Detecting Human-Object Relationships in Videos: Supplementary Materials

Jingwei Ji Rishi Desai Juan Carlos Niebles
Stanford University
{jingweij, rdesai2, jniebles}@cs.stanford.edu

1. Background: Transformer model

Our HORT model employs both intra- and inter- transformers. Here, we will describe the intra-transformer, which essentially has the same architecture as the original transformer from [6]. As discussed in the primary text, the inputs for the transformers are as follows:

$$O'_{i,t} = W_o^T O_{i,t} + \mathcal{PE}(b^o_{i,t}), W_o \in \mathbb{R}^{d_o \times d_{Tx}} \quad (1)$$

$$R'_{j,t} = W_r^T R_{j,t} + \mathcal{PE}(b^r_{j,t}), W_r \in \mathbb{R}^{d_r \times d_{Tx}} \quad (2)$$

$$P'_{k,t} = W_p^T P_{k,t} + \mathcal{PE}(b^p_{k,t}), W_p \in \mathbb{R}^{d_p \times d_{Tx}} \quad (3)$$

For simplicity, we denote $d = d_{Tx}$ across this section.

The **transformer** (intra-transformer) is a composition of an encoder and a decoder, each of which consists of L transformer blocks with parameters $f_{\theta_L} \circ \cdots \circ f_{\theta_1}(x) \in \mathbb{R}^{n \times d}$ for n objects of dimension $\mathbb{R}^d$. A **transformer block** is a parameterized function class $f_\theta : \mathbb{R}^{n \times d} \to \mathbb{R}^{n \times d}$. For input $x \in \mathbb{R}^{n \times d}$, we compute $f_\theta(x) = z$ over each of the L blocks.

Both the encoder layer and decoder layer contain similar sub-modules of multi-head attention, feed-forward networks (FFN), and LayerNorms, but differ in regard to the input of the multi-head attention sub-layer. Each decoder layer involves an additional multi-head attention sub-layer, where the queries come from the output of the previous encoder layer, and the keys and values are from the encoder output, which is named the *encoder memory*.

Below we will describe the architecture of each transformer block in the encoder. For each of the L blocks, we have a matrix Q of queries, K of keys, and V of values. A transformer block contains H parallel heads, indexed $h \in \{1, 2, ..., H\}$. Thus, in transformer block i and head h, we have the following Q, K, and V matrices:

$$Q_h(x_i) = W^T_{h,q} x_i, W_{h,q} \in \mathbb{R}^{d \times k} \quad (4)$$

$$K_h(x_i) = W^T_{h,k} x_i, W_{h,k} \in \mathbb{R}^{d \times k} \quad (5)$$

$$V_h(x_i) = W^T_{h,v} x_i, W_{h,v} \in \mathbb{R}^{d \times k}, \quad (6)$$

where $k = d/H$. We can use the queries, keys, and values to compute the attention weights $a_{h,ij}$ on head h between transformer blocks i and j.

$$a_{h,ij} = \text{softmax}_j\left(\frac{Q_h(x_i) \cdot K_h(x_j)}{\sqrt{k}}\right) \quad (7)$$

where softmax_j means to take the softmax over the d-dimensional vector indexed by j. We can now use the attention weights to further compute the intermediary value u_i.

$$u_i = \sum_{h=1}^{H} W^T_{c,h} \sum_{j=1}^{n} a_{h,ij} V_h(x_j) \quad (8)$$

where $W_{c,h} \in \mathbb{R}^{k \times d}$. x_i is added to the attention layer output u_i as a skip connection. We then apply LayerNorm [1], a feed-forward network containing two linear projections with a ReLU operation between them, and LayerNorm again, to get the final value z_i, the output of transformer block i.

2. Qualitative results

See Figure 1 for qualitative results of our HORT model on Action Genome. In the first row, the person has changed his body posture and the objects he is interacting with. Note that our model less frequently detects the laptop when it is not involved in any interaction. The model also correctly classifies the different relationships between the person and the bed across time (when a person is lying on the bed by his body side, the ground truth label for spatial relationship is ⟨`bed-on the side of-person`⟩).

In the second row, our model correctly predicts the change of the contacting relationship between the person and the box. Occasionally the model makes a mistake in object detection, such as wrongly detecting a pillow in the fourth frame.

The video in the last row suffers from bad lighting conditions, where the chair can hardly be seen only by appearance. However, with the contextual information from the laptop, table, and the human pose, our model infers the existence of the chair in most frames. Moreover, although the shelf at the top of the frame is a clearly unobstructed object,

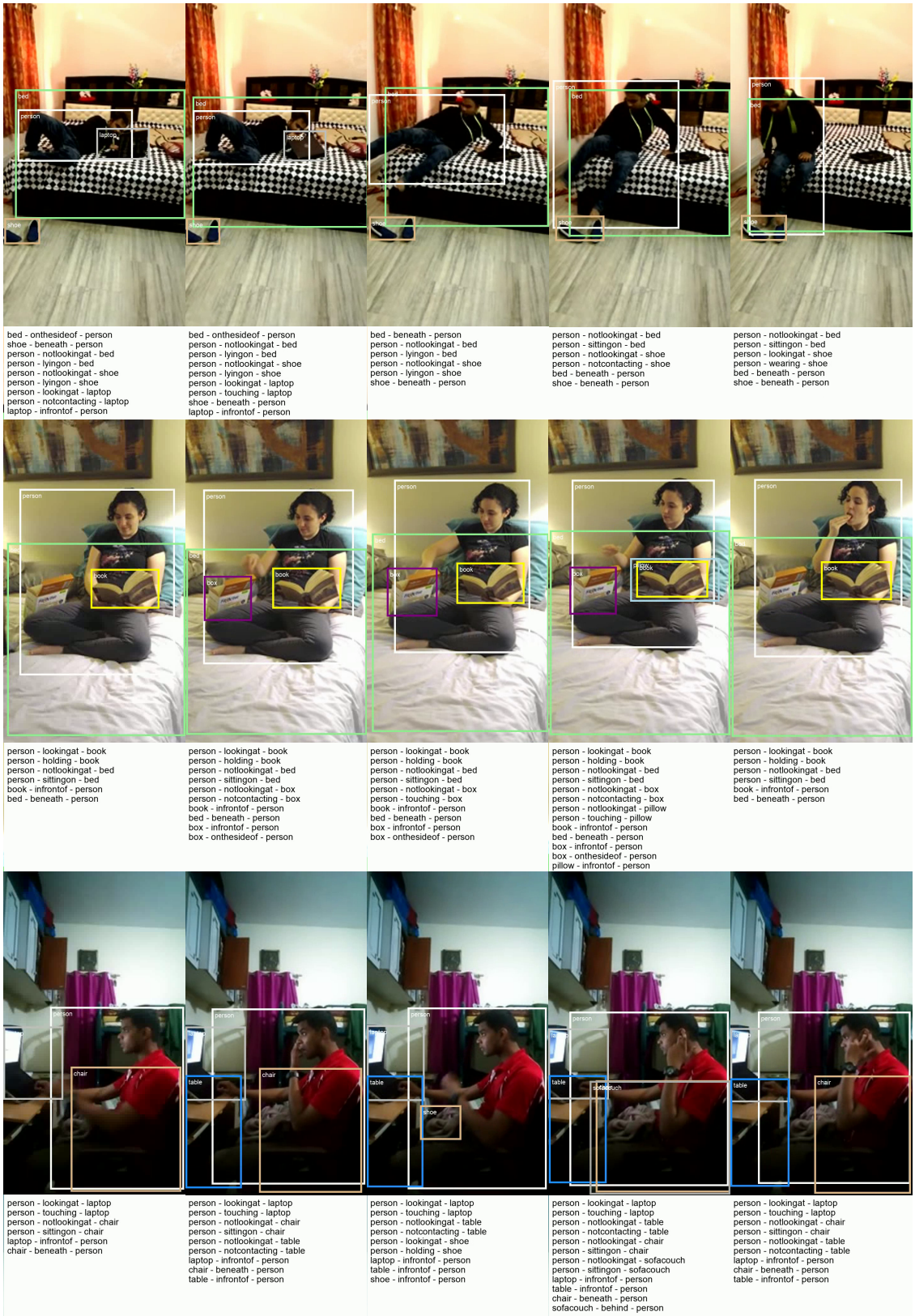

Figure 1: Qualitative results of our HORT model on Action Genome. Despite the imperfection in wrongly detected objects or misclassified relationships, our HORT model can spot interacted objects and infer the temporal dynamics of human-object relationships.

Table 1: Evaluation of scene graph generation on Action Genome with single-relationship constraint. In this experiment, we follow the same single-relationship constraint and use the same cross entropy loss as in [2]. All results of previous baselines are from [2]. Our HORT model still outperforms all baselines in this setting.

Method	PredCls				SGCls				SGGen			
	image		video		image		video		image		video	
	R@20	R@50	R@20	R@50	R@20	R@50	R@20	R@50	R@20	R@50	R@20	R@50
VRD [5]	14.75	14.85	14.51	14.60	13.65	14.69	13.41	14.44	10.28	10.94	10.04	10.70
Freq Prior [9]	32.70	32.84	32.25	32.37	31.52	32.78	31.08	32.32	24.03	24.87	23.49	24.31
IMP [7]	35.15	35.56	34.50	34.86	31.73	34.85	31.09	34.16	23.88	25.52	23.23	24.82
MSDN [4]	35.27	35.64	34.61	34.93	31.89	34.98	31.28	34.28	24.00	25.64	23.39	24.95
Graph R-CNN [8]	35.36	35.74	34.80	35.12	31.94	35.07	31.43	34.46	24.12	25.77	23.59	25.15
RelDN [10]	35.89	36.09	35.36	35.51	33.47	35.84	32.96	35.27	25.00	26.21	24.45	25.63
HORT	**35.94**	**36.11**	**35.43**	**35.57**	**34.13**	**35.97**	**33.64**	**35.45**	**25.45**	**26.35**	**24.91**	**25.80**

Figure 2: Typical failure cases: (a) biased co-occurrence of objects: chairs and tables often co-occur so the model is biased towards predicting such a combination; (b) biased relationships: the person is actually not looking at television while most examples in the training set are ⟨person - looking at - television⟩; (c) confused object detection: blankets and towels are often misclassified as their training examples are very similar.

because the person is not interacting with it, our model does not include the shelf in the output.

We also demonstrate examples of typical failure cases in Figure 2. We have observed three types of common failures.

Biased co-occurrence of objects. Some combinations of objects appear frequently in the training set, thus the model lean to predict such objects together, e.g. chairs and tables. In Figure 2(a), although there is no table in the scene, the model has mistakenly inferred its existence, probably based upon the human pose, the chair and the box.

Biased relationships. When the training set contains too many certain relationships on an object, the model often fails to predict other possible relationships. For example, when a person is watching the TV while occasionally turns his head away, it is hard for the model to capture such a change without enough training examples or an auxiliary gaze detection model.

Confused object detection. A dataset may have different labels on very similar objects. For instance, in Action Genome, "towel" and "blanket" is often hard to distinguish. The object detector can be confused in training, which results in mistakes in the overall HOR detection.

3. HOR detection in Action Genome

Careful readers may have noticed that in the Table 1 of the main text, our reported performance of scene graph generation baselines is universally better than baselines reported in Action Genome [2]. This is due to the following reasons:

1. The original baselines were restricted to output only one relationship label for each human-object pair. This is a typical setup in scene graph generation experiments on Visual Genome [3], while the restriction should be removed for Action Genome HOR detection as multiple relationships can co-exist between each human-object pair.

2. To allow the models to learn from multiple relationships between each pair, we have changed the relationship score activation from softmax to sigmoid and replaced the cross entropy loss to binary cross entropy loss accordingly.

We have applied these changes to all baselines and our model for fair comparisons.

As a reference, we have also trained and tested our HORT model with a cross entropy loss and the single-relationship constraint, so that we can directly compare to measurements reported in [2]. We report this direct comparison in Table 1. Under the setting of [2], the HORT model outperforms all previous baselines again.

References

[1] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016. 1

[2] Jingwei Ji, Ranjay Krishna, Li Fei-Fei, and Juan Carlos Niebles. Action genome: Actions as compositions of spatio-temporal scene graphs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10236–10247, 2020. 3

[3] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, Michael Bernstein, and Li Fei-Fei. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International Journal of Computer Vision*, 123(1):32–73, 2017. 3

[4] Yikang Li, Wanli Ouyang, Bolei Zhou, Kun Wang, and Xiaogang Wang. Scene graph generation from objects, phrases and region captions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1261–1270, 2017. 3

[5] Cewu Lu, Ranjay Krishna, Michael Bernstein, and Li Fei-Fei. Visual relationship detection with language priors. In *European Conference on Computer Vision*, pages 852–869. Springer, 2016. 3

[6] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017. 1

[7] Danfei Xu, Yuke Zhu, Christopher B Choy, and Li Fei-Fei. Scene graph generation by iterative message passing. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, 2017. 3

[8] Jianwei Yang, Jiasen Lu, Stefan Lee, Dhruv Batra, and Devi Parikh. Graph r-cnn for scene graph generation. *arXiv preprint arXiv:1808.00191*, 2018. 3

[9] Rowan Zellers, Mark Yatskar, Sam Thomson, and Yejin Choi. Neural motifs: Scene graph parsing with global context. *arXiv preprint arXiv:1711.06640*, 2017. 3

[10] Ji Zhang, Kevin J Shih, Ahmed Elgammal, Andrew Tao, and Bryan Catanzaro. Graphical contrastive losses for scene graph parsing. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 11535–11543, 2019. 3