

# Appendix

## 1 Dataset Generation

The Household Object Movements from Everyday Routines (HOMER) dataset<sup>1</sup> is composed of routine behaviors for five households, spanning 50 days for the train split and 10 days for test split. The households are based on an identical apartment setting with four rooms containing 108 objects and 33 atomic actions such as find, walk, grab, etc. To create HOMER, we first make a list of activities of daily living related to in-home routines, and then source our dataset using a two-tier strategy, separately sourcing high level activity schedules comprising of the above activities, and the low level action sequences to perform each activity. The 22 high-level activities of daily living we use are as follows:

brushing teeth	cleaning	watching TV	take out trash
showering	laundry	vaccuum cleaning	wash dishes
breakfast	leave home	reading	playing music
dinner	come home	going to the bathroom	listening to music
lunch	socializing	getting dressed	
computer work	taking medication	kitchen cleaning	

In the following sections, we first walk through the process of sourcing and processing each of the Activity Schedules and Activity Scripts separately, and then explain how they are together used to sample the final routines.

### 1.1 Activity Schedules

To obtain realistic activity schedules, we first asked workers on Amazon Mechanical Turk about which hours on a typical day they are likely to be doing each activity. We obtained this data from 25 workers, of which we filter out 4 for providing nonsensical schedules such as brushing and having dinner constantly throughout the day. The remaining 21 candidates include 4 female and 17 male participants, 8 participants aged 25-35, 7 aged 35-45, 3 aged 45-55, and 3 aged over 55. We also reclassified meals so that those happening in the morning were classified as breakfast (even if the raw input marked them as dinner), those in the afternoon were lunch, and in the evening were dinner.

The original samples are noisy and include idiosyncratic preferences of the individuals, so we cluster the samples for each activity to obtain aggregate distributions representing a common underlying habit of the individual samples that form the cluster. We represent each sample using a feature vector containing : 1) the means of a fitted mixture of two gaussians, and 2) the number of hours that activity is done in the morning (before 12:00), in the afternoon (between 12:00 and 18:00), and in the evening (after 18:00). Using these features to represent every sample, we divide the samples for each activity into 4 clusters using k-means clustering. We remove the clusters containing 3 or fewer samples. This results in up to 4 aggregate distributions representing different habits pertaining to that activity. Visual inspection suggests that these clusters represent semantically meaningful habits, such as brushing teeth in the morning v.s. brushing teeth twice a day, and having an early breakfast v.s. a late breakfast v.s. skipping breakfast as shown in Figure 1.

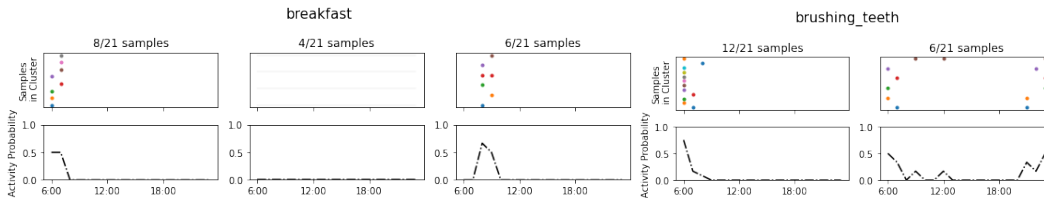


Figure 1: Example clusters of activity schedules representing semantically meaningful habits.

<sup>1</sup>The dataset is available at [https://github.com/GT-RAIL/rail\\_tasks/tree/homer/routines](https://github.com/GT-RAIL/rail_tasks/tree/homer/routines), and along with the code to generate different versions of it.

We combine a habit from each activity to compose complete temporal activity distributions representing fictitious households. We assume the activities to be mutually independent, except the *leave home* and *come home* activities, where the expected time of *leave home* must be before that of *come home* for the combination to be valid. Out of the several possible combinations of habits, we selected households that have distinct characteristics. We measure the distinctness of households by a KL-divergence metric on the temporal activity distributions. To maximize this metric, we use genetic optimization to obtain a set of five valid households, with the fitness function as the average pairwise KL-divergence. The mating function selects a new group of 5 households from any two existing groups, and the random mutation changes a household in a group to a random sample. A pool consists of 20 candidate household sets, and the optimization is run 5 times for 1000 iterations each, after which the best solution, representing a set of five diverse households, is picked.

## 1.2 Activity Scripts

We ask participants to emulate each activity from the aforementioned list on a simulator. We recruited 23 participants to compose step-by-step action sequences for each activity, defining the avatar’s movement, interactions with various objects, and the time duration required for it. In this manner, we obtained 61 scripts in total, covering all of our 22 activities. Following is an example script snippet representing *brushing teeth*, consisting of action sequences as well as the estimated time duration range in minutes needed to do these actions (## <min\_duration>-<max\_duration> in the snippet).

```
## 1-2
[Walk] <bathroom>
[Walk] <toothbrush>
[Find] <toothbrush>
[Grab] <toothbrush>
## 0-1
[Walk] <bathroom_cabinet>
[Find] <bathroom_cabinet>
[Open] <bathroom_cabinet>
[Find] <tooth_paste>
[Grab] <tooth_paste>
[Close] <bathroom_cabinet>
[Pour] <tooth_paste> <toothbrush>
[Find] <bathroom_counter>
[Putback] <tooth_paste> <bathroom_counter>
.....
```

## 1.3 Sampling routines

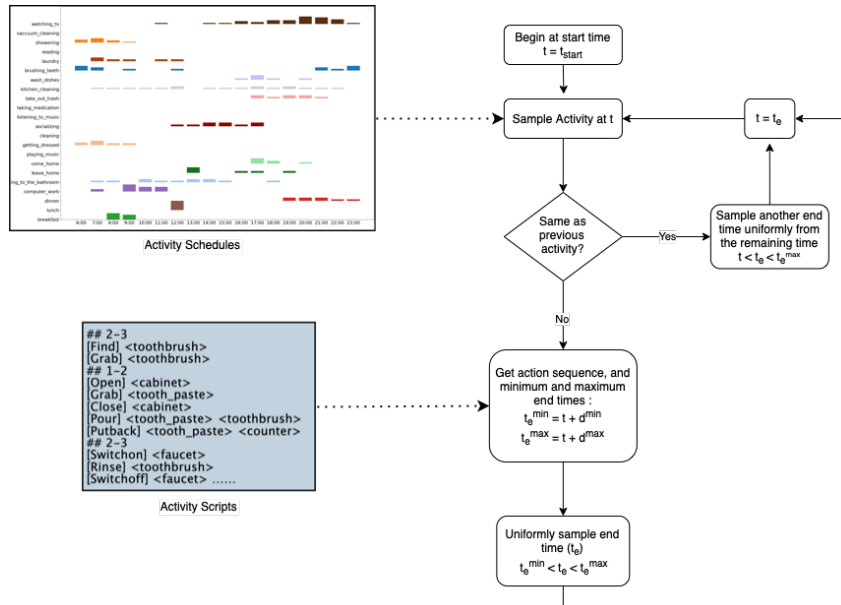


Figure 2: Process of sampling schedules from a household distribution and activity scripts

To generate schedule samples for a household, we use the household's temporal activity distribution and an action script for every activity. We use Monte Carlo sampling to generate sample schedules as outlined in Figure 2. Starting at 6am, we sample an activity from the schedule distribution, and obtain an end time for that activity by uniformly sampling in the duration range from the script. We sample another activity from the schedule distribution at that end time. If the same activity is sampled again, the activity is elongated, and another end time is sampled between that time and the maximum end time, otherwise the new activity is started. Once the start and end times are obtained for the activity, the durations of different actions within it are sampled while respecting the given total activity time. By iteratively sampling activities in this manner until the end time of midnight, we obtain samples of timestamped action sequences representing daily schedules.

The day-long scripts representing routines are executed on the VirtualHome simulator, starting from a fixed initial state in the apartment. This yields sequences of object arrangements resulting from the sampled routines. The object arrangement at a given time is available in a graphical format containing objects with object states at each node and object-object relations as edges. For each sampled routine, a sequence of object arrangement graphs is stored along with the associated time stamps to compose the final dataset.