters as $K = 3$. The elbow method is a heuristic that is used to determine the optimal number of clusters in a dataset after applying a dimensionality reduction method [26]. By using this approach to determine the optimal value for K, we found that $K = 3$ was ideal within the simulated data, and $K = 5$ was ideal within the real-world data for eliciting intuitive preference groups without overfitting to individual people.

Further analysis by qualitative inspection indicated implicit preference types represented by each resultant cluster. By performing cluster-wise inspection to the resulting preference groups, we found that the clusters produced by the autoencoder method were human-interpretable at the scale of data used in this work. We subsequently use this autoencoder method to generate the clusters – or preference groups – in our proposed preference learning approach.

3.5 Models

3.5.1 Baseline

In order to benchmark the performance of both the mixture of experts and meta learning methods, we chose to use an LSTM as the baseline model. We used an LSTM with a sliding window over partial sequences as input. The output of the model is a prediction for the next action towards completing the task sequence. This baseline model was trained on an aggregate dataset of all task sequences at once, rather than first processing the data into preference group clusters.

Our LSTM uses 4 hidden state features, 1 layer, and 19 output classes. The input size was 95 for the simulated data and 90 for the real-world data. We use 80% of data for training and the remaining 20% for evaluation. The model was trained over $40,000$ epochs with a learning rate of 0.0001 using cross entropy loss and the Adam optimizer.

3.5.2 Mixture of Experts

The Mixture of Experts (MoE) paradigm is a type of algorithm where the type of model is chosen based on a sample input [24]. In this work, we utilize and extend upon an MoE method devised by Zhao et al. in [33].

We use a long short term memory (LSTM) network [12] as the base model for our approach. LSTMs are widely used for sequential modeling because they employ an attention mechanism that learns variable-range long-term dependencies by using previous history to inform the current prediction. Since this component aims to learn the temporal context within a sequence, LSTMs are a viable candidate for predicting the most probable next action within a sequential task, such as the ones included in this study. Additionally, we apply a sliding-window modification to the LSTM in order to account for variable-length sequences seen in real-world data. For consistent evaluation, the MoE model uses the same parameters as described previously for the baseline model.

An overview of this method is shown in Figure 3.1. In summary, an autoencoder with 2-dimensional latent space is trained on a dataset of human demonstrations. After applying K-Means on the latent space, the resultant clusters are used in a mixture-of-experts model. This approach selects a model from a policy library of different models, each trained on a subset of data that corresponds to a different preference group.

3.5.3 Meta Learning

Meta learning is a paradigm where a model can ”learn to learn” [8]. In other words, its objective is to improve an underlying learning algorithm after experiencing multiple learning episodes. Training data for meta learning is composed of support sets, which in our approach are the preference groups resulting from the clustering step. This setup enables the model to be trained on the samples provided in the support sets, then test how well it can make predictions on the query set. In this work, we use meta learning as a means of adapting to out-of-distribution behaviors or preference types.

Initial adaptation of this component was led by Michelle Zhao. The implementation for our meta learning approach is extended from the `learn2learn` library [1], which uses MAML as the base learner. In this algorithm, there are two types of parameter updates – the outer loop updates the initialization of the model's parameters and the inner loop uses the outer loop parameters to adapt to samples seen during training. Since this base meta learning formulation performs classification, we modify our model to do multi-class classification of annotations from the action space of 19

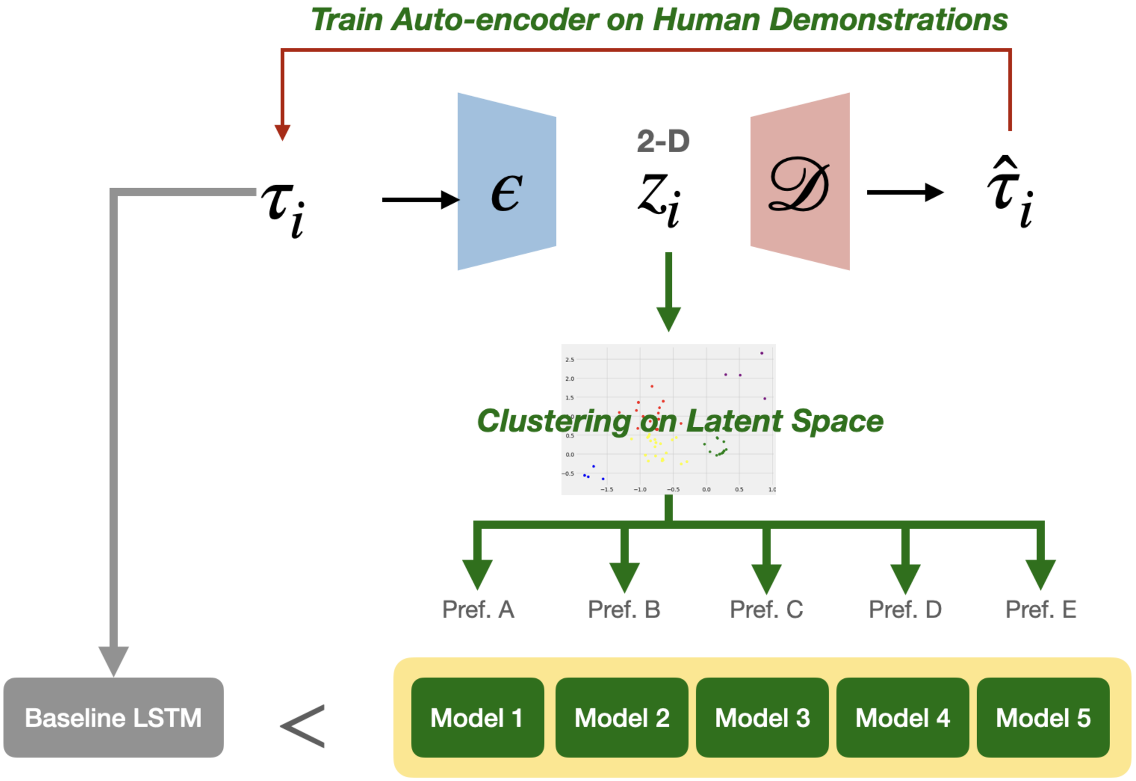


Figure 3.1: Overview of the approach displaying preference type identification with an autoencoder and preference learning via the mixture of experts algorithm. (Figure courtesy of Michelle Zhao.)

annotations using a fully-connected network (FCN). We train the model on a subset of preference groups, and then provide it with the held out groups in order to evaluate its performance on unseen preference types.

The meta learning model uses an input size of 54 with 19 output classes. The architecture is a 4-layer FCN. We used 2 out of 3 support sets for training on the simulated data, and 3 out of 5 for the real-world data. The model was trained over 400 epochs with a meta learning rate of $\alpha = 0.01$ for the outer loop and learning rate of $\beta = 0.001$ for the inner loop using cross entropy loss and the Adam optimizer.

Chapter 4

Real-World Data Collection

4.1 Overview

We conducted an evaluative data collection to obtain a real-world dataset of diverse task demonstrations. The following sections describe the study design. Our learning algorithms were evaluated post-hoc on the annotated collected data.

4.2 Task Design

We selected two different meal preparation tasks to evaluate our method – (1) preparing a salad and (2) preparing a peanut butter and jelly (PB & J) sandwich, using real ingredients and utensils for both. We chose these tasks because they possess inherent variation in manners of how they are performed – e.g. choice of utensil, selection of ingredients, and order of steps. They are also fairly common dishes that are presumably familiar to most people, even those who do not cook regularly.

Furthermore, we designed these tasks based on additional constraints for practicality in a user study setting. Each task does not require the use of a stove, does not require a precise recipe nor niche expertise, and the required ingredients were within a reasonable cost for the scale of data collection. Additionally, these tasks meet IRB-recommended safety considerations which were important factors in recruiting a diverse population of participants for our study. Perhaps most significantly, these

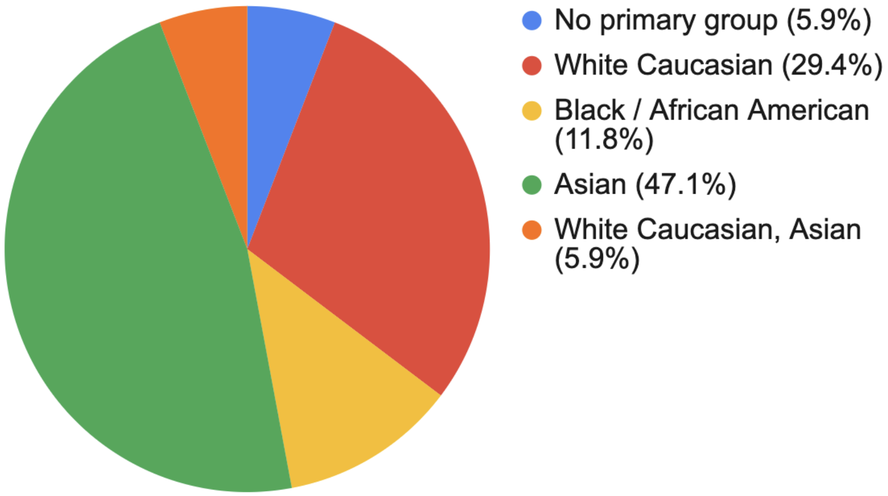


Figure 4.1: Participants' answer to the following question: *How would you describe your primary racial group?*

tasks also have relevant downstream applications to assistance, as they are proxies for regular real-world human-centric routines that people often perform in their own homes.

4.3 Participant Information

The participant demographics were selected to approximate the real-world population. We recruited 17 individuals from Carnegie Mellon University and the broader Pittsburgh community, primarily using the CBDR recruitment platform. Participant ages ranged from $19 - 82$, where notably 6 participants were older adults over the age of 50. Mean age was 37.5 years old with a standard deviation of 21.2 years. The ratio of male- to female-identifying individuals was $11 : 6$. The majority (52.9%) of participants were students. Additional demographic information is provided in the corresponding Figures 4.1, 4.2, 4.3.

4.4 Procedure

Participants were briefed about the study details and completed an informed consent form in accordance with the Institutional Review Board protocol. Participants were

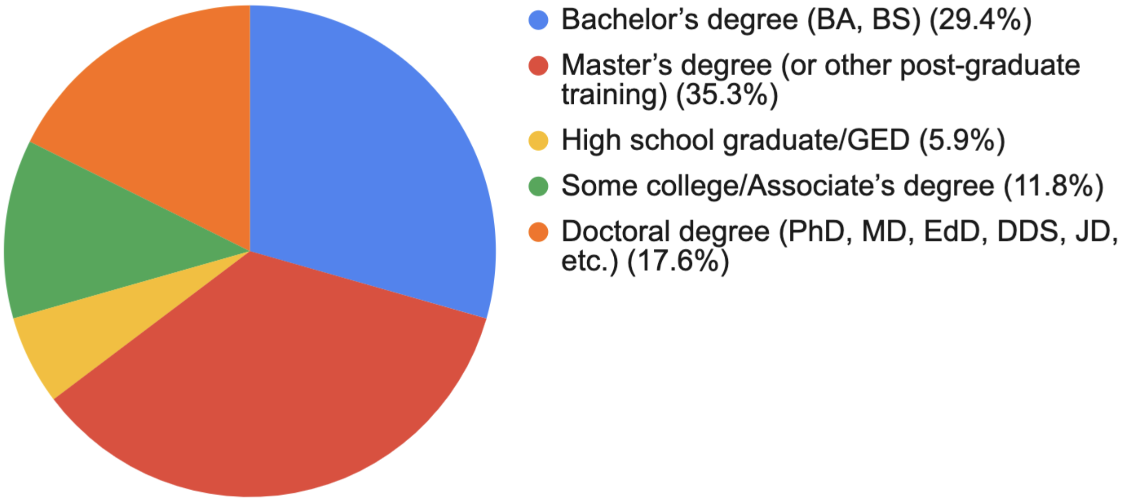


Figure 4.2: Participants’ answer to the following question: *What is your highest level of education?*

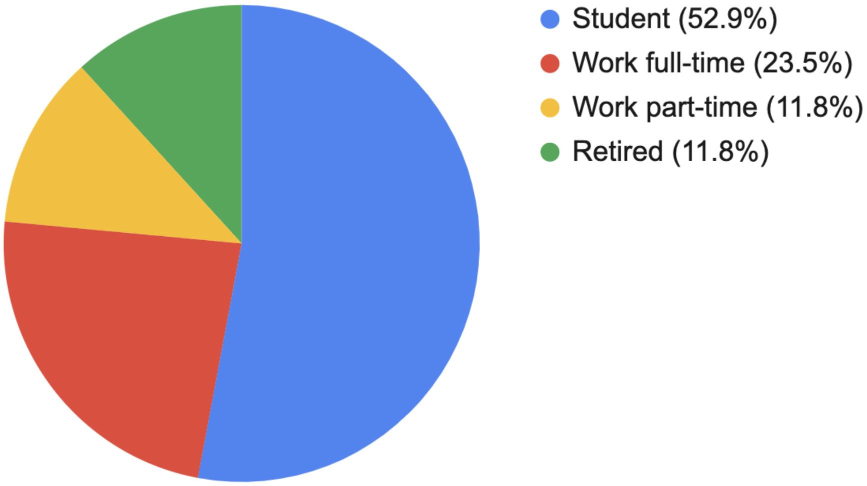


Figure 4.3: Participants’ answer to the following question: *What is your primary occupational status?*

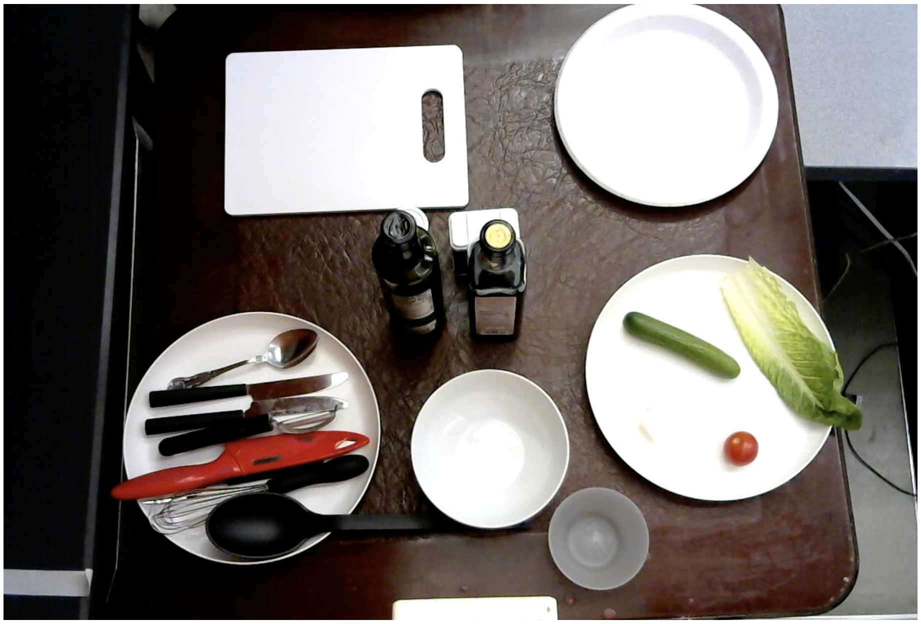


Figure 4.4: Example start configuration for the salad preparation task.

given an opportunity to ask questions before proceeding with the study.

Each participant was prompted to initiate the task using the following statement: ”Please make the {salad, sandwich} naturally as if you were making a meal for yourself in your own home.” Participants performed each of the 2 tasks 5 times each, resulting in 10 instances per participant. Data was collected using a combination of two cameras: one ceiling-mounted above the kitchen countertop area to provide a third-person view, and a second head-mounted GoPro camera worn by the individual for a first-person view. The resulting video data was annotated post-hoc, as described in Section 4.6.

Following the study, each participant completed a post-study questionnaire containing qualitative questions to reflect the measures described in Section 4.5. The initial setup for each of the two tasks is shown in the corresponding Figures 4.4 and 4.5. The displayed examples are from the third-person camera configuration.

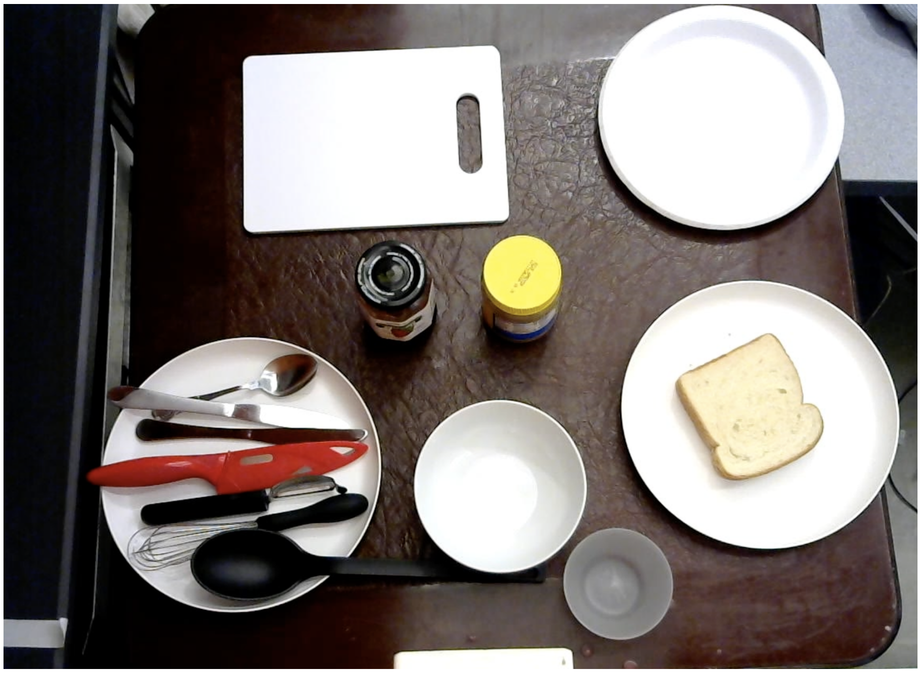


Figure 4.5: Example start configuration for the sandwich preparation task.

4.5 Qualitative Evaluation

Each session concluded with a questionnaire containing the following qualitative factors:

- Participant time spent on personal meal preparation tasks
- Consistency of participant cooking style
- Degree of preference that participant has for how they make their meals

We use these attributes to determine the extent to which each participant is expected to have preferences for the selected tasks.

4.6 Data Annotation

Annotations are discretized as words which represent the possible action space for each task. Figure 4.6 displays sample frames from the dataset. We define the following action space sets for the two tasks:

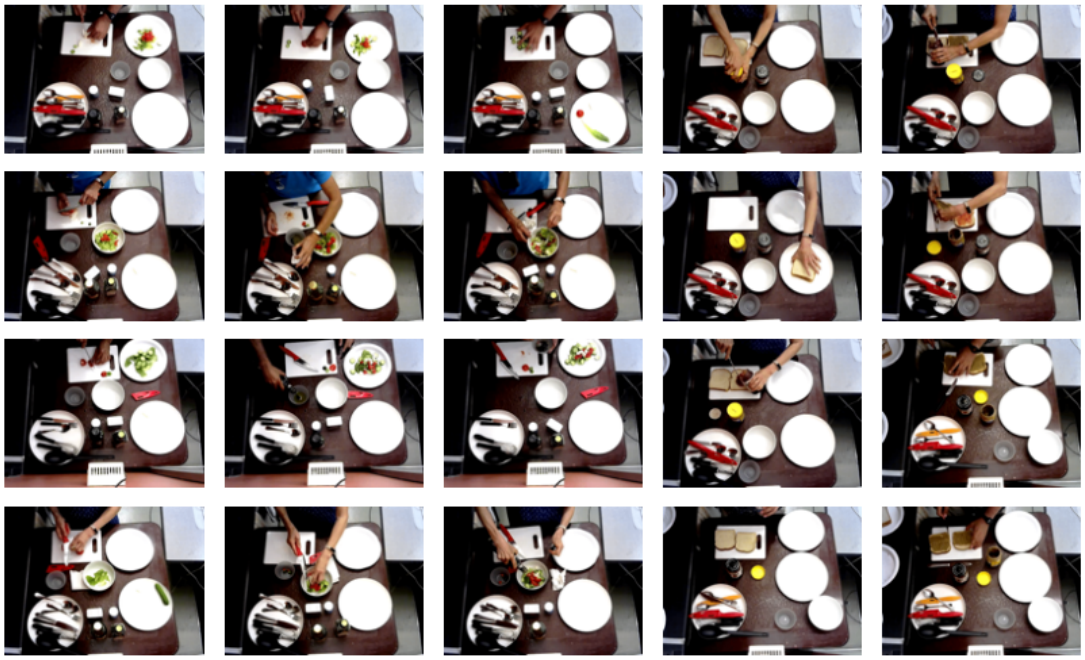


Figure 4.6: Frames captured from the real-world dataset collection.

- Salad task: {start, serve_salad_onto_plate, add_dressing, mix_dressing, add_salt, add_pepper, add_oil, add_vinegar, mix_ingredients, place_tomato_into_bowl, place_cucumber_into_bowl, place_lettuce_into_bowl, place_cheese_into_bowl, cut_cheese, cut_lettuce, cut_tomato, cut_cucumber, peel_cucumber, end}
- Sandwich task: {start, put_pb_on_bread, put_jelly_on_bread, spread_pb_on_bread, spread_jelly_on_bread, put_slices_together, cut_sandwich, serve_sandwich_onto_plate, end}

Thus the total action space size is $N = 19$ for the salad preparation task and $N = 9$ for the sandwich preparation task. We note that this level of granularity for annotations was chosen to fit the requirements of this study, but could be extended to include more fine-grained annotations in future work.

Chapter 5

Results

5.1 Overview

In this chapter, we present the results from evaluating our approaches on both the simulated data and the real-world data; the results are organized into these respective subsections. Within this, we first evaluate our method for learning implicit preference types via an autoencoder. Additionally, we evaluate two approaches for predicting an individual's next action in a task sequence – first where the model is conditioned on the person's preference type using a mixture of experts model, and secondly via meta-learning a better initial parametrization over the different preference type distributions.

5.2 Simulated Data

We generated an augmented dataset of $N = 100,000$ simulated sequences for the salad preparation task using the approach described in 3.2. This dataset is used to produce the results throughout this section.

5.2.1 Preference Learning via Autoencoder

Using the elbow method with silhouette score and testing $K = 2$ to $K = 9$, we found that using $K = 3$ for the autoencoder yielded the optimal number of clusters.

	Baseline (Aggregate)	MoE	Meta Learning
Accuracy (%)	72.28	**77.28**	62.71

Table 5.1: Results displaying the accuracy comparison between the two learning algorithms against the baseline aggregate model; all trained on the simulated dataset.

	Group 0	Group 1	Group 2	MoE
Accuracy (%)	56.16	44.48	56.76	**77.28**

Table 5.2: Results comparing the accuracy between the MoE and group-specific models; all trained on the simulated dataset.

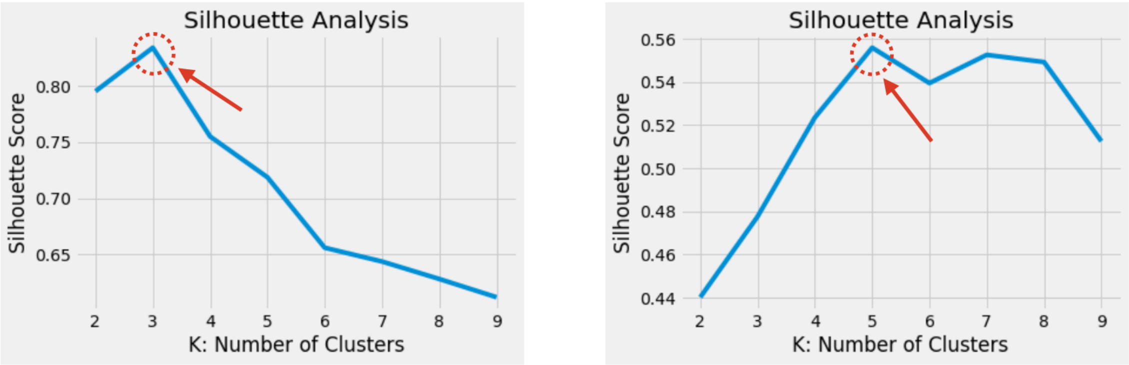
Silhouette score analysis is shown in Figure 5.1.

The resultant clusters are displayed in Figure 5.2. In summary, the three preference types that were found from inspecting each cluster are:

- Preparing dressing before salad
- Preparing salad then dressing, where cucumber is cut before cheese
- Preparing salad then dressing, where cheese is cut before cucumber

5.2.2 Mixture-of-Experts

Accuracy for the mixture-of-experts (MoE) model compared to the baseline and meta learning models is shown in Table 5.1. The results for MoE compared to the three group-specific models are shown in Table 5.2. Group-specific models were trained only on data within a single cluster, i.e. preference type. These findings demonstrate that the MoE model obtained higher accuracy, 77.28%, than both the baseline aggregate model (72.28%) and models trained over individual preference groups. These results indicate that MoE, by conditioning a model on preference type, can more accurately predict what action an individual will perform next after being trained on a diverse distribution of data.



Simulated Data

Real-World Data

Figure 5.1: Silhouette score analysis demonstrating a clear peak at $K = 3$ for the simulated data and $K = 5$ for the real-world data, indicating the optimal number of clusters.

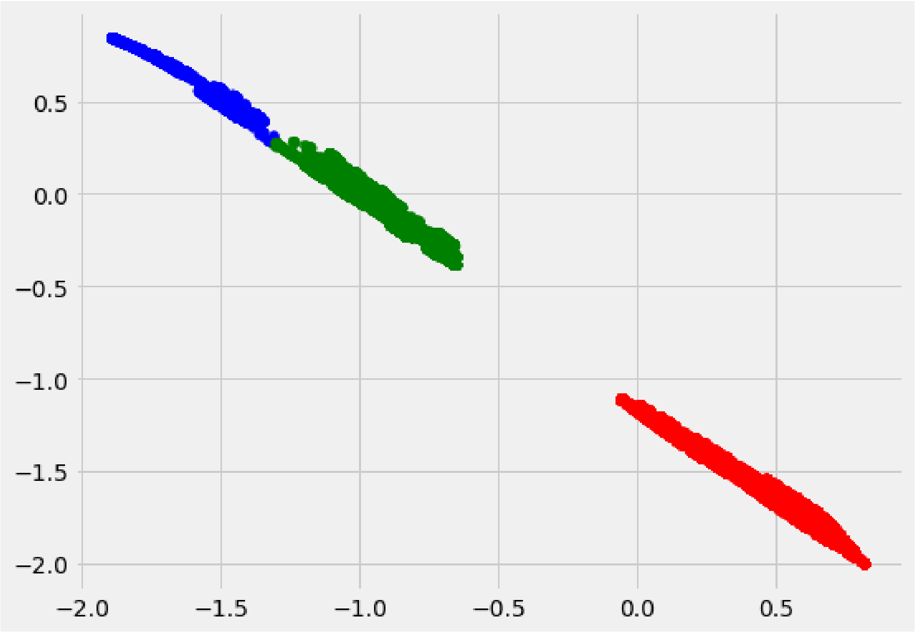


Figure 5.2: Preference groups for $K = 3$ on the simulated salad task data. The red cluster represents the preference type for preparing dressing before salad; the blue cluster represents the preference type for preparing salad then dressing, where cucumber is cut before cheese; the green cluster represents the preference type for preparing salad then dressing, where cheese is cut before the cucumber.

	Baseline (Aggregate)	MoE	Meta Learning
Accuracy (%)	80.82	**94.87**	68.28

Table 5.3: Results displaying the test set accuracy comparison between the two learning algorithms – Mixture of Experts and Meta Learning – against the baseline model. All models trained on the real-world data.

5.2.3 Meta Learning

By using meta learning in order to learn a model over a variety of preference groups (i.e. *support sets* in meta learning terminology), we find that we obtain accuracy of 62.71% that shows lower performance than the baseline model's 72.28%. The meta learning model also results in lower performance than the MoE model when trained over the simulated data.

5.3 Real-World Data

5.3.1 Preference Learning via Autoencoder

Using the autoencoder to learn a lower-dimensional representation over the real-world salad task data, and subsequently applying K-Means clustering, we find that there are $K = 5$ preference groups. This is validated using silhouette score analysis with the elbow method, shown in Figure 5.1. The resulting clusters are shown in Figure 5.3.

In summary, the five preference types found from inspecting each cluster are:

- Cluster 0: Preparing vegetables then cheese, followed by dressing last
- Cluster 1: Preparing vegetables and dressing in parallel
- Cluster 2: Preparing vegetables then dressing
- Cluster 3: Preparing vegetables, then only salt/pepper for dressing
- Cluster 4: Preparing salad without cheese or dressing

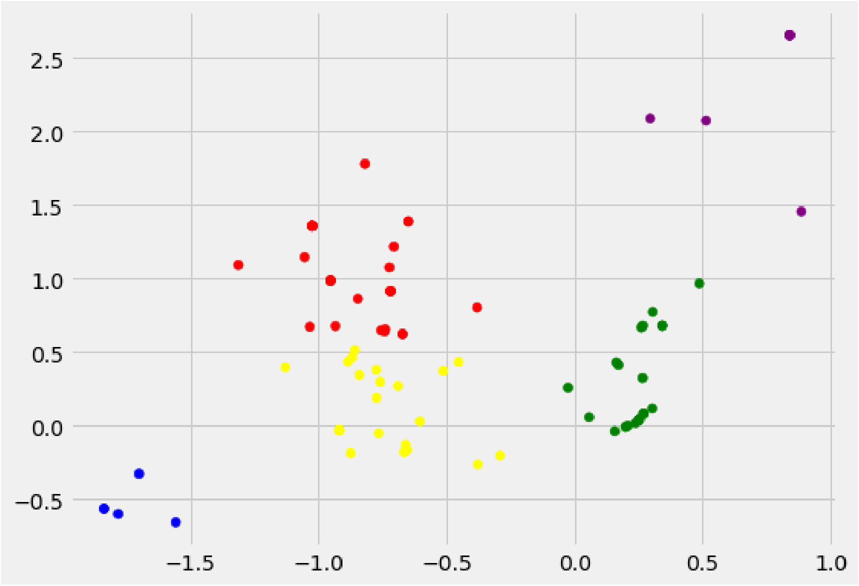


Figure 5.3: Preference groups for $K = 5$ on the real-world salad task data. The semantic labels for these clusters are detailed in Section 5.3.1.

5.3.2 Mixture-of-Experts

Compared to the baseline aggregate model's accuracy of 80.82%, the MoE model achieves 94.87% accuracy as shown in Table 5.3. Similarly as seen within the simulated data, these results suggest that MoE is better able to predict the next action in a task sequence compared to the baseline model due to conditioning on preference type. The MoE model obtained higher accuracy when trained on the real-world data, despite being a smaller dataset than the simulated data.

5.3.3 Meta Learning

We find that the meta learning model is achieves lower predictive accuracy of 68.28% compared to the baseline aggregate model's 80.82% accuracy. As seen within the simulated data results, the meta learning model obtained lower accuracy than the MoE approach. However, as was the case for MoE, our meta learning model also resulted in higher performance when trained on real-world data.

5.4 User Study Qualitative Observations

5.4.1 Overview

We found that the majority (58.8%) of participants reported that they had *some* preferences within their meal preparation habits, yet only 35.3% reported that they had *strong* preferences. The remainder, 5.9%, reported that they do not have preferences within their meal preparation habits. These results are presented in Figure 5.6. Additional participant responses to qualitative survey questions are displayed in Figures 5.4, 5.5, and 5.7.

5.4.2 Salad Task

In the following subsection, we provide a per-participant analysis of corresponding preference type. We additionally describe participant consistency in order to understand whether individuals behave in a manner that reflects our assumption on how to learn preferences in the real world.

Consistency, reported in the right-most column, is determined by whether the participant performs the task in the same manner (reflected by all demonstrations belonging to one cluster). If the majority of participant sequences belong to a single cluster, they are reported as mostly consistent. If the demonstrations belong to three or more clusters, the participant is reported as being inconsistent.

These findings indicate that the majority – 10 out of 17 participants – behaved consistently across each instance of the task. Two participants behaved inconsistently. Out of the five participants that behaved in a mostly consistent manner, two individuals (participants 1 and 16) performed the task differently for only the first instance, then proceeded to execute the task consistently for the remainder of the study session. If individuals who behave consistently and mostly consistently are included in the definition of consistent behavior, then the self-reported participant responses in Figure 5.5 (88.2%, or 15 out of 17 individuals) support these results.

The results suggest that most people behave in a consistent manner for the meal preparation task examined in this study. For participants who do not act consistently, future work in preference learning could address this challenge by investigating new

Participant ID	Instance 1	Instance 2	Instance 3	Instance 4	Instance 5	Consistent
1	Cluster 3	Cluster 1	Cluster 1	Cluster 1	Cluster 1	Mostly
2	Cluster 2	Cluster 2	Cluster 0	Cluster 1	Cluster 1	No
3	Cluster 0	Cluster 0	Cluster 0	Cluster 0	Cluster 0	Yes
4	Cluster 2	Cluster 0	Cluster 3	Cluster 3	Cluster 3	No
5	Cluster 2	Cluster 2	Cluster 2	Cluster 2	Cluster 2	Yes
6	Cluster 1	Cluster 1	Cluster 1	Cluster 1	Cluster 1	Yes
7	Cluster 1	Cluster 1	Cluster 1	Cluster 1	Cluster 1	Yes
8	Cluster 4	Cluster 4	Cluster 4	Cluster 4	Cluster 4	Yes
9	Cluster 1	Cluster 3	Cluster 1	Cluster 3	Cluster 1	Mostly
10	Cluster 2	Cluster 2	Cluster 2	Cluster 2	Cluster 2	Yes
11	Cluster 3	Cluster 3	Cluster 3	Cluster 3	Cluster 3	Yes
12	Cluster 1	Cluster 1	Cluster 1	Cluster 1	Cluster 1	Yes
13	Cluster 1	Cluster 1	Cluster 1	Cluster 1	Cluster 1	Yes
14	Cluster 2	Cluster 2	Cluster 1	Cluster 2	Cluster 1	Mostly
15	Cluster 2	Cluster 2	Cluster 2	Cluster 2	Cluster 2	Yes
16	Cluster 2	Cluster 1	Cluster 1	Cluster 1	Cluster 1	Mostly
17	Cluster 1	Cluster 2	Cluster 1	Cluster 1	Cluster 1	Mostly

Table 5.4: Per-participant preference type analysis.

learning paradigms or learning to adapt over longer time horizons of observing the user performing the task.

5.4.3 Survey Results

The self-reported results from the post-study questionnaire show that the majority of individuals (73.7%) follow the ”salad then dressing” preference type in the salad preparation task. The demonstrations in the actual collected dataset support this result. A majority, 64.7%, of participants reported that they either agree or strongly agree to spending a significant amount of time cooking, and nearly all (94.1%) of participants stated that they had either some or strong preferences for how they perform meal preparation.

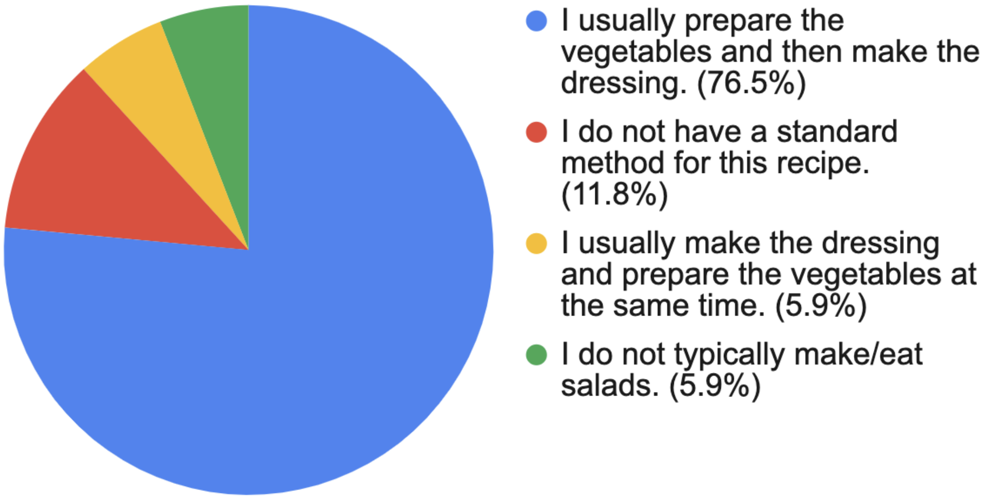


Figure 5.4: Participant responses to the prompt, ”Choose the method which best describes how you typically make salad at home”.

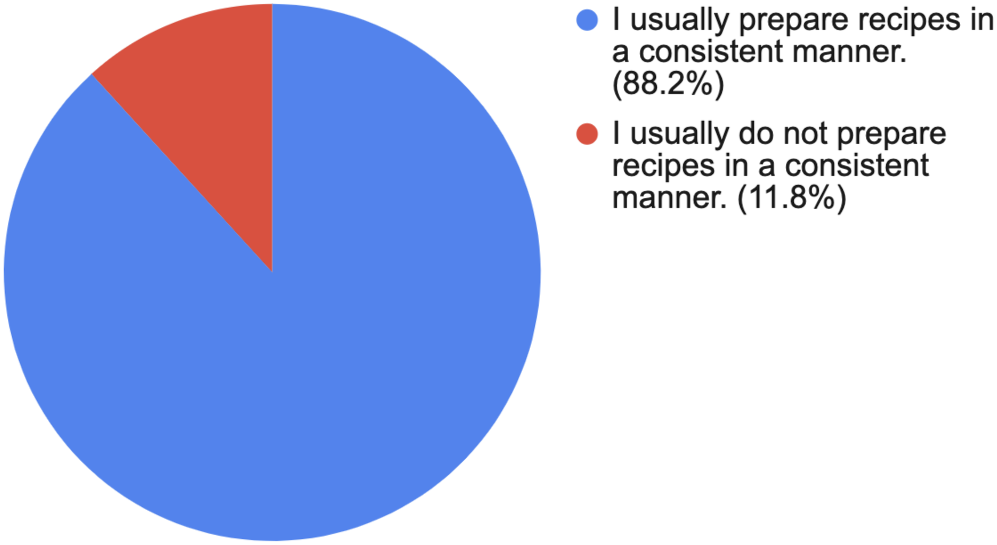


Figure 5.5: Participant responses to the prompt, ”Choose the option that best describes your style in the kitchen”.

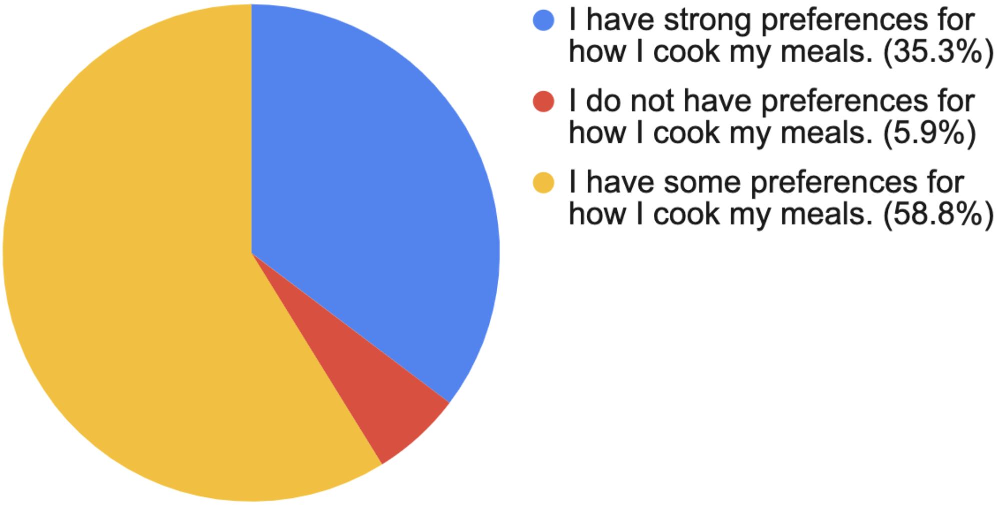


Figure 5.6: Participant responses to the prompt, ”Choose the option that best describes your meal preparation habits”.

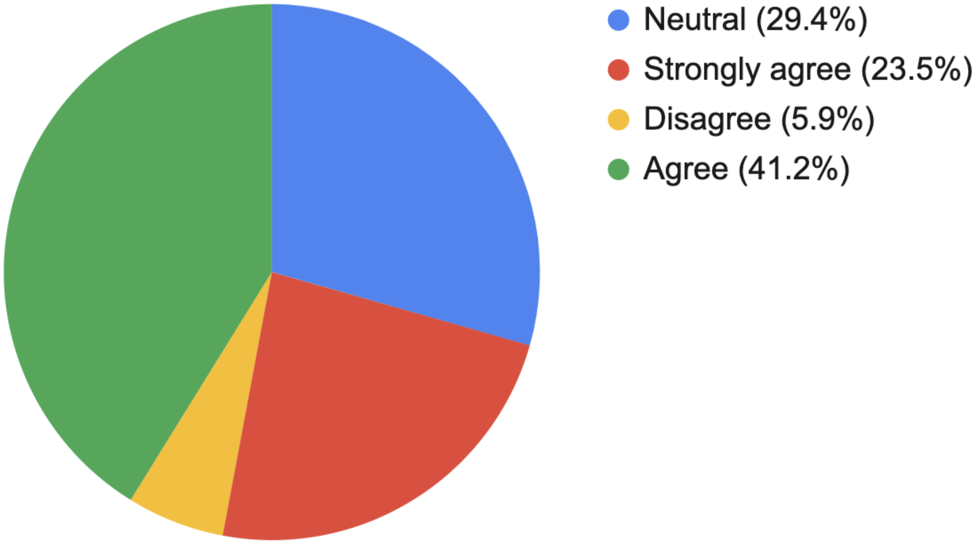


Figure 5.7: Participant responses to the prompt, ”I spend a significant amount of time cooking”.

Chapter 6

Future Work

Potential directions for future work include running a subsequent user study with humans in the loop in order to qualitatively evaluate whether the model reflects individuals' preferences, not solely on the basis of the model's predictive accuracy. It could be insightful to further assess this approach from additional human-centric perspectives (such as trust, relevance, intuitiveness) in order to measure the model's alignment with the human's preferences. Evaluating on other types of tasks, such as the sandwich preparation data provided in this work, could also provide an understanding of how well this approach may work on different problems.

Furthermore, long-term future work could include an embodied agent, such as a social robot, which interacts with the participant through providing verbal suggestions from the model's predictions or providing proactive assistance. This component would also be well-matched with a more comprehensive qualitative evaluation to understand whether the suggestions provided by the assistive robot are indeed considered relevant, intuitive, and helpful by the user.

Additionally, this work could be applied to alternative demographics of participants, such as older adults or those with memory impairment issues. For such applications, it could be insightful to study routines over longer time horizons, such as those that occur over multiple days or weeks. This could also potentially enable insights on monitoring deviations in peoples' routines as a means of detecting health anomalies. Another direction that could arise from this is ensuring that the resultant preference types are semantically understandable, as they are in this work, even as the model's input data becomes more complex. Such safety considerations would be especially important in health-focused applications.

We would also like for future work to include a different, per-participant evaluation scheme for both the mixture of experts and meta learning methods, in which each model is provided with 1 additional demonstration from each individual, and the resulting fine-tuned model is compared to a baseline model without per-participant fine-tuning.

Lastly, this work explores preference learning from a temporal context. In the future, it would be interesting to include additional features, a more fine-grained action space of annotations, or other data modalities in the model to determine whether it is possible to learn a more robust model of individuals' preferences. Such extensions would also require a large dataset of labeled data, with repeated task instances per individual; we hope that our work provides steps in the right direction towards this objective.

Limitations

While learning individuals' preferences is an important problem, there are some critical assumptions we make in this study that should be further investigated in later work, namely: while demonstrations are typically more informative than repeatedly querying the human, not all humans behave rationally. Thus it is important to avoid overestimating rational or consistent behavior when deploying interactive preference-based systems in the future.

By nature of transforming a high-dimensional state space to a lower-dimensional representation, the learned implicit representations may not always be easily interpretable nor semantically understandable preference types. This would be particularly evident for more complex tasks, especially those with less deterministic outcomes.

Additionally, although the study was conducted in the real world, in-person, and in a real kitchen, it is important to note certain factors that may have influenced the participants to act in a manner that is different from how they do in their own homes – for example, the user study kitchen was set up in a simplified manner that could be different from what users are accustomed to. In the future, a followup study could address this by running the full data collection and online evaluation process in individuals' own homes. Furthermore, people often don't repeat things over and over in the real world as they did in our data collection process, thus a more realistic followup study would be conducted over a longer time frame to observe people performing routine household tasks across multiple days or weeks.

Lastly, people do not always have a clear or definitive internal model of their own preferences, so including a more robust ground truth human-in-the-loop evaluation would be an important next step. Accounting for this qualitative component would enable us to determine whether the approaches outlined in this work are indeed better – and subjectively preferred – beyond simply using accuracy as a metric.

Chapter 7

Conclusion

In conclusion, this work presents a methodology for learning and evaluating implicit preference types. We present an approach to generate customizable quantities of simulated human task sequences, starting from existing real-world datasets, which are smaller and challenging to find for preference learning problems. Additionally, we share a 170-video dataset comprised of real humans performing multiple instances of two different real-world, commonplace household tasks. We build upon prior work by applying strategy matching techniques to preference identification from real-world task data, and further demonstrate that learning a representation by transforming trajectories into lower-dimensional output via an autoencoder produces clusters that represent human-interpretable implicit preference types.

Furthermore, we present two learning methods for action prediction conditioned on preference type – mixture of experts and meta learning. We compare our approaches on both the simulated data and real-world data, and evaluate their effectiveness in both data regimes. In the first method, mixture-of-experts, we find that the model is capable of more accurately predicting the next action towards completing a task sequence compared to a baseline model that is agnostic to preference type. In the second approach, meta learning, we find that the model obtains lower predictive accuracy than the baseline aggregate model. We postulate that this is due to the evaluation scheme used in this work, but future directions may extend this by using meta learning to evaluate on a per-individual basis. Additionally, for both learning methods, we find that the models achieve higher performance when trained on real-world data, indicating that for both models being trained on a smaller, yet higher quality and more realistic dataset yields better accuracy than equivalent models trained on simulated data. This highlights the limitations of using simulated data and the importance of using real human data in human-centric domains, such as preference learning in household tasks. While the MoE results are promising and show improvement over the baseline model, we suggest directions for future work that build upon these findings by integrating a human-in-the-loop evaluation in order to consider additional factors, such as subjective preference alignment.

Ultimately, we posit that human preferences are an important factor to consider and integrate into future research. We hope that the contributions outlined in this work pave the way for future applications that integrate preferences into interactive learning scenarios between humans and their robot partners.

Bibliography

[1] Sébastien M. R. Arnold, Praateek Mahajan, Debajyoti Datta, Ian Bunner, and Konstantinos Saitas Zarkias. learn2learn: A library for meta-learning research, 2020. 3.5.3

[2] Andrea Bajcsy, Dylan P. Losey, Marcia K. O'Malley, and Anca D. Dragan. Learning robot objectives from physical human interaction. In Sergey Levine, Vincent Vanhoucke, and Ken Goldberg, editors, *Proceedings of the 1st Annual Conference on Robot Learning*, volume 78 of *Proceedings of Machine Learning Research*, pages 217–226. PMLR, 13–15 Nov 2017. URL https://proceedings.mlr.press/v78/bajcsy17a.html. 2.3.1

[3] Andreea Bobu, Yi Liu, Rohin Shah, Daniel S. Brown, and Anca D. Dragan. Sirl: Similarity-based implicit representation learning. In *Proceedings of the 2023 ACM/IEEE International Conference on Human-Robot Interaction*, HRI '23, page 565–574, New York, NY, USA, 2023. Association for Computing Machinery. ISBN 9781450399647. doi: 10.1145/3568162.3576989. URL https://doi.org/10.1145/3568162.3576989. 1, 2.3.1

[4] W. Bradley Knox and Peter Stone. Tamer: Training an agent manually via evaluative reinforcement. In *2008 7th IEEE International Conference on Development and Learning*, pages 292–297, 2008. doi: 10.1109/DEVLRN.2008.4640845. 1

[5] Micah Carroll, Rohin Shah, Mark K Ho, Tom Griffiths, Sanjit Seshia, Pieter Abbeel, and Anca Dragan. On the utility of learning about humans for human-ai coordination. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper/2019/file/f5b1b89d98b7286673128a5fb112cb9a-Paper.pdf. 1.1

[6] Paul Christiano, Jan Leike, Tom B. Brown, Miljan Martic, Shane Legg, and Dario Amodei. Deep reinforcement learning from human preferences, 2023. 1, 2.3.1

[7] Dima Damen, Hazel Doughty, Giovanni Maria Farinella, Sanja Fidler, Antonino

Furnari, Evangelos Kazakos, Davide Moltisanti, Jonathan Munro, Toby Perrett, Will Price, and Michael Wray. The EPIC-KITCHENS dataset: Collection, challenges and baselines. *CoRR*, abs/2005.00343, 2020. URL https://arxiv.org/abs/2005.00343. 2.6.1

[8] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks, 2017. 2.5, 2.5.1, 3.5.3

[9] Tesca Fitzgerald, Pallavi Koppol, Patrick Callaghan, Russell Quinlan Jun Hei Wong, Reid Simmons, Oliver Kroemer, and Henny Admoni. INQUIRE: INteractive querying for user-aware informative REasoning. In *6th Annual Conference on Robot Learning*, 2022. URL https://openreview.net/forum?id=3CQ3Vt0v99. 2.3.1

[10] Raghav Goyal, Samira Ebrahimi Kahou, Vincent Michalski, Joanna Materzynska, Susanne Westphal, Heuna Kim, Valentin Haenel, Ingo Fründ, Peter Yianilos, Moritz Mueller-Freitag, Florian Hoppe, Christian Thurau, Ingo Bax, and Roland Memisevic. The ”something something” video database for learning and evaluating visual common sense. *CoRR*, abs/1706.04261, 2017. URL http://arxiv.org/abs/1706.04261. 2.6.1

[11] Kristen Grauman, Andrew Westbury, Eugene Byrne, Zachary Chavis, Antonino Furnari, Rohit Girdhar, Jackson Hamburger, Hao Jiang, Miao Liu, Xingyu Liu, Miguel Martin, Tushar Nagarajan, Ilija Radosavovic, Santhosh Kumar Ramakrishnan, Fiona Ryan, Jayant Sharma, Michael Wray, Mengmeng Xu, Eric Zhongcong Xu, Chen Zhao, Siddhant Bansal, Dhruv Batra, Vincent Cartillier, Sean Crane, Tien Do, Morrie Doulaty, Akshay Erapalli, Christoph Feichtenhofer, Adriano Fragomeni, Qichen Fu, Christian Fuegen, Abrham Gebreselasie, Cristina González, James Hillis, Xuhua Huang, Yifei Huang, Wenqi Jia, Weslie Khoo, Jáchym Kolár, Satwik Kottur, Anurag Kumar, Federico Landini, Chao Li, Yanghao Li, Zhenqiang Li, Karttikeya Mangalam, Raghava Modhugu, Jonathan Munro, Tullie Murrell, Takumi Nishiyasu, Will Price, Paola Ruiz Puentes, Merey Ramazanova, Leda Sari, Kiran Somasundaram, Audrey Southerland, Yusuke Sugano, Ruijie Tao, Minh Vo, Yuchen Wang, Xindi Wu, Takuma Yagi, Yunyi Zhu, Pablo Arbelaez, David Crandall, Dima Damen, Giovanni Maria Farinella, Bernard Ghanem, Vamsi Krishna Ithapu, C. V. Jawahar, Hanbyul Joo, Kris Kitani, Haizhou Li, Richard A. Newcombe, Aude Oliva, Hyun Soo Park, James M. Rehg, Yoichi Sato, Jianbo Shi, Mike Zheng Shou, Antonio Torralba, Lorenzo Torresani, Mingfei Yan, and Jitendra Malik. Ego4d: Around the world in 3, 000 hours of egocentric video. *CoRR*, abs/2110.07058, 2021. URL https://arxiv.org/abs/2110.07058. 2.6.1

[12] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Computation*, 9(8):1735–1780, 1997. doi: 10.1162/neco.1997.9.8.1735. 3.5.2

[13] Timothy M. Hospedales, Antreas Antoniou, Paul Micaelli, and Amos J. Storkey. Meta-learning in neural networks: A survey. *CoRR*, abs/2004.05439, 2020. URL https://arxiv.org/abs/2004.05439. 2.5.1

[14] Robert A. Jacobs, Michael I. Jordan, Steven J. Nowlan, and Geoffrey E. Hinton. Adaptive mixtures of local experts. *Neural Computation*, 3(1):79–87, 1991. doi: 10.1162/neco.1991.3.1.79. 2.4

[15] Michael I. Jordan and Robert A. Jacobs. Hierarchical mixtures of experts and the em algorithm. *Neural Computation*, 6(2):181–214, 1994. doi: 10.1162/neco.1994.6.2.181. 2.4

[16] Eric Kolve, Roozbeh Mottaghi, Winson Han, Eli VanderBilt, Luca Weihs, Alvaro Herrasti, Matt Deitke, Kiana Ehsani, Daniel Gordon, Yuke Zhu, Aniruddha Kembhavi, Abhinav Gupta, and Ali Farhadi. Ai2-thor: An interactive 3d environment for visual ai, 2022. 1.3

[17] Fernando De la Torre, Jessica K. Hodgins, Adam W. Bargteil, Xavier Martin, J. Robert Macey, Alex Tusell Collado, and Pep Beltran. Guide to the carnegie mellon university multimodal activity (cmu-mmac) database. 2008. URL https://api.semanticscholar.org/CorpusID:16721121. 2.6.1

[18] Jacky Liang, Viktor Makoviychuk, Ankur Handa, Nuttapong Chentanez, Miles Macklin, and Dieter Fox. Gpu-accelerated robotic simulation for distributed reinforcement learning, 2018. 1.3

[19] World Health Organization. *A blueprint for dementia research*. World Health Organization, 2022. 1.2

[20] Archit Parnami and Minwoo Lee. Learning from few examples: A summary of approaches to few-shot learning, 2022. 2.5

[21] Xavier Puig, Kevin Ra, Marko Boben, Jiaman Li, Tingwu Wang, Sanja Fidler, and Antonio Torralba. Virtualhome: Simulating household activities via programs, 2018. 1.3

[22] D. E. Rumelhart, G. E. Hinton, and R. J. Williams. Learning internal representations by error propagation. page 318–362, 1986. 3.4.1

[23] Jurgen Schmidhuber. Evolutionary principles in self-referential learning. on learning now to learn: The meta-meta-meta...-hook. 14 May 1987. URL http://www.idsia.ch/~juergen/diploma.html. 2.5

[24] Noam Shazeer, Azalia Mirhoseini, Krzysztof Maziarz, Andy Davis, Quoc Le, Geoffrey Hinton, and Jeff Dean. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer, 2017. 3.5.2

[25] Sebastian Stein and Stephen J. McKenna. Combining embedded accelerometers with computer vision for recognizing food preparation activities. In *Proceedings*

of the 2013 ACM International Joint Conference on Pervasive and Ubiquitous Computing, UbiComp '13, page 729–738, New York, NY, USA, 2013. Association for Computing Machinery. ISBN 9781450317702. doi: 10.1145/2493432.2493482. URL https://doi.org/10.1145/2493432.2493482. 2.6

[26] Robert L. Thorndike. Who belongs in the family? *Psychometrika*, 18:267–276, 1953. 3.4.1

[27] Yaqing Wang, Quanming Yao, James Kwok, and Lionel M. Ni. Generalizing from a few examples: A survey on few-shot learning, 2020. 2.5

[28] Aaron Wilson, Alan Fern, and Prasad Tadepalli. A bayesian approach for policy learning from trajectory preference queries. In *Proceedings of the 25th International Conference on Neural Information Processing Systems - Volume 1*, NIPS'12, page 1133–1141, Red Hook, NY, USA, 2012. Curran Associates Inc. 2.3.1

[29] Christian Wirth, Riad Akrour, Gerhard Neumann, and Johannes Fürnkranz. A survey of preference-based reinforcement learning methods. *J. Mach. Learn. Res.*, 18(1):4945–4990, jan 2017. ISSN 1532-4435. 2.1

[30] Timothy Wong and Zhiyuan Luo. Recurrent auto-encoder model for multidimensional time series representation, 2018. URL https://openreview.net/forum?id=r1cLblgCZ. 3.4.1

[31] Mark Woodward, Chelsea Finn, and Karol Hausman. Learning to interactively learn and assist, 2019. 2.3.1

[32] Bryce Woodworth, Francesco Ferrari, Teofilo E. Zosa, and Laurel D. Riek. Preference learning in assistive robotics: Observational repeated inverse reinforcement learning. 85:420–439, 17–18 Aug 2018. URL https://proceedings.mlr.press/v85/woodworth18a.html. 1

[33] Michelle Zhao, Reid Simmons, and Henny Admoni. Coordination with humans via strategy matching. pages 9116–9123, 2022. doi: 10.1109/IROS47612.2022.9982277. 2.4.1, 3.4.1, 3.5.2

[34] Luowei Zhou and Jason J. Corso. Youcookii dataset. 2017. 2.6.1