Figure 1: Semantic distribution of actions of Ego-HOI test sets. We use BERT [9] embeddings to visualize the classes.

1. Visualization

For a more comprehensive analysis, we present more visualizations of video properties:

**Video Property on The Val/Test Set.** We present visualizations of the video properties on the test or valid sets, including action semantics (Figure 1), camera motion (Figure 2), hand pose (Figure 3), blurriness (Figure 4), hand box (Figure 5) and object box (Figure 6). Our proposed dataset One4All-T is more comprehensive on these properties. We also show the semantic similarity matrix of the test sets in Figure 7.

**Data Addition and Removal in All-for-one Setting.** We visualize the samples added or removed during task-specific model enhancement. The data points are represented according to video properties and we show the PCA visualizations of two key properties: hand locations (Figure 8) and object locations (9). It can be observed that the removed data usually locate in the dense area of whole datasets. Besides, the additional samples are not only located in the dense area (since our ablation study shows that similar data brings more model improvement) but also supplement more diverse samples located in the sparse regions.
2. Details of Ego4D-AR

Ego4D-AR (Action Recognition) is derived from the data and annotations of Ego4D [3]. We use the annotations from the long-term action anticipation task from Ego4D, which contains start and end frame indices and the corresponding human action. We adopt the video clips from the hand-object interaction task, whose lengths are 8 seconds. A total of 41,085 clips are used for training and 28,348 for validation. The video clips that are exactly within long-term action segments are assigned with action labels, and the rest are discarded.

After the filtering and label assignment, we obtain the single-label action recognition dataset Ego4D-AR, which has 22,081 training samples and 14,530 validation samples. Ego4D-AR has 77 action classes, where 66 are in the train set and 58 are in the validation set. There are 11 zero-shot classes in the validation set due to the filtering process, resulting in the relatively low performance on One-for-all task (Table 3 in the main text).

3. Implementation Details

3.1. One4All-P and One4All-T

We merge the action classes of EPIC KITCHENS 100, EGTEA-Gaze+, Ego4D-AR, and Something-Else and have an active pool with nearly 500 actions. Then the classes
with the same semantics are merged. The remaining classes for One4All-P and One4All-T are 394 and 204 respectively (the total number of classes are 401 due to some zero-shot classes). Then we use our data selection algorithm derived based on the analysis of the video properties presented in the main paper to build our One4All datasets. We respectively sample 20 and 5 instances per class for One4All-P and One4All-T as the initial dataset. Then we select samples according to the unified video property with weight 5:10:8:10:5 (i.e., action semantics: hand box: hand pose: object box: camera motion: blurriness). The KDE update frequency \( k \) is 2,000/2,500/5,000 for One4All-P-20K/30K/50K and 300/500/1,000 for One4All-T-3K/5K/10K. Note that the larger datasets are built based on the smaller ones (e.g. we add 20 K samples to One4All-P-30K to build One4All-P-50K).

### 3.2. Model and Training Details

In the one-for-all training stage, the model is trained with an Adam optimizer with a warmup learning rate of 2.0e-5 for 5 epochs and a cosine learning rate from 1.0e-4 to 1.0e-6 for 90 epochs. In the all-for-one training stage, the model is finetuned with a learning rate of 5.0e-5.

### 3.3. More Ablation Study

**Loss weight** For loss weights \( \lambda_1, \lambda_2 \), We select the parameter by cross-validation and the ablation study is given in Table 1.

<table>
<thead>
<tr>
<th>( \lambda_1 )</th>
<th>( \lambda_2 )</th>
<th>EGTEA</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.2 (default)</td>
<td>0.1 (default)</td>
<td>70.8</td>
</tr>
<tr>
<td>0.5</td>
<td>0.1</td>
<td>70.8</td>
</tr>
<tr>
<td>0.1</td>
<td>0.1</td>
<td>70.6</td>
</tr>
<tr>
<td>0.2</td>
<td>0.3</td>
<td>70.6</td>
</tr>
<tr>
<td>0.2</td>
<td>0.05</td>
<td>70.5</td>
</tr>
</tbody>
</table>

Table 1: Ablation study of \( \lambda_1, \lambda_2 \).

**Dataset Components** Given that Epic-100 and Sth-Else are the main constituents in our pretraining set, we conducted an experiment with pretraining on only these two datasets. The accuracy on EPIC-100/Sth-Else are 54.1%/44.5%.

**Backbone Model on Epic-100** For a fair comparison to state-of-the-art model on Epic-100, we add an experiment using MeMViT as backbone in our model and achieve 70.7% accuracy, which is achives SOTA on the benchmark.

### 4. Further Discussion

Currently, we managed to enhance the balancedness of video properties of Ego-HOI datasets with our selection algorithm. But due to the limitation of data diversity and the trade-off between multiple video properties, we can not achieve ideal balancedness on all properties, as shown in Figure 5, 3, 6, 2, and 4. In the future, we will extend our video property-based data selection algorithm to *new data collection* and try to use the massive noisy, third-person, or weakly supervised video data. We will also enhance our baseline and leverage the unique video properties of the Ego-HOI task.

### 5. Licences

The data we use are from the following datasets and are all publicly available and only for research use. Our data pre-processing and selection will be made public.

- **EPIC KITCHENS 100** [1]: Link, Creative Commons Attribution-NonCommercial 4.0 International License
- **EGTEA Gaze+** [4]: Link
- **Ego4D** [3]: Link
- **Something Something** [2]: Link
- **Something Else** [5]: Link

And our code is based on the following code repositories. Our code will also be made public.

- **PySlowFast**: Link, Apache-2.0 License
- **OpenMMPose**: Link, Apache-2.0 License
- **CLIP**: Link, MIT license
- **ActionCLIP**: Link, MIT License

### References


